



UML 2.5.1 METAMODEL

19.0 LTR

User Guide

All material contained herein is considered proprietary information owned by No Magic, Inc. and is not to be shared, copied, or reproduced by any means. All information copyright 2006-2018 by No Magic, Inc. All Rights Reserved.

INTRODUCTION

This document presents the UML Meta Model structure and its representation in MagicDraw. For more information about UML Meta Model, see the latest UML specification at <http://www.uml.org>.

The UML Meta Model document lists UML Meta Model elements in alphabetical order. The element description includes table with the following columns: attribute label, attribute name, attribute type, declaring class and sample template expression (VTL).

See the sample of the table below.

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	BaseElement	\$Abstraction[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Abstraction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Abstraction[i].classType

Table – sample of UML Meta Model element description

Attribute Label

The Attribute Label column provides name of property used in the MagicDraw elements specification dialog box.

Attribute Name

The Attribute Name column provides name of property used in the UML Metamodel.

Attribute Type

The Attribute Type column provides name of property's type (another UML Metamodel element).

Declaring Class

The Declaring Class column provides name of property's owner in metamodel hierarchy. Some meta element properties are derived from super elements.

Sample Template Expression (VTL) for reports generation

Sample Template Expression (VTL) is the last column from the table, which gives the expression for reports generation. This expression allows to print value of the element's attribute in a report. For more information about VTL code, please see "MagicDraw Report Wizard UserGuide.pdf", "Template Variables" section.

UML 2.5.1 META MODEL

Table of Contents

1.	ABSTRACTION	10
2.	ACCEPTCALLACTION	11
3.	ACCEPTEVENTACTION	12
4.	ACTION.....	13
5.	ACTIONEXECUTIONSPECIFICATION	14
6.	ACTIONINPUTPIN.....	16
7.	ACTIVITY.....	17
8.	ACTIVITYEDGE	20
9.	ACTIVITYFINALNODE	21
10.	ACTIVITYGROUP.....	22
11.	ACTIVITYNODE	23
12.	ACTIVITYPARAMETERNODE.....	24
13.	ACTIVITYPARTITION	25
14.	ACTOR	26
15.	ADDSTRUCTURALFEATUREVALUEACTION	28
16.	ADDVARIABLEVALUEACTION	30
17.	ANYRECEIVEEVENT.....	31
18.	ARTIFACT	32
19.	ASSOCIATION	34
20.	ASSOCIATIONCLASS	36
21.	BEHAVIOR.....	38
22.	BEHAVIORALFEATURE.....	40
23.	BEHAVIOREDCLASSIFIER.....	41
24.	BEHAVIOREXECUTIONSPECIFICATION.....	43
25.	BROADCASTSIGNALACTION.....	44
26.	CALLACTION	46
27.	CALLBEHAVIORACTION	47
28.	CALLEVENT	49
29.	CALLOPERATIONACTION.....	49

30.	CENTRALBUFFERNODE.....	51
31.	CHANGEEVENT	52
32.	CLASS	53
33.	CLASSIFIER.....	55
34.	CLASSIFIERTEMPLATEPARAMETER	57
35.	CLAUSE	58
36.	CLEARASSOCIATIONACTION.....	58
37.	CLEARSTRUCTURALFEATUREACTION	60
38.	CLEARVARIABLEACTION	62
39.	COLLABORATION	63
40.	COLLABORATIONUSE.....	65
41.	COM.NOMAGIC.EXPRESSIONS.OPERATION	66
42.	COMBINEDFRAGMENT	66
43.	COMMENT.....	67
44.	COMMUNICATIONPATH.....	67
45.	COMPONENT	69
46.	COMPONENTREALIZATION	71
47.	CONDITIONALNODE	72
48.	CONNECTABLEELEMENT	74
49.	CONNECTABLEELEMENTTEMPLATEPARAMETER.....	75
50.	CONNECTIONPOINTREFERENCE	76
51.	CONNECTOR	78
52.	CONNECTOREND.....	79
53.	CONSIDERIGNOREFRAGMENT.....	79
54.	CONSTRAINT	81
55.	CONTINUATION.....	82
56.	CONTROLFLOW.....	83
57.	CONTROLNODE.....	84
58.	CREATELINKACTION.....	85
59.	CREATELINKOBJECTACTION	86
60.	CREATEOBJECTACTION	87
61.	DATASTORENODE.....	89
62.	DATATYPE.....	90
63.	DECISIONNODE.....	92
64.	DEPENDENCY.....	93

65.	DEPLOYEDARTIFACT.....	94
66.	DEPLOYMENT.....	95
67.	DEPLOYMENTSPECIFICATION.....	96
68.	DEPLOYMENTTARGET.....	98
69.	DESTROYLINKACTION.....	99
70.	DESTROYOBJECTACTION.....	100
71.	DESTRUCTIONOCCURRENCESPECIFICATION.....	101
72.	DEVICE.....	103
73.	DIAGRAM.....	105
74.	DIRECTEDRELATIONSHIP.....	106
75.	DURATION.....	106
76.	DURATIONCONSTRAINT.....	107
77.	DURATIONINTERVAL.....	109
78.	DURATIONOBSERVATION.....	110
79.	ELEMENT.....	111
80.	ELEMENTIMPORT.....	111
81.	ELEMENTVALUE.....	112
82.	ENCAPSULATEDCLASSIFIER.....	113
83.	ENUMERATION.....	115
84.	ENUMERATIONLITERAL.....	117
85.	EVENT.....	118
86.	EXCEPTIONHANDLER.....	119
87.	EXECUTABLENODE.....	119
88.	EXECUTIONENVIRONMENT.....	120
89.	EXECUTIONOCCURRENCESPECIFICATION.....	123
90.	EXECUTIONSPECIFICATION.....	124
91.	EXPANSIONNODE.....	125
92.	EXPANSIONREGION.....	126
93.	EXPRESSION.....	128
94.	EXTEND.....	129
95.	EXTENSION.....	130
96.	EXTENSIONEND.....	132
97.	EXTENSIONPOINT.....	134
98.	FEATURE.....	135
99.	FINALNODE.....	135

100. FINALSTATE.....	136
101. FLOWFINALNODE.....	138
102. FORKNODE.....	139
103. FUNCTIONBEHAVIOR.....	140
104. GATE.....	142
105. GENERALIZATION.....	143
106. GENERALIZATIONSET.....	144
107. GENERALORDERING.....	145
108. IMAGE.....	146
109. INCLUDE.....	146
110. INFORMATIONFLOW.....	147
111. INFORMATIONITEM.....	148
112. INITIALNODE.....	150
113. INPUTPIN.....	151
114. INSTANCESPECIFICATION.....	152
115. INSTANCEVALUE.....	154
116. INTERACTION.....	155
117. INTERACTIONCONSTRAINT.....	157
118. INTERACTIONFRAGMENT.....	158
119. INTERACTIONOPERAND.....	159
120. INTERACTIONUSE.....	160
121. INTERFACE.....	161
122. INTERFACEREALIZATION.....	163
123. INTERRUPTIBLEACTIVITYREGION.....	164
124. INTERVAL.....	165
125. INTERVALCONSTRAINT.....	166
126. INVOCATIONACTION.....	167
127. JAVA.LANG.OBJECT.....	169
128. JOINNODE.....	169
129. LIFELINE.....	170
130. LINKACTION.....	171
131. LINKENDCREATIONDATA.....	172
132. LINKENDDATA.....	173
133. LINKENDDESTRUCTIONDATA.....	174
134. LITERALBOOLEAN.....	174

135. LITERALINTEGER	175
136. LITERALNULL	177
137. LITERALREAL	178
138. LITERALSPECIFICATION	179
139. LITERALSTRING	180
140. LITERALUNLIMITEDNATURAL	181
141. LOOPNODE	182
142. MANIFESTATION	184
143. MERGENODE	185
144. MESSAGE	186
145. MESSAGEEND	187
146. MESSAGEEVENT	188
147. MESSAGEOCCURRENCESPECIFICATION	189
148. MODEL	190
149. MULTIPLICITYELEMENT	191
150. NAMEDELEMENT	192
151. NAMESPACE	193
152. NODE	194
153. OBJECTFLOW	196
154. OBJECTNODE	197
155. OBSERVATION	198
156. OCCURRENCESPECIFICATION	199
157. OPAQUEACTION	200
158. OPAQUEBEHAVIOR	202
159. OPAQUEEXPRESSION	204
160. OPERATION	205
161. OPERATIONTEMPLATEPARAMETER	207
162. OUTPUTPIN	208
163. PACKAGE	209
164. PACKAGEABLEELEMENT	211
165. PACKAGEIMPORT	211
166. PACKAGEMERGE	212
167. PARAMETER	213
168. PARAMETERABLEELEMENT	214
169. PARAMETERSSET	215

170. PARTDECOMPOSITION	216
171. PIN.....	217
172. PORT	218
173. PRIMITIVE TYPE	220
174. PROFILE	222
175. PROFILEAPPLICATION	223
176. PROPERTY.....	224
177. PROTOCOLCONFORMANCE	225
178. PROTOCOLSTATEMACHINE	226
179. PROTOCOLTRANSITION	229
180. PSEUDOSTATE	230
181. QUALIFIERVALUE.....	231
182. RAISEEXCEPTIONACTION.....	232
183. READEXTENTACTION.....	233
184. READISCLASSIFIEDOBJECTACTION.....	234
185. READLINKACTION.....	236
186. READLINKOBJECTENDACTION	237
187. READLINKOBJECTENDQUALIFIERACTION	239
188. READSELF ACTION.....	241
189. READSTRUCTURALFEATUREACTION	243
190. READVARIABLEACTION	244
191. REALIZATION.....	246
192. RECEPTION.....	247
193. RECLASSIFYOBJECTACTION.....	248
194. REDEFINABLEELEMENT.....	249
195. REDEFINABLETEMPLATESIGNATURE	250
196. REDUCEACTION	252
197. REGION	253
198. RELATIONSHIP.....	254
199. REMOVESTRUCTURALFEATUREVALUEACTION	255
200. REMOVEVARIABLEVALUEACTION.....	257
201. REPLYACTION.....	259
202. SENDOBJECTACTION.....	260
203. SENDSIGNALACTION	262
204. SEQUENCENODE.....	263

205. SIGNAL	265
206. SIGNAL EVENT	266
207. SLOT	267
208. START CLASSIFIER BEHAVIOR ACTION	268
209. START OBJECT BEHAVIOR ACTION	270
210. STATE	271
211. STATE INVARIANT	273
212. STATE MACHINE	274
213. STEREOTYPE	276
214. STRING EXPRESSION	278
215. STRUCTURAL FEATURE	279
216. STRUCTURAL FEATURE ACTION	280
217. STRUCTURED ACTIVITY NODE	282
218. STRUCTURED CLASSIFIER	284
219. SUBSTITUTION	286
220. TEMPLATEABLE ELEMENT	287
221. TEMPLATE BINDING	287
222. TEMPLATE PARAMETER	288
223. TEMPLATE PARAMETER SUBSTITUTION	289
224. TEMPLATE SIGNATURE	290
225. TEST IDENTITY ACTION	290
226. TIME CONSTRAINT	292
227. TIME EVENT	293
228. TIME EXPRESSION	294
229. TIME INTERVAL	295
230. TIME OBSERVATION	296
231. TRANSITION	297
232. TRIGGER	298
233. TYPE	299
234. TYPED ELEMENT	300
235. UNMARSHALL ACTION	301
236. USAGE	302
237. USE CASE	303
238. VALUE PIN	305
239. VALUE SPECIFICATION	306

240. VALUE SPECIFICATION ACTION	307
241. VARIABLE	309
242. VARIABLE ACTION	310
243. VERTEX	311
244. WRITE LINK ACTION	312
245. WRITE STRUCTURAL FEATURE ACTION	313
246. WRITE VARIABLE ACTION	315

1. Abstraction

Generalization

- [Dependency](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$Abstraction[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Abstraction[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$Abstraction[i].classType</code>
Client	client	NamedElement[]	Dependency	<code>\$Abstraction[i].client</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Abstraction[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$Abstraction[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$Abstraction[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Abstraction[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Abstraction[i].invalid</code>
localID	localID	String	ModelObject	<code>\$Abstraction[i].localID</code>
Mapping	mapping	OpaqueExpression	Abstraction	<code>\$Abstraction[i].mapping</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$Abstraction[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$Abstraction[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Abstraction[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Abstraction[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Abstraction[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Abstraction[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Abstraction[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$Abstraction[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$Abstraction[i].owningPackage</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$Abstraction[i].owningTemplateParameter</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Abstraction[i].qualifiedName</code>
Related Element	relatedElement	Element[]	Relationship	<code>\$Abstraction[i].relatedElement</code>
Source	source	Element[]	DirectedRelationship	<code>\$Abstraction[i].source</code>
Supplier	supplier	NamedElement[]	Dependency	<code>\$Abstraction[i].supplier</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Abstraction[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Abstraction[i].syncElement</code>
Target	target	Element[]	DirectedRelationship	<code>\$Abstraction[i].target</code>
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$Abstraction[i].templateParameter</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$Abstraction[i].visibility</code>

2. AcceptCallAction

Generalization

- AcceptEventAction

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$AcceptCallAction[i].ID
Activity	activity	Activity	ActivityNode	\$AcceptCallAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$AcceptCallAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$AcceptCallAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$AcceptCallAction[i].clientDependency
Context	context	Classifier	Action	\$AcceptCallAction[i].context
isEditable	editable	boolean	BaseElement	\$AcceptCallAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$AcceptCallAction[i].handler
humanName	humanName	String	BaseElement	\$AcceptCallAction[i].humanName
humanType	humanType	String	BaseElement	\$AcceptCallAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$AcceptCallAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$AcceptCallAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$AcceptCallAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$AcceptCallAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$AcceptCallAction[i].incoming
Input	input	InputPin[]	Action	\$AcceptCallAction[i].input
isInvalid	invalid	boolean	ModelObject	\$AcceptCallAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$AcceptCallAction[i].leaf
localID	localID	String	ModelObject	\$AcceptCallAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$AcceptCallAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$AcceptCallAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$AcceptCallAction[i].locallyReentrant
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	\$AcceptCallAction[i].mdExtensions
Name	name	String	NamedElement	\$AcceptCallAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$AcceptCallAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$AcceptCallAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$AcceptCallAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$AcceptCallAction[i].outgoing
Output	output	OutputPin[]	Action	\$AcceptCallAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$AcceptCallAction[i].ownedComment

Owned Element	ownedElement	Element[]	Element	\$AcceptCallAction[i].ownedElement
Owner	owner	Element	Element	\$AcceptCallAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$AcceptCallAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$AcceptCallAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$AcceptCallAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$AcceptCallAction[i].redefinitionContext
Result	result	OutputPin[]	AcceptEventAction	\$AcceptCallAction[i].result
Return Information	returnInformation	OutputPin	AcceptCallAction	\$AcceptCallAction[i].returnInformation
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$AcceptCallAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$AcceptCallAction[i].syncElement
Trigger	trigger	Trigger[]	AcceptEventAction	\$AcceptCallAction[i].trigger
Is Unmarshall	unmarshall	boolean	AcceptEventAction	\$AcceptCallAction[i].unmarshall
Visibility	visibility	VisibilityKind	NamedElement	\$AcceptCallAction[i].visibility

3. AcceptEventAction

Generalization

- [Action](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$AcceptEventAction[i].ID
Activity	activity	Activity	ActivityNode	\$AcceptEventAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$AcceptEventAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$AcceptEventAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$AcceptEventAction[i].clientDependency
Context	context	Classifier	Action	\$AcceptEventAction[i].context
isEditable	editable	boolean	BaseElement	\$AcceptEventAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$AcceptEventAction[i].handler
humanName	humanName	String	BaseElement	\$AcceptEventAction[i].humanName
humanType	humanType	String	BaseElement	\$AcceptEventAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$AcceptEventAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$AcceptEventAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$AcceptEventAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$AcceptEventAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$AcceptEventAction[i].incoming
Input	input	InputPin[]	Action	\$AcceptEventAction[i].input

isInvalid	invalid	boolean	ModelObject	<code>\$AcceptEventAction[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$AcceptEventAction[i].leaf</code>
localID	localID	String	ModelObject	<code>\$AcceptEventAction[i].localID</code>
Local Postcondition	localPostcondition	Constraint[]	Action	<code>\$AcceptEventAction[i].localPostcondition</code>
Local Precondition	localPrecondition	Constraint[]	Action	<code>\$AcceptEventAction[i].localPrecondition</code>
Is Locally Reentrant	locallyReentrant	boolean	Action	<code>\$AcceptEventAction[i].locallyReentrant</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$AcceptEventAction[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$AcceptEventAction[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$AcceptEventAction[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$AcceptEventAction[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$AcceptEventAction[i].objectParent</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$AcceptEventAction[i].outgoing</code>
Output	output	OutputPin[]	Action	<code>\$AcceptEventAction[i].output</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$AcceptEventAction[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$AcceptEventAction[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$AcceptEventAction[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$AcceptEventAction[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$AcceptEventAction[i].redefinedElement</code>
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	<code>\$AcceptEventAction[i].redefinedNode</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$AcceptEventAction[i].redefinitionContext</code>
Result	result	OutputPin[]	AcceptEventAction	<code>\$AcceptEventAction[i].result</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$AcceptEventAction[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$AcceptEventAction[i].syncElement</code>
Trigger	trigger	Trigger[]	AcceptEventAction	<code>\$AcceptEventAction[i].trigger</code>
Is Unmarshall	unmarshall	boolean	AcceptEventAction	<code>\$AcceptEventAction[i].unmarshall</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$AcceptEventAction[i].visibility</code>

4. Action

Generalization

- [ExecutableNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$Action[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$Action[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Action[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$Action[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Action[i].clientDependency</code>

Context	context	Classifier	Action	<code>\$Action[i].context</code>
isEditable	editable	boolean	BaseElement	<code>\$Action[i].editable</code>
Handler	handler	ExceptionHandler[]	ExecutableNode	<code>\$Action[i].handler</code>
humanName	humanName	String	BaseElement	<code>\$Action[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Action[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$Action[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$Action[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$Action[i].inPartition</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$Action[i].inStructuredNode</code>
Incoming	incoming	ActivityEdge[]	ActivityNode	<code>\$Action[i].incoming</code>
Input	input	InputPin[]	Action	<code>\$Action[i].input</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Action[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$Action[i].leaf</code>
localID	localID	String	ModelObject	<code>\$Action[i].localID</code>
Local Postcondition	localPostcondition	Constraint[]	Action	<code>\$Action[i].localPostcondition</code>
Local Precondition	localPrecondition	Constraint[]	Action	<code>\$Action[i].localPrecondition</code>
Is Locally Reentrant	locallyReentrant	boolean	Action	<code>\$Action[i].locallyReentrant</code>
mdExtensions	mdExtensions	MDExtension[]	MDObject	<code>\$Action[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$Action[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Action[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Action[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Action[i].objectParent</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$Action[i].outgoing</code>
Output	output	OutputPin[]	Action	<code>\$Action[i].output</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Action[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Action[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$Action[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Action[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$Action[i].redefinedElement</code>
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	<code>\$Action[i].redefinedNode</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$Action[i].redefinitionContext</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Action[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Action[i].syncElement</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$Action[i].visibility</code>

5. ActionExecutionSpecification

Generalization

- ExecutionSpecification

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$ActionExecutionSpecification[i].ID
Action	action	Action	ActionExecutionSpecification	\$ActionExecutionSpecification[i].action
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ActionExecutionSpecification[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ActionExecutionSpecification[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ActionExecutionSpecification[i].clientDependency
Covered	covered	Lifeline[]	InteractionFragment	\$ActionExecutionSpecification[i].covered
isEditable	editable	boolean	BaseElement	\$ActionExecutionSpecification[i].editable
Enclosing Interaction	enclosingInteraction	Interaction	InteractionFragment	\$ActionExecutionSpecification[i].enclosingInteraction
Enclosing Operand	enclosingOperand	InteractionOperand	InteractionFragment	\$ActionExecutionSpecification[i].enclosingOperand
Finish	finish	OccurrenceSpecification	ExecutionSpecification	\$ActionExecutionSpecification[i].finish
General Ordering	generalOrdering	GeneralOrdering[]	InteractionFragment	\$ActionExecutionSpecification[i].generalOrdering
humanName	humanName	String	BaseElement	\$ActionExecutionSpecification[i].humanName
humanType	humanType	String	BaseElement	\$ActionExecutionSpecification[i].humanType
isInvalid	invalid	boolean	ModelObject	\$ActionExecutionSpecification[i].invalid
localID	localID	String	ModelObject	\$ActionExecutionSpecification[i].localID
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	\$ActionExecutionSpecification[i].mdExtensions
Name	name	String	NamedElement	\$ActionExecutionSpecification[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ActionExecutionSpecification[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ActionExecutionSpecification[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ActionExecutionSpecification[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$ActionExecutionSpecification[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ActionExecutionSpecification[i].ownedElement
Owner	owner	Element	Element	\$ActionExecutionSpecification[i].owner

Qualified Name	qualifiedName	String	NamedElement	<code>\$ActionExecutionSpecification[i].qualifiedName</code>
Start	start	OccurrenceSpecification	ExecutionSpecification	<code>\$ActionExecutionSpecification[i].start</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$ActionExecutionSpecification[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$ActionExecutionSpecification[i].syncElement</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$ActionExecutionSpecification[i].visibility</code>

6. ActionInputPin

Generalization

- [InputPin](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$ActionInputPin[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$ActionInputPin[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$ActionInputPin[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$ActionInputPin[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$ActionInputPin[i].clientDependency</code>
Is Control	control	boolean	Pin	<code>\$ActionInputPin[i].control</code>
Is Control Type	controlType	boolean	ObjectNode	<code>\$ActionInputPin[i].controlType</code>
isEditable	editable	boolean	BaseElement	<code>\$ActionInputPin[i].editable</code>
From Action	fromAction	Action	ActionInputPin	<code>\$ActionInputPin[i].fromAction</code>
humanName	humanName	String	BaseElement	<code>\$ActionInputPin[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$ActionInputPin[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$ActionInputPin[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$ActionInputPin[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$ActionInputPin[i].inPartition</code>
In State	inState	State[]	ObjectNode	<code>\$ActionInputPin[i].inState</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$ActionInputPin[i].inStructuredNode</code>
Incoming	incoming	ActivityEdge[]	ActivityNode	<code>\$ActionInputPin[i].incoming</code>
isInvalid	invalid	boolean	ModelObject	<code>\$ActionInputPin[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$ActionInputPin[i].leaf</code>
localID	localID	String	ModelObject	<code>\$ActionInputPin[i].localID</code>
Lower	lower	int	MultiplicityElement	<code>\$ActionInputPin[i].lower</code>

Lower Value	lowerValue	ValueSpecification	MultiplicityElement	<code>\$ActionInputPin[i].lowerValue</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$ActionInputPin[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$ActionInputPin[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$ActionInputPin[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$ActionInputPin[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$ActionInputPin[i].objectParent</code>
Is Ordered	ordered	boolean	MultiplicityElement	<code>\$ActionInputPin[i].ordered</code>
Ordering	ordering	ObjectNodeOrderingKind	ObjectNode	<code>\$ActionInputPin[i].ordering</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$ActionInputPin[i].outgoing</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$ActionInputPin[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$ActionInputPin[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$ActionInputPin[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$ActionInputPin[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$ActionInputPin[i].redefinedElement</code>
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	<code>\$ActionInputPin[i].redefinedNode</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$ActionInputPin[i].redefinitionContext</code>
Selection	selection	Behavior	ObjectNode	<code>\$ActionInputPin[i].selection</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$ActionInputPin[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$ActionInputPin[i].syncElement</code>
Type	type	Type	TypedElement	<code>\$ActionInputPin[i].type</code>
Is Unique	unique	boolean	MultiplicityElement	<code>\$ActionInputPin[i].unique</code>
Upper	upper	int	MultiplicityElement	<code>\$ActionInputPin[i].upper</code>
Upper Bound	upperBound	ValueSpecification	ObjectNode	<code>\$ActionInputPin[i].upperBound</code>
Upper Value	upperValue	ValueSpecification	MultiplicityElement	<code>\$ActionInputPin[i].upperValue</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$ActionInputPin[i].visibility</code>

7. Activity

Generalization

- [Behavior](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$Activity[i].ID</code>
Class	UMLClass	Class	Classifier	<code>\$Activity[i].UMLClass</code>
Is Abstract	abstract	boolean	Class	<code>\$Activity[i].abstract</code>
Is Active	active	boolean	Class	<code>\$Activity[i].active</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Activity[i].appliedStereotypeInstance</code>
Attribute	attribute	Property[]	Classifier	<code>\$Activity[i].attribute</code>

classType	classType	Class	BaseElement	\$Activity[i].classType
Classifier Behavior	classifierBehavior	Behavior	BehavioredClassifier	\$Activity[i].classifierBehavior
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Activity[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$Activity[i].collaborationUse
Context	context	BehavioredClassifier	Behavior	\$Activity[i].context
Edge	edge	ActivityEdge[]	Activity	\$Activity[i].edge
isEditable	editable	boolean	BaseElement	\$Activity[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$Activity[i].elementImport
Event	event	Event[]	Behavior	\$Activity[i].event
Extension	extension	Extension[]	Class	\$Activity[i].extension
Feature	feature	Feature[]	Classifier	\$Activity[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$Activity[i].finalSpecialization
General	general	Classifier[]	Classifier	\$Activity[i].general
Generalization	generalization	Generalization[]	Classifier	\$Activity[i].generalization
Group	group	ActivityGroup[]	Activity	\$Activity[i].group
humanName	humanName	String	BaseElement	\$Activity[i].humanName
humanType	humanType	String	BaseElement	\$Activity[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Activity[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$Activity[i].inheritedMember
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavioredClassifier	\$Activity[i].interfaceRealization
isInvalid	invalid	boolean	ModelObject	\$Activity[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$Activity[i].leaf
localID	localID	String	ModelObject	\$Activity[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Activity[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Activity[i].member
Name	name	String	NamedElement	\$Activity[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Activity[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Activity[i].namespace
Nested Classifier	nestedClassifier	Classifier[]	Class	\$Activity[i].nestedClassifier
Node	node	ActivityNode[]	Activity	\$Activity[i].node
objectParent	objectParent	BaseElement	BaseElement	\$Activity[i].objectParent
Observation	observation	Observation[]	Behavior	\$Activity[i].observation
Owned Attribute	ownedAttribute	Property[]	Class	\$Activity[i].ownedAttribute
Owned Behavior	ownedBehavior	Behavior[]	BehavioredClassifier	\$Activity[i].ownedBehavior
Owned Comment	ownedComment	Comment[]	Element	\$Activity[i].ownedComment
Owned Connector	ownedConnector	Connector[]	StructuredClassifier	\$Activity[i].ownedConnector
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Activity[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Activity[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$Activity[i].ownedMember
Owned Operation	ownedOperation	Operation[]	Class	\$Activity[i].ownedOperation

Owned Parameter	ownedParameter	Parameter[]	Behavior	\$Activity[i].ownedParameter
Owned Parameter Set	ownedParameterSet	ParameterSet[]	Behavior	\$Activity[i].ownedParameterSet
Owned Port	ownedPort	Port[]	EncapsulatedClassifier	\$Activity[i].ownedPort
Owned Reception	ownedReception	Reception[]	Class	\$Activity[i].ownedReception
Owned Rule	ownedRule	Constraint[]	Namespace	\$Activity[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$Activity[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$Activity[i].ownedUseCase
Owner	owner	Element	Element	\$Activity[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Activity[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Activity[i].owningTemplateParameter
Package	package	Package	Type	\$Activity[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$Activity[i].packageImport
Part	part	Property[]	StructuredClassifier	\$Activity[i].part
Partition	partition	ActivityPartition[]	Activity	\$Activity[i].partition
Postcondition	postcondition	Constraint[]	Behavior	\$Activity[i].postcondition
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$Activity[i].powertypeExtent
Precondition	precondition	Constraint[]	Behavior	\$Activity[i].precondition
Qualified Name	qualifiedName	String	NamedElement	\$Activity[i].qualifiedName
Is Read Only	readOnly	boolean	Activity	\$Activity[i].readOnly
Redefined Behavior	redefinedBehavior	Behavior[]	Behavior	\$Activity[i].redefinedBehavior
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$Activity[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Activity[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$Activity[i].redefinitionContext
Is Reentrant	reentrant	boolean	Behavior	\$Activity[i].reentrant
Representation	representation	CollaborationUse	Classifier	\$Activity[i].representation
Role	role	ConnectableElement[]	StructuredClassifier	\$Activity[i].role
Is Single Execution	singleExecution	boolean	Activity	\$Activity[i].singleExecution
Specification	specification	BehavioralFeature	Behavior	\$Activity[i].specification
Structured Node	structuredNode	StructuredActivityNode[]	Activity	\$Activity[i].structuredNode
Substitution	substitution	Substitution[]	Classifier	\$Activity[i].substitution
Super Class	superClass	Class[]	Class	\$Activity[i].superClass
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Activity[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Activity[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$Activity[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$Activity[i].templateParameter
Use Case	useCase	UseCase[]	Classifier	\$Activity[i].useCase
Variable	variable	Variable[]	Activity	\$Activity[i].variable
Visibility	visibility	VisibilityKind	PackageableElement	\$Activity[i].visibility

8. ActivityEdge

Generalization

- [RedefinableElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$ActivityEdge[i].ID
Activity	activity	Activity	ActivityEdge	\$ActivityEdge[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ActivityEdge[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ActivityEdge[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ActivityEdge[i].clientDependency
isEditable	editable	boolean	BaseElement	\$ActivityEdge[i].editable
Guard	guard	ValueSpecification	ActivityEdge	\$ActivityEdge[i].guard
humanName	humanName	String	BaseElement	\$ActivityEdge[i].humanName
humanType	humanType	String	BaseElement	\$ActivityEdge[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityEdge	\$ActivityEdge[i].inGroup
In Partition	inPartition	ActivityPartition[]	ActivityEdge	\$ActivityEdge[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityEdge	\$ActivityEdge[i].inStructuredNode
Interrupts	interrupts	InterruptibleActivityRegion	ActivityEdge	\$ActivityEdge[i].interrupts
isInvalid	invalid	boolean	ModelObject	\$ActivityEdge[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ActivityEdge[i].leaf
localID	localID	String	ModelObject	\$ActivityEdge[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$ActivityEdge[i].mdExtensions
Name	name	String	NamedElement	\$ActivityEdge[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ActivityEdge[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ActivityEdge[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ActivityEdge[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$ActivityEdge[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ActivityEdge[i].ownedElement
Owner	owner	Element	Element	\$ActivityEdge[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ActivityEdge[i].qualifiedName
Redefined Edge	redefinedEdge	ActivityEdge[]	ActivityEdge	\$ActivityEdge[i].redefinedEdge
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ActivityEdge[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ActivityEdge[i].redefinitionContext
Source	source	ActivityNode	ActivityEdge	\$ActivityEdge[i].source
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ActivityEdge[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ActivityEdge[i].syncElement

Target	target	ActivityNode	ActivityEdge	<code>\$ActivityEdge[i].target</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$ActivityEdge[i].visibility</code>
Weight	weight	ValueSpecification	ActivityEdge	<code>\$ActivityEdge[i].weight</code>

9. ActivityFinalNode

Generalization

- [FinalNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$ActivityFinalNode[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$ActivityFinalNode[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$ActivityFinalNode[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$ActivityFinalNode[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$ActivityFinalNode[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$ActivityFinalNode[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$ActivityFinalNode[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$ActivityFinalNode[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$ActivityFinalNode[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$ActivityFinalNode[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$ActivityFinalNode[i].inPartition</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$ActivityFinalNode[i].inStructuredNode</code>
Incoming	incoming	ActivityEdge[]	ActivityNode	<code>\$ActivityFinalNode[i].incoming</code>
isInvalid	invalid	boolean	ModelObject	<code>\$ActivityFinalNode[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$ActivityFinalNode[i].leaf</code>
localID	localID	String	ModelObject	<code>\$ActivityFinalNode[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$ActivityFinalNode[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$ActivityFinalNode[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$ActivityFinalNode[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$ActivityFinalNode[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$ActivityFinalNode[i].objectParent</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$ActivityFinalNode[i].outgoing</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$ActivityFinalNode[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$ActivityFinalNode[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$ActivityFinalNode[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$ActivityFinalNode[i].qualifiedName</code>

Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ActivityFinalNode[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ActivityFinalNode[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ActivityFinalNode[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ActivityFinalNode[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ActivityFinalNode[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$ActivityFinalNode[i].visibility

10. ActivityGroup

Generalization

- [NamedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$ActivityGroup[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ActivityGroup[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ActivityGroup[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ActivityGroup[i].clientDependency
Contained Edge	containedEdge	ActivityEdge[]	ActivityGroup	\$ActivityGroup[i].containedEdge
Contained Node	containedNode	ActivityNode[]	ActivityGroup	\$ActivityGroup[i].containedNode
isEditable	editable	boolean	BaseElement	\$ActivityGroup[i].editable
humanName	humanName	String	BaseElement	\$ActivityGroup[i].humanName
humanType	humanType	String	BaseElement	\$ActivityGroup[i].humanType
In Activity	inActivity	Activity	ActivityGroup	\$ActivityGroup[i].inActivity
isInvalid	invalid	boolean	ModelObject	\$ActivityGroup[i].invalid
localID	localID	String	ModelObject	\$ActivityGroup[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$ActivityGroup[i].mdExtensions
Name	name	String	NamedElement	\$ActivityGroup[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ActivityGroup[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ActivityGroup[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ActivityGroup[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$ActivityGroup[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ActivityGroup[i].ownedElement
Owner	owner	Element	Element	\$ActivityGroup[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ActivityGroup[i].qualifiedName
Subgroup	subgroup	ActivityGroup[]	ActivityGroup	\$ActivityGroup[i].subgroup
Super Group	superGroup	ActivityGroup	ActivityGroup	\$ActivityGroup[i].superGroup
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ActivityGroup[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ActivityGroup[i].syncElement

Visibility	visibility	VisibilityKind	NamedElement	\$ActivityGroup[i].visibility

11. ActivityNode

Generalization

- RedefinableElement

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$ActivityNode[i].ID
Activity	activity	Activity	ActivityNode	\$ActivityNode[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ActivityNode[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ActivityNode[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ActivityNode[i].clientDependency
isEditable	editable	boolean	BaseElement	\$ActivityNode[i].editable
humanName	humanName	String	BaseElement	\$ActivityNode[i].humanName
humanType	humanType	String	BaseElement	\$ActivityNode[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ActivityNode[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ActivityNode[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ActivityNode[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ActivityNode[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ActivityNode[i].incoming
isInvalid	invalid	boolean	ModelObject	\$ActivityNode[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ActivityNode[i].leaf
localID	localID	String	ModelObject	\$ActivityNode[i].localID
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	\$ActivityNode[i].mdExtensions
Name	name	String	NamedElement	\$ActivityNode[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ActivityNode[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ActivityNode[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ActivityNode[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ActivityNode[i].outgoing
Owned Comment	ownedComment	Comment[]	Element	\$ActivityNode[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ActivityNode[i].ownedElement
Owner	owner	Element	Element	\$ActivityNode[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ActivityNode[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ActivityNode[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ActivityNode[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ActivityNode[i].redefinitionContext

Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$ActivityNode[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$ActivityNode[i].syncElement</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$ActivityNode[i].visibility</code>

12. ActivityParameterNode

Generalization

- [ObjectNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$ActivityParameterNode[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$ActivityParameterNode[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$ActivityParameterNode[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$ActivityParameterNode[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$ActivityParameterNode[i].clientDependency</code>
Is Control Type	controlType	boolean	ObjectNode	<code>\$ActivityParameterNode[i].controlType</code>
isEditable	editable	boolean	BaseElement	<code>\$ActivityParameterNode[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$ActivityParameterNode[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$ActivityParameterNode[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$ActivityParameterNode[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$ActivityParameterNode[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$ActivityParameterNode[i].inPartition</code>
In State	inState	State[]	ObjectNode	<code>\$ActivityParameterNode[i].inState</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$ActivityParameterNode[i].inStructuredNode</code>
Incoming	incoming	ActivityEdge[]	ActivityNode	<code>\$ActivityParameterNode[i].incoming</code>
isInvalid	invalid	boolean	ModelObject	<code>\$ActivityParameterNode[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$ActivityParameterNode[i].leaf</code>
localID	localID	String	ModelObject	<code>\$ActivityParameterNode[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$ActivityParameterNode[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$ActivityParameterNode[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$ActivityParameterNode[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$ActivityParameterNode[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$ActivityParameterNode[i].objectParent</code>
Ordering	ordering	ObjectNodeOrderingKind	ObjectNode	<code>\$ActivityParameterNode[i].ordering</code>

Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ActivityParameterNode[i].outgoing
Owned Comment	ownedComment	Comment[]	Element	\$ActivityParameterNode[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ActivityParameterNode[i].ownedElement
Owner	owner	Element	Element	\$ActivityParameterNode[i].owner
Parameter	parameter	Parameter	ActivityParameterNode	\$ActivityParameterNode[i].parameter
Qualified Name	qualifiedName	String	NamedElement	\$ActivityParameterNode[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ActivityParameterNode[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ActivityParameterNode[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ActivityParameterNode[i].redefinitionContext
Selection	selection	Behavior	ObjectNode	\$ActivityParameterNode[i].selection
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ActivityParameterNode[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ActivityParameterNode[i].syncElement
Type	type	Type	TypedElement	\$ActivityParameterNode[i].type
Upper Bound	upperBound	ValueSpecification	ObjectNode	\$ActivityParameterNode[i].upperBound
Visibility	visibility	VisibilityKind	NamedElement	\$ActivityParameterNode[i].visibility

13. ActivityPartition

Generalization

- [ActivityGroup](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ActivityPartition[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ActivityPartition[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ActivityPartition[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ActivityPartition[i].clientDependency
Contained Edge	containedEdge	ActivityEdge[]	ActivityGroup	\$ActivityPartition[i].containedEdge
Contained Node	containedNode	ActivityNode[]	ActivityGroup	\$ActivityPartition[i].containedNode
Is Dimension	dimension	boolean	ActivityPartition	\$ActivityPartition[i].dimension
Edge	edge	ActivityEdge[]	ActivityPartition	\$ActivityPartition[i].edge
isEditable	editable	boolean	BaseElement	\$ActivityPartition[i].editable
Is External	external	boolean	ActivityPartition	\$ActivityPartition[i].external
humanName	humanName	String	BaseElement	\$ActivityPartition[i].humanName
humanType	humanType	String	BaseElement	\$ActivityPartition[i].humanType

In Activity	inActivity	Activity	ActivityGroup	\$ActivityPartition[i].inActivity
isInvalid	invalid	boolean	ModelObject	\$ActivityPartition[i].invalid
localID	localID	String	ModelObject	\$ActivityPartition[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ActivityPartition[i].mdExtensions
Name	name	String	NamedElement	\$ActivityPartition[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ActivityPartition[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ActivityPartition[i].namespace
Node	node	ActivityNode[]	ActivityPartition	\$ActivityPartition[i].node
objectParent	objectParent	BaseElement	BaseElement	\$ActivityPartition[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$ActivityPartition[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ActivityPartition[i].ownedElement
Owner	owner	Element	Element	\$ActivityPartition[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ActivityPartition[i].qualifiedName
Represents	represents	Element	ActivityPartition	\$ActivityPartition[i].represents
Subgroup	subgroup	ActivityGroup[]	ActivityGroup	\$ActivityPartition[i].subgroup
Subpartition	subpartition	ActivityPartition[]	ActivityPartition	\$ActivityPartition[i].subpartition
Super Group	superGroup	ActivityGroup	ActivityGroup	\$ActivityPartition[i].superGroup
Super Partition	superPartition	ActivityPartition	ActivityPartition	\$ActivityPartition[i].superPartition
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ActivityPartition[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ActivityPartition[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$ActivityPartition[i].visibility

14. Actor

Generalization

- [BehavoredClassifier](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Actor[i].ID
Class	UMLClass	Class	Classifier	\$Actor[i].UMLClass
Is Abstract	abstract	boolean	Classifier	\$Actor[i].abstract
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Actor[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$Actor[i].attribute
classType	classType	Class	BaseElement	\$Actor[i].classType
Classifier Behavior	classifierBehavior	Behavior	BehavoredClassifier	\$Actor[i].classifierBehavior
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Actor[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$Actor[i].collaborationUse
isEditable	editable	boolean	BaseElement	\$Actor[i].editable

Element Import	elementImport	ElementImport[]	Namespace	\$Actor[i].elementImport
Feature	feature	Feature[]	Classifier	\$Actor[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$Actor[i].finalSpecialization
General	general	Classifier[]	Classifier	\$Actor[i].general
Generalization	generalization	Generalization[]	Classifier	\$Actor[i].generalization
humanName	humanName	String	BaseElement	\$Actor[i].humanName
humanType	humanType	String	BaseElement	\$Actor[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Actor[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$Actor[i].inheritedMember
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavioedClassifier	\$Actor[i].interfaceRealization
isInvalid	invalid	boolean	ModelObject	\$Actor[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$Actor[i].leaf
localID	localID	String	ModelObject	\$Actor[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$Actor[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Actor[i].member
Name	name	String	NamedElement	\$Actor[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Actor[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Actor[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Actor[i].objectParent
Owned Behavior	ownedBehavior	Behavior[]	BehavioedClassifier	\$Actor[i].ownedBehavior
Owned Comment	ownedComment	Comment[]	Element	\$Actor[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Actor[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Actor[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$Actor[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$Actor[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$Actor[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$Actor[i].ownedUseCase
Owner	owner	Element	Element	\$Actor[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Actor[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Actor[i].owningTemplateParameter
Package	package	Package	Type	\$Actor[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$Actor[i].packageImport
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$Actor[i].powertypeExtent
Qualified Name	qualifiedName	String	NamedElement	\$Actor[i].qualifiedName
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$Actor[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Actor[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$Actor[i].redefinitionContext
Representation	representation	CollaborationUse	Classifier	\$Actor[i].representation
Substitution	substitution	Substitution[]	Classifier	\$Actor[i].substitution

Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Actor[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Actor[i].syncElement</code>
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	<code>\$Actor[i].templateBinding</code>
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	<code>\$Actor[i].templateParameter</code>
Use Case	useCase	UseCase[]	Classifier	<code>\$Actor[i].useCase</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$Actor[i].visibility</code>

15. AddStructuralFeatureValueAction

Generalization

- [WriteStructuralFeatureAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$AddStructuralFeatureValueAction[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$AddStructuralFeatureValueAction[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$AddStructuralFeatureValueAction[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$AddStructuralFeatureValueAction[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$AddStructuralFeatureValueAction[i].clientDependency</code>
Context	context	Classifier	Action	<code>\$AddStructuralFeatureValueAction[i].context</code>
isEditable	editable	boolean	BaseElement	<code>\$AddStructuralFeatureValueAction[i].editable</code>
Handler	handler	ExceptionHandler[]	ExecutableNode	<code>\$AddStructuralFeatureValueAction[i].handler</code>
humanName	humanName	String	BaseElement	<code>\$AddStructuralFeatureValueAction[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$AddStructuralFeatureValueAction[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$AddStructuralFeatureValueAction[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$AddStructuralFeatureValueAction[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$AddStructuralFeatureValueAction[i].inPartition</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$AddStructuralFeatureValueAction[i].inStruc</code>

				turedNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$AddStructuralFeatureValueAction[i].incoming
Input	input	InputPin[]	Action	\$AddStructuralFeatureValueAction[i].input
Insert At	insertAt	InputPin	AddStructuralFeatureValueAction	\$AddStructuralFeatureValueAction[i].insertAt
isInvalid	invalid	boolean	ModelObject	\$AddStructuralFeatureValueAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$AddStructuralFeatureValueAction[i].leaf
localID	localID	String	ModelObject	\$AddStructuralFeatureValueAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$AddStructuralFeatureValueAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$AddStructuralFeatureValueAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$AddStructuralFeatureValueAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$AddStructuralFeatureValueAction[i].mdExtensions
Name	name	String	NamedElement	\$AddStructuralFeatureValueAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$AddStructuralFeatureValueAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$AddStructuralFeatureValueAction[i].namespace
Object	object	InputPin	StructuralFeatureAction	\$AddStructuralFeatureValueAction[i].object
objectParent	objectParent	BaseElement	BaseElement	\$AddStructuralFeatureValueAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$AddStructuralFeatureValueAction[i].outgoing
Output	output	OutputPin[]	Action	\$AddStructuralFeatureValueAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$AddStructuralFeatureValueAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$AddStructuralFeatureValueAction[i].ownedElement
Owner	owner	Element	Element	\$AddStructuralFeatureValueAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$AddStructuralFeatureValueAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$AddStructuralFeatureValueAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$AddStructuralFeatureValueAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$AddStructuralFeatureValueAction[i].redefinitionContext

Is Replace All	replaceAll	boolean	AddStructuralFeatureValueAction	\$AddStructuralFeatureValueAction[i].replaceAll
Result	result	OutputPin	WriteStructuralFeatureAction	\$AddStructuralFeatureValueAction[i].result
Structural Feature	structuralFeature	StructuralFeature	StructuralFeatureAction	\$AddStructuralFeatureValueAction[i].structuralFeature
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$AddStructuralFeatureValueAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$AddStructuralFeatureValueAction[i].syncElement
Value	value	InputPin	WriteStructuralFeatureAction	\$AddStructuralFeatureValueAction[i].value
Visibility	visibility	VisibilityKind	NamedElement	\$AddStructuralFeatureValueAction[i].visibility

16. AddVariableValueAction

Generalization

- WriteVariableAction

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$AddVariableValueAction[i].ID
Activity	activity	Activity	ActivityNode	\$AddVariableValueAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$AddVariableValueAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$AddVariableValueAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$AddVariableValueAction[i].clientDependency
Context	context	Classifier	Action	\$AddVariableValueAction[i].context
isEditable	editable	boolean	BaseElement	\$AddVariableValueAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$AddVariableValueAction[i].handler
humanName	humanName	String	BaseElement	\$AddVariableValueAction[i].humanName
humanType	humanType	String	BaseElement	\$AddVariableValueAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$AddVariableValueAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$AddVariableValueAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$AddVariableValueAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$AddVariableValueAction[i].inStructuredNode

Incoming	incoming	ActivityEdge[]	ActivityNode	\$AddVariableValueAction[i].incoming
Input	input	InputPin[]	Action	\$AddVariableValueAction[i].input
Insert At	insertAt	InputPin	AddVariableValueAction	\$AddVariableValueAction[i].insertAt
isInvalid	invalid	boolean	ModelObject	\$AddVariableValueAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$AddVariableValueAction[i].leaf
localID	localID	String	ModelObject	\$AddVariableValueAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$AddVariableValueAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$AddVariableValueAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$AddVariableValueAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$AddVariableValueAction[i].mdExtensions
Name	name	String	NamedElement	\$AddVariableValueAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$AddVariableValueAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$AddVariableValueAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$AddVariableValueAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$AddVariableValueAction[i].outgoing
Output	output	OutputPin[]	Action	\$AddVariableValueAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$AddVariableValueAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$AddVariableValueAction[i].ownedElement
Owner	owner	Element	Element	\$AddVariableValueAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$AddVariableValueAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$AddVariableValueAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$AddVariableValueAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$AddVariableValueAction[i].redefinitionContext
Is Replace All	replaceAll	boolean	AddVariableValueAction	\$AddVariableValueAction[i].replaceAll
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$AddVariableValueAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$AddVariableValueAction[i].syncElement
Value	value	InputPin	WriteVariableAction	\$AddVariableValueAction[i].value
Variable	variable	Variable	VariableAction	\$AddVariableValueAction[i].variable
Visibility	visibility	VisibilityKind	NamedElement	\$AddVariableValueAction[i].visibility

17. AnyReceiveEvent

Generalization

- **MessageEvent**

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$AnyReceiveEvent[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$AnyReceiveEvent[i].appliedStereotypeInstance
Behavior	behavior	Behavior	Event	\$AnyReceiveEvent[i].behavior
classType	classType	Class	BaseElement	\$AnyReceiveEvent[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$AnyReceiveEvent[i].clientDependency
isEditable	editable	boolean	BaseElement	\$AnyReceiveEvent[i].editable
humanName	humanName	String	BaseElement	\$AnyReceiveEvent[i].humanName
humanType	humanType	String	BaseElement	\$AnyReceiveEvent[i].humanType
isInvalid	invalid	boolean	ModelObject	\$AnyReceiveEvent[i].invalid
localID	localID	String	ModelObject	\$AnyReceiveEvent[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$AnyReceiveEvent[i].mdExtensions
Name	name	String	NamedElement	\$AnyReceiveEvent[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$AnyReceiveEvent[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$AnyReceiveEvent[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$AnyReceiveEvent[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$AnyReceiveEvent[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$AnyReceiveEvent[i].ownedElement
Owner	owner	Element	Element	\$AnyReceiveEvent[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$AnyReceiveEvent[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$AnyReceiveEvent[i].owningTemplateParameter
Qualified Name	qualifiedName	String	NamedElement	\$AnyReceiveEvent[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$AnyReceiveEvent[i].supplierDependency
Sync Element	syncElement	Element	Element	\$AnyReceiveEvent[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$AnyReceiveEvent[i].templateParameter
Visibility	visibility	VisibilityKind	PackageableElement	\$AnyReceiveEvent[i].visibility

18. Artifact

Generalization

- [Classifier](#)
- [DeployedArtifact](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class)	Sample Template Expression (VTL)

			Owner)	
ID	ID	String	MDOBJECT	\$Artifact[i].ID
Class	UMLClass	Class	Classifier	\$Artifact[i].UMLClass
Is Abstract	abstract	boolean	Classifier	\$Artifact[i].abstract
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Artifact[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$Artifact[i].attribute
classType	classType	Class	BaseElement	\$Artifact[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Artifact[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$Artifact[i].collaborationUse
isEditable	editable	boolean	BaseElement	\$Artifact[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$Artifact[i].elementImport
Feature	feature	Feature[]	Classifier	\$Artifact[i].feature
File Name	fileName	String	Artifact	\$Artifact[i].fileName
Is Final Specialization	finalSpecialization	boolean	Classifier	\$Artifact[i].finalSpecialization
General	general	Classifier[]	Classifier	\$Artifact[i].general
Generalization	generalization	Generalization[]	Classifier	\$Artifact[i].generalization
humanName	humanName	String	BaseElement	\$Artifact[i].humanName
humanType	humanType	String	BaseElement	\$Artifact[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Artifact[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$Artifact[i].inheritedMember
isInvalid	invalid	boolean	ModelObject	\$Artifact[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$Artifact[i].leaf
localID	localID	String	ModelObject	\$Artifact[i].localID
Manifestation	manifestation	Manifestation[]	Artifact	\$Artifact[i].manifestation
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$Artifact[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Artifact[i].member
Name	name	String	NamedElement	\$Artifact[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Artifact[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Artifact[i].namespace
Nested Artifact	nestedArtifact	Artifact[]	Artifact	\$Artifact[i].nestedArtifact
objectParent	objectParent	BaseElement	BaseElement	\$Artifact[i].objectParent
Owned Attribute	ownedAttribute	Property[]	Artifact	\$Artifact[i].ownedAttribute
Owned Comment	ownedComment	Comment[]	Element	\$Artifact[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Artifact[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Artifact[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$Artifact[i].ownedMember
Owned Operation	ownedOperation	Operation[]	Artifact	\$Artifact[i].ownedOperation
Owned Rule	ownedRule	Constraint[]	Namespace	\$Artifact[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$Artifact[i].ownedTemplateSignature

Owned Use Case	ownedUseCase	UseCase[]	Classifier	<code>\$Artifact[i].ownedUseCase</code>
Owner	owner	Element	Element	<code>\$Artifact[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$Artifact[i].owningPackage</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$Artifact[i].owningTemplateParameter</code>
Package	package	Package	Type	<code>\$Artifact[i].package</code>
Package Import	packageImport	PackageImport[]	Namespace	<code>\$Artifact[i].packageImport</code>
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	<code>\$Artifact[i].powertypeExtent</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Artifact[i].qualifiedName</code>
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	<code>\$Artifact[i].redefinedClassifier</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$Artifact[i].redefinedElement</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$Artifact[i].redefinitionContext</code>
Representation	representation	CollaborationUse	Classifier	<code>\$Artifact[i].representation</code>
Substitution	substitution	Substitution[]	Classifier	<code>\$Artifact[i].substitution</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Artifact[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Artifact[i].syncElement</code>
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	<code>\$Artifact[i].templateBinding</code>
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	<code>\$Artifact[i].templateParameter</code>
Use Case	useCase	UseCase[]	Classifier	<code>\$Artifact[i].useCase</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$Artifact[i].visibility</code>

19. Association

Generalization

- [Classifier](#)
- [Relationship](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$Association[i].ID</code>
Class	UMLClass	Class	Classifier	<code>\$Association[i].UMLClass</code>
Is Abstract	abstract	boolean	Classifier	<code>\$Association[i].abstract</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Association[i].appliedStereotypeInstance</code>
Attribute	attribute	Property[]	Classifier	<code>\$Association[i].attribute</code>
classType	classType	Class	BaseElement	<code>\$Association[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Association[i].clientDependency</code>
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	<code>\$Association[i].collaborationUse</code>
Is Derived	derived	boolean	Association	<code>\$Association[i].derived</code>
isEditable	editable	boolean	BaseElement	<code>\$Association[i].editable</code>

Element Import	elementImport	ElementImport[]	Namespace	\$Association[i].elementImport
End Type	endType	Type[]	Association	\$Association[i].endType
Feature	feature	Feature[]	Classifier	\$Association[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$Association[i].finalSpecialization
General	general	Classifier[]	Classifier	\$Association[i].general
Generalization	generalization	Generalization[]	Classifier	\$Association[i].generalization
humanName	humanName	String	BaseElement	\$Association[i].humanName
humanType	humanType	String	BaseElement	\$Association[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Association[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$Association[i].inheritedMember
isInvalid	invalid	boolean	ModelObject	\$Association[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$Association[i].leaf
localID	localID	String	ModelObject	\$Association[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Association[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Association[i].member
Member End	memberEnd	Property[]	Association	\$Association[i].memberEnd
Name	name	String	NamedElement	\$Association[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Association[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Association[i].namespace
Navigable Owned End	navigableOwnedEnd	Property[]	Association	\$Association[i].navigableOwnedEnd
objectParent	objectParent	BaseElement	BaseElement	\$Association[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Association[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Association[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Association[i].ownedElement
Owned End	ownedEnd	Property[]	Association	\$Association[i].ownedEnd
Owned Member	ownedMember	NamedElement[]	Namespace	\$Association[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$Association[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$Association[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$Association[i].ownedUseCase
Owner	owner	Element	Element	\$Association[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Association[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Association[i].owningTemplateParameter
Package	package	Package	Type	\$Association[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$Association[i].packageImport
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$Association[i].powertypeExtent
Qualified Name	qualifiedName	String	NamedElement	\$Association[i].qualifiedName
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$Association[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Association[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$Association[i].redefinitionContext

Related Element	relatedElement	Element[]	Relationship	\$Association[i].relatedElement
Representation	representation	CollaborationUse	Classifier	\$Association[i].representation
Substitution	substitution	Substitution[]	Classifier	\$Association[i].substitution
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Association[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Association[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$Association[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$Association[i].templateParameter
Use Case	useCase	UseCase[]	Classifier	\$Association[i].useCase
Visibility	visibility	VisibilityKind	PackageableElement	\$Association[i].visibility

20. AssociationClass

Generalization

- [Association](#)
- [Class](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$AssociationClass[i].ID
Class	UMLClass	Class	Classifier	\$AssociationClass[i].UMLClass
Is Abstract	abstract	boolean	Class	\$AssociationClass[i].abstract
Is Active	active	boolean	Class	\$AssociationClass[i].active
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$AssociationClass[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$AssociationClass[i].attribute
classType	classType	Class	BaseElement	\$AssociationClass[i].classType
Classifier Behavior	classifierBehavior	Behavior	BehavioredClassifier	\$AssociationClass[i].classifierBehavior
Client Dependency	clientDependency	Dependency[]	NamedElement	\$AssociationClass[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$AssociationClass[i].collaborationUse
Is Derived	derived	boolean	Association	\$AssociationClass[i].derived
isEditable	editable	boolean	BaseElement	\$AssociationClass[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$AssociationClass[i].elementImport
End Type	endType	Type[]	Association	\$AssociationClass[i].endType
Extension	extension	Extension[]	Class	\$AssociationClass[i].extension
Feature	feature	Feature[]	Classifier	\$AssociationClass[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$AssociationClass[i].finalSpecialization
General	general	Classifier[]	Classifier	\$AssociationClass[i].general
Generalization	generalization	Generalization[]	Classifier	\$AssociationClass[i].generalization

humanName	humanName	String	BaseElement	\$AssociationClass[i].humanName
humanType	humanType	String	BaseElement	\$AssociationClass[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$AssociationClass[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$AssociationClass[i].inheritedMember
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavioredClassifier	\$AssociationClass[i].interfaceRealization
isInvalid	invalid	boolean	ModelObject	\$AssociationClass[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$AssociationClass[i].leaf
localID	localID	String	ModelObject	\$AssociationClass[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$AssociationClass[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$AssociationClass[i].member
Member End	memberEnd	Property[]	Association	\$AssociationClass[i].memberEnd
Name	name	String	NamedElement	\$AssociationClass[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$AssociationClass[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$AssociationClass[i].namespace
Navigable Owned End	navigableOwnedEnd	Property[]	Association	\$AssociationClass[i].navigableOwnedEnd
Nested Classifier	nestedClassifier	Classifier[]	Class	\$AssociationClass[i].nestedClassifier
objectParent	objectParent	BaseElement	BaseElement	\$AssociationClass[i].objectParent
Owned Attribute	ownedAttribute	Property[]	Class	\$AssociationClass[i].ownedAttribute
Owned Behavior	ownedBehavior	Behavior[]	BehavioredClassifier	\$AssociationClass[i].ownedBehavior
Owned Comment	ownedComment	Comment[]	Element	\$AssociationClass[i].ownedComment
Owned Connector	ownedConnector	Connector[]	StructuredClassifier	\$AssociationClass[i].ownedConnector
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$AssociationClass[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$AssociationClass[i].ownedElement
Owned End	ownedEnd	Property[]	Association	\$AssociationClass[i].ownedEnd
Owned Member	ownedMember	NamedElement[]	Namespace	\$AssociationClass[i].ownedMember
Owned Operation	ownedOperation	Operation[]	Class	\$AssociationClass[i].ownedOperation
Owned Port	ownedPort	Port[]	EncapsulatedClassifier	\$AssociationClass[i].ownedPort
Owned Reception	ownedReception	Reception[]	Class	\$AssociationClass[i].ownedReception
Owned Rule	ownedRule	Constraint[]	Namespace	\$AssociationClass[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$AssociationClass[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$AssociationClass[i].ownedUseCase
Owner	owner	Element	Element	\$AssociationClass[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$AssociationClass[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$AssociationClass[i].owningTemplateParameter
Package	package	Package	Type	\$AssociationClass[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$AssociationClass[i].packageImport
Part	part	Property[]	StructuredClassifier	\$AssociationClass[i].part
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$AssociationClass[i].powertypeExtent

Qualified Name	qualifiedName	String	NamedElement	\$AssociationClass[i].qualifiedName
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$AssociationClass[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$AssociationClass[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$AssociationClass[i].redefinitionContext
Related Element	relatedElement	Element[]	Relationship	\$AssociationClass[i].relatedElement
Representation	representation	CollaborationUse	Classifier	\$AssociationClass[i].representation
Role	role	ConnectableElement[]	StructuredClassifier	\$AssociationClass[i].role
Substitution	substitution	Substitution[]	Classifier	\$AssociationClass[i].substitution
Super Class	superClass	Class[]	Class	\$AssociationClass[i].superClass
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$AssociationClass[i].supplierDependency
Sync Element	syncElement	Element	Element	\$AssociationClass[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$AssociationClass[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$AssociationClass[i].templateParameter
Use Case	useCase	UseCase[]	Classifier	\$AssociationClass[i].useCase
Visibility	visibility	VisibilityKind	PackageableElement	\$AssociationClass[i].visibility

21. Behavior

Generalization

- Class

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Behavior[i].ID
Class	UMLClass	Class	Classifier	\$Behavior[i].UMLClass
Is Abstract	abstract	boolean	Class	\$Behavior[i].abstract
Is Active	active	boolean	Class	\$Behavior[i].active
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Behavior[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$Behavior[i].attribute
classType	classType	Class	BaseElement	\$Behavior[i].classType
Classifier Behavior	classifierBehavior	Behavior	BehavioredClassifier	\$Behavior[i].classifierBehavior
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Behavior[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$Behavior[i].collaborationUse
Context	context	BehavioredClassifier	Behavior	\$Behavior[i].context
isEditable	editable	boolean	BaseElement	\$Behavior[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$Behavior[i].elementImport
Event	event	Event[]	Behavior	\$Behavior[i].event
Extension	extension	Extension[]	Class	\$Behavior[i].extension

Feature	feature	Feature[]	Classifier	\$Behavior[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$Behavior[i].finalSpecialization
General	general	Classifier[]	Classifier	\$Behavior[i].general
Generalization	generalization	Generalization[]	Classifier	\$Behavior[i].generalization
humanName	humanName	String	BaseElement	\$Behavior[i].humanName
humanType	humanType	String	BaseElement	\$Behavior[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Behavior[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$Behavior[i].inheritedMember
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavioredClassifier	\$Behavior[i].interfaceRealization
isInvalid	invalid	boolean	ModelObject	\$Behavior[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$Behavior[i].leaf
localID	localID	String	ModelObject	\$Behavior[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Behavior[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Behavior[i].member
Name	name	String	NamedElement	\$Behavior[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Behavior[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Behavior[i].namespace
Nested Classifier	nestedClassifier	Classifier[]	Class	\$Behavior[i].nestedClassifier
objectParent	objectParent	BaseElement	BaseElement	\$Behavior[i].objectParent
Observation	observation	Observation[]	Behavior	\$Behavior[i].observation
Owned Attribute	ownedAttribute	Property[]	Class	\$Behavior[i].ownedAttribute
Owned Behavior	ownedBehavior	Behavior[]	BehavioredClassifier	\$Behavior[i].ownedBehavior
Owned Comment	ownedComment	Comment[]	Element	\$Behavior[i].ownedComment
Owned Connector	ownedConnector	Connector[]	StructuredClassifier	\$Behavior[i].ownedConnector
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Behavior[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Behavior[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$Behavior[i].ownedMember
Owned Operation	ownedOperation	Operation[]	Class	\$Behavior[i].ownedOperation
Owned Parameter	ownedParameter	Parameter[]	Behavior	\$Behavior[i].ownedParameter
Owned Parameter Set	ownedParameterSet	ParameterSet[]	Behavior	\$Behavior[i].ownedParameterSet
Owned Port	ownedPort	Port[]	EncapsulatedClassifier	\$Behavior[i].ownedPort
Owned Reception	ownedReception	Reception[]	Class	\$Behavior[i].ownedReception
Owned Rule	ownedRule	Constraint[]	Namespace	\$Behavior[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$Behavior[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$Behavior[i].ownedUseCase
Owner	owner	Element	Element	\$Behavior[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Behavior[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Behavior[i].owningTemplateParameter
Package	package	Package	Type	\$Behavior[i].package

Package Import	packageImport	PackageImport[]	Namespace	\$Behavior[i].packageImport
Part	part	Property[]	StructuredClassifier	\$Behavior[i].part
Postcondition	postcondition	Constraint[]	Behavior	\$Behavior[i].postcondition
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$Behavior[i].powertypeExtent
Precondition	precondition	Constraint[]	Behavior	\$Behavior[i].precondition
Qualified Name	qualifiedName	String	NamedElement	\$Behavior[i].qualifiedName
Redefined Behavior	redefinedBehavior	Behavior[]	Behavior	\$Behavior[i].redefinedBehavior
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$Behavior[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Behavior[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$Behavior[i].redefinitionContext
Is Reentrant	reentrant	boolean	Behavior	\$Behavior[i].reentrant
Representation	representation	CollaborationUse	Classifier	\$Behavior[i].representation
Role	role	ConnectableElement[]	StructuredClassifier	\$Behavior[i].role
Specification	specification	BehavioralFeature	Behavior	\$Behavior[i].specification
Substitution	substitution	Substitution[]	Classifier	\$Behavior[i].substitution
Super Class	superClass	Class[]	Class	\$Behavior[i].superClass
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Behavior[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Behavior[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$Behavior[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$Behavior[i].templateParameter
Use Case	useCase	UseCase[]	Classifier	\$Behavior[i].useCase
Visibility	visibility	VisibilityKind	PackageableElement	\$Behavior[i].visibility

22. BehavioralFeature

Generalization

- [Feature](#)
- [Namespace](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	id	String	MDOObject	\$BehavioralFeature[i].id
Is Abstract	abstract	boolean	BehavioralFeature	\$BehavioralFeature[i].abstract
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$BehavioralFeature[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$BehavioralFeature[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$BehavioralFeature[i].clientDependency
Concurrency	concurrency	CallConcurrencyKind	BehavioralFeature	\$BehavioralFeature[i].concurrency

isEditable	editable	boolean	BaseElement	\$BehavioralFeature[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$BehavioralFeature[i].elementImport
Featuring Classifier	featuringClassifier	Classifier	Feature	\$BehavioralFeature[i].featuringClassifier
humanName	humanName	String	BaseElement	\$BehavioralFeature[i].humanName
humanType	humanType	String	BaseElement	\$BehavioralFeature[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$BehavioralFeature[i].importedMember
isInvalid	invalid	boolean	ModelObject	\$BehavioralFeature[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$BehavioralFeature[i].leaf
localID	localID	String	ModelObject	\$BehavioralFeature[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$BehavioralFeature[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$BehavioralFeature[i].member
Method	method	Behavior[]	BehavioralFeature	\$BehavioralFeature[i].method
Name	name	String	NamedElement	\$BehavioralFeature[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$BehavioralFeature[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$BehavioralFeature[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$BehavioralFeature[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$BehavioralFeature[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$BehavioralFeature[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$BehavioralFeature[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$BehavioralFeature[i].ownedMember
Owned Parameter	ownedParameter	Parameter[]	BehavioralFeature	\$BehavioralFeature[i].ownedParameter
Owned Parameter Set	ownedParameterSet	ParameterSet[]	BehavioralFeature	\$BehavioralFeature[i].ownedParameterSet
Owned Rule	ownedRule	Constraint[]	Namespace	\$BehavioralFeature[i].ownedRule
Owner	owner	Element	Element	\$BehavioralFeature[i].owner
Package Import	packageImport	PackageImport[]	Namespace	\$BehavioralFeature[i].packageImport
Qualified Name	qualifiedName	String	NamedElement	\$BehavioralFeature[i].qualifiedName
Raised Exception	raisedException	Type[]	BehavioralFeature	\$BehavioralFeature[i].raisedException
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$BehavioralFeature[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$BehavioralFeature[i].redefinitionContext
Is Static	static	boolean	Feature	\$BehavioralFeature[i].static
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$BehavioralFeature[i].supplierDependency
Sync Element	syncElement	Element	Element	\$BehavioralFeature[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$BehavioralFeature[i].visibility

23. BehavoredClassifier

Generalization

- [Classifier](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MObject	\$BehavioredClassifier[i].ID
Class	UMLClass	Class	Classifier	\$BehavioredClassifier[i].UMLClass
Is Abstract	abstract	boolean	Classifier	\$BehavioredClassifier[i].abstract
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$BehavioredClassifier[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$BehavioredClassifier[i].attribute
classType	classType	Class	BaseElement	\$BehavioredClassifier[i].classType
Classifier Behavior	classifierBehavior	Behavior	BehavioredClassifier	\$BehavioredClassifier[i].classifierBehavior
Client Dependency	clientDependency	Dependency[]	NamedElement	\$BehavioredClassifier[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$BehavioredClassifier[i].collaborationUse
isEditable	editable	boolean	BaseElement	\$BehavioredClassifier[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$BehavioredClassifier[i].elementImport
Feature	feature	Feature[]	Classifier	\$BehavioredClassifier[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$BehavioredClassifier[i].finalSpecialization
General	general	Classifier[]	Classifier	\$BehavioredClassifier[i].general
Generalization	generalization	Generalization[]	Classifier	\$BehavioredClassifier[i].generalization
humanName	humanName	String	BaseElement	\$BehavioredClassifier[i].humanName
humanType	humanType	String	BaseElement	\$BehavioredClassifier[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$BehavioredClassifier[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$BehavioredClassifier[i].inheritedMember
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavioredClassifier	\$BehavioredClassifier[i].interfaceRealization
isInvalid	invalid	boolean	ModelObject	\$BehavioredClassifier[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$BehavioredClassifier[i].leaf
localID	localID	String	ModelObject	\$BehavioredClassifier[i].localID
mdExtensions	mdExtensions	MDExtension[]	MObject	\$BehavioredClassifier[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$BehavioredClassifier[i].member
Name	name	String	NamedElement	\$BehavioredClassifier[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$BehavioredClassifier[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$BehavioredClassifier[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$BehavioredClassifier[i].objectParent
Owned Behavior	ownedBehavior	Behavior[]	BehavioredClassifier	\$BehavioredClassifier[i].ownedBehavior
Owned Comment	ownedComment	Comment[]	Element	\$BehavioredClassifier[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$BehavioredClassifier[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$BehavioredClassifier[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$BehavioredClassifier[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$BehavioredClassifier[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$BehavioredClassifier[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$BehavioredClassifier[i].ownedUseCase

Owner	owner	Element	Element	<code>\$BehavioredClassifier[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$BehavioredClassifier[i].owningPackage</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$BehavioredClassifier[i].owningTemplateParameter</code>
Package	package	Package	Type	<code>\$BehavioredClassifier[i].package</code>
Package Import	packageImport	PackageImport[]	Namespace	<code>\$BehavioredClassifier[i].packageImport</code>
PowerType Extent	powerTypeExtent	GeneralizationSet[]	Classifier	<code>\$BehavioredClassifier[i].powerTypeExtent</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$BehavioredClassifier[i].qualifiedName</code>
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	<code>\$BehavioredClassifier[i].redefinedClassifier</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$BehavioredClassifier[i].redefinedElement</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$BehavioredClassifier[i].redefinitionContext</code>
Representation	representation	CollaborationUse	Classifier	<code>\$BehavioredClassifier[i].representation</code>
Substitution	substitution	Substitution[]	Classifier	<code>\$BehavioredClassifier[i].substitution</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$BehavioredClassifier[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$BehavioredClassifier[i].syncElement</code>
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	<code>\$BehavioredClassifier[i].templateBinding</code>
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	<code>\$BehavioredClassifier[i].templateParameter</code>
Use Case	useCase	UseCase[]	Classifier	<code>\$BehavioredClassifier[i].useCase</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$BehavioredClassifier[i].visibility</code>

24. BehaviorExecutionSpecification

Generalization

- [ExecutionSpecification](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$BehaviorExecutionSpecification[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$BehaviorExecutionSpecification[i].appliedStereotypeInstance</code>
Behavior	behavior	Behavior	BehaviorExecutionSpecification	<code>\$BehaviorExecutionSpecification[i].behavior</code>
classType	classType	Class	BaseElement	<code>\$BehaviorExecutionSpecification[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$BehaviorExecutionSpecification[i].clientDependency</code>
Covered	covered	Lifeline[]	InteractionFragment	<code>\$BehaviorExecutionSpecification[i].covered</code>

isEditable	editable	boolean	BaseElement	\$BehaviorExecutionSpecification[i].editable
Enclosing Interaction	enclosingInteraction	Interaction	InteractionFragment	\$BehaviorExecutionSpecification[i].enclosingInteraction
Enclosing Operand	enclosingOperand	InteractionOperand	InteractionFragment	\$BehaviorExecutionSpecification[i].enclosingOperand
Finish	finish	OccurrenceSpecification	ExecutionSpecification	\$BehaviorExecutionSpecification[i].finish
General Ordering	generalOrdering	GeneralOrdering[]	InteractionFragment	\$BehaviorExecutionSpecification[i].generalOrdering
humanName	humanName	String	BaseElement	\$BehaviorExecutionSpecification[i].humanName
humanType	humanType	String	BaseElement	\$BehaviorExecutionSpecification[i].humanType
isInvalid	invalid	boolean	ModelObject	\$BehaviorExecutionSpecification[i].invalid
localID	localID	String	ModelObject	\$BehaviorExecutionSpecification[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$BehaviorExecutionSpecification[i].mdExtensions
Name	name	String	NamedElement	\$BehaviorExecutionSpecification[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$BehaviorExecutionSpecification[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$BehaviorExecutionSpecification[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$BehaviorExecutionSpecification[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$BehaviorExecutionSpecification[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$BehaviorExecutionSpecification[i].ownedElement
Owner	owner	Element	Element	\$BehaviorExecutionSpecification[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$BehaviorExecutionSpecification[i].qualifiedName
Start	start	OccurrenceSpecification	ExecutionSpecification	\$BehaviorExecutionSpecification[i].start
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$BehaviorExecutionSpecification[i].supplierDependency
Sync Element	syncElement	Element	Element	\$BehaviorExecutionSpecification[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$BehaviorExecutionSpecification[i].visibility

25. BroadcastSignalAction

Generalization

- InvocationAction

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$BroadcastSignalAction[i].ID
Activity	activity	Activity	ActivityNode	\$BroadcastSignalAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$BroadcastSignalAction[i].appliedStereotypeInstance
Argument	argument	InputPin[]	InvocationAction	\$BroadcastSignalAction[i].argument
classType	classType	Class	BaseElement	\$BroadcastSignalAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$BroadcastSignalAction[i].clientDependency
Context	context	Classifier	Action	\$BroadcastSignalAction[i].context
isEditable	editable	boolean	BaseElement	\$BroadcastSignalAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$BroadcastSignalAction[i].handler
humanName	humanName	String	BaseElement	\$BroadcastSignalAction[i].humanName
humanType	humanType	String	BaseElement	\$BroadcastSignalAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$BroadcastSignalAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$BroadcastSignalAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$BroadcastSignalAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$BroadcastSignalAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$BroadcastSignalAction[i].incoming
Input	input	InputPin[]	Action	\$BroadcastSignalAction[i].input
isInvalid	invalid	boolean	ModelObject	\$BroadcastSignalAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$BroadcastSignalAction[i].leaf
localID	localID	String	ModelObject	\$BroadcastSignalAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$BroadcastSignalAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$BroadcastSignalAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$BroadcastSignalAction[i].locallyReentrant
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	\$BroadcastSignalAction[i].mdExtensions
Name	name	String	NamedElement	\$BroadcastSignalAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$BroadcastSignalAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$BroadcastSignalAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$BroadcastSignalAction[i].objectParent
On Port	onPort	Port	InvocationAction	\$BroadcastSignalAction[i].onPort
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$BroadcastSignalAction[i].outgoing
Output	output	OutputPin[]	Action	\$BroadcastSignalAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$BroadcastSignalAction[i].ownedComment

Owned Element	ownedElement	Element[]	Element	\$BroadcastSignalAction[i].ownedElement
Owner	owner	Element	Element	\$BroadcastSignalAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$BroadcastSignalAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$BroadcastSignalAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$BroadcastSignalAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$BroadcastSignalAction[i].redefinitionContext
Signal	signal	Signal	BroadcastSignalAction	\$BroadcastSignalAction[i].signal
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$BroadcastSignalAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$BroadcastSignalAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$BroadcastSignalAction[i].visibility

26. CallAction

Generalization

- [InvocationAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$CallAction[i].ID
Activity	activity	Activity	ActivityNode	\$CallAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$CallAction[i].appliedStereotypeInstance
Argument	argument	InputPin[]	InvocationAction	\$CallAction[i].argument
classType	classType	Class	BaseElement	\$CallAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$CallAction[i].clientDependency
Context	context	Classifier	Action	\$CallAction[i].context
isEditable	editable	boolean	BaseElement	\$CallAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$CallAction[i].handler
humanName	humanName	String	BaseElement	\$CallAction[i].humanName
humanType	humanType	String	BaseElement	\$CallAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$CallAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$CallAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$CallAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$CallAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$CallAction[i].incoming
Input	input	InputPin[]	Action	\$CallAction[i].input
isInvalid	invalid	boolean	ModelObject	\$CallAction[i].invalid

Is Leaf	leaf	boolean	RedefinableElement	<code>\$CallAction[i].leaf</code>
localID	localID	String	ModelObject	<code>\$CallAction[i].localID</code>
Local Postcondition	localPostcondition	Constraint[]	Action	<code>\$CallAction[i].localPostcondition</code>
Local Precondition	localPrecondition	Constraint[]	Action	<code>\$CallAction[i].localPrecondition</code>
Is Locally Reentrant	locallyReentrant	boolean	Action	<code>\$CallAction[i].locallyReentrant</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$CallAction[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$CallAction[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$CallAction[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$CallAction[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$CallAction[i].objectParent</code>
On Port	onPort	Port	InvocationAction	<code>\$CallAction[i].onPort</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$CallAction[i].outgoing</code>
Output	output	OutputPin[]	Action	<code>\$CallAction[i].output</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$CallAction[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$CallAction[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$CallAction[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$CallAction[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$CallAction[i].redefinedElement</code>
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	<code>\$CallAction[i].redefinedNode</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$CallAction[i].redefinitionContext</code>
Result	result	OutputPin[]	CallAction	<code>\$CallAction[i].result</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$CallAction[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$CallAction[i].syncElement</code>
Is Synchronous	synchronous	boolean	CallAction	<code>\$CallAction[i].synchronous</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$CallAction[i].visibility</code>

27. CallBehaviorAction

Generalization

- [CallAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$CallBehaviorAction[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$CallBehaviorAction[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$CallBehaviorAction[i].appliedStereotypeInstance</code>
Argument	argument	InputPin[]	InvocationAction	<code>\$CallBehaviorAction[i].argument</code>
Behavior	behavior	Behavior	CallBehaviorAction	<code>\$CallBehaviorAction[i].behavior</code>

classType	classType	Class	BaseElement	\$CallBehaviorAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$CallBehaviorAction[i].clientDependency
Context	context	Classifier	Action	\$CallBehaviorAction[i].context
isEditable	editable	boolean	BaseElement	\$CallBehaviorAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$CallBehaviorAction[i].handler
humanName	humanName	String	BaseElement	\$CallBehaviorAction[i].humanName
humanType	humanType	String	BaseElement	\$CallBehaviorAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$CallBehaviorAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$CallBehaviorAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$CallBehaviorAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$CallBehaviorAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$CallBehaviorAction[i].incoming
Input	input	InputPin[]	Action	\$CallBehaviorAction[i].input
isInvalid	invalid	boolean	ModelObject	\$CallBehaviorAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$CallBehaviorAction[i].leaf
localID	localID	String	ModelObject	\$CallBehaviorAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$CallBehaviorAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$CallBehaviorAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$CallBehaviorAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$CallBehaviorAction[i].mdExtensions
Name	name	String	NamedElement	\$CallBehaviorAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$CallBehaviorAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$CallBehaviorAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$CallBehaviorAction[i].objectParent
On Port	onPort	Port	InvocationAction	\$CallBehaviorAction[i].onPort
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$CallBehaviorAction[i].outgoing
Output	output	OutputPin[]	Action	\$CallBehaviorAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$CallBehaviorAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$CallBehaviorAction[i].ownedElement
Owner	owner	Element	Element	\$CallBehaviorAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$CallBehaviorAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$CallBehaviorAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$CallBehaviorAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$CallBehaviorAction[i].redefinitionContext
Result	result	OutputPin[]	CallAction	\$CallBehaviorAction[i].result
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$CallBehaviorAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$CallBehaviorAction[i].syncElement
Is Synchronous	synchronous	boolean	CallAction	\$CallBehaviorAction[i].synchronous
Visibility	visibility	VisibilityKind	NamedElement	\$CallBehaviorAction[i].visibility

28. CallEvent

Generalization

- [MessageEvent](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$CallEvent[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$CallEvent[i].appliedStereotypeInstance</code>
Behavior	behavior	Behavior	Event	<code>\$CallEvent[i].behavior</code>
classType	classType	Class	BaseElement	<code>\$CallEvent[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$CallEvent[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$CallEvent[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$CallEvent[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$CallEvent[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$CallEvent[i].invalid</code>
localID	localID	String	ModelObject	<code>\$CallEvent[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$CallEvent[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$CallEvent[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$CallEvent[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$CallEvent[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$CallEvent[i].objectParent</code>
Operation	operation	Operation	CallEvent	<code>\$CallEvent[i].operation</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$CallEvent[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$CallEvent[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$CallEvent[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$CallEvent[i].owningPackage</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$CallEvent[i].owningTemplateParameter</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$CallEvent[i].qualifiedName</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$CallEvent[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$CallEvent[i].syncElement</code>
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$CallEvent[i].templateParameter</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$CallEvent[i].visibility</code>

29. CallOperationAction

Generalization

- [CallAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$CallOperationAction[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$CallOperationAction[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$CallOperationAction[i].appliedStereotypeInstance</code>
Argument	argument	InputPin[]	InvocationAction	<code>\$CallOperationAction[i].argument</code>
classType	classType	Class	BaseElement	<code>\$CallOperationAction[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$CallOperationAction[i].clientDependency</code>
Context	context	Classifier	Action	<code>\$CallOperationAction[i].context</code>
isEditable	editable	boolean	BaseElement	<code>\$CallOperationAction[i].editable</code>
Handler	handler	ExceptionHandler[]	ExecutableNode	<code>\$CallOperationAction[i].handler</code>
humanName	humanName	String	BaseElement	<code>\$CallOperationAction[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$CallOperationAction[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$CallOperationAction[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$CallOperationAction[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$CallOperationAction[i].inPartition</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$CallOperationAction[i].inStructuredNode</code>
Incoming	incoming	ActivityEdge[]	ActivityNode	<code>\$CallOperationAction[i].incoming</code>
Input	input	InputPin[]	Action	<code>\$CallOperationAction[i].input</code>
isInvalid	invalid	boolean	ModelObject	<code>\$CallOperationAction[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$CallOperationAction[i].leaf</code>
localID	localID	String	ModelObject	<code>\$CallOperationAction[i].localID</code>
Local Postcondition	localPostcondition	Constraint[]	Action	<code>\$CallOperationAction[i].localPostcondition</code>
Local Precondition	localPrecondition	Constraint[]	Action	<code>\$CallOperationAction[i].localPrecondition</code>
Is Locally Reentrant	locallyReentrant	boolean	Action	<code>\$CallOperationAction[i].locallyReentrant</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$CallOperationAction[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$CallOperationAction[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$CallOperationAction[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$CallOperationAction[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$CallOperationAction[i].objectParent</code>
On Port	onPort	Port	InvocationAction	<code>\$CallOperationAction[i].onPort</code>
Operation	operation	Operation	CallOperationAction	<code>\$CallOperationAction[i].operation</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$CallOperationAction[i].outgoing</code>
Output	output	OutputPin[]	Action	<code>\$CallOperationAction[i].output</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$CallOperationAction[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$CallOperationAction[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$CallOperationAction[i].owner</code>

Qualified Name	qualifiedName	String	NamedElement	\$CallOperationAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$CallOperationAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$CallOperationAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$CallOperationAction[i].redefinitionContext
Result	result	OutputPin[]	CallAction	\$CallOperationAction[i].result
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$CallOperationAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$CallOperationAction[i].syncElement
Is Synchronous	synchronous	boolean	CallAction	\$CallOperationAction[i].synchronous
Target	target	InputPin	CallOperationAction	\$CallOperationAction[i].target
Visibility	visibility	VisibilityKind	NamedElement	\$CallOperationAction[i].visibility

30. CentralBufferNode

Generalization

- ObjectNode

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$CentralBufferNode[i].ID
Activity	activity	Activity	ActivityNode	\$CentralBufferNode[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$CentralBufferNode[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$CentralBufferNode[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$CentralBufferNode[i].clientDependency
Is Control Type	controlType	boolean	ObjectNode	\$CentralBufferNode[i].controlType
isEditable	editable	boolean	BaseElement	\$CentralBufferNode[i].editable
humanName	humanName	String	BaseElement	\$CentralBufferNode[i].humanName
humanType	humanType	String	BaseElement	\$CentralBufferNode[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$CentralBufferNode[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$CentralBufferNode[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$CentralBufferNode[i].inPartition
In State	inState	State[]	ObjectNode	\$CentralBufferNode[i].inState
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$CentralBufferNode[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$CentralBufferNode[i].incoming
isInvalid	invalid	boolean	ModelObject	\$CentralBufferNode[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$CentralBufferNode[i].leaf
localID	localID	String	ModelObject	\$CentralBufferNode[i].localID

mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$CentralBufferNode[i].mdExtensions
Name	name	String	NamedElement	\$CentralBufferNode[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$CentralBufferNode[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$CentralBufferNode[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$CentralBufferNode[i].objectParent
Ordering	ordering	ObjectNodeOrderingKind	ObjectNode	\$CentralBufferNode[i].ordering
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$CentralBufferNode[i].outgoing
Owned Comment	ownedComment	Comment[]	Element	\$CentralBufferNode[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$CentralBufferNode[i].ownedElement
Owner	owner	Element	Element	\$CentralBufferNode[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$CentralBufferNode[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$CentralBufferNode[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$CentralBufferNode[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$CentralBufferNode[i].redefinitionContext
Selection	selection	Behavior	ObjectNode	\$CentralBufferNode[i].selection
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$CentralBufferNode[i].supplierDependency
Sync Element	syncElement	Element	Element	\$CentralBufferNode[i].syncElement
Type	type	Type	TypedElement	\$CentralBufferNode[i].type
Upper Bound	upperBound	ValueSpecification	ObjectNode	\$CentralBufferNode[i].upperBound
Visibility	visibility	VisibilityKind	NamedElement	\$CentralBufferNode[i].visibility

31. ChangeEvent

Generalization

- [Event](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ChangeEvent[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ChangeEvent[i].appliedStereotypeInstance
Behavior	behavior	Behavior	Event	\$ChangeEvent[i].behavior
Change Expression	changeExpression	ValueSpecification	ChangeEvent	\$ChangeEvent[i].changeExpression
classType	classType	Class	BaseElement	\$ChangeEvent[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ChangeEvent[i].clientDependency
isEditable	editable	boolean	BaseElement	\$ChangeEvent[i].editable
humanName	humanName	String	BaseElement	\$ChangeEvent[i].humanName
humanType	humanType	String	BaseElement	\$ChangeEvent[i].humanType
isInvalid	invalid	boolean	ModelObject	\$ChangeEvent[i].invalid
localID	localID	String	ModelObject	\$ChangeEvent[i].localID

mdExtensions	mdExtensions	MDEExtension[]	MDOObject	<code>\$ChangeEvent[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$ChangeEvent[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$ChangeEvent[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$ChangeEvent[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$ChangeEvent[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$ChangeEvent[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$ChangeEvent[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$ChangeEvent[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$ChangeEvent[i].owningPackage</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$ChangeEvent[i].owningTemplateParameter</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$ChangeEvent[i].qualifiedName</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$ChangeEvent[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$ChangeEvent[i].syncElement</code>
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$ChangeEvent[i].templateParameter</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$ChangeEvent[i].visibility</code>

32. Class

Generalization

- [BehavoredClassifier](#)
- [EncapsulatedClassifier](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$Class[i].ID</code>
Class	UMLClass	Class	Classifier	<code>\$Class[i].UMLClass</code>
Is Abstract	abstract	boolean	Class	<code>\$Class[i].abstract</code>
Is Active	active	boolean	Class	<code>\$Class[i].active</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Class[i].appliedStereotypeInstance</code>
Attribute	attribute	Property[]	Classifier	<code>\$Class[i].attribute</code>
classType	classType	Class	BaseElement	<code>\$Class[i].classType</code>
Classifier Behavior	classifierBehavior	Behavior	BehavoredClassifier	<code>\$Class[i].classifierBehavior</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Class[i].clientDependency</code>
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	<code>\$Class[i].collaborationUse</code>
isEditable	editable	boolean	BaseElement	<code>\$Class[i].editable</code>
Element Import	elementImport	ElementImport[]	Namespace	<code>\$Class[i].elementImport</code>
Extension	extension	Extension[]	Class	<code>\$Class[i].extension</code>
Feature	feature	Feature[]	Classifier	<code>\$Class[i].feature</code>

Is Final Specialization	finalSpecialization	boolean	Classifier	\$Class[i].finalSpecialization
General	general	Classifier[]	Classifier	\$Class[i].general
Generalization	generalization	Generalization[]	Classifier	\$Class[i].generalization
humanName	humanName	String	BaseElement	\$Class[i].humanName
humanType	humanType	String	BaseElement	\$Class[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Class[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$Class[i].inheritedMember
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavioredClassifier	\$Class[i].interfaceRealization
isInvalid	invalid	boolean	ModelObject	\$Class[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$Class[i].leaf
localID	localID	String	ModelObject	\$Class[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Class[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Class[i].member
Name	name	String	NamedElement	\$Class[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Class[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Class[i].namespace
Nested Classifier	nestedClassifier	Classifier[]	Class	\$Class[i].nestedClassifier
objectParent	objectParent	BaseElement	BaseElement	\$Class[i].objectParent
Owned Attribute	ownedAttribute	Property[]	Class	\$Class[i].ownedAttribute
Owned Behavior	ownedBehavior	Behavior[]	BehavioredClassifier	\$Class[i].ownedBehavior
Owned Comment	ownedComment	Comment[]	Element	\$Class[i].ownedComment
Owned Connector	ownedConnector	Connector[]	StructuredClassifier	\$Class[i].ownedConnector
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Class[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Class[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$Class[i].ownedMember
Owned Operation	ownedOperation	Operation[]	Class	\$Class[i].ownedOperation
Owned Port	ownedPort	Port[]	EncapsulatedClassifier	\$Class[i].ownedPort
Owned Reception	ownedReception	Reception[]	Class	\$Class[i].ownedReception
Owned Rule	ownedRule	Constraint[]	Namespace	\$Class[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$Class[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$Class[i].ownedUseCase
Owner	owner	Element	Element	\$Class[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Class[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Class[i].owningTemplateParameter
Package	package	Package	Type	\$Class[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$Class[i].packageImport
Part	part	Property[]	StructuredClassifier	\$Class[i].part
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$Class[i].powertypeExtent
Qualified Name	qualifiedName	String	NamedElement	\$Class[i].qualifiedName

Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	<code>\$Class[i].redefinedClassifier</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$Class[i].redefinedElement</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$Class[i].redefinitionContext</code>
Representation	representation	CollaborationUse	Classifier	<code>\$Class[i].representation</code>
Role	role	ConnectableElement[]	StructuredClassifier	<code>\$Class[i].role</code>
Substitution	substitution	Substitution[]	Classifier	<code>\$Class[i].substitution</code>
Super Class	superClass	Class[]	Class	<code>\$Class[i].superClass</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Class[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Class[i].syncElement</code>
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	<code>\$Class[i].templateBinding</code>
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	<code>\$Class[i].templateParameter</code>
Use Case	useCase	UseCase[]	Classifier	<code>\$Class[i].useCase</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$Class[i].visibility</code>

33. Classifier

Generalization

- [Namespace](#)
- [RedefinableElement](#)
- [TemplateableElement](#)
- [Type](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$Classifier[i].ID</code>
Class	UMLClass	Class	Classifier	<code>\$Classifier[i].UMLClass</code>
Is Abstract	abstract	boolean	Classifier	<code>\$Classifier[i].abstract</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Classifier[i].appliedStereotypeInstance</code>
Attribute	attribute	Property[]	Classifier	<code>\$Classifier[i].attribute</code>
classType	classType	Class	BaseElement	<code>\$Classifier[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Classifier[i].clientDependency</code>
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	<code>\$Classifier[i].collaborationUse</code>
isEditable	editable	boolean	BaseElement	<code>\$Classifier[i].editable</code>
Element Import	elementImport	ElementImport[]	Namespace	<code>\$Classifier[i].elementImport</code>
Feature	feature	Feature[]	Classifier	<code>\$Classifier[i].feature</code>
Is Final Specialization	finalSpecialization	boolean	Classifier	<code>\$Classifier[i].finalSpecialization</code>
General	general	Classifier[]	Classifier	<code>\$Classifier[i].general</code>
Generalization	generalization	Generalization[]	Classifier	<code>\$Classifier[i].generalization</code>

humanName	humanName	String	BaseElement	\$Classifier[i].humanName
humanType	humanType	String	BaseElement	\$Classifier[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Classifier[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$Classifier[i].inheritedMember
isInvalid	invalid	boolean	ModelObject	\$Classifier[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$Classifier[i].leaf
localID	localID	String	ModelObject	\$Classifier[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Classifier[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Classifier[i].member
Name	name	String	NamedElement	\$Classifier[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Classifier[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Classifier[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Classifier[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Classifier[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Classifier[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Classifier[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$Classifier[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$Classifier[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$Classifier[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$Classifier[i].ownedUseCase
Owner	owner	Element	Element	\$Classifier[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Classifier[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Classifier[i].owningTemplateParameter
Package	package	Package	Type	\$Classifier[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$Classifier[i].packageImport
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$Classifier[i].powertypeExtent
Qualified Name	qualifiedName	String	NamedElement	\$Classifier[i].qualifiedName
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$Classifier[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Classifier[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$Classifier[i].redefinitionContext
Representation	representation	CollaborationUse	Classifier	\$Classifier[i].representation
Substitution	substitution	Substitution[]	Classifier	\$Classifier[i].substitution
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Classifier[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Classifier[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$Classifier[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$Classifier[i].templateParameter
Use Case	useCase	UseCase[]	Classifier	\$Classifier[i].useCase
Visibility	visibility	VisibilityKind	PackageableElement	\$Classifier[i].visibility

34. ClassifierTemplateParameter

Generalization

- [TemplateParameter](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$ClassifierTemplateParameter[i].ID</code>
isAllowSubstitutable	allowSubstitutable	boolean	ClassifierTemplateParameter	<code>\$ClassifierTemplateParameter[i].allowSubstitutable</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$ClassifierTemplateParameter[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$ClassifierTemplateParameter[i].classType</code>
Constraining Classifier	constrainingClassifier	Classifier[]	ClassifierTemplateParameter	<code>\$ClassifierTemplateParameter[i].constrainingClassifier</code>
Default	default	ParameterableElement	TemplateParameter	<code>\$ClassifierTemplateParameter[i].default</code>
isEditable	editable	boolean	BaseElement	<code>\$ClassifierTemplateParameter[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$ClassifierTemplateParameter[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$ClassifierTemplateParameter[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$ClassifierTemplateParameter[i].invalid</code>
localID	localID	String	ModelObject	<code>\$ClassifierTemplateParameter[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$ClassifierTemplateParameter[i].mdExtensions</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$ClassifierTemplateParameter[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$ClassifierTemplateParameter[i].ownedComment</code>
Owned Default	ownedDefault	ParameterableElement	TemplateParameter	<code>\$ClassifierTemplateParameter[i].ownedDefault</code>
Owned Element	ownedElement	Element[]	Element	<code>\$ClassifierTemplateParameter[i].ownedElement</code>
Owned Parametered Element	ownedParameteredElement	ParameterableElement	TemplateParameter	<code>\$ClassifierTemplateParameter[i].ownedParameteredElement</code>
Owner	owner	Element	Element	<code>\$ClassifierTemplateParameter[i].owner</code>
Parametered Element	parameteredElement	Classifier	ClassifierTemplateParameter	<code>\$ClassifierTemplateParameter[i].parameteredElement</code>
Signature	signature	TemplateSignature	TemplateParameter	<code>\$ClassifierTemplateParameter[i].signature</code>
Sync Element	syncElement	Element	Element	<code>\$ClassifierTemplateParameter[i].syncElement</code>

				nt

35. Clause

Generalization

- [Element](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$Clause[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Clause[i].appliedStereotypeInstance
Body	body	ExecutableNode[]	Clause	\$Clause[i].body
Body Output	bodyOutput	OutputPin[]	Clause	\$Clause[i].bodyOutput
classType	classType	Class	BaseElement	\$Clause[i].classType
Decider	decider	OutputPin	Clause	\$Clause[i].decider
isEditable	editable	boolean	BaseElement	\$Clause[i].editable
humanName	humanName	String	BaseElement	\$Clause[i].humanName
humanType	humanType	String	BaseElement	\$Clause[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Clause[i].invalid
localID	localID	String	ModelObject	\$Clause[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$Clause[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$Clause[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Clause[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Clause[i].ownedElement
Owner	owner	Element	Element	\$Clause[i].owner
Predecessor Clause	predecessorClause	Clause[]	Clause	\$Clause[i].predecessorClause
Successor Clause	successorClause	Clause[]	Clause	\$Clause[i].successorClause
Sync Element	syncElement	Element	Element	\$Clause[i].syncElement
Test	test	ExecutableNode[]	Clause	\$Clause[i].test

36. ClearAssociationAction

Generalization

- [Action](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$ClearAssociationAction[i].ID

Activity	activity	Activity	ActivityNode	\$ClearAssociationAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ClearAssociationAction[i].appliedStereotypeInstance
Association	association	Association	ClearAssociationAction	\$ClearAssociationAction[i].association
classType	classType	Class	BaseElement	\$ClearAssociationAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ClearAssociationAction[i].clientDependency
Context	context	Classifier	Action	\$ClearAssociationAction[i].context
isEditable	editable	boolean	BaseElement	\$ClearAssociationAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$ClearAssociationAction[i].handler
humanName	humanName	String	BaseElement	\$ClearAssociationAction[i].humanName
humanType	humanType	String	BaseElement	\$ClearAssociationAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ClearAssociationAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ClearAssociationAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ClearAssociationAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ClearAssociationAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ClearAssociationAction[i].incoming
Input	input	InputPin[]	Action	\$ClearAssociationAction[i].input
isInvalid	invalid	boolean	ModelObject	\$ClearAssociationAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ClearAssociationAction[i].leaf
localID	localID	String	ModelObject	\$ClearAssociationAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$ClearAssociationAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$ClearAssociationAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$ClearAssociationAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$ClearAssociationAction[i].mdExtensions
Name	name	String	NamedElement	\$ClearAssociationAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ClearAssociationAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ClearAssociationAction[i].namespace
Object	object	InputPin	ClearAssociationAction	\$ClearAssociationAction[i].object
objectParent	objectParent	BaseElement	BaseElement	\$ClearAssociationAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ClearAssociationAction[i].outgoing
Output	output	OutputPin[]	Action	\$ClearAssociationAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$ClearAssociationAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ClearAssociationAction[i].ownedElement
Owner	owner	Element	Element	\$ClearAssociationAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ClearAssociationAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ClearAssociationAction[i].redefinedElement

				t
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ClearAssociationAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ClearAssociationAction[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ClearAssociationAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ClearAssociationAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$ClearAssociationAction[i].visibility

37. ClearStructuralFeatureAction

Generalization

- [StructuralFeatureAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ClearStructuralFeatureAction[i].ID
Activity	activity	Activity	ActivityNode	\$ClearStructuralFeatureAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ClearStructuralFeatureAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ClearStructuralFeatureAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ClearStructuralFeatureAction[i].clientDependency
Context	context	Classifier	Action	\$ClearStructuralFeatureAction[i].context
isEditable	editable	boolean	BaseElement	\$ClearStructuralFeatureAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$ClearStructuralFeatureAction[i].handler
humanName	humanName	String	BaseElement	\$ClearStructuralFeatureAction[i].humanName
humanType	humanType	String	BaseElement	\$ClearStructuralFeatureAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ClearStructuralFeatureAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ClearStructuralFeatureAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ClearStructuralFeatureAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ClearStructuralFeatureAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ClearStructuralFeatureAction[i].incoming
Input	input	InputPin[]	Action	\$ClearStructuralFeatureAction[i].input
isInvalid	invalid	boolean	ModelObject	\$ClearStructuralFeatureAction[i].invalid

Is Leaf	leaf	boolean	RedefinableElement	\$ClearStructuralFeatureAction[i].leaf
localID	localID	String	ModelObject	\$ClearStructuralFeatureAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$ClearStructuralFeatureAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$ClearStructuralFeatureAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$ClearStructuralFeatureAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ClearStructuralFeatureAction[i].mdExtensions
Name	name	String	NamedElement	\$ClearStructuralFeatureAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ClearStructuralFeatureAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ClearStructuralFeatureAction[i].namespace
Object	object	InputPin	StructuralFeatureAction	\$ClearStructuralFeatureAction[i].object
objectParent	objectParent	BaseElement	BaseElement	\$ClearStructuralFeatureAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ClearStructuralFeatureAction[i].outgoing
Output	output	OutputPin[]	Action	\$ClearStructuralFeatureAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$ClearStructuralFeatureAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ClearStructuralFeatureAction[i].ownedElement
Owner	owner	Element	Element	\$ClearStructuralFeatureAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ClearStructuralFeatureAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ClearStructuralFeatureAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ClearStructuralFeatureAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ClearStructuralFeatureAction[i].redefinitionContext
Result	result	OutputPin	ClearStructuralFeatureAction	\$ClearStructuralFeatureAction[i].result
Structural Feature	structuralFeature	StructuralFeature	StructuralFeatureAction	\$ClearStructuralFeatureAction[i].structuralFeature
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ClearStructuralFeatureAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ClearStructuralFeatureAction[i].syncElement

Visibility	visibility	VisibilityKind	NamedElement	\$ClearStructuralFeatureAction[i].visibility

38. ClearVariableAction

Generalization

- [VariableAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ClearVariableAction[i].ID
Activity	activity	Activity	ActivityNode	\$ClearVariableAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ClearVariableAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ClearVariableAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ClearVariableAction[i].clientDependency
Context	context	Classifier	Action	\$ClearVariableAction[i].context
isEditable	editable	boolean	BaseElement	\$ClearVariableAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$ClearVariableAction[i].handler
humanName	humanName	String	BaseElement	\$ClearVariableAction[i].humanName
humanType	humanType	String	BaseElement	\$ClearVariableAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ClearVariableAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ClearVariableAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ClearVariableAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ClearVariableAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ClearVariableAction[i].incoming
Input	input	InputPin[]	Action	\$ClearVariableAction[i].input
isInvalid	invalid	boolean	ModelObject	\$ClearVariableAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ClearVariableAction[i].leaf
localID	localID	String	ModelObject	\$ClearVariableAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$ClearVariableAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$ClearVariableAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$ClearVariableAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ClearVariableAction[i].mdExtensions
Name	name	String	NamedElement	\$ClearVariableAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ClearVariableAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ClearVariableAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ClearVariableAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ClearVariableAction[i].outgoing

Output	output	OutputPin[]	Action	\$ClearVariableAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$ClearVariableAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ClearVariableAction[i].ownedElement
Owner	owner	Element	Element	\$ClearVariableAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ClearVariableAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ClearVariableAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ClearVariableAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ClearVariableAction[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ClearVariableAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ClearVariableAction[i].syncElement
Variable	variable	Variable	VariableAction	\$ClearVariableAction[i].variable
Visibility	visibility	VisibilityKind	NamedElement	\$ClearVariableAction[i].visibility

39. Collaboration

Generalization

- [BehavoredClassifier](#)
- [StructuredClassifier](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Collaboration[i].ID
Class	UMLClass	Class	Classifier	\$Collaboration[i].UMLClass
Is Abstract	abstract	boolean	Classifier	\$Collaboration[i].abstract
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Collaboration[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$Collaboration[i].attribute
classType	classType	Class	BaseElement	\$Collaboration[i].classType
Classifier Behavior	classifierBehavior	Behavior	BehavoredClassifier	\$Collaboration[i].classifierBehavior
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Collaboration[i].clientDependency
Collaboration Role	collaborationRole	ConnectableElement[]	Collaboration	\$Collaboration[i].collaborationRole
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$Collaboration[i].collaborationUse
isEditable	editable	boolean	BaseElement	\$Collaboration[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$Collaboration[i].elementImport
Feature	feature	Feature[]	Classifier	\$Collaboration[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$Collaboration[i].finalSpecialization
General	general	Classifier[]	Classifier	\$Collaboration[i].general
Generalization	generalization	Generalization[]	Classifier	\$Collaboration[i].generalization
humanName	humanName	String	BaseElement	\$Collaboration[i].humanName
humanType	humanType	String	BaseElement	\$Collaboration[i].humanType

Imported Member	importedMember	PackageableElement[]	Namespace	\$Collaboration[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$Collaboration[i].inheritedMember
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavioredClassifier	\$Collaboration[i].interfaceRealization
isInvalid	invalid	boolean	ModelObject	\$Collaboration[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$Collaboration[i].leaf
localID	localID	String	ModelObject	\$Collaboration[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Collaboration[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Collaboration[i].member
Name	name	String	NamedElement	\$Collaboration[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Collaboration[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Collaboration[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Collaboration[i].objectParent
Owned Attribute	ownedAttribute	Property[]	StructuredClassifier	\$Collaboration[i].ownedAttribute
Owned Behavior	ownedBehavior	Behavior[]	BehavioredClassifier	\$Collaboration[i].ownedBehavior
Owned Comment	ownedComment	Comment[]	Element	\$Collaboration[i].ownedComment
Owned Connector	ownedConnector	Connector[]	StructuredClassifier	\$Collaboration[i].ownedConnector
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Collaboration[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Collaboration[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$Collaboration[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$Collaboration[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$Collaboration[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$Collaboration[i].ownedUseCase
Owner	owner	Element	Element	\$Collaboration[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Collaboration[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Collaboration[i].owningTemplateParameter
Package	package	Package	Type	\$Collaboration[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$Collaboration[i].packageImport
Part	part	Property[]	StructuredClassifier	\$Collaboration[i].part
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$Collaboration[i].powertypeExtent
Qualified Name	qualifiedName	String	NamedElement	\$Collaboration[i].qualifiedName
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$Collaboration[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Collaboration[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$Collaboration[i].redefinitionContext
Representation	representation	CollaborationUse	Classifier	\$Collaboration[i].representation
Role	role	ConnectableElement[]	StructuredClassifier	\$Collaboration[i].role
Substitution	substitution	Substitution[]	Classifier	\$Collaboration[i].substitution
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Collaboration[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Collaboration[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$Collaboration[i].templateBinding

Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	<code>\$Collaboration[i].templateParameter</code>
Use Case	useCase	UseCase[]	Classifier	<code>\$Collaboration[i].useCase</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$Collaboration[i].visibility</code>

40. CollaborationUse

Generalization

- [NamedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$CollaborationUse[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$CollaborationUse[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$CollaborationUse[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$CollaborationUse[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$CollaborationUse[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$CollaborationUse[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$CollaborationUse[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$CollaborationUse[i].invalid</code>
localID	localID	String	ModelObject	<code>\$CollaborationUse[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$CollaborationUse[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$CollaborationUse[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$CollaborationUse[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$CollaborationUse[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$CollaborationUse[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$CollaborationUse[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$CollaborationUse[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$CollaborationUse[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$CollaborationUse[i].qualifiedName</code>
Role Binding	roleBinding	Dependency[]	CollaborationUse	<code>\$CollaborationUse[i].roleBinding</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$CollaborationUse[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$CollaborationUse[i].syncElement</code>
Type	type	Collaboration	CollaborationUse	<code>\$CollaborationUse[i].type</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$CollaborationUse[i].visibility</code>

41. com.nomagic.expressions.Operation

Generalization

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)

42. CombinedFragment

Generalization

- InteractionFragment

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$CombinedFragment[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$CombinedFragment[i].appliedStereotypeInstance
Cfragment Gate	cfragmentGate	Gate[]	CombinedFragment	\$CombinedFragment[i].cfragmentGate
classType	classType	Class	BaseElement	\$CombinedFragment[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$CombinedFragment[i].clientDependency
Covered	covered	Lifeline[]	InteractionFragment	\$CombinedFragment[i].covered
isEditable	editable	boolean	BaseElement	\$CombinedFragment[i].editable
Enclosing Interaction	enclosingInteraction	Interaction	InteractionFragment	\$CombinedFragment[i].enclosingInteraction
Enclosing Operand	enclosingOperand	InteractionOperand	InteractionFragment	\$CombinedFragment[i].enclosingOperand
General Ordering	generalOrdering	GeneralOrdering[]	InteractionFragment	\$CombinedFragment[i].generalOrdering
humanName	humanName	String	BaseElement	\$CombinedFragment[i].humanName
humanType	humanType	String	BaseElement	\$CombinedFragment[i].humanType
Interaction Operator	interactionOperator	InteractionOperatorKind	CombinedFragment	\$CombinedFragment[i].interactionOperator
isInvalid	invalid	boolean	ModelObject	\$CombinedFragment[i].invalid
localID	localID	String	ModelObject	\$CombinedFragment[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$CombinedFragment[i].mdExtensions
Name	name	String	NamedElement	\$CombinedFragment[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$CombinedFragment[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$CombinedFragment[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$CombinedFragment[i].objectParent
Operand	operand	InteractionOperand[]	CombinedFragment	\$CombinedFragment[i].operand
Owned Comment	ownedComment	Comment[]	Element	\$CombinedFragment[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$CombinedFragment[i].ownedElement

Owner	owner	Element	Element	<code>\$CombinedFragment[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$CombinedFragment[i].qualifiedName</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$CombinedFragment[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$CombinedFragment[i].syncElement</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$CombinedFragment[i].visibility</code>

43. Comment

Generalization

- [Element](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$Comment[i].ID</code>
Annotated Element	annotatedElement	Element[]	Comment	<code>\$Comment[i].annotatedElement</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Comment[i].appliedStereotypeInstance</code>
Body	body	String	Comment	<code>\$Comment[i].body</code>
classType	classType	Class	BaseElement	<code>\$Comment[i].classType</code>
isEditable	editable	boolean	BaseElement	<code>\$Comment[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$Comment[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Comment[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Comment[i].invalid</code>
localID	localID	String	ModelObject	<code>\$Comment[i].localID</code>
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	<code>\$Comment[i].mdExtensions</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Comment[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Comment[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Comment[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$Comment[i].owner</code>
Owning Element	owningElement	Element	Comment	<code>\$Comment[i].owningElement</code>
Sync Element	syncElement	Element	Element	<code>\$Comment[i].syncElement</code>

44. CommunicationPath

Generalization

- [Association](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)

ID	ID	String	MDOObject	\$CommunicationPath[i].ID
Class	UMLClass	Class	Classifier	\$CommunicationPath[i].UMLClass
Is Abstract	abstract	boolean	Classifier	\$CommunicationPath[i].abstract
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$CommunicationPath[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$CommunicationPath[i].attribute
classType	classType	Class	BaseElement	\$CommunicationPath[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$CommunicationPath[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$CommunicationPath[i].collaborationUse
Is Derived	derived	boolean	Association	\$CommunicationPath[i].derived
isEditable	editable	boolean	BaseElement	\$CommunicationPath[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$CommunicationPath[i].elementImport
End Type	endType	Type[]	Association	\$CommunicationPath[i].endType
Feature	feature	Feature[]	Classifier	\$CommunicationPath[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$CommunicationPath[i].finalSpecialization
General	general	Classifier[]	Classifier	\$CommunicationPath[i].general
Generalization	generalization	Generalization[]	Classifier	\$CommunicationPath[i].generalization
humanName	humanName	String	BaseElement	\$CommunicationPath[i].humanName
humanType	humanType	String	BaseElement	\$CommunicationPath[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$CommunicationPath[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$CommunicationPath[i].inheritedMember
isInvalid	invalid	boolean	ModelObject	\$CommunicationPath[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$CommunicationPath[i].leaf
localID	localID	String	ModelObject	\$CommunicationPath[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$CommunicationPath[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$CommunicationPath[i].member
Member End	memberEnd	Property[]	Association	\$CommunicationPath[i].memberEnd
Name	name	String	NamedElement	\$CommunicationPath[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$CommunicationPath[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$CommunicationPath[i].namespace
Navigable Owned End	navigableOwnedEnd	Property[]	Association	\$CommunicationPath[i].navigableOwnedEnd
objectParent	objectParent	BaseElement	BaseElement	\$CommunicationPath[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$CommunicationPath[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$CommunicationPath[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$CommunicationPath[i].ownedElement
Owned End	ownedEnd	Property[]	Association	\$CommunicationPath[i].ownedEnd
Owned Member	ownedMember	NamedElement[]	Namespace	\$CommunicationPath[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$CommunicationPath[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSign	Classifier	\$CommunicationPath[i].ownedTemplateSig

		ature		nature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$CommunicationPath[i].ownedUseCase
Owner	owner	Element	Element	\$CommunicationPath[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$CommunicationPath[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$CommunicationPath[i].owningTemplateParameter
Package	package	Package	Type	\$CommunicationPath[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$CommunicationPath[i].packageImport
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$CommunicationPath[i].powertypeExtent
Qualified Name	qualifiedName	String	NamedElement	\$CommunicationPath[i].qualifiedName
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$CommunicationPath[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$CommunicationPath[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$CommunicationPath[i].redefinitionContext
Related Element	relatedElement	Element[]	Relationship	\$CommunicationPath[i].relatedElement
Representation	representation	CollaborationUse	Classifier	\$CommunicationPath[i].representation
Substitution	substitution	Substitution[]	Classifier	\$CommunicationPath[i].substitution
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$CommunicationPath[i].supplierDependency
Sync Element	syncElement	Element	Element	\$CommunicationPath[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$CommunicationPath[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$CommunicationPath[i].templateParameter
Use Case	useCase	UseCase[]	Classifier	\$CommunicationPath[i].useCase
Visibility	visibility	VisibilityKind	PackageableElement	\$CommunicationPath[i].visibility

45. Component

Generalization

- Class

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$Component[i].ID
Class	UMLClass	Class	Classifier	\$Component[i].UMLClass
Is Abstract	abstract	boolean	Class	\$Component[i].abstract
Is Active	active	boolean	Class	\$Component[i].active
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Component[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$Component[i].attribute
classType	classType	Class	BaseElement	\$Component[i].classType

Classifier Behavior	classifierBehavior	Behavior	BehavioredClassifier	<code>\$Component[i].classifierBehavior</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Component[i].clientDependency</code>
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	<code>\$Component[i].collaborationUse</code>
isEditable	editable	boolean	BaseElement	<code>\$Component[i].editable</code>
Element Import	elementImport	ElementImport[]	Namespace	<code>\$Component[i].elementImport</code>
Extension	extension	Extension[]	Class	<code>\$Component[i].extension</code>
Feature	feature	Feature[]	Classifier	<code>\$Component[i].feature</code>
Is Final Specialization	finalSpecialization	boolean	Classifier	<code>\$Component[i].finalSpecialization</code>
General	general	Classifier[]	Classifier	<code>\$Component[i].general</code>
Generalization	generalization	Generalization[]	Classifier	<code>\$Component[i].generalization</code>
humanName	humanName	String	BaseElement	<code>\$Component[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Component[i].humanType</code>
Imported Member	importedMember	PackageableElement[]	Namespace	<code>\$Component[i].importedMember</code>
Is Indirectly Instantiated	indirectlyInstantiated	boolean	Component	<code>\$Component[i].indirectlyInstantiated</code>
Inherited Member	inheritedMember	NamedElement[]	Classifier	<code>\$Component[i].inheritedMember</code>
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavioredClassifier	<code>\$Component[i].interfaceRealization</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Component[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$Component[i].leaf</code>
localID	localID	String	ModelObject	<code>\$Component[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDObject	<code>\$Component[i].mdExtensions</code>
Member	member	NamedElement[]	Namespace	<code>\$Component[i].member</code>
Name	name	String	NamedElement	<code>\$Component[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Component[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Component[i].namespace</code>
Nested Classifier	nestedClassifier	Classifier[]	Class	<code>\$Component[i].nestedClassifier</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Component[i].objectParent</code>
Owned Attribute	ownedAttribute	Property[]	Class	<code>\$Component[i].ownedAttribute</code>
Owned Behavior	ownedBehavior	Behavior[]	BehavioredClassifier	<code>\$Component[i].ownedBehavior</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Component[i].ownedComment</code>
Owned Connector	ownedConnector	Connector[]	StructuredClassifier	<code>\$Component[i].ownedConnector</code>
Owned Diagram	ownedDiagram	Diagram[]	Namespace	<code>\$Component[i].ownedDiagram</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Component[i].ownedElement</code>
Owned Member	ownedMember	NamedElement[]	Namespace	<code>\$Component[i].ownedMember</code>
Owned Operation	ownedOperation	Operation[]	Class	<code>\$Component[i].ownedOperation</code>
Owned Port	ownedPort	Port[]	EncapsulatedClassifier	<code>\$Component[i].ownedPort</code>
Owned Reception	ownedReception	Reception[]	Class	<code>\$Component[i].ownedReception</code>
Owned Rule	ownedRule	Constraint[]	Namespace	<code>\$Component[i].ownedRule</code>
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	<code>\$Component[i].ownedTemplateSignature</code>
Owned Use Case	ownedUseCase	UseCase[]	Classifier	<code>\$Component[i].ownedUseCase</code>

Owner	owner	Element	Element	<code>\$Component[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$Component[i].owningPackage</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$Component[i].owningTemplateParameter</code>
Package	package	Package	Type	<code>\$Component[i].package</code>
Package Import	packageImport	PackageImport[]	Namespace	<code>\$Component[i].packageImport</code>
Packaged Element	packagedElement	PackageableElement[]	Component	<code>\$Component[i].packagedElement</code>
Part	part	Property[]	StructuredClassifier	<code>\$Component[i].part</code>
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	<code>\$Component[i].powertypeExtent</code>
Provided	provided	Interface[]	Component	<code>\$Component[i].provided</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Component[i].qualifiedName</code>
Realization	realization	ComponentRealization[]	Component	<code>\$Component[i].realization</code>
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	<code>\$Component[i].redefinedClassifier</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$Component[i].redefinedElement</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$Component[i].redefinitionContext</code>
Representation	representation	CollaborationUse	Classifier	<code>\$Component[i].representation</code>
Required	required	Interface[]	Component	<code>\$Component[i].required</code>
Role	role	ConnectableElement[]	StructuredClassifier	<code>\$Component[i].role</code>
Substitution	substitution	Substitution[]	Classifier	<code>\$Component[i].substitution</code>
Super Class	superClass	Class[]	Class	<code>\$Component[i].superClass</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Component[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Component[i].syncElement</code>
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	<code>\$Component[i].templateBinding</code>
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	<code>\$Component[i].templateParameter</code>
Use Case	useCase	UseCase[]	Classifier	<code>\$Component[i].useCase</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$Component[i].visibility</code>

46. ComponentRealization

Generalization

- [Realization](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$ComponentRealization[i].ID</code>
Abstraction	abstraction	Component	ComponentRealization	<code>\$ComponentRealization[i].abstraction</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$ComponentRealization[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$ComponentRealization[i].classType</code>

Client	client	NamedElement[]	Dependency	<code>\$ComponentRealization[i].client</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$ComponentRealization[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$ComponentRealization[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$ComponentRealization[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$ComponentRealization[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$ComponentRealization[i].invalid</code>
localID	localID	String	ModelObject	<code>\$ComponentRealization[i].localID</code>
Mapping	mapping	OpaqueExpression	Abstraction	<code>\$ComponentRealization[i].mapping</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$ComponentRealization[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$ComponentRealization[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$ComponentRealization[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$ComponentRealization[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$ComponentRealization[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$ComponentRealization[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$ComponentRealization[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$ComponentRealization[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$ComponentRealization[i].owningPackage</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$ComponentRealization[i].owningTemplateParameter</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$ComponentRealization[i].qualifiedName</code>
Realizing Classifier	realizingClassifier	Classifier[]	ComponentRealization	<code>\$ComponentRealization[i].realizingClassifier</code>
Related Element	relatedElement	Element[]	Relationship	<code>\$ComponentRealization[i].relatedElement</code>
Source	source	Element[]	DirectedRelationship	<code>\$ComponentRealization[i].source</code>
Supplier	supplier	NamedElement[]	Dependency	<code>\$ComponentRealization[i].supplier</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$ComponentRealization[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$ComponentRealization[i].syncElement</code>
Target	target	Element[]	DirectedRelationship	<code>\$ComponentRealization[i].target</code>
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$ComponentRealization[i].templateParameter</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$ComponentRealization[i].visibility</code>

47. ConditionalNode

Generalization

- [StructuredActivityNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class)	Sample Template Expression (VTL)

			Owner)	
ID	ID	String	MDOBJECT	\$ConditionalNode[i].ID
Activity	activity	Activity	StructuredActivityNode	\$ConditionalNode[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ConditionalNode[i].appliedStereotypeInstance
Is Assured	assured	boolean	ConditionalNode	\$ConditionalNode[i].assured
classType	classType	Class	BaseElement	\$ConditionalNode[i].classType
Clause	clause	Clause[]	ConditionalNode	\$ConditionalNode[i].clause
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ConditionalNode[i].clientDependency
Contained Edge	containedEdge	ActivityEdge[]	ActivityGroup	\$ConditionalNode[i].containedEdge
Contained Node	containedNode	ActivityNode[]	ActivityGroup	\$ConditionalNode[i].containedNode
Context	context	Classifier	Action	\$ConditionalNode[i].context
Is Determinate	determinate	boolean	ConditionalNode	\$ConditionalNode[i].determinate
Edge	edge	ActivityEdge[]	StructuredActivityNode	\$ConditionalNode[i].edge
isEditable	editable	boolean	BaseElement	\$ConditionalNode[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$ConditionalNode[i].elementImport
Handler	handler	ExceptionHandler[]	ExecutableNode	\$ConditionalNode[i].handler
humanName	humanName	String	BaseElement	\$ConditionalNode[i].humanName
humanType	humanType	String	BaseElement	\$ConditionalNode[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$ConditionalNode[i].importedMember
In Activity	inActivity	Activity	ActivityGroup	\$ConditionalNode[i].inActivity
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ConditionalNode[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ConditionalNode[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ConditionalNode[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ConditionalNode[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ConditionalNode[i].incoming
Input	input	InputPin[]	Action	\$ConditionalNode[i].input
isInvalid	invalid	boolean	ModelObject	\$ConditionalNode[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ConditionalNode[i].leaf
localID	localID	String	ModelObject	\$ConditionalNode[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$ConditionalNode[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$ConditionalNode[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$ConditionalNode[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$ConditionalNode[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$ConditionalNode[i].member
isMustIsolate	mustIsolate	boolean	StructuredActivityNode	\$ConditionalNode[i].mustIsolate
Name	name	String	NamedElement	\$ConditionalNode[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ConditionalNode[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ConditionalNode[i].namespace

Node	node	ActivityNode[]	StructuredActivityNode	\$ConditionalNode[i].node
objectParent	objectParent	BaseElement	BaseElement	\$ConditionalNode[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ConditionalNode[i].outgoing
Output	output	OutputPin[]	Action	\$ConditionalNode[i].output
Owned Comment	ownedComment	Comment[]	Element	\$ConditionalNode[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$ConditionalNode[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$ConditionalNode[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$ConditionalNode[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$ConditionalNode[i].ownedRule
Owner	owner	Element	Element	\$ConditionalNode[i].owner
Package Import	packageImport	PackageImport[]	Namespace	\$ConditionalNode[i].packageImport
Qualified Name	qualifiedName	String	NamedElement	\$ConditionalNode[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ConditionalNode[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ConditionalNode[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ConditionalNode[i].redefinitionContext
Result	result	OutputPin[]	ConditionalNode	\$ConditionalNode[i].result
Structured Node Input	structuredNodeInput	InputPin[]	StructuredActivityNode	\$ConditionalNode[i].structuredNodeInput
Structured Node Output	structuredNodeOutput	OutputPin[]	StructuredActivityNode	\$ConditionalNode[i].structuredNodeOutput
Subgroup	subgroup	ActivityGroup[]	ActivityGroup	\$ConditionalNode[i].subgroup
Super Group	superGroup	ActivityGroup	ActivityGroup	\$ConditionalNode[i].superGroup
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ConditionalNode[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ConditionalNode[i].syncElement
Variable	variable	Variable[]	StructuredActivityNode	\$ConditionalNode[i].variable
Visibility	visibility	VisibilityKind	NamedElement	\$ConditionalNode[i].visibility

48. ConnectableElement

Generalization

- [ParameterableElement](#)
- [TypedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ConnectableElement[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ConnectableElement[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ConnectableElement[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ConnectableElement[i].clientDependency
isEditable	editable	boolean	BaseElement	\$ConnectableElement[i].editable

End	end	ConnectorEnd[]	ConnectableElement	\$ConnectableElement[i].end
humanName	humanName	String	BaseElement	\$ConnectableElement[i].humanName
humanType	humanType	String	BaseElement	\$ConnectableElement[i].humanType
isInvalid	invalid	boolean	ModelObject	\$ConnectableElement[i].invalid
localID	localID	String	ModelObject	\$ConnectableElement[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ConnectableElement[i].mdExtensions
Name	name	String	NamedElement	\$ConnectableElement[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ConnectableElement[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ConnectableElement[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ConnectableElement[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$ConnectableElement[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ConnectableElement[i].ownedElement
Owner	owner	Element	Element	\$ConnectableElement[i].owner
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$ConnectableElement[i].owningTemplateParameter
Qualified Name	qualifiedName	String	NamedElement	\$ConnectableElement[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ConnectableElement[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ConnectableElement[i].syncElement
Template Parameter	templateParameter	ConnectableElementTemplateParameter	ConnectableElement	\$ConnectableElement[i].templateParameter
Type	type	Type	TypedElement	\$ConnectableElement[i].type
Visibility	visibility	VisibilityKind	NamedElement	\$ConnectableElement[i].visibility

49. ConnectableElementTemplateParameter

Generalization

- [TemplateParameter](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ConnectableElementTemplateParameter[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ConnectableElementTemplateParameter[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ConnectableElementTemplateParameter[i].classType
Default	default	ParameterableElement	TemplateParameter	\$ConnectableElementTemplateParameter[i].default

isEditable	editable	boolean	BaseElement	\$ConnectableElementTemplateParameter[i].editable
humanName	humanName	String	BaseElement	\$ConnectableElementTemplateParameter[i].humanName
humanType	humanType	String	BaseElement	\$ConnectableElementTemplateParameter[i].humanType
isInvalid	invalid	boolean	ModelObject	\$ConnectableElementTemplateParameter[i].invalid
localID	localID	String	ModelObject	\$ConnectableElementTemplateParameter[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$ConnectableElementTemplateParameter[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$ConnectableElementTemplateParameter[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$ConnectableElementTemplateParameter[i].ownedComment
Owned Default	ownedDefault	ParameterableElement	TemplateParameter	\$ConnectableElementTemplateParameter[i].ownedDefault
Owned Element	ownedElement	Element[]	Element	\$ConnectableElementTemplateParameter[i].ownedElement
Owned Parametered Element	ownedParameteredElement	ParameterableElement	TemplateParameter	\$ConnectableElementTemplateParameter[i].ownedParameteredElement
Owner	owner	Element	Element	\$ConnectableElementTemplateParameter[i].owner
Parametered Element	parameteredElement	ConnectableElement	ConnectableElementTemplateParameter	\$ConnectableElementTemplateParameter[i].parameteredElement
Signature	signature	TemplateSignature	TemplateParameter	\$ConnectableElementTemplateParameter[i].signature
Sync Element	syncElement	Element	Element	\$ConnectableElementTemplateParameter[i].syncElement

50. ConnectionPointReference

Generalization

- Vertex

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$ConnectionPointReference[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ConnectionPointReference[i].appliedStere

				otypeInstance
classType	classType	Class	BaseElement	\$ConnectionPointReference[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ConnectionPointReference[i].clientDependency
Container	container	Region	Vertex	\$ConnectionPointReference[i].container
isEditable	editable	boolean	BaseElement	\$ConnectionPointReference[i].editable
Entry	entry	Pseudostate[]	ConnectionPointReference	\$ConnectionPointReference[i].entry
Exit	exit	Pseudostate[]	ConnectionPointReference	\$ConnectionPointReference[i].exit
humanName	humanName	String	BaseElement	\$ConnectionPointReference[i].humanName
humanType	humanType	String	BaseElement	\$ConnectionPointReference[i].humanType
Incoming	incoming	Transition[]	Vertex	\$ConnectionPointReference[i].incoming
isInvalid	invalid	boolean	ModelObject	\$ConnectionPointReference[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ConnectionPointReference[i].leaf
localID	localID	String	ModelObject	\$ConnectionPointReference[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ConnectionPointReference[i].mdExtensions
Name	name	String	NamedElement	\$ConnectionPointReference[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ConnectionPointReference[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ConnectionPointReference[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ConnectionPointReference[i].objectParent
Outgoing	outgoing	Transition[]	Vertex	\$ConnectionPointReference[i].outgoing
Owned Comment	ownedComment	Comment[]	Element	\$ConnectionPointReference[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ConnectionPointReference[i].ownedElement
Owner	owner	Element	Element	\$ConnectionPointReference[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ConnectionPointReference[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ConnectionPointReference[i].redefinedElement
Redefined Vertex	redefinedVertex	Vertex	Vertex	\$ConnectionPointReference[i].redefinedVertex
Redefinition Context	redefinitionContext	Classifier[]	Vertex	\$ConnectionPointReference[i].redefinitionContext
State	state	State	ConnectionPointReference	\$ConnectionPointReference[i].state
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ConnectionPointReference[i].supplierDependency

Sync Element	syncElement	Element	Element	\$ConnectionPointReference[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$ConnectionPointReference[i].visibility

51. Connector

Generalization

- [Feature](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$Connector[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Connector[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Connector[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Connector[i].clientDependency
Contract	contract	Behavior[]	Connector	\$Connector[i].contract
isEditable	editable	boolean	BaseElement	\$Connector[i].editable
End	end	ConnectorEnd[]	Connector	\$Connector[i].end
Featuring Classifier	featuringClassifier	Classifier	Feature	\$Connector[i].featuringClassifier
humanName	humanName	String	BaseElement	\$Connector[i].humanName
humanType	humanType	String	BaseElement	\$Connector[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Connector[i].invalid
Kind	kind	ConnectorKind	Connector	\$Connector[i].kind
Is Leaf	leaf	boolean	RedefinableElement	\$Connector[i].leaf
localID	localID	String	ModelObject	\$Connector[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$Connector[i].mdExtensions
Name	name	String	NamedElement	\$Connector[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Connector[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Connector[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Connector[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Connector[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Connector[i].ownedElement
Owner	owner	Element	Element	\$Connector[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$Connector[i].qualifiedName
Redefined Connector	redefinedConnector	Connector[]	Connector	\$Connector[i].redefinedConnector
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Connector[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$Connector[i].redefinitionContext
Is Static	static	boolean	Feature	\$Connector[i].static
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Connector[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Connector[i].syncElement

Type	type	Association	Connector	<code>\$Connector[i].type</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$Connector[i].visibility</code>

52. ConnectorEnd

Generalization

- [MultiplicityElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$ConnectorEnd[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$ConnectorEnd[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$ConnectorEnd[i].classType</code>
Defining End	definingEnd	Property	ConnectorEnd	<code>\$ConnectorEnd[i].definingEnd</code>
isEditable	editable	boolean	BaseElement	<code>\$ConnectorEnd[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$ConnectorEnd[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$ConnectorEnd[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$ConnectorEnd[i].invalid</code>
localID	localID	String	ModelObject	<code>\$ConnectorEnd[i].localID</code>
Lower	lower	int	MultiplicityElement	<code>\$ConnectorEnd[i].lower</code>
Lower Value	lowerValue	ValueSpecification	MultiplicityElement	<code>\$ConnectorEnd[i].lowerValue</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$ConnectorEnd[i].mdExtensions</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$ConnectorEnd[i].objectParent</code>
Is Ordered	ordered	boolean	MultiplicityElement	<code>\$ConnectorEnd[i].ordered</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$ConnectorEnd[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$ConnectorEnd[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$ConnectorEnd[i].owner</code>
Part With Port	partWithPort	Property	ConnectorEnd	<code>\$ConnectorEnd[i].partWithPort</code>
Role	role	ConnectableElement	ConnectorEnd	<code>\$ConnectorEnd[i].role</code>
Sync Element	syncElement	Element	Element	<code>\$ConnectorEnd[i].syncElement</code>
Is Unique	unique	boolean	MultiplicityElement	<code>\$ConnectorEnd[i].unique</code>
Upper	upper	int	MultiplicityElement	<code>\$ConnectorEnd[i].upper</code>
Upper Value	upperValue	ValueSpecification	MultiplicityElement	<code>\$ConnectorEnd[i].upperValue</code>

53. ConsiderIgnoreFragment

Generalization

- CombinedFragment

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$ConsiderIgnoreFragment[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ConsiderIgnoreFragment[i].appliedStereotypeInstance
Cfragment Gate	cfragmentGate	Gate[]	CombinedFragment	\$ConsiderIgnoreFragment[i].cfragmentGate
classType	classType	Class	BaseElement	\$ConsiderIgnoreFragment[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ConsiderIgnoreFragment[i].clientDependency
Covered	covered	Lifeline[]	InteractionFragment	\$ConsiderIgnoreFragment[i].covered
isEditable	editable	boolean	BaseElement	\$ConsiderIgnoreFragment[i].editable
Enclosing Interaction	enclosingInteraction	Interaction	InteractionFragment	\$ConsiderIgnoreFragment[i].enclosingInteraction
Enclosing Operand	enclosingOperand	InteractionOperand	InteractionFragment	\$ConsiderIgnoreFragment[i].enclosingOperand
General Ordering	generalOrdering	GeneralOrdering[]	InteractionFragment	\$ConsiderIgnoreFragment[i].generalOrdering
humanName	humanName	String	BaseElement	\$ConsiderIgnoreFragment[i].humanName
humanType	humanType	String	BaseElement	\$ConsiderIgnoreFragment[i].humanType
Interaction Operator	interactionOperator	InteractionOperatorKind	CombinedFragment	\$ConsiderIgnoreFragment[i].interactionOperator
isInvalid	invalid	boolean	ModelObject	\$ConsiderIgnoreFragment[i].invalid
localID	localID	String	ModelObject	\$ConsiderIgnoreFragment[i].localID
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	\$ConsiderIgnoreFragment[i].mdExtensions
Message	message	NamedElement[]	ConsiderIgnoreFragment	\$ConsiderIgnoreFragment[i].message
Name	name	String	NamedElement	\$ConsiderIgnoreFragment[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ConsiderIgnoreFragment[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ConsiderIgnoreFragment[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ConsiderIgnoreFragment[i].objectParent
Operand	operand	InteractionOperand[]	CombinedFragment	\$ConsiderIgnoreFragment[i].operand
Owned Comment	ownedComment	Comment[]	Element	\$ConsiderIgnoreFragment[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ConsiderIgnoreFragment[i].ownedElement
Owner	owner	Element	Element	\$ConsiderIgnoreFragment[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ConsiderIgnoreFragment[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ConsiderIgnoreFragment[i].supplierDependency

Sync Element	syncElement	Element	Element	<code>\$ConsiderIgnoreFragment[i].syncElement</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$ConsiderIgnoreFragment[i].visibility</code>

54. Constraint

Generalization

- [PackageableElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$Constraint[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Constraint[i].appliedStereotypeInstance</code>
Body Context	bodyContext	Operation	Constraint	<code>\$Constraint[i].bodyContext</code>
classType	classType	Class	BaseElement	<code>\$Constraint[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Constraint[i].clientDependency</code>
Constrained Element	constrainedElement	Element[]	Constraint	<code>\$Constraint[i].constrainedElement</code>
Context	context	Namespace	Constraint	<code>\$Constraint[i].context</code>
isEditable	editable	boolean	BaseElement	<code>\$Constraint[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$Constraint[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Constraint[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Constraint[i].invalid</code>
localID	localID	String	ModelObject	<code>\$Constraint[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$Constraint[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$Constraint[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Constraint[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Constraint[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Constraint[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Constraint[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Constraint[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$Constraint[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$Constraint[i].owningPackage</code>
Owning State	owningState	State	Constraint	<code>\$Constraint[i].owningState</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$Constraint[i].owningTemplateParameter</code>
Owning Transition	owningTransition	ProtocolTransition	Constraint	<code>\$Constraint[i].owningTransition</code>
Post Context	postContext	Operation	Constraint	<code>\$Constraint[i].postContext</code>
Pre Context	preContext	Operation	Constraint	<code>\$Constraint[i].preContext</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Constraint[i].qualifiedName</code>
Specification	specification	ValueSpecification	Constraint	<code>\$Constraint[i].specification</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Constraint[i].supplierDependency</code>

Sync Element	syncElement	Element	Element	<code>\$Constraint[i].syncElement</code>
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$Constraint[i].templateParameter</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$Constraint[i].visibility</code>

55. Continuation

Generalization

- [InteractionFragment](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$Continuation[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Continuation[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$Continuation[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Continuation[i].clientDependency</code>
Covered	covered	Lifeline[]	InteractionFragment	<code>\$Continuation[i].covered</code>
isEditable	editable	boolean	BaseElement	<code>\$Continuation[i].editable</code>
Enclosing Interaction	enclosingInteraction	Interaction	InteractionFragment	<code>\$Continuation[i].enclosingInteraction</code>
Enclosing Operand	enclosingOperand	InteractionOperand	InteractionFragment	<code>\$Continuation[i].enclosingOperand</code>
General Ordering	generalOrdering	GeneralOrdering[]	InteractionFragment	<code>\$Continuation[i].generalOrdering</code>
humanName	humanName	String	BaseElement	<code>\$Continuation[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Continuation[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Continuation[i].invalid</code>
localID	localID	String	ModelObject	<code>\$Continuation[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$Continuation[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$Continuation[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Continuation[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Continuation[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Continuation[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Continuation[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Continuation[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$Continuation[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Continuation[i].qualifiedName</code>
isSetting	setting	boolean	Continuation	<code>\$Continuation[i].setting</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Continuation[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Continuation[i].syncElement</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$Continuation[i].visibility</code>

56. ControlFlow

Generalization

- [ActivityEdge](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$ControlFlow[i].ID
Activity	activity	Activity	ActivityEdge	\$ControlFlow[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ControlFlow[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ControlFlow[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ControlFlow[i].clientDependency
isEditable	editable	boolean	BaseElement	\$ControlFlow[i].editable
Guard	guard	ValueSpecification	ActivityEdge	\$ControlFlow[i].guard
humanName	humanName	String	BaseElement	\$ControlFlow[i].humanName
humanType	humanType	String	BaseElement	\$ControlFlow[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityEdge	\$ControlFlow[i].inGroup
In Partition	inPartition	ActivityPartition[]	ActivityEdge	\$ControlFlow[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityEdge	\$ControlFlow[i].inStructuredNode
Interrupts	interrupts	InterruptibleActivityRegion	ActivityEdge	\$ControlFlow[i].interrupts
isInvalid	invalid	boolean	ModelObject	\$ControlFlow[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ControlFlow[i].leaf
localID	localID	String	ModelObject	\$ControlFlow[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$ControlFlow[i].mdExtensions
Name	name	String	NamedElement	\$ControlFlow[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ControlFlow[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ControlFlow[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ControlFlow[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$ControlFlow[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ControlFlow[i].ownedElement
Owner	owner	Element	Element	\$ControlFlow[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ControlFlow[i].qualifiedName
Redefined Edge	redefinedEdge	ActivityEdge[]	ActivityEdge	\$ControlFlow[i].redefinedEdge
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ControlFlow[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ControlFlow[i].redefinitionContext
Source	source	ActivityNode	ActivityEdge	\$ControlFlow[i].source
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ControlFlow[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ControlFlow[i].syncElement

Target	target	ActivityNode	ActivityEdge	<code>\$ControlFlow[i].target</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$ControlFlow[i].visibility</code>
Weight	weight	ValueSpecification	ActivityEdge	<code>\$ControlFlow[i].weight</code>

57. ControlNode

Generalization

- [ActivityNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$ControlNode[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$ControlNode[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$ControlNode[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$ControlNode[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$ControlNode[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$ControlNode[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$ControlNode[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$ControlNode[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$ControlNode[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$ControlNode[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$ControlNode[i].inPartition</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$ControlNode[i].inStructuredNode</code>
Incoming	incoming	ActivityEdge[]	ActivityNode	<code>\$ControlNode[i].incoming</code>
isInvalid	invalid	boolean	ModelObject	<code>\$ControlNode[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$ControlNode[i].leaf</code>
localID	localID	String	ModelObject	<code>\$ControlNode[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$ControlNode[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$ControlNode[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$ControlNode[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$ControlNode[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$ControlNode[i].objectParent</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$ControlNode[i].outgoing</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$ControlNode[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$ControlNode[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$ControlNode[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$ControlNode[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$ControlNode[i].redefinedElement</code>

Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ControlNode[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ControlNode[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ControlNode[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ControlNode[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$ControlNode[i].visibility

58. CreateLinkAction

Generalization

- [WriteLinkAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$CreateLinkAction[i].ID
Activity	activity	Activity	ActivityNode	\$CreateLinkAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$CreateLinkAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$CreateLinkAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$CreateLinkAction[i].clientDependency
Context	context	Classifier	Action	\$CreateLinkAction[i].context
isEditable	editable	boolean	BaseElement	\$CreateLinkAction[i].editable
End Data	endData	LinkEndData[]	CreateLinkAction	\$CreateLinkAction[i].endData
Handler	handler	ExceptionHandler[]	ExecutableNode	\$CreateLinkAction[i].handler
humanName	humanName	String	BaseElement	\$CreateLinkAction[i].humanName
humanType	humanType	String	BaseElement	\$CreateLinkAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$CreateLinkAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$CreateLinkAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$CreateLinkAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$CreateLinkAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$CreateLinkAction[i].incoming
Input	input	InputPin[]	Action	\$CreateLinkAction[i].input
Input Value	inputValue	InputPin[]	LinkAction	\$CreateLinkAction[i].inputValue
isInvalid	invalid	boolean	ModelObject	\$CreateLinkAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$CreateLinkAction[i].leaf
localID	localID	String	ModelObject	\$CreateLinkAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$CreateLinkAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$CreateLinkAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$CreateLinkAction[i].locallyReentrant

mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$CreateLinkAction[i].mdExtensions
Name	name	String	NamedElement	\$CreateLinkAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$CreateLinkAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$CreateLinkAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$CreateLinkAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$CreateLinkAction[i].outgoing
Output	output	OutputPin[]	Action	\$CreateLinkAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$CreateLinkAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$CreateLinkAction[i].ownedElement
Owner	owner	Element	Element	\$CreateLinkAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$CreateLinkAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$CreateLinkAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$CreateLinkAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$CreateLinkAction[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$CreateLinkAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$CreateLinkAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$CreateLinkAction[i].visibility

59. CreateLinkObjectAction

Generalization

- [CreateLinkAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$CreateLinkObjectAction[i].ID
Activity	activity	Activity	ActivityNode	\$CreateLinkObjectAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$CreateLinkObjectAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$CreateLinkObjectAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$CreateLinkObjectAction[i].clientDependency
Context	context	Classifier	Action	\$CreateLinkObjectAction[i].context
isEditable	editable	boolean	BaseElement	\$CreateLinkObjectAction[i].editable
End Data	endData	LinkEndData[]	CreateLinkAction	\$CreateLinkObjectAction[i].endData
Handler	handler	ExceptionHandler[]	ExecutableNode	\$CreateLinkObjectAction[i].handler
humanName	humanName	String	BaseElement	\$CreateLinkObjectAction[i].humanName
humanType	humanType	String	BaseElement	\$CreateLinkObjectAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$CreateLinkObjectAction[i].inGroup

In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$CreateLinkObjectAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$CreateLinkObjectAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$CreateLinkObjectAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$CreateLinkObjectAction[i].incoming
Input	input	InputPin[]	Action	\$CreateLinkObjectAction[i].input
Input Value	inputValue	InputPin[]	LinkAction	\$CreateLinkObjectAction[i].inputValue
isInvalid	invalid	boolean	ModelObject	\$CreateLinkObjectAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$CreateLinkObjectAction[i].leaf
localID	localID	String	ModelObject	\$CreateLinkObjectAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$CreateLinkObjectAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$CreateLinkObjectAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$CreateLinkObjectAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$CreateLinkObjectAction[i].mdExtensions
Name	name	String	NamedElement	\$CreateLinkObjectAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$CreateLinkObjectAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$CreateLinkObjectAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$CreateLinkObjectAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$CreateLinkObjectAction[i].outgoing
Output	output	OutputPin[]	Action	\$CreateLinkObjectAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$CreateLinkObjectAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$CreateLinkObjectAction[i].ownedElement
Owner	owner	Element	Element	\$CreateLinkObjectAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$CreateLinkObjectAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$CreateLinkObjectAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$CreateLinkObjectAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$CreateLinkObjectAction[i].redefinitionContext
Result	result	OutputPin	CreateLinkObjectAction	\$CreateLinkObjectAction[i].result
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$CreateLinkObjectAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$CreateLinkObjectAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$CreateLinkObjectAction[i].visibility

60. CreateObjectAction

Generalization

- Action

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$CreateObjectAction[i].ID
Activity	activity	Activity	ActivityNode	\$CreateObjectAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$CreateObjectAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$CreateObjectAction[i].classType
Classifier	classifier	Classifier	CreateObjectAction	\$CreateObjectAction[i].classifier
Client Dependency	clientDependency	Dependency[]	NamedElement	\$CreateObjectAction[i].clientDependency
Context	context	Classifier	Action	\$CreateObjectAction[i].context
isEditable	editable	boolean	BaseElement	\$CreateObjectAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$CreateObjectAction[i].handler
humanName	humanName	String	BaseElement	\$CreateObjectAction[i].humanName
humanType	humanType	String	BaseElement	\$CreateObjectAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$CreateObjectAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$CreateObjectAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$CreateObjectAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$CreateObjectAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$CreateObjectAction[i].incoming
Input	input	InputPin[]	Action	\$CreateObjectAction[i].input
isInvalid	invalid	boolean	ModelObject	\$CreateObjectAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$CreateObjectAction[i].leaf
localID	localID	String	ModelObject	\$CreateObjectAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$CreateObjectAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$CreateObjectAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$CreateObjectAction[i].locallyReentrant
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	\$CreateObjectAction[i].mdExtensions
Name	name	String	NamedElement	\$CreateObjectAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$CreateObjectAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$CreateObjectAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$CreateObjectAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$CreateObjectAction[i].outgoing
Output	output	OutputPin[]	Action	\$CreateObjectAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$CreateObjectAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$CreateObjectAction[i].ownedElement
Owner	owner	Element	Element	\$CreateObjectAction[i].owner

Qualified Name	qualifiedName	String	NamedElement	\$CreateObjectAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$CreateObjectAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$CreateObjectAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$CreateObjectAction[i].redefinitionContext
Result	result	OutputPin	CreateObjectAction	\$CreateObjectAction[i].result
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$CreateObjectAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$CreateObjectAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$CreateObjectAction[i].visibility

61. DataStoreNode

Generalization

- CentralBufferNode

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$DataStoreNode[i].ID
Activity	activity	Activity	ActivityNode	\$DataStoreNode[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$DataStoreNode[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$DataStoreNode[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$DataStoreNode[i].clientDependency
Is Control Type	controlType	boolean	ObjectNode	\$DataStoreNode[i].controlType
isEditable	editable	boolean	BaseElement	\$DataStoreNode[i].editable
humanName	humanName	String	BaseElement	\$DataStoreNode[i].humanName
humanType	humanType	String	BaseElement	\$DataStoreNode[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$DataStoreNode[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$DataStoreNode[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$DataStoreNode[i].inPartition
In State	inState	State[]	ObjectNode	\$DataStoreNode[i].inState
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$DataStoreNode[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$DataStoreNode[i].incoming
isInvalid	invalid	boolean	ModelObject	\$DataStoreNode[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$DataStoreNode[i].leaf
localID	localID	String	ModelObject	\$DataStoreNode[i].localID
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	\$DataStoreNode[i].mdExtensions
Name	name	String	NamedElement	\$DataStoreNode[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$DataStoreNode[i].nameExpression

Namespace	namespace	Namespace	NamedElement	<code>\$DataStoreNode[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$DataStoreNode[i].objectParent</code>
Ordering	ordering	ObjectNodeOrderingKind	ObjectNode	<code>\$DataStoreNode[i].ordering</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$DataStoreNode[i].outgoing</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$DataStoreNode[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$DataStoreNode[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$DataStoreNode[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$DataStoreNode[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$DataStoreNode[i].redefinedElement</code>
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	<code>\$DataStoreNode[i].redefinedNode</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$DataStoreNode[i].redefinitionContext</code>
Selection	selection	Behavior	ObjectNode	<code>\$DataStoreNode[i].selection</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$DataStoreNode[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$DataStoreNode[i].syncElement</code>
Type	type	Type	TypedElement	<code>\$DataStoreNode[i].type</code>
Upper Bound	upperBound	ValueSpecification	ObjectNode	<code>\$DataStoreNode[i].upperBound</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$DataStoreNode[i].visibility</code>

62. DataType

Generalization

- [Classifier](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$DataType[i].ID</code>
Class	UMLClass	Class	Classifier	<code>\$DataType[i].UMLClass</code>
Is Abstract	abstract	boolean	Classifier	<code>\$DataType[i].abstract</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$DataType[i].appliedStereotypeInstance</code>
Attribute	attribute	Property[]	Classifier	<code>\$DataType[i].attribute</code>
classType	classType	Class	BaseElement	<code>\$DataType[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$DataType[i].clientDependency</code>
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	<code>\$DataType[i].collaborationUse</code>
isEditable	editable	boolean	BaseElement	<code>\$DataType[i].editable</code>
Element Import	elementImport	ElementImport[]	Namespace	<code>\$DataType[i].elementImport</code>
Feature	feature	Feature[]	Classifier	<code>\$DataType[i].feature</code>
Is Final Specialization	finalSpecialization	boolean	Classifier	<code>\$DataType[i].finalSpecialization</code>
General	general	Classifier[]	Classifier	<code>\$DataType[i].general</code>
Generalization	generalization	Generalization[]	Classifier	<code>\$DataType[i].generalization</code>

humanName	humanName	String	BaseElement	\$DataType[i].humanName
humanType	humanType	String	BaseElement	\$DataType[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$DataType[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$DataType[i].inheritedMember
isInvalid	invalid	boolean	ModelObject	\$DataType[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$DataType[i].leaf
localID	localID	String	ModelObject	\$DataType[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$DataType[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$DataType[i].member
Name	name	String	NamedElement	\$DataType[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$DataType[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$DataType[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$DataType[i].objectParent
Owned Attribute	ownedAttribute	Property[]	DataType	\$DataType[i].ownedAttribute
Owned Comment	ownedComment	Comment[]	Element	\$DataType[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$DataType[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$DataType[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$DataType[i].ownedMember
Owned Operation	ownedOperation	Operation[]	DataType	\$DataType[i].ownedOperation
Owned Rule	ownedRule	Constraint[]	Namespace	\$DataType[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$DataType[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$DataType[i].ownedUseCase
Owner	owner	Element	Element	\$DataType[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$DataType[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$DataType[i].owningTemplateParameter
Package	package	Package	Type	\$DataType[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$DataType[i].packageImport
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$DataType[i].powertypeExtent
Qualified Name	qualifiedName	String	NamedElement	\$DataType[i].qualifiedName
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$DataType[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$DataType[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$DataType[i].redefinitionContext
Representation	representation	CollaborationUse	Classifier	\$DataType[i].representation
Substitution	substitution	Substitution[]	Classifier	\$DataType[i].substitution
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$DataType[i].supplierDependency
Sync Element	syncElement	Element	Element	\$DataType[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$DataType[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$DataType[i].templateParameter

Use Case	useCase	UseCase[]	Classifier	\$DataType[i].useCase
Visibility	visibility	VisibilityKind	PackageableElement	\$DataType[i].visibility

63. DecisionNode

Generalization

- [ControlNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$DecisionNode[i].ID
Activity	activity	Activity	ActivityNode	\$DecisionNode[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$DecisionNode[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$DecisionNode[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$DecisionNode[i].clientDependency
Decision Input	decisionInput	Behavior	DecisionNode	\$DecisionNode[i].decisionInput
Decision Input Flow	decisionInputFlow	ObjectFlow	DecisionNode	\$DecisionNode[i].decisionInputFlow
isEditable	editable	boolean	BaseElement	\$DecisionNode[i].editable
humanName	humanName	String	BaseElement	\$DecisionNode[i].humanName
humanType	humanType	String	BaseElement	\$DecisionNode[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$DecisionNode[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$DecisionNode[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$DecisionNode[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$DecisionNode[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$DecisionNode[i].incoming
isInvalid	invalid	boolean	ModelObject	\$DecisionNode[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$DecisionNode[i].leaf
localID	localID	String	ModelObject	\$DecisionNode[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$DecisionNode[i].mdExtensions
Name	name	String	NamedElement	\$DecisionNode[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$DecisionNode[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$DecisionNode[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$DecisionNode[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$DecisionNode[i].outgoing
Owned Comment	ownedComment	Comment[]	Element	\$DecisionNode[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$DecisionNode[i].ownedElement
Owner	owner	Element	Element	\$DecisionNode[i].owner

Qualified Name	qualifiedName	String	NamedElement	\$DecisionNode[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$DecisionNode[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$DecisionNode[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$DecisionNode[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$DecisionNode[i].supplierDependency
Sync Element	syncElement	Element	Element	\$DecisionNode[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$DecisionNode[i].visibility

64. Dependency

Generalization

- [DirectedRelationship](#)
- [PackageableElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MObject	\$Dependency[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Dependency[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Dependency[i].classType
Client	client	NamedElement[]	Dependency	\$Dependency[i].client
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Dependency[i].clientDependency
isEditable	editable	boolean	BaseElement	\$Dependency[i].editable
humanName	humanName	String	BaseElement	\$Dependency[i].humanName
humanType	humanType	String	BaseElement	\$Dependency[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Dependency[i].invalid
localID	localID	String	ModelObject	\$Dependency[i].localID
mdExtensions	mdExtensions	MDExtension[]	MObject	\$Dependency[i].mdExtensions
Name	name	String	NamedElement	\$Dependency[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Dependency[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Dependency[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Dependency[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Dependency[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Dependency[i].ownedElement
Owner	owner	Element	Element	\$Dependency[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Dependency[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Dependency[i].owningTemplateParameter
Qualified Name	qualifiedName	String	NamedElement	\$Dependency[i].qualifiedName
Related Element	relatedElement	Element[]	Relationship	\$Dependency[i].relatedElement
Source	source	Element[]	DirectedRelationship	\$Dependency[i].source

Supplier	supplier	NamedElement[]	Dependency	\$Dependency[i].supplier
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Dependency[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Dependency[i].syncElement
Target	target	Element[]	DirectedRelationship	\$Dependency[i].target
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$Dependency[i].templateParameter
Visibility	visibility	VisibilityKind	PackageableElement	\$Dependency[i].visibility

65. DeployedArtifact

Generalization

- [NamedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$DeployedArtifact[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$DeployedArtifact[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$DeployedArtifact[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$DeployedArtifact[i].clientDependency
isEditable	editable	boolean	BaseElement	\$DeployedArtifact[i].editable
humanName	humanName	String	BaseElement	\$DeployedArtifact[i].humanName
humanType	humanType	String	BaseElement	\$DeployedArtifact[i].humanType
isInvalid	invalid	boolean	ModelObject	\$DeployedArtifact[i].invalid
localID	localID	String	ModelObject	\$DeployedArtifact[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$DeployedArtifact[i].mdExtensions
Name	name	String	NamedElement	\$DeployedArtifact[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$DeployedArtifact[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$DeployedArtifact[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$DeployedArtifact[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$DeployedArtifact[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$DeployedArtifact[i].ownedElement
Owner	owner	Element	Element	\$DeployedArtifact[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$DeployedArtifact[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$DeployedArtifact[i].supplierDependency
Sync Element	syncElement	Element	Element	\$DeployedArtifact[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$DeployedArtifact[i].visibility

66. Deployment

Generalization

- [Dependency](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$Deployment[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Deployment[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Deployment[i].classType
Client	client	NamedElement[]	Dependency	\$Deployment[i].client
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Deployment[i].clientDependency
Configuration	configuration	DeploymentSpecification[]	Deployment	\$Deployment[i].configuration
Deployed Artifact	deployedArtifact	DeployedArtifact[]	Deployment	\$Deployment[i].deployedArtifact
isEditable	editable	boolean	BaseElement	\$Deployment[i].editable
humanName	humanName	String	BaseElement	\$Deployment[i].humanName
humanType	humanType	String	BaseElement	\$Deployment[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Deployment[i].invalid
localID	localID	String	ModelObject	\$Deployment[i].localID
Location	location	DeploymentTarget	Deployment	\$Deployment[i].location
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$Deployment[i].mdExtensions
Name	name	String	NamedElement	\$Deployment[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Deployment[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Deployment[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Deployment[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Deployment[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Deployment[i].ownedElement
Owner	owner	Element	Element	\$Deployment[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Deployment[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Deployment[i].owningTemplateParameter
Qualified Name	qualifiedName	String	NamedElement	\$Deployment[i].qualifiedName
Related Element	relatedElement	Element[]	Relationship	\$Deployment[i].relatedElement
Source	source	Element[]	DirectedRelationship	\$Deployment[i].source
Supplier	supplier	NamedElement[]	Dependency	\$Deployment[i].supplier
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Deployment[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Deployment[i].syncElement
Target	target	Element[]	DirectedRelationship	\$Deployment[i].target
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$Deployment[i].templateParameter

Visibility	visibility	VisibilityKind	PackageableElement	\$Deployment[i].visibility

67. DeploymentSpecification

Generalization

- [Artifact](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$DeploymentSpecification[i].ID
Class	UMLClass	Class	Classifier	\$DeploymentSpecification[i].UMLClass
Is Abstract	abstract	boolean	Classifier	\$DeploymentSpecification[i].abstract
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$DeploymentSpecification[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$DeploymentSpecification[i].attribute
classType	classType	Class	BaseElement	\$DeploymentSpecification[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$DeploymentSpecification[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$DeploymentSpecification[i].collaborationUse
Deployment	deployment	Deployment	DeploymentSpecification	\$DeploymentSpecification[i].deployment
Deployment Location	deploymentLocation	String	DeploymentSpecification	\$DeploymentSpecification[i].deploymentLocation
isEditable	editable	boolean	BaseElement	\$DeploymentSpecification[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$DeploymentSpecification[i].elementImport
Execution Location	executionLocation	String	DeploymentSpecification	\$DeploymentSpecification[i].executionLocation
Feature	feature	Feature[]	Classifier	\$DeploymentSpecification[i].feature
File Name	fileName	String	Artifact	\$DeploymentSpecification[i].fileName
Is Final Specialization	finalSpecialization	boolean	Classifier	\$DeploymentSpecification[i].finalSpecialization
General	general	Classifier[]	Classifier	\$DeploymentSpecification[i].general
Generalization	generalization	Generalization[]	Classifier	\$DeploymentSpecification[i].generalization
humanName	humanName	String	BaseElement	\$DeploymentSpecification[i].humanName
humanType	humanType	String	BaseElement	\$DeploymentSpecification[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$DeploymentSpecification[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$DeploymentSpecification[i].inheritedMember

isInvalid	invalid	boolean	ModelObject	\$DeploymentSpecification[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$DeploymentSpecification[i].leaf
localID	localID	String	ModelObject	\$DeploymentSpecification[i].localID
Manifestation	manifestation	Manifestation[]	Artifact	\$DeploymentSpecification[i].manifestation
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$DeploymentSpecification[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$DeploymentSpecification[i].member
Name	name	String	NamedElement	\$DeploymentSpecification[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$DeploymentSpecification[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$DeploymentSpecification[i].namespace
Nested Artifact	nestedArtifact	Artifact[]	Artifact	\$DeploymentSpecification[i].nestedArtifact
objectParent	objectParent	BaseElement	BaseElement	\$DeploymentSpecification[i].objectParent
Owned Attribute	ownedAttribute	Property[]	Artifact	\$DeploymentSpecification[i].ownedAttribute
Owned Comment	ownedComment	Comment[]	Element	\$DeploymentSpecification[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$DeploymentSpecification[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$DeploymentSpecification[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$DeploymentSpecification[i].ownedMember
Owned Operation	ownedOperation	Operation[]	Artifact	\$DeploymentSpecification[i].ownedOperation
Owned Rule	ownedRule	Constraint[]	Namespace	\$DeploymentSpecification[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$DeploymentSpecification[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$DeploymentSpecification[i].ownedUseCase
Owner	owner	Element	Element	\$DeploymentSpecification[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$DeploymentSpecification[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$DeploymentSpecification[i].owningTemplateParameter
Package	package	Package	Type	\$DeploymentSpecification[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$DeploymentSpecification[i].packageImport
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$DeploymentSpecification[i].powertypeExtent
Qualified Name	qualifiedName	String	NamedElement	\$DeploymentSpecification[i].qualifiedName
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$DeploymentSpecification[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$DeploymentSpecification[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$DeploymentSpecification[i].redefinitionContext

Representation	representation	CollaborationUse	Classifier	<code>\$DeploymentSpecification[i].representation</code>
Substitution	substitution	Substitution[]	Classifier	<code>\$DeploymentSpecification[i].substitution</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$DeploymentSpecification[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$DeploymentSpecification[i].syncElement</code>
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	<code>\$DeploymentSpecification[i].templateBinding</code>
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	<code>\$DeploymentSpecification[i].templateParameter</code>
Use Case	useCase	UseCase[]	Classifier	<code>\$DeploymentSpecification[i].useCase</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$DeploymentSpecification[i].visibility</code>

68. DeploymentTarget

Generalization

- [NamedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$DeploymentTarget[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$DeploymentTarget[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$DeploymentTarget[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$DeploymentTarget[i].clientDependency</code>
Deployed Element	deployedElement	PackageableElement[]	DeploymentTarget	<code>\$DeploymentTarget[i].deployedElement</code>
Deployment	deployment	Deployment[]	DeploymentTarget	<code>\$DeploymentTarget[i].deployment</code>
isEditable	editable	boolean	BaseElement	<code>\$DeploymentTarget[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$DeploymentTarget[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$DeploymentTarget[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$DeploymentTarget[i].invalid</code>
localID	localID	String	ModelObject	<code>\$DeploymentTarget[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$DeploymentTarget[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$DeploymentTarget[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$DeploymentTarget[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$DeploymentTarget[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$DeploymentTarget[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$DeploymentTarget[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$DeploymentTarget[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$DeploymentTarget[i].owner</code>

Qualified Name	qualifiedName	String	NamedElement	\$DeploymentTarget[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$DeploymentTarget[i].supplierDependency
Sync Element	syncElement	Element	Element	\$DeploymentTarget[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$DeploymentTarget[i].visibility

69. DestroyLinkAction

Generalization

- [WriteLinkAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$DestroyLinkAction[i].ID
Activity	activity	Activity	ActivityNode	\$DestroyLinkAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$DestroyLinkAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$DestroyLinkAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$DestroyLinkAction[i].clientDependency
Context	context	Classifier	Action	\$DestroyLinkAction[i].context
isEditable	editable	boolean	BaseElement	\$DestroyLinkAction[i].editable
End Data	endData	LinkEndData[]	DestroyLinkAction	\$DestroyLinkAction[i].endData
Handler	handler	ExceptionHandler[]	ExecutableNode	\$DestroyLinkAction[i].handler
humanName	humanName	String	BaseElement	\$DestroyLinkAction[i].humanName
humanType	humanType	String	BaseElement	\$DestroyLinkAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$DestroyLinkAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$DestroyLinkAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$DestroyLinkAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$DestroyLinkAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$DestroyLinkAction[i].incoming
Input	input	InputPin[]	Action	\$DestroyLinkAction[i].input
Input Value	inputValue	InputPin[]	LinkAction	\$DestroyLinkAction[i].inputValue
isInvalid	invalid	boolean	ModelObject	\$DestroyLinkAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$DestroyLinkAction[i].leaf
localID	localID	String	ModelObject	\$DestroyLinkAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$DestroyLinkAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$DestroyLinkAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$DestroyLinkAction[i].locallyReentrant
mdExtensions	mdExtensions	MDEExtension[]	MDOObject	\$DestroyLinkAction[i].mdExtensions

Name	name	String	NamedElement	<code>\$DestroyLinkAction[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$DestroyLinkAction[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$DestroyLinkAction[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$DestroyLinkAction[i].objectParent</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$DestroyLinkAction[i].outgoing</code>
Output	output	OutputPin[]	Action	<code>\$DestroyLinkAction[i].output</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$DestroyLinkAction[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$DestroyLinkAction[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$DestroyLinkAction[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$DestroyLinkAction[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$DestroyLinkAction[i].redefinedElement</code>
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	<code>\$DestroyLinkAction[i].redefinedNode</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$DestroyLinkAction[i].redefinitionContext</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$DestroyLinkAction[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$DestroyLinkAction[i].syncElement</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$DestroyLinkAction[i].visibility</code>

70. DestroyObjectAction

Generalization

- [Action](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$DestroyObjectAction[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$DestroyObjectAction[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$DestroyObjectAction[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$DestroyObjectAction[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$DestroyObjectAction[i].clientDependency</code>
Context	context	Classifier	Action	<code>\$DestroyObjectAction[i].context</code>
Is Destroy Links	destroyLinks	boolean	DestroyObjectAction	<code>\$DestroyObjectAction[i].destroyLinks</code>
Is Destroy Owned Objects	destroyOwnedObjects	boolean	DestroyObjectAction	<code>\$DestroyObjectAction[i].destroyOwnedObjects</code>
isEditable	editable	boolean	BaseElement	<code>\$DestroyObjectAction[i].editable</code>
Handler	handler	ExceptionHandler[]	ExecutableNode	<code>\$DestroyObjectAction[i].handler</code>
humanName	humanName	String	BaseElement	<code>\$DestroyObjectAction[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$DestroyObjectAction[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$DestroyObjectAction[i].inGroup</code>

In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$DestroyObjectAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$DestroyObjectAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$DestroyObjectAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$DestroyObjectAction[i].incoming
Input	input	InputPin[]	Action	\$DestroyObjectAction[i].input
isInvalid	invalid	boolean	ModelObject	\$DestroyObjectAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$DestroyObjectAction[i].leaf
localID	localID	String	ModelObject	\$DestroyObjectAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$DestroyObjectAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$DestroyObjectAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$DestroyObjectAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$DestroyObjectAction[i].mdExtensions
Name	name	String	NamedElement	\$DestroyObjectAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$DestroyObjectAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$DestroyObjectAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$DestroyObjectAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$DestroyObjectAction[i].outgoing
Output	output	OutputPin[]	Action	\$DestroyObjectAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$DestroyObjectAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$DestroyObjectAction[i].ownedElement
Owner	owner	Element	Element	\$DestroyObjectAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$DestroyObjectAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$DestroyObjectAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$DestroyObjectAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$DestroyObjectAction[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$DestroyObjectAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$DestroyObjectAction[i].syncElement
Target	target	InputPin	DestroyObjectAction	\$DestroyObjectAction[i].target
Visibility	visibility	VisibilityKind	NamedElement	\$DestroyObjectAction[i].visibility

71. DestructionOccurrenceSpecification

Generalization

- [MessageOccurrenceSpecification](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)

ID	ID	String	MObject	\$DestructionOccurrenceSpecification[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$DestructionOccurrenceSpecification[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$DestructionOccurrenceSpecification[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$DestructionOccurrenceSpecification[i].clientDependency
Covered	covered	Lifeline[]	OccurrenceSpecification	\$DestructionOccurrenceSpecification[i].covered
isEditable	editable	boolean	BaseElement	\$DestructionOccurrenceSpecification[i].editable
Enclosing Interaction	enclosingInteraction	Interaction	InteractionFragment	\$DestructionOccurrenceSpecification[i].enclosingInteraction
Enclosing Operand	enclosingOperand	InteractionOperand	InteractionFragment	\$DestructionOccurrenceSpecification[i].enclosingOperand
General Ordering	generalOrdering	GeneralOrdering[]	InteractionFragment	\$DestructionOccurrenceSpecification[i].generalOrdering
humanName	humanName	String	BaseElement	\$DestructionOccurrenceSpecification[i].humanName
humanType	humanType	String	BaseElement	\$DestructionOccurrenceSpecification[i].humanType
isInvalid	invalid	boolean	ModelObject	\$DestructionOccurrenceSpecification[i].invalid
localID	localID	String	ModelObject	\$DestructionOccurrenceSpecification[i].localID
mdExtensions	mdExtensions	MDExtension[]	MObject	\$DestructionOccurrenceSpecification[i].mdExtensions
Message	message	Message	MessageEnd	\$DestructionOccurrenceSpecification[i].message
Name	name	String	NamedElement	\$DestructionOccurrenceSpecification[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$DestructionOccurrenceSpecification[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$DestructionOccurrenceSpecification[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$DestructionOccurrenceSpecification[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$DestructionOccurrenceSpecification[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$DestructionOccurrenceSpecification[i].ownedElement

Owner	owner	Element	Element	<code>\$DestructionOccurrenceSpecification[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$DestructionOccurrenceSpecification[i].qualifiedName</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$DestructionOccurrenceSpecification[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$DestructionOccurrenceSpecification[i].syncElement</code>
To After	toAfter	GeneralOrdering[]	OccurrenceSpecification	<code>\$DestructionOccurrenceSpecification[i].toAfter</code>
To Before	toBefore	GeneralOrdering[]	OccurrenceSpecification	<code>\$DestructionOccurrenceSpecification[i].toBefore</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$DestructionOccurrenceSpecification[i].visibility</code>

72. Device

Generalization

- [Node](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$Device[i].ID</code>
Class	UMLClass	Class	Classifier	<code>\$Device[i].UMLClass</code>
Is Abstract	abstract	boolean	Class	<code>\$Device[i].abstract</code>
Is Active	active	boolean	Class	<code>\$Device[i].active</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Device[i].appliedStereotypeInstance</code>
Attribute	attribute	Property[]	Classifier	<code>\$Device[i].attribute</code>
classType	classType	Class	BaseElement	<code>\$Device[i].classType</code>
Classifier Behavior	classifierBehavior	Behavior	BehavioredClassifier	<code>\$Device[i].classifierBehavior</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Device[i].clientDependency</code>
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	<code>\$Device[i].collaborationUse</code>
Deployed Element	deployedElement	PackageableElement[]	DeploymentTarget	<code>\$Device[i].deployedElement</code>
Deployment	deployment	Deployment[]	DeploymentTarget	<code>\$Device[i].deployment</code>
isEditable	editable	boolean	BaseElement	<code>\$Device[i].editable</code>
Element Import	elementImport	ElementImport[]	Namespace	<code>\$Device[i].elementImport</code>
Extension	extension	Extension[]	Class	<code>\$Device[i].extension</code>
Feature	feature	Feature[]	Classifier	<code>\$Device[i].feature</code>
Is Final Specialization	finalSpecialization	boolean	Classifier	<code>\$Device[i].finalSpecialization</code>

General	general	Classifier[]	Classifier	<code>\$Device[i].general</code>
Generalization	generalization	Generalization[]	Classifier	<code>\$Device[i].generalization</code>
humanName	humanName	String	BaseElement	<code>\$Device[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Device[i].humanType</code>
Imported Member	importedMember	PackageableElement[]	Namespace	<code>\$Device[i].importedMember</code>
Inherited Member	inheritedMember	NamedElement[]	Classifier	<code>\$Device[i].inheritedMember</code>
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavioredClassifier	<code>\$Device[i].interfaceRealization</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Device[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$Device[i].leaf</code>
localID	localID	String	ModelObject	<code>\$Device[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$Device[i].mdExtensions</code>
Member	member	NamedElement[]	Namespace	<code>\$Device[i].member</code>
Name	name	String	NamedElement	<code>\$Device[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Device[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Device[i].namespace</code>
Nested Classifier	nestedClassifier	Classifier[]	Class	<code>\$Device[i].nestedClassifier</code>
Nested Node	nestedNode	Node[]	Node	<code>\$Device[i].nestedNode</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Device[i].objectParent</code>
Owned Attribute	ownedAttribute	Property[]	Class	<code>\$Device[i].ownedAttribute</code>
Owned Behavior	ownedBehavior	Behavior[]	BehavioredClassifier	<code>\$Device[i].ownedBehavior</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Device[i].ownedComment</code>
Owned Connector	ownedConnector	Connector[]	StructuredClassifier	<code>\$Device[i].ownedConnector</code>
Owned Diagram	ownedDiagram	Diagram[]	Namespace	<code>\$Device[i].ownedDiagram</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Device[i].ownedElement</code>
Owned Member	ownedMember	NamedElement[]	Namespace	<code>\$Device[i].ownedMember</code>
Owned Operation	ownedOperation	Operation[]	Class	<code>\$Device[i].ownedOperation</code>
Owned Port	ownedPort	Port[]	EncapsulatedClassifier	<code>\$Device[i].ownedPort</code>
Owned Reception	ownedReception	Reception[]	Class	<code>\$Device[i].ownedReception</code>
Owned Rule	ownedRule	Constraint[]	Namespace	<code>\$Device[i].ownedRule</code>
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	<code>\$Device[i].ownedTemplateSignature</code>
Owned Use Case	ownedUseCase	UseCase[]	Classifier	<code>\$Device[i].ownedUseCase</code>
Owner	owner	Element	Element	<code>\$Device[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$Device[i].owningPackage</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$Device[i].owningTemplateParameter</code>
Package	package	Package	Type	<code>\$Device[i].package</code>
Package Import	packageImport	PackageImport[]	Namespace	<code>\$Device[i].packageImport</code>
Part	part	Property[]	StructuredClassifier	<code>\$Device[i].part</code>
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	<code>\$Device[i].powertypeExtent</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Device[i].qualifiedName</code>

Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$Device[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Device[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$Device[i].redefinitionContext
Representation	representation	CollaborationUse	Classifier	\$Device[i].representation
Role	role	ConnectableElement[]	StructuredClassifier	\$Device[i].role
Substitution	substitution	Substitution[]	Classifier	\$Device[i].substitution
Super Class	superClass	Class[]	Class	\$Device[i].superClass
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Device[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Device[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$Device[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$Device[i].templateParameter
Use Case	useCase	UseCase[]	Classifier	\$Device[i].useCase
Visibility	visibility	VisibilityKind	PackageableElement	\$Device[i].visibility

73. Diagram

Generalization

- [NamedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$Diagram[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Diagram[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Diagram[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Diagram[i].clientDependency
Context	context	Element	Diagram	\$Diagram[i].context
isEditable	editable	boolean	BaseElement	\$Diagram[i].editable
humanName	humanName	String	BaseElement	\$Diagram[i].humanName
humanType	humanType	String	BaseElement	\$Diagram[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Diagram[i].invalid
localID	localID	String	ModelObject	\$Diagram[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$Diagram[i].mdExtensions
Name	name	String	NamedElement	\$Diagram[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Diagram[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Diagram[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Diagram[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Diagram[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Diagram[i].ownedElement

Owner	owner	Element	Element	\$Diagram[i].owner
Owner Of Diagram	ownerOfDiagram	Namespace	Diagram	\$Diagram[i].ownerOfDiagram
Qualified Name	qualifiedName	String	NamedElement	\$Diagram[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Diagram[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Diagram[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$Diagram[i].visibility

74. DirectedRelationship

Generalization

- [Relationship](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$DirectedRelationship[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$DirectedRelationship[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$DirectedRelationship[i].classType
isEditable	editable	boolean	BaseElement	\$DirectedRelationship[i].editable
humanName	humanName	String	BaseElement	\$DirectedRelationship[i].humanName
humanType	humanType	String	BaseElement	\$DirectedRelationship[i].humanType
isInvalid	invalid	boolean	ModelObject	\$DirectedRelationship[i].invalid
localID	localID	String	ModelObject	\$DirectedRelationship[i].localID
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	\$DirectedRelationship[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$DirectedRelationship[i].objectParent
Owned Comment	ownedComment	COMMENT[]	Element	\$DirectedRelationship[i].ownedComment
Owned Element	ownedElement	ELEMENT[]	Element	\$DirectedRelationship[i].ownedElement
Owner	owner	Element	Element	\$DirectedRelationship[i].owner
Related Element	relatedElement	ELEMENT[]	Relationship	\$DirectedRelationship[i].relatedElement
Source	source	ELEMENT[]	DirectedRelationship	\$DirectedRelationship[i].source
Sync Element	syncElement	Element	Element	\$DirectedRelationship[i].syncElement
Target	target	ELEMENT[]	DirectedRelationship	\$DirectedRelationship[i].target

75. Duration

Generalization

- [ValueSpecification](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$Duration[i].ID</code>
Applied Stereotype Instance	<code>appliedStereotypeInstance</code>	InstanceSpecification	Element	<code>\$Duration[i].appliedStereotypeInstance</code>
classType	<code>classType</code>	Class	BaseElement	<code>\$Duration[i].classType</code>
Client Dependency	<code>clientDependency</code>	Dependency[]	NamedElement	<code>\$Duration[i].clientDependency</code>
isEditable	<code>editable</code>	boolean	BaseElement	<code>\$Duration[i].editable</code>
Expr	<code>expr</code>	ValueSpecification	Duration	<code>\$Duration[i].expr</code>
Expression	<code>expression</code>	Expression	ValueSpecification	<code>\$Duration[i].expression</code>
humanName	<code>humanName</code>	String	BaseElement	<code>\$Duration[i].humanName</code>
humanType	<code>humanType</code>	String	BaseElement	<code>\$Duration[i].humanType</code>
isInvalid	<code>invalid</code>	boolean	ModelObject	<code>\$Duration[i].invalid</code>
localID	<code>localID</code>	String	ModelObject	<code>\$Duration[i].localID</code>
mdExtensions	<code>mdExtensions</code>	MDEXTENSION[]	MDOBJECT	<code>\$Duration[i].mdExtensions</code>
Name	<code>name</code>	String	NamedElement	<code>\$Duration[i].name</code>
Name Expression	<code>nameExpression</code>	StringExpression	NamedElement	<code>\$Duration[i].nameExpression</code>
Namespace	<code>namespace</code>	Namespace	NamedElement	<code>\$Duration[i].namespace</code>
objectParent	<code>objectParent</code>	BaseElement	BaseElement	<code>\$Duration[i].objectParent</code>
Observation	<code>observation</code>	OBSERVATION[]	Duration	<code>\$Duration[i].observation</code>
Owned Comment	<code>ownedComment</code>	COMMENT[]	Element	<code>\$Duration[i].ownedComment</code>
Owned Element	<code>ownedElement</code>	ELEMENT[]	Element	<code>\$Duration[i].ownedElement</code>
Owner	<code>owner</code>	Element	Element	<code>\$Duration[i].owner</code>
Owning Constraint	<code>owningConstraint</code>	Constraint	ValueSpecification	<code>\$Duration[i].owningConstraint</code>
Owning Instance Spec	<code>owningInstanceSpec</code>	InstanceSpecification	ValueSpecification	<code>\$Duration[i].owningInstanceSpec</code>
Owning Lower	<code>owningLower</code>	MultiplicityElement	ValueSpecification	<code>\$Duration[i].owningLower</code>
Owning Package	<code>owningPackage</code>	Package	PackageableElement	<code>\$Duration[i].owningPackage</code>
Owning Parameter	<code>owningParameter</code>	Parameter	ValueSpecification	<code>\$Duration[i].owningParameter</code>
Owning Property	<code>owningProperty</code>	Property	ValueSpecification	<code>\$Duration[i].owningProperty</code>
Owning Slot	<code>owningSlot</code>	Slot	ValueSpecification	<code>\$Duration[i].owningSlot</code>
Owning Template Parameter	<code>owningTemplateParameter</code>	TemplateParameter	ParameterableElement	<code>\$Duration[i].owningTemplateParameter</code>
Owning Upper	<code>owningUpper</code>	MultiplicityElement	ValueSpecification	<code>\$Duration[i].owningUpper</code>
Qualified Name	<code>qualifiedName</code>	String	NamedElement	<code>\$Duration[i].qualifiedName</code>
Supplier Dependency	<code>supplierDependency</code>	Dependency[]	NamedElement	<code>\$Duration[i].supplierDependency</code>
Sync Element	<code>syncElement</code>	Element	Element	<code>\$Duration[i].syncElement</code>
Template Parameter	<code>templateParameter</code>	TemplateParameter	ParameterableElement	<code>\$Duration[i].templateParameter</code>
Type	<code>type</code>	Type	TypedElement	<code>\$Duration[i].type</code>
Visibility	<code>visibility</code>	VisibilityKind	PackageableElement	<code>\$Duration[i].visibility</code>

76. DurationConstraint

Generalization

- IntervalConstraint

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$DurationConstraint[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$DurationConstraint[i].appliedStereotypeInstance</code>
Body Context	bodyContext	Operation	Constraint	<code>\$DurationConstraint[i].bodyContext</code>
classType	classType	Class	BaseElement	<code>\$DurationConstraint[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$DurationConstraint[i].clientDependency</code>
Constrained Element	constrainedElement	Element[]	Constraint	<code>\$DurationConstraint[i].constrainedElement</code>
Context	context	Namespace	Constraint	<code>\$DurationConstraint[i].context</code>
isEditable	editable	boolean	BaseElement	<code>\$DurationConstraint[i].editable</code>
First Event	firstEvent	Boolean[]	DurationConstraint	<code>\$DurationConstraint[i].firstEvent</code>
humanName	humanName	String	BaseElement	<code>\$DurationConstraint[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$DurationConstraint[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$DurationConstraint[i].invalid</code>
localID	localID	String	ModelObject	<code>\$DurationConstraint[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$DurationConstraint[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$DurationConstraint[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$DurationConstraint[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$DurationConstraint[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$DurationConstraint[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$DurationConstraint[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$DurationConstraint[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$DurationConstraint[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$DurationConstraint[i].owningPackage</code>
Owning State	owningState	State	Constraint	<code>\$DurationConstraint[i].owningState</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$DurationConstraint[i].owningTemplateParameter</code>
Owning Transition	owningTransition	ProtocolTransition	Constraint	<code>\$DurationConstraint[i].owningTransition</code>
Post Context	postContext	Operation	Constraint	<code>\$DurationConstraint[i].postContext</code>
Pre Context	preContext	Operation	Constraint	<code>\$DurationConstraint[i].preContext</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$DurationConstraint[i].qualifiedName</code>
Specification	specification	DurationInterval	DurationConstraint	<code>\$DurationConstraint[i].specification</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$DurationConstraint[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$DurationConstraint[i].syncElement</code>
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$DurationConstraint[i].templateParameter</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$DurationConstraint[i].visibility</code>

77. DurationInterval

Generalization

- Interval

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$DurationInterval[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$DurationInterval[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$DurationInterval[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$DurationInterval[i].clientDependency
isEditable	editable	boolean	BaseElement	\$DurationInterval[i].editable
Expression	expression	Expression	ValueSpecification	\$DurationInterval[i].expression
humanName	humanName	String	BaseElement	\$DurationInterval[i].humanName
humanType	humanType	String	BaseElement	\$DurationInterval[i].humanType
isInvalid	invalid	boolean	ModelObject	\$DurationInterval[i].invalid
localID	localID	String	ModelObject	\$DurationInterval[i].localID
Max	max	Duration	DurationInterval	\$DurationInterval[i].max
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$DurationInterval[i].mdExtensions
Min	min	Duration	DurationInterval	\$DurationInterval[i].min
Name	name	String	NamedElement	\$DurationInterval[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$DurationInterval[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$DurationInterval[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$DurationInterval[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$DurationInterval[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$DurationInterval[i].ownedElement
Owner	owner	Element	Element	\$DurationInterval[i].owner
Owning Constraint	owningConstraint	Constraint	ValueSpecification	\$DurationInterval[i].owningConstraint
Owning Instance Spec	owningInstanceSpec	InstanceSpecification	ValueSpecification	\$DurationInterval[i].owningInstanceSpec
Owning Lower	owningLower	MultiplicityElement	ValueSpecification	\$DurationInterval[i].owningLower
Owning Package	owningPackage	Package	PackageableElement	\$DurationInterval[i].owningPackage
Owning Parameter	owningParameter	Parameter	ValueSpecification	\$DurationInterval[i].owningParameter
Owning Property	owningProperty	Property	ValueSpecification	\$DurationInterval[i].owningProperty
Owning Slot	owningSlot	Slot	ValueSpecification	\$DurationInterval[i].owningSlot
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$DurationInterval[i].owningTemplateParameter
Owning Upper	owningUpper	MultiplicityElement	ValueSpecification	\$DurationInterval[i].owningUpper
Qualified Name	qualifiedName	String	NamedElement	\$DurationInterval[i].qualifiedName

Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$DurationInterval[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$DurationInterval[i].syncElement</code>
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$DurationInterval[i].templateParameter</code>
Type	type	Type	TypedElement	<code>\$DurationInterval[i].type</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$DurationInterval[i].visibility</code>

78. DurationObservation

Generalization

- [Observation](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$DurationObservation[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$DurationObservation[i].appliedStereotypeInstance</code>
Behavior	behavior	Behavior	Observation	<code>\$DurationObservation[i].behavior</code>
classType	classType	Class	BaseElement	<code>\$DurationObservation[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$DurationObservation[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$DurationObservation[i].editable</code>
Event	event	NamedElement[]	DurationObservation	<code>\$DurationObservation[i].event</code>
First Event	firstEvent	Boolean[]	DurationObservation	<code>\$DurationObservation[i].firstEvent</code>
humanName	humanName	String	BaseElement	<code>\$DurationObservation[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$DurationObservation[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$DurationObservation[i].invalid</code>
localID	localID	String	ModelObject	<code>\$DurationObservation[i].localID</code>
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	<code>\$DurationObservation[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$DurationObservation[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$DurationObservation[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$DurationObservation[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$DurationObservation[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$DurationObservation[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$DurationObservation[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$DurationObservation[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$DurationObservation[i].owningPackage</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$DurationObservation[i].owningTemplateParameter</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$DurationObservation[i].qualifiedName</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$DurationObservation[i].supplierDependency</code>

				y
Sync Element	syncElement	Element	Element	\$DurationObservation[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$DurationObservation[i].templateParameter
Visibility	visibility	VisibilityKind	PackageableElement	\$DurationObservation[i].visibility

79. Element

Generalization

- [ModelObject](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Element[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Element[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Element[i].classType
isEditable	editable	boolean	BaseElement	\$Element[i].editable
humanName	humanName	String	BaseElement	\$Element[i].humanName
humanType	humanType	String	BaseElement	\$Element[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Element[i].invalid
localID	localID	String	ModelObject	\$Element[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Element[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$Element[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Element[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Element[i].ownedElement
Owner	owner	Element	Element	\$Element[i].owner
Sync Element	syncElement	Element	Element	\$Element[i].syncElement

80. ElementImport

Generalization

- [DirectedRelationship](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ElementImport[i].ID
Alias	alias	String	ElementImport	\$ElementImport[i].alias
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ElementImport[i].appliedStereotypeInstance

classType	classType	Class	BaseElement	\$ElementImport[i].classType
isEditable	editable	boolean	BaseElement	\$ElementImport[i].editable
humanName	humanName	String	BaseElement	\$ElementImport[i].humanName
humanType	humanType	String	BaseElement	\$ElementImport[i].humanType
Imported Element	importedElement	PackageableElement	ElementImport	\$ElementImport[i].importedElement
Importing Namespace	importingNamespace	Namespace	ElementImport	\$ElementImport[i].importingNamespace
isInvalid	invalid	boolean	ModelObject	\$ElementImport[i].invalid
localID	localID	String	ModelObject	\$ElementImport[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ElementImport[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$ElementImport[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$ElementImport[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ElementImport[i].ownedElement
Owner	owner	Element	Element	\$ElementImport[i].owner
Related Element	relatedElement	Element[]	Relationship	\$ElementImport[i].relatedElement
Source	source	Element[]	DirectedRelationship	\$ElementImport[i].source
Sync Element	syncElement	Element	Element	\$ElementImport[i].syncElement
Target	target	Element[]	DirectedRelationship	\$ElementImport[i].target
Visibility	visibility	VisibilityKind	ElementImport	\$ElementImport[i].visibility

81. ElementValue

Generalization

- [ValueSpecification](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ElementValue[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ElementValue[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ElementValue[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ElementValue[i].clientDependency
isEditable	editable	boolean	BaseElement	\$ElementValue[i].editable
Element	element	Element	ElementValue	\$ElementValue[i].element
Expression	expression	Expression	ValueSpecification	\$ElementValue[i].expression
humanName	humanName	String	BaseElement	\$ElementValue[i].humanName
humanType	humanType	String	BaseElement	\$ElementValue[i].humanType
isInvalid	invalid	boolean	ModelObject	\$ElementValue[i].invalid
localID	localID	String	ModelObject	\$ElementValue[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ElementValue[i].mdExtensions

Name	name	String	NamedElement	\$ElementValue[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ElementValue[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ElementValue[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ElementValue[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$ElementValue[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ElementValue[i].ownedElement
Owner	owner	Element	Element	\$ElementValue[i].owner
Owning Constraint	owningConstraint	Constraint	ValueSpecification	\$ElementValue[i].owningConstraint
Owning Instance Spec	owningInstanceSpec	InstanceSpecification	ValueSpecification	\$ElementValue[i].owningInstanceSpec
Owning Lower	owningLower	MultiplicityElement	ValueSpecification	\$ElementValue[i].owningLower
Owning Package	owningPackage	Package	PackageableElement	\$ElementValue[i].owningPackage
Owning Parameter	owningParameter	Parameter	ValueSpecification	\$ElementValue[i].owningParameter
Owning Property	owningProperty	Property	ValueSpecification	\$ElementValue[i].owningProperty
Owning Slot	owningSlot	Slot	ValueSpecification	\$ElementValue[i].owningSlot
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$ElementValue[i].owningTemplateParameter
Owning Upper	owningUpper	MultiplicityElement	ValueSpecification	\$ElementValue[i].owningUpper
Qualified Name	qualifiedName	String	NamedElement	\$ElementValue[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ElementValue[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ElementValue[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$ElementValue[i].templateParameter
Type	type	Type	TypedElement	\$ElementValue[i].type
Visibility	visibility	VisibilityKind	PackageableElement	\$ElementValue[i].visibility

82. EncapsulatedClassifier

Generalization

- [StructuredClassifier](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$EncapsulatedClassifier[i].ID
Class	UMLClass	Class	Classifier	\$EncapsulatedClassifier[i].UMLClass
Is Abstract	abstract	boolean	Classifier	\$EncapsulatedClassifier[i].abstract
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$EncapsulatedClassifier[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$EncapsulatedClassifier[i].attribute
classType	classType	Class	BaseElement	\$EncapsulatedClassifier[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$EncapsulatedClassifier[i].clientDependenc

				y
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$EncapsulatedClassifier[i].collaborationUse
isEditable	editable	boolean	BaseElement	\$EncapsulatedClassifier[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$EncapsulatedClassifier[i].elementImport
Feature	feature	Feature[]	Classifier	\$EncapsulatedClassifier[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$EncapsulatedClassifier[i].finalSpecialization
General	general	Classifier[]	Classifier	\$EncapsulatedClassifier[i].general
Generalization	generalization	Generalization[]	Classifier	\$EncapsulatedClassifier[i].generalization
humanName	humanName	String	BaseElement	\$EncapsulatedClassifier[i].humanName
humanType	humanType	String	BaseElement	\$EncapsulatedClassifier[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$EncapsulatedClassifier[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$EncapsulatedClassifier[i].inheritedMember
isInvalid	invalid	boolean	ModelObject	\$EncapsulatedClassifier[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$EncapsulatedClassifier[i].leaf
localID	localID	String	ModelObject	\$EncapsulatedClassifier[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$EncapsulatedClassifier[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$EncapsulatedClassifier[i].member
Name	name	String	NamedElement	\$EncapsulatedClassifier[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$EncapsulatedClassifier[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$EncapsulatedClassifier[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$EncapsulatedClassifier[i].objectParent
Owned Attribute	ownedAttribute	Property[]	StructuredClassifier	\$EncapsulatedClassifier[i].ownedAttribute
Owned Comment	ownedComment	Comment[]	Element	\$EncapsulatedClassifier[i].ownedComment
Owned Connector	ownedConnector	Connector[]	StructuredClassifier	\$EncapsulatedClassifier[i].ownedConnector
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$EncapsulatedClassifier[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$EncapsulatedClassifier[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$EncapsulatedClassifier[i].ownedMember
Owned Port	ownedPort	Port[]	EncapsulatedClassifier	\$EncapsulatedClassifier[i].ownedPort
Owned Rule	ownedRule	Constraint[]	Namespace	\$EncapsulatedClassifier[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$EncapsulatedClassifier[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$EncapsulatedClassifier[i].ownedUseCase
Owner	owner	Element	Element	\$EncapsulatedClassifier[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$EncapsulatedClassifier[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$EncapsulatedClassifier[i].owningTemplateParameter
Package	package	Package	Type	\$EncapsulatedClassifier[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$EncapsulatedClassifier[i].packageImport
Part	part	Property[]	StructuredClassifier	\$EncapsulatedClassifier[i].part

PowerType Extent	powerTypeExtent	GeneralizationSet[]	Classifier	<code>\$EncapsulatedClassifier[i].powerTypeExtent</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$EncapsulatedClassifier[i].qualifiedName</code>
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	<code>\$EncapsulatedClassifier[i].redefinedClassifier</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$EncapsulatedClassifier[i].redefinedElement</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$EncapsulatedClassifier[i].redefinitionContext</code>
Representation	representation	CollaborationUse	Classifier	<code>\$EncapsulatedClassifier[i].representation</code>
Role	role	ConnectableElement[]	StructuredClassifier	<code>\$EncapsulatedClassifier[i].role</code>
Substitution	substitution	Substitution[]	Classifier	<code>\$EncapsulatedClassifier[i].substitution</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$EncapsulatedClassifier[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$EncapsulatedClassifier[i].syncElement</code>
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	<code>\$EncapsulatedClassifier[i].templateBinding</code>
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	<code>\$EncapsulatedClassifier[i].templateParameter</code>
Use Case	useCase	UseCase[]	Classifier	<code>\$EncapsulatedClassifier[i].useCase</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$EncapsulatedClassifier[i].visibility</code>

83. Enumeration

Generalization

- [DataType](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$Enumeration[i].ID</code>
Class	UMLClass	Class	Classifier	<code>\$Enumeration[i].UMLClass</code>
Is Abstract	abstract	boolean	Classifier	<code>\$Enumeration[i].abstract</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Enumeration[i].appliedStereotypeInstance</code>
Attribute	attribute	Property[]	Classifier	<code>\$Enumeration[i].attribute</code>
classType	classType	Class	BaseElement	<code>\$Enumeration[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Enumeration[i].clientDependency</code>
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	<code>\$Enumeration[i].collaborationUse</code>
isEditable	editable	boolean	BaseElement	<code>\$Enumeration[i].editable</code>
Element Import	elementImport	ElementImport[]	Namespace	<code>\$Enumeration[i].elementImport</code>
Feature	feature	Feature[]	Classifier	<code>\$Enumeration[i].feature</code>
Is Final Specialization	finalSpecialization	boolean	Classifier	<code>\$Enumeration[i].finalSpecialization</code>

General	general	Classifier[]	Classifier	\$Enumeration[i].general
Generalization	generalization	Generalization[]	Classifier	\$Enumeration[i].generalization
humanName	humanName	String	BaseElement	\$Enumeration[i].humanName
humanType	humanType	String	BaseElement	\$Enumeration[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Enumeration[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$Enumeration[i].inheritedMember
isInvalid	invalid	boolean	ModelObject	\$Enumeration[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$Enumeration[i].leaf
localID	localID	String	ModelObject	\$Enumeration[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Enumeration[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Enumeration[i].member
Name	name	String	NamedElement	\$Enumeration[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Enumeration[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Enumeration[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Enumeration[i].objectParent
Owned Attribute	ownedAttribute	Property[]	DataType	\$Enumeration[i].ownedAttribute
Owned Comment	ownedComment	Comment[]	Element	\$Enumeration[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Enumeration[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Enumeration[i].ownedElement
Owned Literal	ownedLiteral	EnumerationLiteral[]	Enumeration	\$Enumeration[i].ownedLiteral
Owned Member	ownedMember	NamedElement[]	Namespace	\$Enumeration[i].ownedMember
Owned Operation	ownedOperation	Operation[]	DataType	\$Enumeration[i].ownedOperation
Owned Rule	ownedRule	Constraint[]	Namespace	\$Enumeration[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$Enumeration[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$Enumeration[i].ownedUseCase
Owner	owner	Element	Element	\$Enumeration[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Enumeration[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Enumeration[i].owningTemplateParameter
Package	package	Package	Type	\$Enumeration[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$Enumeration[i].packageImport
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$Enumeration[i].powertypeExtent
Qualified Name	qualifiedName	String	NamedElement	\$Enumeration[i].qualifiedName
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$Enumeration[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Enumeration[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$Enumeration[i].redefinitionContext
Representation	representation	CollaborationUse	Classifier	\$Enumeration[i].representation
Substitution	substitution	Substitution[]	Classifier	\$Enumeration[i].substitution
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Enumeration[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Enumeration[i].syncElement

Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$Enumeration[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$Enumeration[i].templateParameter
Use Case	useCase	UseCase[]	Classifier	\$Enumeration[i].useCase
Visibility	visibility	VisibilityKind	PackageableElement	\$Enumeration[i].visibility

84. EnumerationLiteral

Generalization

- [InstanceSpecification](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$EnumerationLiteral[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$EnumerationLiteral[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$EnumerationLiteral[i].classType
Classifier	classifier	Classifier[]	EnumerationLiteral	\$EnumerationLiteral[i].classifier
Client Dependency	clientDependency	Dependency[]	NamedElement	\$EnumerationLiteral[i].clientDependency
Deployed Element	deployedElement	PackageableElement[]	DeploymentTarget	\$EnumerationLiteral[i].deployedElement
Deployment	deployment	Deployment[]	DeploymentTarget	\$EnumerationLiteral[i].deployment
isEditable	editable	boolean	BaseElement	\$EnumerationLiteral[i].editable
Enumeration	enumeration	Enumeration	EnumerationLiteral	\$EnumerationLiteral[i].enumeration
humanName	humanName	String	BaseElement	\$EnumerationLiteral[i].humanName
humanType	humanType	String	BaseElement	\$EnumerationLiteral[i].humanType
isInvalid	invalid	boolean	ModelObject	\$EnumerationLiteral[i].invalid
localID	localID	String	ModelObject	\$EnumerationLiteral[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$EnumerationLiteral[i].mdExtensions
Name	name	String	NamedElement	\$EnumerationLiteral[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$EnumerationLiteral[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$EnumerationLiteral[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$EnumerationLiteral[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$EnumerationLiteral[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$EnumerationLiteral[i].ownedElement
Owner	owner	Element	Element	\$EnumerationLiteral[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$EnumerationLiteral[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$EnumerationLiteral[i].owningTemplateParameter
Qualified Name	qualifiedName	String	NamedElement	\$EnumerationLiteral[i].qualifiedName

Slot	slot	Slot[]	InstanceSpecification	\$EnumerationLiteral[i].slot
Specification	specification	ValueSpecification	InstanceSpecification	\$EnumerationLiteral[i].specification
Stereotyped Element	stereotypedElement	Element	InstanceSpecification	\$EnumerationLiteral[i].stereotypedElement
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$EnumerationLiteral[i].supplierDependency
Sync Element	syncElement	Element	Element	\$EnumerationLiteral[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$EnumerationLiteral[i].templateParameter
Visibility	visibility	VisibilityKind	PackageableElement	\$EnumerationLiteral[i].visibility

85. Event

Generalization

- [PackageableElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Event[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Event[i].appliedStereotypeInstance
Behavior	behavior	Behavior	Event	\$Event[i].behavior
classType	classType	Class	BaseElement	\$Event[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Event[i].clientDependency
isEditable	editable	boolean	BaseElement	\$Event[i].editable
humanName	humanName	String	BaseElement	\$Event[i].humanName
humanType	humanType	String	BaseElement	\$Event[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Event[i].invalid
localID	localID	String	ModelObject	\$Event[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Event[i].mdExtensions
Name	name	String	NamedElement	\$Event[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Event[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Event[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Event[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Event[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Event[i].ownedElement
Owner	owner	Element	Element	\$Event[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Event[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Event[i].owningTemplateParameter
Qualified Name	qualifiedName	String	NamedElement	\$Event[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Event[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Event[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$Event[i].templateParameter

Visibility	visibility	VisibilityKind	PackageableElement	\$Event[i].visibility

86. ExceptionHandler

Generalization

- [Element](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ExceptionHandler[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ExceptionHandler[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ExceptionHandler[i].classType
isEditable	editable	boolean	BaseElement	\$ExceptionHandler[i].editable
Exception Input	exceptionInput	ObjectNode	ExceptionHandler	\$ExceptionHandler[i].exceptionInput
Exception Type	exceptionType	Classifier[]	ExceptionHandler	\$ExceptionHandler[i].exceptionType
Handler Body	handlerBody	ExecutableNode	ExceptionHandler	\$ExceptionHandler[i].handlerBody
humanName	humanName	String	BaseElement	\$ExceptionHandler[i].humanName
humanType	humanType	String	BaseElement	\$ExceptionHandler[i].humanType
isInvalid	invalid	boolean	ModelObject	\$ExceptionHandler[i].invalid
localID	localID	String	ModelObject	\$ExceptionHandler[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ExceptionHandler[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$ExceptionHandler[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$ExceptionHandler[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ExceptionHandler[i].ownedElement
Owner	owner	Element	Element	\$ExceptionHandler[i].owner
Protected Node	protectedNode	ExecutableNode	ExceptionHandler	\$ExceptionHandler[i].protectedNode
Sync Element	syncElement	Element	Element	\$ExceptionHandler[i].syncElement

87. ExecutableNode

Generalization

- [ActivityNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ExecutableNode[i].ID
Activity	activity	Activity	ActivityNode	\$ExecutableNode[i].activity

Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$ExecutableNode[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$ExecutableNode[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$ExecutableNode[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$ExecutableNode[i].editable</code>
Handler	handler	ExceptionHandler[]	ExecutableNode	<code>\$ExecutableNode[i].handler</code>
humanName	humanName	String	BaseElement	<code>\$ExecutableNode[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$ExecutableNode[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$ExecutableNode[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$ExecutableNode[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$ExecutableNode[i].inPartition</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$ExecutableNode[i].inStructuredNode</code>
Incoming	incoming	ActivityEdge[]	ActivityNode	<code>\$ExecutableNode[i].incoming</code>
isInvalid	invalid	boolean	ModelObject	<code>\$ExecutableNode[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$ExecutableNode[i].leaf</code>
localID	localID	String	ModelObject	<code>\$ExecutableNode[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDObject	<code>\$ExecutableNode[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$ExecutableNode[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$ExecutableNode[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$ExecutableNode[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$ExecutableNode[i].objectParent</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$ExecutableNode[i].outgoing</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$ExecutableNode[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$ExecutableNode[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$ExecutableNode[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$ExecutableNode[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$ExecutableNode[i].redefinedElement</code>
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	<code>\$ExecutableNode[i].redefinedNode</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$ExecutableNode[i].redefinitionContext</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$ExecutableNode[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$ExecutableNode[i].syncElement</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$ExecutableNode[i].visibility</code>

88. ExecutionEnvironment

Generalization

- [Node](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$ExecutionEnvironment[i].ID</code>
Class	UMLClass	Class	Classifier	<code>\$ExecutionEnvironment[i].UMLClass</code>
Is Abstract	abstract	boolean	Class	<code>\$ExecutionEnvironment[i].abstract</code>
Is Active	active	boolean	Class	<code>\$ExecutionEnvironment[i].active</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$ExecutionEnvironment[i].appliedStereotypeInstance</code>
Attribute	attribute	Property[]	Classifier	<code>\$ExecutionEnvironment[i].attribute</code>
classType	classType	Class	BaseElement	<code>\$ExecutionEnvironment[i].classType</code>
Classifier Behavior	classifierBehavior	Behavior	BehavedClassifier	<code>\$ExecutionEnvironment[i].classifierBehavior</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$ExecutionEnvironment[i].clientDependency</code>
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	<code>\$ExecutionEnvironment[i].collaborationUse</code>
Deployed Element	deployedElement	PackageableElement[]	DeploymentTarget	<code>\$ExecutionEnvironment[i].deployedElement</code>
Deployment	deployment	Deployment[]	DeploymentTarget	<code>\$ExecutionEnvironment[i].deployment</code>
isEditable	editable	boolean	BaseElement	<code>\$ExecutionEnvironment[i].editable</code>
Element Import	elementImport	ElementImport[]	Namespace	<code>\$ExecutionEnvironment[i].elementImport</code>
Extension	extension	Extension[]	Class	<code>\$ExecutionEnvironment[i].extension</code>
Feature	feature	Feature[]	Classifier	<code>\$ExecutionEnvironment[i].feature</code>
Is Final Specialization	finalSpecialization	boolean	Classifier	<code>\$ExecutionEnvironment[i].finalSpecialization</code>
General	general	Classifier[]	Classifier	<code>\$ExecutionEnvironment[i].general</code>
Generalization	generalization	Generalization[]	Classifier	<code>\$ExecutionEnvironment[i].generalization</code>
humanName	humanName	String	BaseElement	<code>\$ExecutionEnvironment[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$ExecutionEnvironment[i].humanType</code>
Imported Member	importedMember	PackageableElement[]	Namespace	<code>\$ExecutionEnvironment[i].importedMember</code>
Inherited Member	inheritedMember	NamedElement[]	Classifier	<code>\$ExecutionEnvironment[i].inheritedMember</code>
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavedClassifier	<code>\$ExecutionEnvironment[i].interfaceRealization</code>
isInvalid	invalid	boolean	ModelObject	<code>\$ExecutionEnvironment[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$ExecutionEnvironment[i].leaf</code>
localID	localID	String	ModelObject	<code>\$ExecutionEnvironment[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$ExecutionEnvironment[i].mdExtensions</code>
Member	member	NamedElement[]	Namespace	<code>\$ExecutionEnvironment[i].member</code>
Name	name	String	NamedElement	<code>\$ExecutionEnvironment[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$ExecutionEnvironment[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$ExecutionEnvironment[i].namespace</code>
Nested Classifier	nestedClassifier	Classifier[]	Class	<code>\$ExecutionEnvironment[i].nestedClassifier</code>
Nested Node	nestedNode	Node[]	Node	<code>\$ExecutionEnvironment[i].nestedNode</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$ExecutionEnvironment[i].objectParent</code>
Owned Attribute	ownedAttribute	Property[]	Class	<code>\$ExecutionEnvironment[i].ownedAttribute</code>

Owned Behavior	ownedBehavior	Behavior[]	BehavioredClassifier	\$ExecutionEnvironment[i].ownedBehavior
Owned Comment	ownedComment	Comment[]	Element	\$ExecutionEnvironment[i].ownedComment
Owned Connector	ownedConnector	Connector[]	StructuredClassifier	\$ExecutionEnvironment[i].ownedConnector
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$ExecutionEnvironment[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$ExecutionEnvironment[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$ExecutionEnvironment[i].ownedMember
Owned Operation	ownedOperation	Operation[]	Class	\$ExecutionEnvironment[i].ownedOperation
Owned Port	ownedPort	Port[]	EncapsulatedClassifier	\$ExecutionEnvironment[i].ownedPort
Owned Reception	ownedReception	Reception[]	Class	\$ExecutionEnvironment[i].ownedReception
Owned Rule	ownedRule	Constraint[]	Namespace	\$ExecutionEnvironment[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$ExecutionEnvironment[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$ExecutionEnvironment[i].ownedUseCase
Owner	owner	Element	Element	\$ExecutionEnvironment[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$ExecutionEnvironment[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$ExecutionEnvironment[i].owningTemplateParameter
Package	package	Package	Type	\$ExecutionEnvironment[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$ExecutionEnvironment[i].packageImport
Part	part	Property[]	StructuredClassifier	\$ExecutionEnvironment[i].part
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$ExecutionEnvironment[i].powertypeExtent
Qualified Name	qualifiedName	String	NamedElement	\$ExecutionEnvironment[i].qualifiedName
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$ExecutionEnvironment[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ExecutionEnvironment[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ExecutionEnvironment[i].redefinitionContext
Representation	representation	CollaborationUse	Classifier	\$ExecutionEnvironment[i].representation
Role	role	ConnectableElement[]	StructuredClassifier	\$ExecutionEnvironment[i].role
Substitution	substitution	Substitution[]	Classifier	\$ExecutionEnvironment[i].substitution
Super Class	superClass	Class[]	Class	\$ExecutionEnvironment[i].superClass
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ExecutionEnvironment[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ExecutionEnvironment[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$ExecutionEnvironment[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$ExecutionEnvironment[i].templateParameter
Use Case	useCase	UseCase[]	Classifier	\$ExecutionEnvironment[i].useCase
Visibility	visibility	VisibilityKind	PackageableElement	\$ExecutionEnvironment[i].visibility

89. ExecutionOccurrenceSpecification

Generalization

- OccurrenceSpecification

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$ExecutionOccurrenceSpecification[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ExecutionOccurrenceSpecification[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ExecutionOccurrenceSpecification[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ExecutionOccurrenceSpecification[i].clientDependency
Covered	covered	Lifeline[]	OccurrenceSpecification	\$ExecutionOccurrenceSpecification[i].covered
isEditable	editable	boolean	BaseElement	\$ExecutionOccurrenceSpecification[i].editable
Enclosing Interaction	enclosingInteraction	Interaction	InteractionFragment	\$ExecutionOccurrenceSpecification[i].enclosingInteraction
Enclosing Operand	enclosingOperand	InteractionOperand	InteractionFragment	\$ExecutionOccurrenceSpecification[i].enclosingOperand
Execution	execution	ExecutionSpecification	ExecutionOccurrenceSpecification	\$ExecutionOccurrenceSpecification[i].execution
General Ordering	generalOrdering	GeneralOrdering[]	InteractionFragment	\$ExecutionOccurrenceSpecification[i].generalOrdering
humanName	humanName	String	BaseElement	\$ExecutionOccurrenceSpecification[i].humanName
humanType	humanType	String	BaseElement	\$ExecutionOccurrenceSpecification[i].humanType
isInvalid	invalid	boolean	ModelObject	\$ExecutionOccurrenceSpecification[i].invalid
localID	localID	String	ModelObject	\$ExecutionOccurrenceSpecification[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$ExecutionOccurrenceSpecification[i].mdExtensions
Name	name	String	NamedElement	\$ExecutionOccurrenceSpecification[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ExecutionOccurrenceSpecification[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ExecutionOccurrenceSpecification[i].name

				space
objectParent	objectParent	BaseElement	BaseElement	\$ExecutionOccurrenceSpecification[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$ExecutionOccurrenceSpecification[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ExecutionOccurrenceSpecification[i].ownedElement
Owner	owner	Element	Element	\$ExecutionOccurrenceSpecification[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ExecutionOccurrenceSpecification[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ExecutionOccurrenceSpecification[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ExecutionOccurrenceSpecification[i].syncElement
To After	toAfter	GeneralOrdering[]	OccurrenceSpecification	\$ExecutionOccurrenceSpecification[i].toAfter
To Before	toBefore	GeneralOrdering[]	OccurrenceSpecification	\$ExecutionOccurrenceSpecification[i].toBefore
Visibility	visibility	VisibilityKind	NamedElement	\$ExecutionOccurrenceSpecification[i].visibility

90. ExecutionSpecification

Generalization

- [InteractionFragment](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ExecutionSpecification[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ExecutionSpecification[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ExecutionSpecification[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ExecutionSpecification[i].clientDependency
Covered	covered	Lifeline[]	InteractionFragment	\$ExecutionSpecification[i].covered
isEditable	editable	boolean	BaseElement	\$ExecutionSpecification[i].editable
Enclosing Interaction	enclosingInteraction	Interaction	InteractionFragment	\$ExecutionSpecification[i].enclosingInteraction
Enclosing Operand	enclosingOperand	InteractionOperand	InteractionFragment	\$ExecutionSpecification[i].enclosingOperand

Finish	finish	OccurrenceSpecification	ExecutionSpecification	\$ExecutionSpecification[i].finish
General Ordering	generalOrdering	GeneralOrdering[]	InteractionFragment	\$ExecutionSpecification[i].generalOrdering
humanName	humanName	String	BaseElement	\$ExecutionSpecification[i].humanName
humanType	humanType	String	BaseElement	\$ExecutionSpecification[i].humanType
isInvalid	invalid	boolean	ModelObject	\$ExecutionSpecification[i].invalid
localID	localID	String	ModelObject	\$ExecutionSpecification[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ExecutionSpecification[i].mdExtensions
Name	name	String	NamedElement	\$ExecutionSpecification[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ExecutionSpecification[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ExecutionSpecification[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ExecutionSpecification[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$ExecutionSpecification[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ExecutionSpecification[i].ownedElement
Owner	owner	Element	Element	\$ExecutionSpecification[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ExecutionSpecification[i].qualifiedName
Start	start	OccurrenceSpecification	ExecutionSpecification	\$ExecutionSpecification[i].start
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ExecutionSpecification[i].supplierDepende ncy
Sync Element	syncElement	Element	Element	\$ExecutionSpecification[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$ExecutionSpecification[i].visibility

91. ExpansionNode

Generalization

- [ObjectNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ExpansionNode[i].ID
Activity	activity	Activity	ActivityNode	\$ExpansionNode[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ExpansionNode[i].appliedStereotypeInstan ce
classType	classType	Class	BaseElement	\$ExpansionNode[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ExpansionNode[i].clientDependency
Is Control Type	controlType	boolean	ObjectNode	\$ExpansionNode[i].controlType
isEditable	editable	boolean	BaseElement	\$ExpansionNode[i].editable
humanName	humanName	String	BaseElement	\$ExpansionNode[i].humanName
humanType	humanType	String	BaseElement	\$ExpansionNode[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ExpansionNode[i].inGroup

In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ExpansionNode[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ExpansionNode[i].inPartition
In State	inState	State[]	ObjectNode	\$ExpansionNode[i].inState
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ExpansionNode[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ExpansionNode[i].incoming
isInvalid	invalid	boolean	ModelObject	\$ExpansionNode[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ExpansionNode[i].leaf
localID	localID	String	ModelObject	\$ExpansionNode[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ExpansionNode[i].mdExtensions
Name	name	String	NamedElement	\$ExpansionNode[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ExpansionNode[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ExpansionNode[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ExpansionNode[i].objectParent
Ordering	ordering	ObjectNodeOrderingKind	ObjectNode	\$ExpansionNode[i].ordering
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ExpansionNode[i].outgoing
Owned Comment	ownedComment	Comment[]	Element	\$ExpansionNode[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ExpansionNode[i].ownedElement
Owner	owner	Element	Element	\$ExpansionNode[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ExpansionNode[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ExpansionNode[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ExpansionNode[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ExpansionNode[i].redefinitionContext
Region As Input	regionAsInput	ExpansionRegion	ExpansionNode	\$ExpansionNode[i].regionAsInput
Region As Output	regionAsOutput	ExpansionRegion	ExpansionNode	\$ExpansionNode[i].regionAsOutput
Selection	selection	Behavior	ObjectNode	\$ExpansionNode[i].selection
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ExpansionNode[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ExpansionNode[i].syncElement
Type	type	Type	TypedElement	\$ExpansionNode[i].type
Upper Bound	upperBound	ValueSpecification	ObjectNode	\$ExpansionNode[i].upperBound
Visibility	visibility	VisibilityKind	NamedElement	\$ExpansionNode[i].visibility

92. ExpansionRegion

Generalization

- [StructuredActivityNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)

ID	ID	String	MDOObject	\$ExpansionRegion[i].ID
Activity	activity	Activity	StructuredActivityNode	\$ExpansionRegion[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ExpansionRegion[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ExpansionRegion[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ExpansionRegion[i].clientDependency
Contained Edge	containedEdge	ActivityEdge[]	ActivityGroup	\$ExpansionRegion[i].containedEdge
Contained Node	containedNode	ActivityNode[]	ActivityGroup	\$ExpansionRegion[i].containedNode
Context	context	Classifier	Action	\$ExpansionRegion[i].context
Edge	edge	ActivityEdge[]	StructuredActivityNode	\$ExpansionRegion[i].edge
isEditable	editable	boolean	BaseElement	\$ExpansionRegion[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$ExpansionRegion[i].elementImport
Handler	handler	ExceptionHandler[]	ExecutableNode	\$ExpansionRegion[i].handler
humanName	humanName	String	BaseElement	\$ExpansionRegion[i].humanName
humanType	humanType	String	BaseElement	\$ExpansionRegion[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$ExpansionRegion[i].importedMember
In Activity	inActivity	Activity	ActivityGroup	\$ExpansionRegion[i].inActivity
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ExpansionRegion[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ExpansionRegion[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ExpansionRegion[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ExpansionRegion[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ExpansionRegion[i].incoming
Input	input	InputPin[]	Action	\$ExpansionRegion[i].input
Input Element	inputElement	ExpansionNode[]	ExpansionRegion	\$ExpansionRegion[i].inputElement
isInvalid	invalid	boolean	ModelObject	\$ExpansionRegion[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ExpansionRegion[i].leaf
localID	localID	String	ModelObject	\$ExpansionRegion[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$ExpansionRegion[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$ExpansionRegion[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$ExpansionRegion[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ExpansionRegion[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$ExpansionRegion[i].member
Mode	mode	ExpansionKind	ExpansionRegion	\$ExpansionRegion[i].mode
isMustIsolate	mustIsolate	boolean	StructuredActivityNode	\$ExpansionRegion[i].mustIsolate
Name	name	String	NamedElement	\$ExpansionRegion[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ExpansionRegion[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ExpansionRegion[i].namespace
Node	node	ActivityNode[]	StructuredActivityNode	\$ExpansionRegion[i].node
objectParent	objectParent	BaseElement	BaseElement	\$ExpansionRegion[i].objectParent

Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ExpansionRegion[i].outgoing
Output	output	OutputPin[]	Action	\$ExpansionRegion[i].output
Output Element	outputElement	ExpansionNode[]	ExpansionRegion	\$ExpansionRegion[i].outputElement
Owned Comment	ownedComment	Comment[]	Element	\$ExpansionRegion[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$ExpansionRegion[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$ExpansionRegion[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$ExpansionRegion[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$ExpansionRegion[i].ownedRule
Owner	owner	Element	Element	\$ExpansionRegion[i].owner
Package Import	packageImport	PackageImport[]	Namespace	\$ExpansionRegion[i].packageImport
Qualified Name	qualifiedName	String	NamedElement	\$ExpansionRegion[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ExpansionRegion[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ExpansionRegion[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ExpansionRegion[i].redefinitionContext
Structured Node Input	structuredNodeInput	InputPin[]	StructuredActivityNode	\$ExpansionRegion[i].structuredNodeInput
Structured Node Output	structuredNodeOutput	OutputPin[]	StructuredActivityNode	\$ExpansionRegion[i].structuredNodeOutput
Subgroup	subgroup	ActivityGroup[]	ActivityGroup	\$ExpansionRegion[i].subgroup
Super Group	superGroup	ActivityGroup	ActivityGroup	\$ExpansionRegion[i].superGroup
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ExpansionRegion[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ExpansionRegion[i].syncElement
Variable	variable	Variable[]	StructuredActivityNode	\$ExpansionRegion[i].variable
Visibility	visibility	VisibilityKind	NamedElement	\$ExpansionRegion[i].visibility

93. Expression

Generalization

- [ValueSpecification](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Expression[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Expression[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Expression[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Expression[i].clientDependency
isEditable	editable	boolean	BaseElement	\$Expression[i].editable
Expression	expression	Expression	ValueSpecification	\$Expression[i].expression
humanName	humanName	String	BaseElement	\$Expression[i].humanName
humanType	humanType	String	BaseElement	\$Expression[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Expression[i].invalid

localID	localID	String	ModelObject	<code>\$Expression[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$Expression[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$Expression[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Expression[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Expression[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Expression[i].objectParent</code>
Operand	operand	ValueSpecification[]	Expression	<code>\$Expression[i].operand</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Expression[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Expression[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$Expression[i].owner</code>
Owning Constraint	owningConstraint	Constraint	ValueSpecification	<code>\$Expression[i].owningConstraint</code>
Owning Instance Spec	owningInstanceSpec	InstanceSpecification	ValueSpecification	<code>\$Expression[i].owningInstanceSpec</code>
Owning Lower	owningLower	MultiplicityElement	ValueSpecification	<code>\$Expression[i].owningLower</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$Expression[i].owningPackage</code>
Owning Parameter	owningParameter	Parameter	ValueSpecification	<code>\$Expression[i].owningParameter</code>
Owning Property	owningProperty	Property	ValueSpecification	<code>\$Expression[i].owningProperty</code>
Owning Slot	owningSlot	Slot	ValueSpecification	<code>\$Expression[i].owningSlot</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$Expression[i].owningTemplateParameter</code>
Owning Upper	owningUpper	MultiplicityElement	ValueSpecification	<code>\$Expression[i].owningUpper</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Expression[i].qualifiedName</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Expression[i].supplierDependency</code>
Symbol	symbol	String	Expression	<code>\$Expression[i].symbol</code>
Sync Element	syncElement	Element	Element	<code>\$Expression[i].syncElement</code>
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$Expression[i].templateParameter</code>
Type	type	Type	TypedElement	<code>\$Expression[i].type</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$Expression[i].visibility</code>

94. Extend

Generalization

- [DirectedRelationship](#)
- [NamedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$Extend[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Extend[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$Extend[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Extend[i].clientDependency</code>

Condition	condition	Constraint	Extend	<code>\$Extend[i].condition</code>
isEditable	editable	boolean	BaseElement	<code>\$Extend[i].editable</code>
Extended Case	extendedCase	UseCase	Extend	<code>\$Extend[i].extendedCase</code>
Extension	extension	UseCase	Extend	<code>\$Extend[i].extension</code>
Extension Location	extensionLocation	ExtensionPoint[]	Extend	<code>\$Extend[i].extensionLocation</code>
humanName	humanName	String	BaseElement	<code>\$Extend[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Extend[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Extend[i].invalid</code>
localID	localID	String	ModelObject	<code>\$Extend[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$Extend[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$Extend[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Extend[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Extend[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Extend[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Extend[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Extend[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$Extend[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Extend[i].qualifiedName</code>
Related Element	relatedElement	Element[]	Relationship	<code>\$Extend[i].relatedElement</code>
Source	source	Element[]	DirectedRelationship	<code>\$Extend[i].source</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Extend[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Extend[i].syncElement</code>
Target	target	Element[]	DirectedRelationship	<code>\$Extend[i].target</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$Extend[i].visibility</code>

95. Extension

Generalization

- [Association](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$Extension[i].ID</code>
Class	UMLClass	Class	Classifier	<code>\$Extension[i].UMLClass</code>
Is Abstract	abstract	boolean	Classifier	<code>\$Extension[i].abstract</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Extension[i].appliedStereotypeInstance</code>
Attribute	attribute	Property[]	Classifier	<code>\$Extension[i].attribute</code>
classType	classType	Class	BaseElement	<code>\$Extension[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Extension[i].clientDependency</code>

Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$Extension[i].collaborationUse
Is Derived	derived	boolean	Association	\$Extension[i].derived
isEditable	editable	boolean	BaseElement	\$Extension[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$Extension[i].elementImport
End Type	endType	Type[]	Association	\$Extension[i].endType
Feature	feature	Feature[]	Classifier	\$Extension[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$Extension[i].finalSpecialization
General	general	Classifier[]	Classifier	\$Extension[i].general
Generalization	generalization	Generalization[]	Classifier	\$Extension[i].generalization
humanName	humanName	String	BaseElement	\$Extension[i].humanName
humanType	humanType	String	BaseElement	\$Extension[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Extension[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$Extension[i].inheritedMember
isInvalid	invalid	boolean	ModelObject	\$Extension[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$Extension[i].leaf
localID	localID	String	ModelObject	\$Extension[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$Extension[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Extension[i].member
Member End	memberEnd	Property[]	Association	\$Extension[i].memberEnd
Metaclass	metaclass	Class	Extension	\$Extension[i].metaclass
Name	name	String	NamedElement	\$Extension[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Extension[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Extension[i].namespace
Navigable Owned End	navigableOwnedEnd	Property[]	Association	\$Extension[i].navigableOwnedEnd
objectParent	objectParent	BaseElement	BaseElement	\$Extension[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Extension[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Extension[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Extension[i].ownedElement
Owned End	ownedEnd	Property[]	Extension	\$Extension[i].ownedEnd
Owned Member	ownedMember	NamedElement[]	Namespace	\$Extension[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$Extension[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$Extension[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$Extension[i].ownedUseCase
Owner	owner	Element	Element	\$Extension[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Extension[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Extension[i].owningTemplateParameter
Package	package	Package	Type	\$Extension[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$Extension[i].packageImport
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$Extension[i].powertypeExtent

Qualified Name	qualifiedName	String	NamedElement	\$Extension[i].qualifiedName
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$Extension[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Extension[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$Extension[i].redefinitionContext
Related Element	relatedElement	Element[]	Relationship	\$Extension[i].relatedElement
Representation	representation	CollaborationUse	Classifier	\$Extension[i].representation
Is Required	required	boolean	Extension	\$Extension[i].required
Substitution	substitution	Substitution[]	Classifier	\$Extension[i].substitution
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Extension[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Extension[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$Extension[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$Extension[i].templateParameter
Use Case	useCase	UseCase[]	Classifier	\$Extension[i].useCase
Visibility	visibility	VisibilityKind	PackageableElement	\$Extension[i].visibility

96. ExtensionEnd

Generalization

- Property

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ExtensionEnd[i].ID
Class	UMLClass	Class	Property	\$ExtensionEnd[i].UMLClass
Aggregation	aggregation	AggregationKind	Property	\$ExtensionEnd[i].aggregation
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ExtensionEnd[i].appliedStereotypeInstance
Association	association	Association	Property	\$ExtensionEnd[i].association
Association End	associationEnd	Property	Property	\$ExtensionEnd[i].associationEnd
classType	classType	Class	BaseElement	\$ExtensionEnd[i].classType
Classifier	classifier	Classifier	Property	\$ExtensionEnd[i].classifier
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ExtensionEnd[i].clientDependency
Is Composite	composite	boolean	Property	\$ExtensionEnd[i].composite
Datatype	datatype	DataType	Property	\$ExtensionEnd[i].datatype
Default Value	defaultValue	ValueSpecification	Property	\$ExtensionEnd[i].defaultValue
Deployed Element	deployedElement	PackageableElement[]	DeploymentTarget	\$ExtensionEnd[i].deployedElement
Deployment	deployment	Deployment[]	DeploymentTarget	\$ExtensionEnd[i].deployment
Is Derived	derived	boolean	Property	\$ExtensionEnd[i].derived

Is Derived Union	derivedUnion	boolean	Property	\$ExtensionEnd[i].derivedUnion
isEditable	editable	boolean	BaseElement	\$ExtensionEnd[i].editable
End	end	ConnectorEnd[]	ConnectableElement	\$ExtensionEnd[i].end
Featuring Classifier	featuringClassifier	Classifier	Feature	\$ExtensionEnd[i].featuringClassifier
humanName	humanName	String	BaseElement	\$ExtensionEnd[i].humanName
humanType	humanType	String	BaseElement	\$ExtensionEnd[i].humanType
Interface	interface	Interface	Property	\$ExtensionEnd[i].interface
isInvalid	invalid	boolean	ModelObject	\$ExtensionEnd[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ExtensionEnd[i].leaf
localID	localID	String	ModelObject	\$ExtensionEnd[i].localID
Lower	lower	int	ExtensionEnd	\$ExtensionEnd[i].lower
Lower Value	lowerValue	ValueSpecification	MultiplicityElement	\$ExtensionEnd[i].lowerValue
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ExtensionEnd[i].mdExtensions
Name	name	String	NamedElement	\$ExtensionEnd[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ExtensionEnd[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ExtensionEnd[i].namespace
isNavigable	navigable	boolean	Property	\$ExtensionEnd[i].navigable
objectParent	objectParent	BaseElement	BaseElement	\$ExtensionEnd[i].objectParent
Opposite	opposite	Property	Property	\$ExtensionEnd[i].opposite
Is Ordered	ordered	boolean	MultiplicityElement	\$ExtensionEnd[i].ordered
Owned Comment	ownedComment	Comment[]	Element	\$ExtensionEnd[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ExtensionEnd[i].ownedElement
Owner	owner	Element	Element	\$ExtensionEnd[i].owner
Owning Association	owningAssociation	Association	Property	\$ExtensionEnd[i].owningAssociation
Owning Signal	owningSignal	Signal	Property	\$ExtensionEnd[i].owningSignal
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$ExtensionEnd[i].owningTemplateParameter
Qualified Name	qualifiedName	String	NamedElement	\$ExtensionEnd[i].qualifiedName
Qualifier	qualifier	Property[]	Property	\$ExtensionEnd[i].qualifier
Is Read Only	readOnly	boolean	StructuralFeature	\$ExtensionEnd[i].readOnly
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ExtensionEnd[i].redefinedElement
Redefined Property	redefinedProperty	Property[]	Property	\$ExtensionEnd[i].redefinedProperty
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ExtensionEnd[i].redefinitionContext
Is Static	static	boolean	Feature	\$ExtensionEnd[i].static
Subsetted Property	subsettedProperty	Property[]	Property	\$ExtensionEnd[i].subsettedProperty
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ExtensionEnd[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ExtensionEnd[i].syncElement
Template Parameter	templateParameter	ConnectableElementTemplateParameter	ConnectableElement	\$ExtensionEnd[i].templateParameter
Type	type	Stereotype	ExtensionEnd	\$ExtensionEnd[i].type

Is Unique	unique	boolean	MultiplicityElement	\$ExtensionEnd[i].unique
Upper	upper	int	MultiplicityElement	\$ExtensionEnd[i].upper
Upper Value	upperValue	ValueSpecification	MultiplicityElement	\$ExtensionEnd[i].upperValue
Visibility	visibility	VisibilityKind	NamedElement	\$ExtensionEnd[i].visibility

97. ExtensionPoint

Generalization

- [RedefinableElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ExtensionPoint[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ExtensionPoint[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ExtensionPoint[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ExtensionPoint[i].clientDependency
isEditable	editable	boolean	BaseElement	\$ExtensionPoint[i].editable
Extension	extension	Extend[]	ExtensionPoint	\$ExtensionPoint[i].extension
humanName	humanName	String	BaseElement	\$ExtensionPoint[i].humanName
humanType	humanType	String	BaseElement	\$ExtensionPoint[i].humanType
isInvalid	invalid	boolean	ModelObject	\$ExtensionPoint[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ExtensionPoint[i].leaf
localID	localID	String	ModelObject	\$ExtensionPoint[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ExtensionPoint[i].mdExtensions
Name	name	String	NamedElement	\$ExtensionPoint[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ExtensionPoint[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ExtensionPoint[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ExtensionPoint[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$ExtensionPoint[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ExtensionPoint[i].ownedElement
Owner	owner	Element	Element	\$ExtensionPoint[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ExtensionPoint[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ExtensionPoint[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ExtensionPoint[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ExtensionPoint[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ExtensionPoint[i].syncElement
Use Case	useCase	UseCase	ExtensionPoint	\$ExtensionPoint[i].useCase
Visibility	visibility	VisibilityKind	NamedElement	\$ExtensionPoint[i].visibility

98. Feature

Generalization

- [RedefinableElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$Feature[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Feature[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$Feature[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Feature[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$Feature[i].editable</code>
Featuring Classifier	featuringClassifier	Classifier	Feature	<code>\$Feature[i].featuringClassifier</code>
humanName	humanName	String	BaseElement	<code>\$Feature[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Feature[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Feature[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$Feature[i].leaf</code>
localID	localID	String	ModelObject	<code>\$Feature[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$Feature[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$Feature[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Feature[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Feature[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Feature[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Feature[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Feature[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$Feature[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Feature[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$Feature[i].redefinedElement</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$Feature[i].redefinitionContext</code>
Is Static	static	boolean	Feature	<code>\$Feature[i].static</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Feature[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Feature[i].syncElement</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$Feature[i].visibility</code>

99. FinalNode

Generalization

- [ControlNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$FinalNode[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$FinalNode[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$FinalNode[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$FinalNode[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$FinalNode[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$FinalNode[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$FinalNode[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$FinalNode[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$FinalNode[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$FinalNode[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$FinalNode[i].inPartition</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$FinalNode[i].inStructuredNode</code>
Incoming	incoming	ActivityEdge[]	ActivityNode	<code>\$FinalNode[i].incoming</code>
isInvalid	invalid	boolean	ModelObject	<code>\$FinalNode[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$FinalNode[i].leaf</code>
localID	localID	String	ModelObject	<code>\$FinalNode[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$FinalNode[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$FinalNode[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$FinalNode[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$FinalNode[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$FinalNode[i].objectParent</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$FinalNode[i].outgoing</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$FinalNode[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$FinalNode[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$FinalNode[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$FinalNode[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$FinalNode[i].redefinedElement</code>
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	<code>\$FinalNode[i].redefinedNode</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$FinalNode[i].redefinitionContext</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$FinalNode[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$FinalNode[i].syncElement</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$FinalNode[i].visibility</code>

100. FinalState

Generalization

- State

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$FinalState[i].ID</code>
Applied Stereotype Instance	<code>appliedStereotypeInstance</code>	InstanceSpecification	Element	<code>\$FinalState[i].appliedStereotypeInstance</code>
classType	<code>classType</code>	Class	BaseElement	<code>\$FinalState[i].classType</code>
Client Dependency	<code>clientDependency</code>	Dependency[]	NamedElement	<code>\$FinalState[i].clientDependency</code>
Is Composite	<code>composite</code>	boolean	State	<code>\$FinalState[i].composite</code>
Connection	<code>connection</code>	ConnectionPointReference[]	State	<code>\$FinalState[i].connection</code>
Connection Point	<code>connectionPoint</code>	Pseudostate[]	State	<code>\$FinalState[i].connectionPoint</code>
Container	<code>container</code>	Region	Vertex	<code>\$FinalState[i].container</code>
Deferrable Trigger	<code>deferrableTrigger</code>	Trigger[]	State	<code>\$FinalState[i].deferrableTrigger</code>
Do Activity	<code>doActivity</code>	Behavior	State	<code>\$FinalState[i].doActivity</code>
isEditable	<code>editable</code>	boolean	BaseElement	<code>\$FinalState[i].editable</code>
Element Import	<code>elementImport</code>	ElementImport[]	Namespace	<code>\$FinalState[i].elementImport</code>
Entry	<code>entry</code>	Behavior	State	<code>\$FinalState[i].entry</code>
Exit	<code>exit</code>	Behavior	State	<code>\$FinalState[i].exit</code>
humanName	<code>humanName</code>	String	BaseElement	<code>\$FinalState[i].humanName</code>
humanType	<code>humanType</code>	String	BaseElement	<code>\$FinalState[i].humanType</code>
Imported Member	<code>importedMember</code>	PackageableElement[]	Namespace	<code>\$FinalState[i].importedMember</code>
Incoming	<code>incoming</code>	Transition[]	Vertex	<code>\$FinalState[i].incoming</code>
isInvalid	<code>invalid</code>	boolean	ModelObject	<code>\$FinalState[i].invalid</code>
Is Leaf	<code>leaf</code>	boolean	RedefinableElement	<code>\$FinalState[i].leaf</code>
localID	<code>localID</code>	String	ModelObject	<code>\$FinalState[i].localID</code>
mdExtensions	<code>mdExtensions</code>	MDExtension[]	MDOObject	<code>\$FinalState[i].mdExtensions</code>
Member	<code>member</code>	NamedElement[]	Namespace	<code>\$FinalState[i].member</code>
Name	<code>name</code>	String	NamedElement	<code>\$FinalState[i].name</code>
Name Expression	<code>nameExpression</code>	StringExpression	NamedElement	<code>\$FinalState[i].nameExpression</code>
Namespace	<code>namespace</code>	Namespace	NamedElement	<code>\$FinalState[i].namespace</code>
objectParent	<code>objectParent</code>	BaseElement	BaseElement	<code>\$FinalState[i].objectParent</code>
Is Orthogonal	<code>orthogonal</code>	boolean	State	<code>\$FinalState[i].orthogonal</code>
Outgoing	<code>outgoing</code>	Transition[]	Vertex	<code>\$FinalState[i].outgoing</code>
Owned Comment	<code>ownedComment</code>	Comment[]	Element	<code>\$FinalState[i].ownedComment</code>
Owned Diagram	<code>ownedDiagram</code>	Diagram[]	Namespace	<code>\$FinalState[i].ownedDiagram</code>
Owned Element	<code>ownedElement</code>	Element[]	Element	<code>\$FinalState[i].ownedElement</code>
Owned Member	<code>ownedMember</code>	NamedElement[]	Namespace	<code>\$FinalState[i].ownedMember</code>
Owned Rule	<code>ownedRule</code>	Constraint[]	Namespace	<code>\$FinalState[i].ownedRule</code>

Owner	owner	Element	Element	<code>\$FinalState[i].owner</code>
Package Import	packageImport	PackageImport[]	Namespace	<code>\$FinalState[i].packageImport</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$FinalState[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$FinalState[i].redefinedElement</code>
Redefined Vertex	redefinedVertex	Vertex	Vertex	<code>\$FinalState[i].redefinedVertex</code>
Redefinition Context	redefinitionContext	Classifier[]	Vertex	<code>\$FinalState[i].redefinitionContext</code>
Region	region	Region[]	State	<code>\$FinalState[i].region</code>
Is Simple	simple	boolean	State	<code>\$FinalState[i].simple</code>
State Invariant	stateInvariant	Constraint	State	<code>\$FinalState[i].stateInvariant</code>
Submachine	submachine	StateMachine	State	<code>\$FinalState[i].submachine</code>
Is Submachine State	submachineState	boolean	State	<code>\$FinalState[i].submachineState</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$FinalState[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$FinalState[i].syncElement</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$FinalState[i].visibility</code>

101. FlowFinalNode

Generalization

- [FinalNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$FlowFinalNode[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$FlowFinalNode[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$FlowFinalNode[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$FlowFinalNode[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$FlowFinalNode[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$FlowFinalNode[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$FlowFinalNode[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$FlowFinalNode[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$FlowFinalNode[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$FlowFinalNode[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$FlowFinalNode[i].inPartition</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$FlowFinalNode[i].inStructuredNode</code>
Incoming	incoming	ActivityEdge[]	ActivityNode	<code>\$FlowFinalNode[i].incoming</code>
isInvalid	invalid	boolean	ModelObject	<code>\$FlowFinalNode[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$FlowFinalNode[i].leaf</code>

localID	localID	String	ModelObject	\$FlowFinalNode[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$FlowFinalNode[i].mdExtensions
Name	name	String	NamedElement	\$FlowFinalNode[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$FlowFinalNode[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$FlowFinalNode[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$FlowFinalNode[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$FlowFinalNode[i].outgoing
Owned Comment	ownedComment	Comment[]	Element	\$FlowFinalNode[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$FlowFinalNode[i].ownedElement
Owner	owner	Element	Element	\$FlowFinalNode[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$FlowFinalNode[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$FlowFinalNode[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$FlowFinalNode[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$FlowFinalNode[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$FlowFinalNode[i].supplierDependency
Sync Element	syncElement	Element	Element	\$FlowFinalNode[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$FlowFinalNode[i].visibility

102. ForkNode

Generalization

- [ControlNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ForkNode[i].ID
Activity	activity	Activity	ActivityNode	\$ForkNode[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ForkNode[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ForkNode[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ForkNode[i].clientDependency
isEditable	editable	boolean	BaseElement	\$ForkNode[i].editable
humanName	humanName	String	BaseElement	\$ForkNode[i].humanName
humanType	humanType	String	BaseElement	\$ForkNode[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ForkNode[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ForkNode[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ForkNode[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ForkNode[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ForkNode[i].incoming

isInvalid	invalid	boolean	ModelObject	\$ForkNode[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ForkNode[i].leaf
localID	localID	String	ModelObject	\$ForkNode[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ForkNode[i].mdExtensions
Name	name	String	NamedElement	\$ForkNode[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ForkNode[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ForkNode[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ForkNode[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ForkNode[i].outgoing
Owned Comment	ownedComment	Comment[]	Element	\$ForkNode[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ForkNode[i].ownedElement
Owner	owner	Element	Element	\$ForkNode[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ForkNode[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ForkNode[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ForkNode[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ForkNode[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ForkNode[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ForkNode[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$ForkNode[i].visibility

103. FunctionBehavior

Generalization

- [OpaqueBehavior](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$FunctionBehavior[i].ID
Class	UMLClass	Class	Classifier	\$FunctionBehavior[i].UMLClass
Is Abstract	abstract	boolean	Class	\$FunctionBehavior[i].abstract
Is Active	active	boolean	Class	\$FunctionBehavior[i].active
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$FunctionBehavior[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$FunctionBehavior[i].attribute
Body	body	String[]	OpaqueBehavior	\$FunctionBehavior[i].body
classType	classType	Class	BaseElement	\$FunctionBehavior[i].classType
Classifier Behavior	classifierBehavior	Behavior	BehavioredClassifier	\$FunctionBehavior[i].classifierBehavior
Client Dependency	clientDependency	Dependency[]	NamedElement	\$FunctionBehavior[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$FunctionBehavior[i].collaborationUse

Context	context	BehavioredClassifier	Behavior	<code>\$FunctionBehavior[i].context</code>
isEditable	editable	boolean	BaseElement	<code>\$FunctionBehavior[i].editable</code>
Element Import	elementImport	ElementImport[]	Namespace	<code>\$FunctionBehavior[i].elementImport</code>
Event	event	Event[]	Behavior	<code>\$FunctionBehavior[i].event</code>
Extension	extension	Extension[]	Class	<code>\$FunctionBehavior[i].extension</code>
Feature	feature	Feature[]	Classifier	<code>\$FunctionBehavior[i].feature</code>
Is Final Specialization	finalSpecialization	boolean	Classifier	<code>\$FunctionBehavior[i].finalSpecialization</code>
General	general	Classifier[]	Classifier	<code>\$FunctionBehavior[i].general</code>
Generalization	generalization	Generalization[]	Classifier	<code>\$FunctionBehavior[i].generalization</code>
humanName	humanName	String	BaseElement	<code>\$FunctionBehavior[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$FunctionBehavior[i].humanType</code>
Imported Member	importedMember	PackageableElement[]	Namespace	<code>\$FunctionBehavior[i].importedMember</code>
Inherited Member	inheritedMember	NamedElement[]	Classifier	<code>\$FunctionBehavior[i].inheritedMember</code>
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavioredClassifier	<code>\$FunctionBehavior[i].interfaceRealization</code>
isInvalid	invalid	boolean	ModelObject	<code>\$FunctionBehavior[i].invalid</code>
Language	language	String[]	OpaqueBehavior	<code>\$FunctionBehavior[i].language</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$FunctionBehavior[i].leaf</code>
localID	localID	String	ModelObject	<code>\$FunctionBehavior[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$FunctionBehavior[i].mdExtensions</code>
Member	member	NamedElement[]	Namespace	<code>\$FunctionBehavior[i].member</code>
Name	name	String	NamedElement	<code>\$FunctionBehavior[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$FunctionBehavior[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$FunctionBehavior[i].namespace</code>
Nested Classifier	nestedClassifier	Classifier[]	Class	<code>\$FunctionBehavior[i].nestedClassifier</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$FunctionBehavior[i].objectParent</code>
Observation	observation	Observation[]	Behavior	<code>\$FunctionBehavior[i].observation</code>
Owned Attribute	ownedAttribute	Property[]	Class	<code>\$FunctionBehavior[i].ownedAttribute</code>
Owned Behavior	ownedBehavior	Behavior[]	BehavioredClassifier	<code>\$FunctionBehavior[i].ownedBehavior</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$FunctionBehavior[i].ownedComment</code>
Owned Connector	ownedConnector	Connector[]	StructuredClassifier	<code>\$FunctionBehavior[i].ownedConnector</code>
Owned Diagram	ownedDiagram	Diagram[]	Namespace	<code>\$FunctionBehavior[i].ownedDiagram</code>
Owned Element	ownedElement	Element[]	Element	<code>\$FunctionBehavior[i].ownedElement</code>
Owned Member	ownedMember	NamedElement[]	Namespace	<code>\$FunctionBehavior[i].ownedMember</code>
Owned Operation	ownedOperation	Operation[]	Class	<code>\$FunctionBehavior[i].ownedOperation</code>
Owned Parameter	ownedParameter	Parameter[]	Behavior	<code>\$FunctionBehavior[i].ownedParameter</code>
Owned Parameter Set	ownedParameterSet	ParameterSet[]	Behavior	<code>\$FunctionBehavior[i].ownedParameterSet</code>
Owned Port	ownedPort	Port[]	EncapsulatedClassifier	<code>\$FunctionBehavior[i].ownedPort</code>
Owned Reception	ownedReception	Reception[]	Class	<code>\$FunctionBehavior[i].ownedReception</code>
Owned Rule	ownedRule	Constraint[]	Namespace	<code>\$FunctionBehavior[i].ownedRule</code>
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSign	Classifier	<code>\$FunctionBehavior[i].ownedTemplateSignat</code>

		ature		ure
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$FunctionBehavior[i].ownedUseCase
Owner	owner	Element	Element	\$FunctionBehavior[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$FunctionBehavior[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$FunctionBehavior[i].owningTemplateParameter
Package	package	Package	Type	\$FunctionBehavior[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$FunctionBehavior[i].packageImport
Part	part	Property[]	StructuredClassifier	\$FunctionBehavior[i].part
Postcondition	postcondition	Constraint[]	Behavior	\$FunctionBehavior[i].postcondition
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$FunctionBehavior[i].powertypeExtent
Precondition	precondition	Constraint[]	Behavior	\$FunctionBehavior[i].precondition
Qualified Name	qualifiedName	String	NamedElement	\$FunctionBehavior[i].qualifiedName
Redefined Behavior	redefinedBehavior	Behavior[]	Behavior	\$FunctionBehavior[i].redefinedBehavior
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$FunctionBehavior[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$FunctionBehavior[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$FunctionBehavior[i].redefinitionContext
Is Reentrant	reentrant	boolean	Behavior	\$FunctionBehavior[i].reentrant
Representation	representation	CollaborationUse	Classifier	\$FunctionBehavior[i].representation
Role	role	ConnectableElement[]	StructuredClassifier	\$FunctionBehavior[i].role
Specification	specification	BehavioralFeature	Behavior	\$FunctionBehavior[i].specification
Substitution	substitution	Substitution[]	Classifier	\$FunctionBehavior[i].substitution
Super Class	superClass	Class[]	Class	\$FunctionBehavior[i].superClass
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$FunctionBehavior[i].supplierDependency
Sync Element	syncElement	Element	Element	\$FunctionBehavior[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$FunctionBehavior[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$FunctionBehavior[i].templateParameter
Use Case	useCase	UseCase[]	Classifier	\$FunctionBehavior[i].useCase
Visibility	visibility	VisibilityKind	PackageableElement	\$FunctionBehavior[i].visibility

104. Gate

Generalization

- MessageEnd

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJect	\$Gate[i].ID

Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Gate[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$Gate[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Gate[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$Gate[i].editable</code>
Formal Gate	formalGate	Gate	Gate	<code>\$Gate[i].formalGate</code>
humanName	humanName	String	BaseElement	<code>\$Gate[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Gate[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Gate[i].invalid</code>
localID	localID	String	ModelObject	<code>\$Gate[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$Gate[i].mdExtensions</code>
Message	message	Message	MessageEnd	<code>\$Gate[i].message</code>
Name	name	String	NamedElement	<code>\$Gate[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Gate[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Gate[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Gate[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Gate[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Gate[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$Gate[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Gate[i].qualifiedName</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Gate[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Gate[i].syncElement</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$Gate[i].visibility</code>

105. Generalization

Generalization

- [DirectedRelationship](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$Generalization[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Generalization[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$Generalization[i].classType</code>
isEditable	editable	boolean	BaseElement	<code>\$Generalization[i].editable</code>
General	general	Classifier	Generalization	<code>\$Generalization[i].general</code>
Generalization Set	generalizationSet	GeneralizationSet[]	Generalization	<code>\$Generalization[i].generalizationSet</code>
humanName	humanName	String	BaseElement	<code>\$Generalization[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Generalization[i].humanType</code>

isInvalid	invalid	boolean	ModelObject	<code>\$Generalization[i].invalid</code>
localID	localID	String	ModelObject	<code>\$Generalization[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$Generalization[i].mdExtensions</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Generalization[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Generalization[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Generalization[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$Generalization[i].owner</code>
Related Element	relatedElement	Element[]	Relationship	<code>\$Generalization[i].relatedElement</code>
Source	source	Element[]	DirectedRelationship	<code>\$Generalization[i].source</code>
Specific	specific	Classifier	Generalization	<code>\$Generalization[i].specific</code>
Is Substitutable	substitutable	boolean	Generalization	<code>\$Generalization[i].substitutable</code>
Sync Element	syncElement	Element	Element	<code>\$Generalization[i].syncElement</code>
Target	target	Element[]	DirectedRelationship	<code>\$Generalization[i].target</code>

106. GeneralizationSet

Generalization

- [PackageableElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$GeneralizationSet[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$GeneralizationSet[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$GeneralizationSet[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$GeneralizationSet[i].clientDependency</code>
Is Covering	covering	boolean	GeneralizationSet	<code>\$GeneralizationSet[i].covering</code>
Is Disjoint	disjoint	boolean	GeneralizationSet	<code>\$GeneralizationSet[i].disjoint</code>
isEditable	editable	boolean	BaseElement	<code>\$GeneralizationSet[i].editable</code>
Generalization	generalization	Generalization[]	GeneralizationSet	<code>\$GeneralizationSet[i].generalization</code>
humanName	humanName	String	BaseElement	<code>\$GeneralizationSet[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$GeneralizationSet[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$GeneralizationSet[i].invalid</code>
localID	localID	String	ModelObject	<code>\$GeneralizationSet[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$GeneralizationSet[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$GeneralizationSet[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$GeneralizationSet[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$GeneralizationSet[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$GeneralizationSet[i].objectParent</code>

Owned Comment	ownedComment	Comment[]	Element	\$GeneralizationSet[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$GeneralizationSet[i].ownedElement
Owner	owner	Element	Element	\$GeneralizationSet[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$GeneralizationSet[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$GeneralizationSet[i].owningTemplateParameter
Powertype	powertype	Classifier	GeneralizationSet	\$GeneralizationSet[i].powertype
Qualified Name	qualifiedName	String	NamedElement	\$GeneralizationSet[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$GeneralizationSet[i].supplierDependency
Sync Element	syncElement	Element	Element	\$GeneralizationSet[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$GeneralizationSet[i].templateParameter
Visibility	visibility	VisibilityKind	PackageableElement	\$GeneralizationSet[i].visibility

107. GeneralOrdering

Generalization

- [NamedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$GeneralOrdering[i].ID
After	after	OccurrenceSpecification	GeneralOrdering	\$GeneralOrdering[i].after
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$GeneralOrdering[i].appliedStereotypeInstance
Before	before	OccurrenceSpecification	GeneralOrdering	\$GeneralOrdering[i].before
classType	classType	Class	BaseElement	\$GeneralOrdering[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$GeneralOrdering[i].clientDependency
isEditable	editable	boolean	BaseElement	\$GeneralOrdering[i].editable
humanName	humanName	String	BaseElement	\$GeneralOrdering[i].humanName
humanType	humanType	String	BaseElement	\$GeneralOrdering[i].humanType
isInvalid	invalid	boolean	ModelObject	\$GeneralOrdering[i].invalid
localID	localID	String	ModelObject	\$GeneralOrdering[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$GeneralOrdering[i].mdExtensions
Name	name	String	NamedElement	\$GeneralOrdering[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$GeneralOrdering[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$GeneralOrdering[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$GeneralOrdering[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$GeneralOrdering[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$GeneralOrdering[i].ownedElement

Owner	owner	Element	Element	\$GeneralOrdering[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$GeneralOrdering[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$GeneralOrdering[i].supplierDependency
Sync Element	syncElement	Element	Element	\$GeneralOrdering[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$GeneralOrdering[i].visibility

108. Image

Generalization

- [Element](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$Image[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Image[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Image[i].classType
Content	content	String	Image	\$Image[i].content
isEditable	editable	boolean	BaseElement	\$Image[i].editable
Format	format	String	Image	\$Image[i].format
humanName	humanName	String	BaseElement	\$Image[i].humanName
humanType	humanType	String	BaseElement	\$Image[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Image[i].invalid
localID	localID	String	ModelObject	\$Image[i].localID
Location	location	String	Image	\$Image[i].location
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$Image[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$Image[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Image[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Image[i].ownedElement
Owner	owner	Element	Element	\$Image[i].owner
Sync Element	syncElement	Element	Element	\$Image[i].syncElement

109. Include

Generalization

- [DirectedRelationship](#)
- [NamedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class)	Sample Template Expression (VTL)

			Owner)	
ID	ID	String	MDOBJECT	\$Include[i].ID
Addition	addition	UseCase	Include	\$Include[i].addition
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Include[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Include[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Include[i].clientDependency
isEditable	editable	boolean	BaseElement	\$Include[i].editable
humanName	humanName	String	BaseElement	\$Include[i].humanName
humanType	humanType	String	BaseElement	\$Include[i].humanType
Including Case	includingCase	UseCase	Include	\$Include[i].includingCase
isInvalid	invalid	boolean	ModelObject	\$Include[i].invalid
localID	localID	String	ModelObject	\$Include[i].localID
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	\$Include[i].mdExtensions
Name	name	String	NamedElement	\$Include[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Include[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Include[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Include[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Include[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Include[i].ownedElement
Owner	owner	Element	Element	\$Include[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$Include[i].qualifiedName
Related Element	relatedElement	Element[]	Relationship	\$Include[i].relatedElement
Source	source	Element[]	DirectedRelationship	\$Include[i].source
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Include[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Include[i].syncElement
Target	target	Element[]	DirectedRelationship	\$Include[i].target
Visibility	visibility	VisibilityKind	NamedElement	\$Include[i].visibility

110. InformationFlow

Generalization

- DirectedRelationship
- PackageableElement

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$InformationFlow[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$InformationFlow[i].appliedStereotypeInstance

classType	classType	Class	BaseElement	\$InformationFlow[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$InformationFlow[i].clientDependency
Conveyed	conveyed	Classifier[]	InformationFlow	\$InformationFlow[i].conveyed
isEditable	editable	boolean	BaseElement	\$InformationFlow[i].editable
humanName	humanName	String	BaseElement	\$InformationFlow[i].humanName
humanType	humanType	String	BaseElement	\$InformationFlow[i].humanType
Information Source	informationSource	NamedElement[]	InformationFlow	\$InformationFlow[i].informationSource
Information Target	informationTarget	NamedElement[]	InformationFlow	\$InformationFlow[i].informationTarget
isInvalid	invalid	boolean	ModelObject	\$InformationFlow[i].invalid
localID	localID	String	ModelObject	\$InformationFlow[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$InformationFlow[i].mdExtensions
Name	name	String	NamedElement	\$InformationFlow[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$InformationFlow[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$InformationFlow[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$InformationFlow[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$InformationFlow[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$InformationFlow[i].ownedElement
Owner	owner	Element	Element	\$InformationFlow[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$InformationFlow[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$InformationFlow[i].owningTemplateParameter
Qualified Name	qualifiedName	String	NamedElement	\$InformationFlow[i].qualifiedName
Realization	realization	Relationship[]	InformationFlow	\$InformationFlow[i].realization
Realizing Activity Edge	realizingActivityEdge	ActivityEdge[]	InformationFlow	\$InformationFlow[i].realizingActivityEdge
Realizing Connector	realizingConnector	Connector[]	InformationFlow	\$InformationFlow[i].realizingConnector
Realizing Message	realizingMessage	Message[]	InformationFlow	\$InformationFlow[i].realizingMessage
Related Element	relatedElement	Element[]	Relationship	\$InformationFlow[i].relatedElement
Source	source	Element[]	DirectedRelationship	\$InformationFlow[i].source
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$InformationFlow[i].supplierDependency
Sync Element	syncElement	Element	Element	\$InformationFlow[i].syncElement
Target	target	Element[]	DirectedRelationship	\$InformationFlow[i].target
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$InformationFlow[i].templateParameter
Visibility	visibility	VisibilityKind	PackageableElement	\$InformationFlow[i].visibility

111. InformationItem

Generalization

- [Classifier](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$InformationItem[i].ID</code>
Class	UMLClass	Class	Classifier	<code>\$InformationItem[i].UMLClass</code>
Is Abstract	abstract	boolean	Classifier	<code>\$InformationItem[i].abstract</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$InformationItem[i].appliedStereotypeInstance</code>
Attribute	attribute	Property[]	Classifier	<code>\$InformationItem[i].attribute</code>
classType	classType	Class	BaseElement	<code>\$InformationItem[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$InformationItem[i].clientDependency</code>
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	<code>\$InformationItem[i].collaborationUse</code>
isEditable	editable	boolean	BaseElement	<code>\$InformationItem[i].editable</code>
Element Import	elementImport	ElementImport[]	Namespace	<code>\$InformationItem[i].elementImport</code>
Feature	feature	Feature[]	Classifier	<code>\$InformationItem[i].feature</code>
Is Final Specialization	finalSpecialization	boolean	Classifier	<code>\$InformationItem[i].finalSpecialization</code>
General	general	Classifier[]	Classifier	<code>\$InformationItem[i].general</code>
Generalization	generalization	Generalization[]	Classifier	<code>\$InformationItem[i].generalization</code>
humanName	humanName	String	BaseElement	<code>\$InformationItem[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$InformationItem[i].humanType</code>
Imported Member	importedMember	PackageableElement[]	Namespace	<code>\$InformationItem[i].importedMember</code>
Inherited Member	inheritedMember	NamedElement[]	Classifier	<code>\$InformationItem[i].inheritedMember</code>
isInvalid	invalid	boolean	ModelObject	<code>\$InformationItem[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$InformationItem[i].leaf</code>
localID	localID	String	ModelObject	<code>\$InformationItem[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$InformationItem[i].mdExtensions</code>
Member	member	NamedElement[]	Namespace	<code>\$InformationItem[i].member</code>
Name	name	String	NamedElement	<code>\$InformationItem[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$InformationItem[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$InformationItem[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$InformationItem[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$InformationItem[i].ownedComment</code>
Owned Diagram	ownedDiagram	Diagram[]	Namespace	<code>\$InformationItem[i].ownedDiagram</code>
Owned Element	ownedElement	Element[]	Element	<code>\$InformationItem[i].ownedElement</code>
Owned Member	ownedMember	NamedElement[]	Namespace	<code>\$InformationItem[i].ownedMember</code>
Owned Rule	ownedRule	Constraint[]	Namespace	<code>\$InformationItem[i].ownedRule</code>
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	<code>\$InformationItem[i].ownedTemplateSignature</code>
Owned Use Case	ownedUseCase	UseCase[]	Classifier	<code>\$InformationItem[i].ownedUseCase</code>
Owner	owner	Element	Element	<code>\$InformationItem[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$InformationItem[i].owningPackage</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$InformationItem[i].owningTemplateParameter</code>

				ter
Package	package	Package	Type	<code>\$InformationItem[i].package</code>
Package Import	packageImport	PackageImport[]	Namespace	<code>\$InformationItem[i].packageImport</code>
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	<code>\$InformationItem[i].powertypeExtent</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$InformationItem[i].qualifiedName</code>
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	<code>\$InformationItem[i].redefinedClassifier</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$InformationItem[i].redefinedElement</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$InformationItem[i].redefinitionContext</code>
Representation	representation	CollaborationUse	Classifier	<code>\$InformationItem[i].representation</code>
Represented	represented	Classifier[]	InformationItem	<code>\$InformationItem[i].represented</code>
Substitution	substitution	Substitution[]	Classifier	<code>\$InformationItem[i].substitution</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$InformationItem[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$InformationItem[i].syncElement</code>
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	<code>\$InformationItem[i].templateBinding</code>
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	<code>\$InformationItem[i].templateParameter</code>
Use Case	useCase	UseCase[]	Classifier	<code>\$InformationItem[i].useCase</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$InformationItem[i].visibility</code>

112. InitialNode

Generalization

- [ControlNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$InitialNode[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$InitialNode[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$InitialNode[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$InitialNode[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$InitialNode[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$InitialNode[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$InitialNode[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$InitialNode[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$InitialNode[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$InitialNode[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$InitialNode[i].inPartition</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$InitialNode[i].inStructuredNode</code>

Incoming	incoming	ActivityEdge[]	ActivityNode	\$InitialNode[i].incoming
isInvalid	invalid	boolean	ModelObject	\$InitialNode[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$InitialNode[i].leaf
localID	localID	String	ModelObject	\$InitialNode[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$InitialNode[i].mdExtensions
Name	name	String	NamedElement	\$InitialNode[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$InitialNode[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$InitialNode[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$InitialNode[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$InitialNode[i].outgoing
Owned Comment	ownedComment	Comment[]	Element	\$InitialNode[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$InitialNode[i].ownedElement
Owner	owner	Element	Element	\$InitialNode[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$InitialNode[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$InitialNode[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$InitialNode[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$InitialNode[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$InitialNode[i].supplierDependency
Sync Element	syncElement	Element	Element	\$InitialNode[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$InitialNode[i].visibility

113. InputPin

Generalization

- [Pin](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$InputPin[i].ID
Activity	activity	Activity	ActivityNode	\$InputPin[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$InputPin[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$InputPin[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$InputPin[i].clientDependency
Is Control	control	boolean	Pin	\$InputPin[i].control
Is Control Type	controlType	boolean	ObjectNode	\$InputPin[i].controlType
isEditable	editable	boolean	BaseElement	\$InputPin[i].editable
humanName	humanName	String	BaseElement	\$InputPin[i].humanName
humanType	humanType	String	BaseElement	\$InputPin[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$InputPin[i].inGroup

In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$InputPin[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$InputPin[i].inPartition
In State	inState	State[]	ObjectNode	\$InputPin[i].inState
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$InputPin[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$InputPin[i].incoming
isInvalid	invalid	boolean	ModelObject	\$InputPin[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$InputPin[i].leaf
localID	localID	String	ModelObject	\$InputPin[i].localID
Lower	lower	int	MultiplicityElement	\$InputPin[i].lower
Lower Value	lowerValue	ValueSpecification	MultiplicityElement	\$InputPin[i].lowerValue
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$InputPin[i].mdExtensions
Name	name	String	NamedElement	\$InputPin[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$InputPin[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$InputPin[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$InputPin[i].objectParent
Is Ordered	ordered	boolean	MultiplicityElement	\$InputPin[i].ordered
Ordering	ordering	ObjectNodeOrderingKind	ObjectNode	\$InputPin[i].ordering
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$InputPin[i].outgoing
Owned Comment	ownedComment	Comment[]	Element	\$InputPin[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$InputPin[i].ownedElement
Owner	owner	Element	Element	\$InputPin[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$InputPin[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$InputPin[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$InputPin[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$InputPin[i].redefinitionContext
Selection	selection	Behavior	ObjectNode	\$InputPin[i].selection
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$InputPin[i].supplierDependency
Sync Element	syncElement	Element	Element	\$InputPin[i].syncElement
Type	type	Type	TypedElement	\$InputPin[i].type
Is Unique	unique	boolean	MultiplicityElement	\$InputPin[i].unique
Upper	upper	int	MultiplicityElement	\$InputPin[i].upper
Upper Bound	upperBound	ValueSpecification	ObjectNode	\$InputPin[i].upperBound
Upper Value	upperValue	ValueSpecification	MultiplicityElement	\$InputPin[i].upperValue
Visibility	visibility	VisibilityKind	NamedElement	\$InputPin[i].visibility

114. InstanceSpecification

Generalization

- [DeployedArtifact](#)
- [DeploymentTarget](#)
- [PackageableElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$InstanceSpecification[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$InstanceSpecification[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$InstanceSpecification[i].classType</code>
Classifier	classifier	Classifier[]	InstanceSpecification	<code>\$InstanceSpecification[i].classifier</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$InstanceSpecification[i].clientDependency</code>
Deployed Element	deployedElement	PackageableElement[]	DeploymentTarget	<code>\$InstanceSpecification[i].deployedElement</code>
Deployment	deployment	Deployment[]	DeploymentTarget	<code>\$InstanceSpecification[i].deployment</code>
isEditable	editable	boolean	BaseElement	<code>\$InstanceSpecification[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$InstanceSpecification[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$InstanceSpecification[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$InstanceSpecification[i].invalid</code>
localID	localID	String	ModelObject	<code>\$InstanceSpecification[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$InstanceSpecification[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$InstanceSpecification[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$InstanceSpecification[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$InstanceSpecification[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$InstanceSpecification[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$InstanceSpecification[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$InstanceSpecification[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$InstanceSpecification[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$InstanceSpecification[i].owningPackage</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$InstanceSpecification[i].owningTemplateParameter</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$InstanceSpecification[i].qualifiedName</code>
Slot	slot	Slot[]	InstanceSpecification	<code>\$InstanceSpecification[i].slot</code>
Specification	specification	ValueSpecification	InstanceSpecification	<code>\$InstanceSpecification[i].specification</code>
Stereotyped Element	stereotypedElement	Element	InstanceSpecification	<code>\$InstanceSpecification[i].stereotypedElement</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$InstanceSpecification[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$InstanceSpecification[i].syncElement</code>
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$InstanceSpecification[i].templateParameter</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$InstanceSpecification[i].visibility</code>

115. InstanceValue

Generalization

- ValueSpecification

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$InstanceValue[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$InstanceValue[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$InstanceValue[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$InstanceValue[i].clientDependency
isEditable	editable	boolean	BaseElement	\$InstanceValue[i].editable
Expression	expression	Expression	ValueSpecification	\$InstanceValue[i].expression
humanName	humanName	String	BaseElement	\$InstanceValue[i].humanName
humanType	humanType	String	BaseElement	\$InstanceValue[i].humanType
Instance	instance	InstanceSpecification	InstanceValue	\$InstanceValue[i].instance
isInvalid	invalid	boolean	ModelObject	\$InstanceValue[i].invalid
localID	localID	String	ModelObject	\$InstanceValue[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$InstanceValue[i].mdExtensions
Name	name	String	NamedElement	\$InstanceValue[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$InstanceValue[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$InstanceValue[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$InstanceValue[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$InstanceValue[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$InstanceValue[i].ownedElement
Owner	owner	Element	Element	\$InstanceValue[i].owner
Owning Constraint	owningConstraint	Constraint	ValueSpecification	\$InstanceValue[i].owningConstraint
Owning Instance Spec	owningInstanceSpec	InstanceSpecification	ValueSpecification	\$InstanceValue[i].owningInstanceSpec
Owning Lower	owningLower	MultiplicityElement	ValueSpecification	\$InstanceValue[i].owningLower
Owning Package	owningPackage	Package	PackageableElement	\$InstanceValue[i].owningPackage
Owning Parameter	owningParameter	Parameter	ValueSpecification	\$InstanceValue[i].owningParameter
Owning Property	owningProperty	Property	ValueSpecification	\$InstanceValue[i].owningProperty
Owning Slot	owningSlot	Slot	ValueSpecification	\$InstanceValue[i].owningSlot
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$InstanceValue[i].owningTemplateParameter
Owning Upper	owningUpper	MultiplicityElement	ValueSpecification	\$InstanceValue[i].owningUpper
Qualified Name	qualifiedName	String	NamedElement	\$InstanceValue[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$InstanceValue[i].supplierDependency

Sync Element	syncElement	Element	Element	\$InstanceValue[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$InstanceValue[i].templateParameter
Type	type	Type	TypedElement	\$InstanceValue[i].type
Visibility	visibility	VisibilityKind	PackageableElement	\$InstanceValue[i].visibility

116. Interaction

Generalization

- [Behavior](#)
- [InteractionFragment](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Interaction[i].ID
Class	UMLClass	Class	Classifier	\$Interaction[i].UMLClass
Is Abstract	abstract	boolean	Class	\$Interaction[i].abstract
Action	action	Action[]	Interaction	\$Interaction[i].action
Is Active	active	boolean	Class	\$Interaction[i].active
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Interaction[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$Interaction[i].attribute
classType	classType	Class	BaseElement	\$Interaction[i].classType
Classifier Behavior	classifierBehavior	Behavior	BehavioredClassifier	\$Interaction[i].classifierBehavior
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Interaction[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$Interaction[i].collaborationUse
Context	context	BehavioredClassifier	Behavior	\$Interaction[i].context
Covered	covered	Lifeline[]	InteractionFragment	\$Interaction[i].covered
isEditable	editable	boolean	BaseElement	\$Interaction[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$Interaction[i].elementImport
Enclosing Interaction	enclosingInteraction	Interaction	InteractionFragment	\$Interaction[i].enclosingInteraction
Enclosing Operand	enclosingOperand	InteractionOperand	InteractionFragment	\$Interaction[i].enclosingOperand
Event	event	Event[]	Behavior	\$Interaction[i].event
Extension	extension	Extension[]	Class	\$Interaction[i].extension
Feature	feature	Feature[]	Classifier	\$Interaction[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$Interaction[i].finalSpecialization
Formal Gate	formalGate	Gate[]	Interaction	\$Interaction[i].formalGate
Fragment	fragment	InteractionFragment[]	Interaction	\$Interaction[i].fragment
General	general	Classifier[]	Classifier	\$Interaction[i].general
General Ordering	generalOrdering	GeneralOrdering[]	InteractionFragment	\$Interaction[i].generalOrdering
Generalization	generalization	Generalization[]	Classifier	\$Interaction[i].generalization

humanName	humanName	String	BaseElement	\$Interaction[i].humanName
humanType	humanType	String	BaseElement	\$Interaction[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Interaction[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$Interaction[i].inheritedMember
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavioredClassifier	\$Interaction[i].interfaceRealization
isInvalid	invalid	boolean	ModelObject	\$Interaction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$Interaction[i].leaf
Lifeline	lifeline	Lifeline[]	Interaction	\$Interaction[i].lifeline
localID	localID	String	ModelObject	\$Interaction[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Interaction[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Interaction[i].member
Message	message	Message[]	Interaction	\$Interaction[i].message
Name	name	String	NamedElement	\$Interaction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Interaction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Interaction[i].namespace
Nested Classifier	nestedClassifier	Classifier[]	Class	\$Interaction[i].nestedClassifier
objectParent	objectParent	BaseElement	BaseElement	\$Interaction[i].objectParent
Observation	observation	Observation[]	Behavior	\$Interaction[i].observation
Owned Attribute	ownedAttribute	Property[]	Class	\$Interaction[i].ownedAttribute
Owned Behavior	ownedBehavior	Behavior[]	BehavioredClassifier	\$Interaction[i].ownedBehavior
Owned Comment	ownedComment	Comment[]	Element	\$Interaction[i].ownedComment
Owned Connector	ownedConnector	Connector[]	StructuredClassifier	\$Interaction[i].ownedConnector
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Interaction[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Interaction[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$Interaction[i].ownedMember
Owned Operation	ownedOperation	Operation[]	Class	\$Interaction[i].ownedOperation
Owned Parameter	ownedParameter	Parameter[]	Behavior	\$Interaction[i].ownedParameter
Owned Parameter Set	ownedParameterSet	ParameterSet[]	Behavior	\$Interaction[i].ownedParameterSet
Owned Port	ownedPort	Port[]	EncapsulatedClassifier	\$Interaction[i].ownedPort
Owned Reception	ownedReception	Reception[]	Class	\$Interaction[i].ownedReception
Owned Rule	ownedRule	Constraint[]	Namespace	\$Interaction[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$Interaction[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$Interaction[i].ownedUseCase
Owner	owner	Element	Element	\$Interaction[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Interaction[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Interaction[i].owningTemplateParameter
Package	package	Package	Type	\$Interaction[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$Interaction[i].packageImport
Part	part	Property[]	StructuredClassifier	\$Interaction[i].part

Postcondition	postcondition	Constraint[]	Behavior	\$Interaction[i].postcondition
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$Interaction[i].powertypeExtent
Precondition	precondition	Constraint[]	Behavior	\$Interaction[i].precondition
Qualified Name	qualifiedName	String	NamedElement	\$Interaction[i].qualifiedName
Redefined Behavior	redefinedBehavior	Behavior[]	Behavior	\$Interaction[i].redefinedBehavior
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$Interaction[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Interaction[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$Interaction[i].redefinitionContext
Is Reentrant	reentrant	boolean	Behavior	\$Interaction[i].reentrant
Representation	representation	CollaborationUse	Classifier	\$Interaction[i].representation
Role	role	ConnectableElement[]	StructuredClassifier	\$Interaction[i].role
Specification	specification	BehavioralFeature	Behavior	\$Interaction[i].specification
Substitution	substitution	Substitution[]	Classifier	\$Interaction[i].substitution
Super Class	superClass	Class[]	Class	\$Interaction[i].superClass
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Interaction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Interaction[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$Interaction[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$Interaction[i].templateParameter
Use Case	useCase	UseCase[]	Classifier	\$Interaction[i].useCase
Visibility	visibility	VisibilityKind	PackageableElement	\$Interaction[i].visibility

117. InteractionConstraint

Generalization

- [Constraint](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$InteractionConstraint[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$InteractionConstraint[i].appliedStereotypeInstance
Body Context	bodyContext	Operation	Constraint	\$InteractionConstraint[i].bodyContext
classType	classType	Class	BaseElement	\$InteractionConstraint[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$InteractionConstraint[i].clientDependency
Constrained Element	constrainedElement	Element[]	Constraint	\$InteractionConstraint[i].constrainedElement
Context	context	Namespace	Constraint	\$InteractionConstraint[i].context
isEditable	editable	boolean	BaseElement	\$InteractionConstraint[i].editable

humanName	humanName	String	BaseElement	\$InteractionConstraint[i].humanName
humanType	humanType	String	BaseElement	\$InteractionConstraint[i].humanType
isInvalid	invalid	boolean	ModelObject	\$InteractionConstraint[i].invalid
localID	localID	String	ModelObject	\$InteractionConstraint[i].localID
Maxint	maxint	ValueSpecification	InteractionConstraint	\$InteractionConstraint[i].maxint
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$InteractionConstraint[i].mdExtensions
Minint	minint	ValueSpecification	InteractionConstraint	\$InteractionConstraint[i].minint
Name	name	String	NamedElement	\$InteractionConstraint[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$InteractionConstraint[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$InteractionConstraint[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$InteractionConstraint[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$InteractionConstraint[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$InteractionConstraint[i].ownedElement
Owner	owner	Element	Element	\$InteractionConstraint[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$InteractionConstraint[i].owningPackage
Owning State	owningState	State	Constraint	\$InteractionConstraint[i].owningState
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$InteractionConstraint[i].owningTemplateParameter
Owning Transition	owningTransition	ProtocolTransition	Constraint	\$InteractionConstraint[i].owningTransition
Post Context	postContext	Operation	Constraint	\$InteractionConstraint[i].postContext
Pre Context	preContext	Operation	Constraint	\$InteractionConstraint[i].preContext
Qualified Name	qualifiedName	String	NamedElement	\$InteractionConstraint[i].qualifiedName
Specification	specification	ValueSpecification	Constraint	\$InteractionConstraint[i].specification
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$InteractionConstraint[i].supplierDependency
Sync Element	syncElement	Element	Element	\$InteractionConstraint[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$InteractionConstraint[i].templateParameter
Visibility	visibility	VisibilityKind	PackageableElement	\$InteractionConstraint[i].visibility

118. InteractionFragment

Generalization

- [NamedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$InteractionFragment[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$InteractionFragment[i].appliedStereotypeInstance

classType	classType	Class	BaseElement	\$InteractionFragment[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$InteractionFragment[i].clientDependency
Covered	covered	Lifeline[]	InteractionFragment	\$InteractionFragment[i].covered
isEditable	editable	boolean	BaseElement	\$InteractionFragment[i].editable
Enclosing Interaction	enclosingInteraction	Interaction	InteractionFragment	\$InteractionFragment[i].enclosingInteraction
Enclosing Operand	enclosingOperand	InteractionOperand	InteractionFragment	\$InteractionFragment[i].enclosingOperand
General Ordering	generalOrdering	GeneralOrdering[]	InteractionFragment	\$InteractionFragment[i].generalOrdering
humanName	humanName	String	BaseElement	\$InteractionFragment[i].humanName
humanType	humanType	String	BaseElement	\$InteractionFragment[i].humanType
isInvalid	invalid	boolean	ModelObject	\$InteractionFragment[i].invalid
localID	localID	String	ModelObject	\$InteractionFragment[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$InteractionFragment[i].mdExtensions
Name	name	String	NamedElement	\$InteractionFragment[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$InteractionFragment[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$InteractionFragment[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$InteractionFragment[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$InteractionFragment[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$InteractionFragment[i].ownedElement
Owner	owner	Element	Element	\$InteractionFragment[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$InteractionFragment[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$InteractionFragment[i].supplierDependency
Sync Element	syncElement	Element	Element	\$InteractionFragment[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$InteractionFragment[i].visibility

119. InteractionOperand

Generalization

- [InteractionFragment](#)
- [Namespace](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$InteractionOperand[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$InteractionOperand[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$InteractionOperand[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$InteractionOperand[i].clientDependency
Covered	covered	Lifeline[]	InteractionFragment	\$InteractionOperand[i].covered

isEditable	editable	boolean	BaseElement	\$InteractionOperand[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$InteractionOperand[i].elementImport
Enclosing Interaction	enclosingInteraction	Interaction	InteractionFragment	\$InteractionOperand[i].enclosingInteraction
Enclosing Operand	enclosingOperand	InteractionOperand	InteractionFragment	\$InteractionOperand[i].enclosingOperand
Fragment	fragment	InteractionFragment[]	InteractionOperand	\$InteractionOperand[i].fragment
General Ordering	generalOrdering	GeneralOrdering[]	InteractionFragment	\$InteractionOperand[i].generalOrdering
Guard	guard	InteractionConstraint	InteractionOperand	\$InteractionOperand[i].guard
humanName	humanName	String	BaseElement	\$InteractionOperand[i].humanName
humanType	humanType	String	BaseElement	\$InteractionOperand[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$InteractionOperand[i].importedMember
isInvalid	invalid	boolean	ModelObject	\$InteractionOperand[i].invalid
localID	localID	String	ModelObject	\$InteractionOperand[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$InteractionOperand[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$InteractionOperand[i].member
Name	name	String	NamedElement	\$InteractionOperand[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$InteractionOperand[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$InteractionOperand[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$InteractionOperand[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$InteractionOperand[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$InteractionOperand[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$InteractionOperand[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$InteractionOperand[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$InteractionOperand[i].ownedRule
Owner	owner	Element	Element	\$InteractionOperand[i].owner
Package Import	packageImport	PackageImport[]	Namespace	\$InteractionOperand[i].packageImport
Qualified Name	qualifiedName	String	NamedElement	\$InteractionOperand[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$InteractionOperand[i].supplierDependency
Sync Element	syncElement	Element	Element	\$InteractionOperand[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$InteractionOperand[i].visibility

120. InteractionUse

Generalization

- [InteractionFragment](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$InteractionUse[i].ID
Actual Gate	actualGate	Gate[]	InteractionUse	\$InteractionUse[i].actualGate

Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$InteractionUse[i].appliedStereotypeInstance
Argument	argument	ValueSpecification[]	InteractionUse	\$InteractionUse[i].argument
classType	classType	Class	BaseElement	\$InteractionUse[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$InteractionUse[i].clientDependency
Covered	covered	Lifeline[]	InteractionFragment	\$InteractionUse[i].covered
isEditable	editable	boolean	BaseElement	\$InteractionUse[i].editable
Enclosing Interaction	enclosingInteraction	Interaction	InteractionFragment	\$InteractionUse[i].enclosingInteraction
Enclosing Operand	enclosingOperand	InteractionOperand	InteractionFragment	\$InteractionUse[i].enclosingOperand
General Ordering	generalOrdering	GeneralOrdering[]	InteractionFragment	\$InteractionUse[i].generalOrdering
humanName	humanName	String	BaseElement	\$InteractionUse[i].humanName
humanType	humanType	String	BaseElement	\$InteractionUse[i].humanType
isInvalid	invalid	boolean	ModelObject	\$InteractionUse[i].invalid
localID	localID	String	ModelObject	\$InteractionUse[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$InteractionUse[i].mdExtensions
Name	name	String	NamedElement	\$InteractionUse[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$InteractionUse[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$InteractionUse[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$InteractionUse[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$InteractionUse[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$InteractionUse[i].ownedElement
Owner	owner	Element	Element	\$InteractionUse[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$InteractionUse[i].qualifiedName
Refers To	refersTo	Interaction	InteractionUse	\$InteractionUse[i].refersTo
Return Value	returnValue	ValueSpecification	InteractionUse	\$InteractionUse[i].returnValue
Return Value Recipient	returnValueRecipient	Property	InteractionUse	\$InteractionUse[i].returnValueRecipient
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$InteractionUse[i].supplierDependency
Sync Element	syncElement	Element	Element	\$InteractionUse[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$InteractionUse[i].visibility

121. Interface

Generalization

- [Classifier](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$Interface[i].ID
Class	UMLClass	Class	Classifier	\$Interface[i].UMLClass

Is Abstract	abstract	boolean	Classifier	\$Interface[i].abstract
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Interface[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$Interface[i].attribute
classType	classType	Class	BaseElement	\$Interface[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Interface[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$Interface[i].collaborationUse
isEditable	editable	boolean	BaseElement	\$Interface[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$Interface[i].elementImport
Feature	feature	Feature[]	Classifier	\$Interface[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$Interface[i].finalSpecialization
General	general	Classifier[]	Classifier	\$Interface[i].general
Generalization	generalization	Generalization[]	Classifier	\$Interface[i].generalization
humanName	humanName	String	BaseElement	\$Interface[i].humanName
humanType	humanType	String	BaseElement	\$Interface[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Interface[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$Interface[i].inheritedMember
isInvalid	invalid	boolean	ModelObject	\$Interface[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$Interface[i].leaf
localID	localID	String	ModelObject	\$Interface[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$Interface[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Interface[i].member
Name	name	String	NamedElement	\$Interface[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Interface[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Interface[i].namespace
Nested Classifier	nestedClassifier	Classifier[]	Interface	\$Interface[i].nestedClassifier
objectParent	objectParent	BaseElement	BaseElement	\$Interface[i].objectParent
Owned Attribute	ownedAttribute	Property[]	Interface	\$Interface[i].ownedAttribute
Owned Comment	ownedComment	Comment[]	Element	\$Interface[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Interface[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Interface[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$Interface[i].ownedMember
Owned Operation	ownedOperation	Operation[]	Interface	\$Interface[i].ownedOperation
Owned Reception	ownedReception	Reception[]	Interface	\$Interface[i].ownedReception
Owned Rule	ownedRule	Constraint[]	Namespace	\$Interface[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$Interface[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$Interface[i].ownedUseCase
Owner	owner	Element	Element	\$Interface[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Interface[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Interface[i].owningTemplateParameter

Package	package	Package	Type	\$Interface[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$Interface[i].packageImport
PowerType Extent	powerTypeExtent	GeneralizationSet[]	Classifier	\$Interface[i].powerTypeExtent
Protocol	protocol	ProtocolStateMachine	Interface	\$Interface[i].protocol
Qualified Name	qualifiedName	String	NamedElement	\$Interface[i].qualifiedName
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$Interface[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Interface[i].redefinedElement
Redefined Interface	redefinedInterface	Interface[]	Interface	\$Interface[i].redefinedInterface
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$Interface[i].redefinitionContext
Representation	representation	CollaborationUse	Classifier	\$Interface[i].representation
Substitution	substitution	Substitution[]	Classifier	\$Interface[i].substitution
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Interface[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Interface[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$Interface[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$Interface[i].templateParameter
Use Case	useCase	UseCase[]	Classifier	\$Interface[i].useCase
Visibility	visibility	VisibilityKind	PackageableElement	\$Interface[i].visibility

122. InterfaceRealization

Generalization

- Realization

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$InterfaceRealization[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$InterfaceRealization[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$InterfaceRealization[i].classType
Client	client	NamedElement[]	Dependency	\$InterfaceRealization[i].client
Client Dependency	clientDependency	Dependency[]	NamedElement	\$InterfaceRealization[i].clientDependency
Contract	contract	Interface	InterfaceRealization	\$InterfaceRealization[i].contract
isEditable	editable	boolean	BaseElement	\$InterfaceRealization[i].editable
humanName	humanName	String	BaseElement	\$InterfaceRealization[i].humanName
humanType	humanType	String	BaseElement	\$InterfaceRealization[i].humanType
Implementing Classifier	implementingClassifier	BehavioredClassifier	InterfaceRealization	\$InterfaceRealization[i].implementingClassifier
isInvalid	invalid	boolean	ModelObject	\$InterfaceRealization[i].invalid

localID	localID	String	ModelObject	\$InterfaceRealization[i].localID
Mapping	mapping	OpaqueExpression	Abstraction	\$InterfaceRealization[i].mapping
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$InterfaceRealization[i].mdExtensions
Name	name	String	NamedElement	\$InterfaceRealization[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$InterfaceRealization[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$InterfaceRealization[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$InterfaceRealization[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$InterfaceRealization[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$InterfaceRealization[i].ownedElement
Owner	owner	Element	Element	\$InterfaceRealization[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$InterfaceRealization[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$InterfaceRealization[i].owningTemplateParameter
Qualified Name	qualifiedName	String	NamedElement	\$InterfaceRealization[i].qualifiedName
Related Element	relatedElement	Element[]	Relationship	\$InterfaceRealization[i].relatedElement
Source	source	Element[]	DirectedRelationship	\$InterfaceRealization[i].source
Supplier	supplier	NamedElement[]	Dependency	\$InterfaceRealization[i].supplier
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$InterfaceRealization[i].supplierDependency
Sync Element	syncElement	Element	Element	\$InterfaceRealization[i].syncElement
Target	target	Element[]	DirectedRelationship	\$InterfaceRealization[i].target
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$InterfaceRealization[i].templateParameter
Visibility	visibility	VisibilityKind	PackageableElement	\$InterfaceRealization[i].visibility

123. InterruptibleActivityRegion

Generalization

- [ActivityGroup](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$InterruptibleActivityRegion[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$InterruptibleActivityRegion[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$InterruptibleActivityRegion[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$InterruptibleActivityRegion[i].clientDependency
Contained Edge	containedEdge	ActivityEdge[]	ActivityGroup	\$InterruptibleActivityRegion[i].containedEdge
Contained Node	containedNode	ActivityNode[]	ActivityGroup	\$InterruptibleActivityRegion[i].containedNode

				e
isEditable	editable	boolean	BaseElement	\$InterruptibleActivityRegion[i].editable
humanName	humanName	String	BaseElement	\$InterruptibleActivityRegion[i].humanName
humanType	humanType	String	BaseElement	\$InterruptibleActivityRegion[i].humanType
In Activity	inActivity	Activity	ActivityGroup	\$InterruptibleActivityRegion[i].inActivity
Interrupting Edge	interruptingEdge	ActivityEdge[]	InterruptibleActivityRegion	\$InterruptibleActivityRegion[i].interruptingEdge
isInvalid	invalid	boolean	ModelObject	\$InterruptibleActivityRegion[i].invalid
localID	localID	String	ModelObject	\$InterruptibleActivityRegion[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$InterruptibleActivityRegion[i].mdExtensions
Name	name	String	NamedElement	\$InterruptibleActivityRegion[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$InterruptibleActivityRegion[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$InterruptibleActivityRegion[i].namespace
Node	node	ActivityNode[]	InterruptibleActivityRegion	\$InterruptibleActivityRegion[i].node
objectParent	objectParent	BaseElement	BaseElement	\$InterruptibleActivityRegion[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$InterruptibleActivityRegion[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$InterruptibleActivityRegion[i].ownedElement
Owner	owner	Element	Element	\$InterruptibleActivityRegion[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$InterruptibleActivityRegion[i].qualifiedName
Subgroup	subgroup	ActivityGroup[]	ActivityGroup	\$InterruptibleActivityRegion[i].subgroup
Super Group	superGroup	ActivityGroup	ActivityGroup	\$InterruptibleActivityRegion[i].superGroup
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$InterruptibleActivityRegion[i].supplierDependency
Sync Element	syncElement	Element	Element	\$InterruptibleActivityRegion[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$InterruptibleActivityRegion[i].visibility

124. Interval

Generalization

- ValueSpecification

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Interval[i].ID

Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Interval[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Interval[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Interval[i].clientDependency
isEditable	editable	boolean	BaseElement	\$Interval[i].editable
Expression	expression	Expression	ValueSpecification	\$Interval[i].expression
humanName	humanName	String	BaseElement	\$Interval[i].humanName
humanType	humanType	String	BaseElement	\$Interval[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Interval[i].invalid
localID	localID	String	ModelObject	\$Interval[i].localID
Max	max	ValueSpecification	Interval	\$Interval[i].max
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$Interval[i].mdExtensions
Min	min	ValueSpecification	Interval	\$Interval[i].min
Name	name	String	NamedElement	\$Interval[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Interval[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Interval[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Interval[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Interval[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Interval[i].ownedElement
Owner	owner	Element	Element	\$Interval[i].owner
Owning Constraint	owningConstraint	Constraint	ValueSpecification	\$Interval[i].owningConstraint
Owning Instance Spec	owningInstanceSpec	InstanceSpecification	ValueSpecification	\$Interval[i].owningInstanceSpec
Owning Lower	owningLower	MultiplicityElement	ValueSpecification	\$Interval[i].owningLower
Owning Package	owningPackage	Package	PackageableElement	\$Interval[i].owningPackage
Owning Parameter	owningParameter	Parameter	ValueSpecification	\$Interval[i].owningParameter
Owning Property	owningProperty	Property	ValueSpecification	\$Interval[i].owningProperty
Owning Slot	owningSlot	Slot	ValueSpecification	\$Interval[i].owningSlot
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Interval[i].owningTemplateParameter
Owning Upper	owningUpper	MultiplicityElement	ValueSpecification	\$Interval[i].owningUpper
Qualified Name	qualifiedName	String	NamedElement	\$Interval[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Interval[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Interval[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$Interval[i].templateParameter
Type	type	Type	TypedElement	\$Interval[i].type
Visibility	visibility	VisibilityKind	PackageableElement	\$Interval[i].visibility

125. IntervalConstraint

Generalization

- [Constraint](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$IntervalConstraint[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$IntervalConstraint[i].appliedStereotypeInstance</code>
Body Context	bodyContext	Operation	Constraint	<code>\$IntervalConstraint[i].bodyContext</code>
classType	classType	Class	BaseElement	<code>\$IntervalConstraint[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$IntervalConstraint[i].clientDependency</code>
Constrained Element	constrainedElement	Element[]	Constraint	<code>\$IntervalConstraint[i].constrainedElement</code>
Context	context	Namespace	Constraint	<code>\$IntervalConstraint[i].context</code>
isEditable	editable	boolean	BaseElement	<code>\$IntervalConstraint[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$IntervalConstraint[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$IntervalConstraint[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$IntervalConstraint[i].invalid</code>
localID	localID	String	ModelObject	<code>\$IntervalConstraint[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$IntervalConstraint[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$IntervalConstraint[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$IntervalConstraint[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$IntervalConstraint[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$IntervalConstraint[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$IntervalConstraint[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$IntervalConstraint[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$IntervalConstraint[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$IntervalConstraint[i].owningPackage</code>
Owning State	owningState	State	Constraint	<code>\$IntervalConstraint[i].owningState</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$IntervalConstraint[i].owningTemplateParameter</code>
Owning Transition	owningTransition	ProtocolTransition	Constraint	<code>\$IntervalConstraint[i].owningTransition</code>
Post Context	postContext	Operation	Constraint	<code>\$IntervalConstraint[i].postContext</code>
Pre Context	preContext	Operation	Constraint	<code>\$IntervalConstraint[i].preContext</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$IntervalConstraint[i].qualifiedName</code>
Specification	specification	Interval	IntervalConstraint	<code>\$IntervalConstraint[i].specification</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$IntervalConstraint[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$IntervalConstraint[i].syncElement</code>
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$IntervalConstraint[i].templateParameter</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$IntervalConstraint[i].visibility</code>

126. InvocationAction

Generalization

- Action

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$InvocationAction[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$InvocationAction[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$InvocationAction[i].appliedStereotypeInstance</code>
Argument	argument	InputPin[]	InvocationAction	<code>\$InvocationAction[i].argument</code>
classType	classType	Class	BaseElement	<code>\$InvocationAction[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$InvocationAction[i].clientDependency</code>
Context	context	Classifier	Action	<code>\$InvocationAction[i].context</code>
isEditable	editable	boolean	BaseElement	<code>\$InvocationAction[i].editable</code>
Handler	handler	ExceptionHandler[]	ExecutableNode	<code>\$InvocationAction[i].handler</code>
humanName	humanName	String	BaseElement	<code>\$InvocationAction[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$InvocationAction[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$InvocationAction[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$InvocationAction[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$InvocationAction[i].inPartition</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$InvocationAction[i].inStructuredNode</code>
Incoming	incoming	ActivityEdge[]	ActivityNode	<code>\$InvocationAction[i].incoming</code>
Input	input	InputPin[]	Action	<code>\$InvocationAction[i].input</code>
isInvalid	invalid	boolean	ModelObject	<code>\$InvocationAction[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$InvocationAction[i].leaf</code>
localID	localID	String	ModelObject	<code>\$InvocationAction[i].localID</code>
Local Postcondition	localPostcondition	Constraint[]	Action	<code>\$InvocationAction[i].localPostcondition</code>
Local Precondition	localPrecondition	Constraint[]	Action	<code>\$InvocationAction[i].localPrecondition</code>
Is Locally Reentrant	locallyReentrant	boolean	Action	<code>\$InvocationAction[i].locallyReentrant</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$InvocationAction[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$InvocationAction[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$InvocationAction[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$InvocationAction[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$InvocationAction[i].objectParent</code>
On Port	onPort	Port	InvocationAction	<code>\$InvocationAction[i].onPort</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$InvocationAction[i].outgoing</code>
Output	output	OutputPin[]	Action	<code>\$InvocationAction[i].output</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$InvocationAction[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$InvocationAction[i].ownedElement</code>

Owner	owner	Element	Element	<code>\$InvocationAction[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$InvocationAction[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$InvocationAction[i].redefinedElement</code>
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	<code>\$InvocationAction[i].redefinedNode</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$InvocationAction[i].redefinitionContext</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$InvocationAction[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$InvocationAction[i].syncElement</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$InvocationAction[i].visibility</code>

127. java.lang.Object

Generalization

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)

128. JoinNode

Generalization

- [ControlNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$JoinNode[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$JoinNode[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$JoinNode[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$JoinNode[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$JoinNode[i].clientDependency</code>
Is Combine Duplicate	combineDuplicate	boolean	JoinNode	<code>\$JoinNode[i].combineDuplicate</code>
isEditable	editable	boolean	BaseElement	<code>\$JoinNode[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$JoinNode[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$JoinNode[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$JoinNode[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$JoinNode[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$JoinNode[i].inPartition</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$JoinNode[i].inStructuredNode</code>
Incoming	incoming	ActivityEdge[]	ActivityNode	<code>\$JoinNode[i].incoming</code>

isInvalid	invalid	boolean	ModelObject	\$JoinNode[i].invalid
Join Spec	joinSpec	ValueSpecification	JoinNode	\$JoinNode[i].joinSpec
Is Leaf	leaf	boolean	RedefinableElement	\$JoinNode[i].leaf
localID	localID	String	ModelObject	\$JoinNode[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$JoinNode[i].mdExtensions
Name	name	String	NamedElement	\$JoinNode[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$JoinNode[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$JoinNode[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$JoinNode[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$JoinNode[i].outgoing
Owned Comment	ownedComment	Comment[]	Element	\$JoinNode[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$JoinNode[i].ownedElement
Owner	owner	Element	Element	\$JoinNode[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$JoinNode[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$JoinNode[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$JoinNode[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$JoinNode[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$JoinNode[i].supplierDependency
Sync Element	syncElement	Element	Element	\$JoinNode[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$JoinNode[i].visibility

129. Lifeline

Generalization

- [NamedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Lifeline[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Lifeline[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Lifeline[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Lifeline[i].clientDependency
Covered By	coveredBy	InteractionFragment[]	Lifeline	\$Lifeline[i].coveredBy
Decomposed As	decomposedAs	PartDecomposition	Lifeline	\$Lifeline[i].decomposedAs
isEditable	editable	boolean	BaseElement	\$Lifeline[i].editable
humanName	humanName	String	BaseElement	\$Lifeline[i].humanName
humanType	humanType	String	BaseElement	\$Lifeline[i].humanType
Interaction	interaction	Interaction	Lifeline	\$Lifeline[i].interaction
isInvalid	invalid	boolean	ModelObject	\$Lifeline[i].invalid

localID	localID	String	ModelObject	\$Lifeline[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Lifeline[i].mdExtensions
Name	name	String	NamedElement	\$Lifeline[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Lifeline[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Lifeline[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Lifeline[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Lifeline[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Lifeline[i].ownedElement
Owner	owner	Element	Element	\$Lifeline[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$Lifeline[i].qualifiedName
Represents	represents	ConnectableElement	Lifeline	\$Lifeline[i].represents
Selector	selector	ValueSpecification	Lifeline	\$Lifeline[i].selector
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Lifeline[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Lifeline[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$Lifeline[i].visibility

130. LinkAction

Generalization

- [Action](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$LinkAction[i].ID
Activity	activity	Activity	ActivityNode	\$LinkAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$LinkAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$LinkAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$LinkAction[i].clientDependency
Context	context	Classifier	Action	\$LinkAction[i].context
isEditable	editable	boolean	BaseElement	\$LinkAction[i].editable
End Data	endData	LinkEndData[]	LinkAction	\$LinkAction[i].endData
Handler	handler	ExceptionHandler[]	ExecutableNode	\$LinkAction[i].handler
humanName	humanName	String	BaseElement	\$LinkAction[i].humanName
humanType	humanType	String	BaseElement	\$LinkAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$LinkAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$LinkAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$LinkAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$LinkAction[i].inStructuredNode

Incoming	incoming	ActivityEdge[]	ActivityNode	\$LinkAction[i].incoming
Input	input	InputPin[]	Action	\$LinkAction[i].input
Input Value	inputValue	InputPin[]	LinkAction	\$LinkAction[i].inputValue
isInvalid	invalid	boolean	ModelObject	\$LinkAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$LinkAction[i].leaf
localID	localID	String	ModelObject	\$LinkAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$LinkAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$LinkAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$LinkAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$LinkAction[i].mdExtensions
Name	name	String	NamedElement	\$LinkAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$LinkAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$LinkAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$LinkAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$LinkAction[i].outgoing
Output	output	OutputPin[]	Action	\$LinkAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$LinkAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$LinkAction[i].ownedElement
Owner	owner	Element	Element	\$LinkAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$LinkAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$LinkAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$LinkAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$LinkAction[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$LinkAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$LinkAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$LinkAction[i].visibility

131. LinkEndCreationData

Generalization

- [LinkEndData](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDObject	\$LinkEndCreationData[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$LinkEndCreationData[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$LinkEndCreationData[i].classType
isEditable	editable	boolean	BaseElement	\$LinkEndCreationData[i].editable

End	end	Property	LinkEndData	<code>\$LinkEndCreationData[i].end</code>
humanName	humanName	String	BaseElement	<code>\$LinkEndCreationData[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$LinkEndCreationData[i].humanType</code>
Insert At	insertAt	InputPin	LinkEndCreationData	<code>\$LinkEndCreationData[i].insertAt</code>
isInvalid	invalid	boolean	ModelObject	<code>\$LinkEndCreationData[i].invalid</code>
localID	localID	String	ModelObject	<code>\$LinkEndCreationData[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$LinkEndCreationData[i].mdExtensions</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$LinkEndCreationData[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$LinkEndCreationData[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$LinkEndCreationData[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$LinkEndCreationData[i].owner</code>
Qualifier	qualifier	QualifierValue[]	LinkEndData	<code>\$LinkEndCreationData[i].qualifier</code>
Is Replace All	replaceAll	boolean	LinkEndCreationData	<code>\$LinkEndCreationData[i].replaceAll</code>
Sync Element	syncElement	Element	Element	<code>\$LinkEndCreationData[i].syncElement</code>
Value	value	InputPin	LinkEndData	<code>\$LinkEndCreationData[i].value</code>

132. LinkEndData

Generalization

- [Element](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$LinkEndData[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$LinkEndData[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$LinkEndData[i].classType</code>
isEditable	editable	boolean	BaseElement	<code>\$LinkEndData[i].editable</code>
End	end	Property	LinkEndData	<code>\$LinkEndData[i].end</code>
humanName	humanName	String	BaseElement	<code>\$LinkEndData[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$LinkEndData[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$LinkEndData[i].invalid</code>
localID	localID	String	ModelObject	<code>\$LinkEndData[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$LinkEndData[i].mdExtensions</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$LinkEndData[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$LinkEndData[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$LinkEndData[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$LinkEndData[i].owner</code>
Qualifier	qualifier	QualifierValue[]	LinkEndData	<code>\$LinkEndData[i].qualifier</code>
Sync Element	syncElement	Element	Element	<code>\$LinkEndData[i].syncElement</code>

Value	value	InputPin	LinkEndData	\$LinkEndData[i].value

133. LinkEndDestructionData

Generalization

- [LinkEndData](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$LinkEndDestructionData[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$LinkEndDestructionData[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$LinkEndDestructionData[i].classType
Destroy At	destroyAt	InputPin	LinkEndDestructionData	\$LinkEndDestructionData[i].destroyAt
Is Destroy Duplicates	destroyDuplicates	boolean	LinkEndDestructionData	\$LinkEndDestructionData[i].destroyDuplicates
isEditable	editable	boolean	BaseElement	\$LinkEndDestructionData[i].editable
End	end	Property	LinkEndData	\$LinkEndDestructionData[i].end
humanName	humanName	String	BaseElement	\$LinkEndDestructionData[i].humanName
humanType	humanType	String	BaseElement	\$LinkEndDestructionData[i].humanType
isInvalid	invalid	boolean	ModelObject	\$LinkEndDestructionData[i].invalid
localID	localID	String	ModelObject	\$LinkEndDestructionData[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$LinkEndDestructionData[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$LinkEndDestructionData[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$LinkEndDestructionData[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$LinkEndDestructionData[i].ownedElement
Owner	owner	Element	Element	\$LinkEndDestructionData[i].owner
Qualifier	qualifier	QualifierValue[]	LinkEndData	\$LinkEndDestructionData[i].qualifier
Sync Element	syncElement	Element	Element	\$LinkEndDestructionData[i].syncElement
Value	value	InputPin	LinkEndData	\$LinkEndDestructionData[i].value

134. LiteralBoolean

Generalization

- [LiteralSpecification](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class)	Sample Template Expression (VTL)

			Owner)	
ID	ID	String	MDOBJECT	\$LiteralBoolean[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$LiteralBoolean[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$LiteralBoolean[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$LiteralBoolean[i].clientDependency
isEditable	editable	boolean	BaseElement	\$LiteralBoolean[i].editable
Expression	expression	Expression	ValueSpecification	\$LiteralBoolean[i].expression
humanName	humanName	String	BaseElement	\$LiteralBoolean[i].humanName
humanType	humanType	String	BaseElement	\$LiteralBoolean[i].humanType
isInvalid	invalid	boolean	ModelObject	\$LiteralBoolean[i].invalid
localID	localID	String	ModelObject	\$LiteralBoolean[i].localID
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	\$LiteralBoolean[i].mdExtensions
Name	name	String	NamedElement	\$LiteralBoolean[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$LiteralBoolean[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$LiteralBoolean[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$LiteralBoolean[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$LiteralBoolean[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$LiteralBoolean[i].ownedElement
Owner	owner	Element	Element	\$LiteralBoolean[i].owner
Owning Constraint	owningConstraint	Constraint	ValueSpecification	\$LiteralBoolean[i].owningConstraint
Owning Instance Spec	owningInstanceSpec	InstanceSpecification	ValueSpecification	\$LiteralBoolean[i].owningInstanceSpec
Owning Lower	owningLower	MultiplicityElement	ValueSpecification	\$LiteralBoolean[i].owningLower
Owning Package	owningPackage	Package	PackageableElement	\$LiteralBoolean[i].owningPackage
Owning Parameter	owningParameter	Parameter	ValueSpecification	\$LiteralBoolean[i].owningParameter
Owning Property	owningProperty	Property	ValueSpecification	\$LiteralBoolean[i].owningProperty
Owning Slot	owningSlot	Slot	ValueSpecification	\$LiteralBoolean[i].owningSlot
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$LiteralBoolean[i].owningTemplateParameter
Owning Upper	owningUpper	MultiplicityElement	ValueSpecification	\$LiteralBoolean[i].owningUpper
Qualified Name	qualifiedName	String	NamedElement	\$LiteralBoolean[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$LiteralBoolean[i].supplierDependency
Sync Element	syncElement	Element	Element	\$LiteralBoolean[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$LiteralBoolean[i].templateParameter
Type	type	Type	TypedElement	\$LiteralBoolean[i].type
isValue	value	boolean	LiteralBoolean	\$LiteralBoolean[i].value
Visibility	visibility	VisibilityKind	PackageableElement	\$LiteralBoolean[i].visibility

135. LiteralInteger

Generalization

- LiteralSpecification

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$LiteralInteger[i].ID</code>
Applied Stereotype Instance	<code>appliedStereotypeInstance</code>	InstanceSpecification	Element	<code>\$LiteralInteger[i].appliedStereotypeInstance</code>
classType	<code>classType</code>	Class	BaseElement	<code>\$LiteralInteger[i].classType</code>
Client Dependency	<code>clientDependency</code>	Dependency[]	NamedElement	<code>\$LiteralInteger[i].clientDependency</code>
isEditable	<code>editable</code>	boolean	BaseElement	<code>\$LiteralInteger[i].editable</code>
Expression	<code>expression</code>	Expression	ValueSpecification	<code>\$LiteralInteger[i].expression</code>
humanName	<code>humanName</code>	String	BaseElement	<code>\$LiteralInteger[i].humanName</code>
humanType	<code>humanType</code>	String	BaseElement	<code>\$LiteralInteger[i].humanType</code>
isInvalid	<code>invalid</code>	boolean	ModelObject	<code>\$LiteralInteger[i].invalid</code>
localID	<code>localID</code>	String	ModelObject	<code>\$LiteralInteger[i].localID</code>
mdExtensions	<code>mdExtensions</code>	MDExtension[]	MDOObject	<code>\$LiteralInteger[i].mdExtensions</code>
Name	<code>name</code>	String	NamedElement	<code>\$LiteralInteger[i].name</code>
Name Expression	<code>nameExpression</code>	StringExpression	NamedElement	<code>\$LiteralInteger[i].nameExpression</code>
Namespace	<code>namespace</code>	Namespace	NamedElement	<code>\$LiteralInteger[i].namespace</code>
objectParent	<code>objectParent</code>	BaseElement	BaseElement	<code>\$LiteralInteger[i].objectParent</code>
Owned Comment	<code>ownedComment</code>	Comment[]	Element	<code>\$LiteralInteger[i].ownedComment</code>
Owned Element	<code>ownedElement</code>	Element[]	Element	<code>\$LiteralInteger[i].ownedElement</code>
Owner	<code>owner</code>	Element	Element	<code>\$LiteralInteger[i].owner</code>
Owning Constraint	<code>owningConstraint</code>	Constraint	ValueSpecification	<code>\$LiteralInteger[i].owningConstraint</code>
Owning Instance Spec	<code>owningInstanceSpec</code>	InstanceSpecification	ValueSpecification	<code>\$LiteralInteger[i].owningInstanceSpec</code>
Owning Lower	<code>owningLower</code>	MultiplicityElement	ValueSpecification	<code>\$LiteralInteger[i].owningLower</code>
Owning Package	<code>owningPackage</code>	Package	PackageableElement	<code>\$LiteralInteger[i].owningPackage</code>
Owning Parameter	<code>owningParameter</code>	Parameter	ValueSpecification	<code>\$LiteralInteger[i].owningParameter</code>
Owning Property	<code>owningProperty</code>	Property	ValueSpecification	<code>\$LiteralInteger[i].owningProperty</code>
Owning Slot	<code>owningSlot</code>	Slot	ValueSpecification	<code>\$LiteralInteger[i].owningSlot</code>
Owning Template Parameter	<code>owningTemplateParameter</code>	TemplateParameter	ParameterableElement	<code>\$LiteralInteger[i].owningTemplateParameter</code>
Owning Upper	<code>owningUpper</code>	MultiplicityElement	ValueSpecification	<code>\$LiteralInteger[i].owningUpper</code>
Qualified Name	<code>qualifiedName</code>	String	NamedElement	<code>\$LiteralInteger[i].qualifiedName</code>
Supplier Dependency	<code>supplierDependency</code>	Dependency[]	NamedElement	<code>\$LiteralInteger[i].supplierDependency</code>
Sync Element	<code>syncElement</code>	Element	Element	<code>\$LiteralInteger[i].syncElement</code>
Template Parameter	<code>templateParameter</code>	TemplateParameter	ParameterableElement	<code>\$LiteralInteger[i].templateParameter</code>
Type	<code>type</code>	Type	TypedElement	<code>\$LiteralInteger[i].type</code>
Value	<code>value</code>	int	LiteralInteger	<code>\$LiteralInteger[i].value</code>
Visibility	<code>visibility</code>	VisibilityKind	PackageableElement	<code>\$LiteralInteger[i].visibility</code>

136. LiteralNull

Generalization

- [LiteralSpecification](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$LiteralNull[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$LiteralNull[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$LiteralNull[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$LiteralNull[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$LiteralNull[i].editable</code>
Expression	expression	Expression	ValueSpecification	<code>\$LiteralNull[i].expression</code>
humanName	humanName	String	BaseElement	<code>\$LiteralNull[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$LiteralNull[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$LiteralNull[i].invalid</code>
localID	localID	String	ModelObject	<code>\$LiteralNull[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$LiteralNull[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$LiteralNull[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$LiteralNull[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$LiteralNull[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$LiteralNull[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$LiteralNull[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$LiteralNull[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$LiteralNull[i].owner</code>
Owning Constraint	owningConstraint	Constraint	ValueSpecification	<code>\$LiteralNull[i].owningConstraint</code>
Owning Instance Spec	owningInstanceSpec	InstanceSpecification	ValueSpecification	<code>\$LiteralNull[i].owningInstanceSpec</code>
Owning Lower	owningLower	MultiplicityElement	ValueSpecification	<code>\$LiteralNull[i].owningLower</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$LiteralNull[i].owningPackage</code>
Owning Parameter	owningParameter	Parameter	ValueSpecification	<code>\$LiteralNull[i].owningParameter</code>
Owning Property	owningProperty	Property	ValueSpecification	<code>\$LiteralNull[i].owningProperty</code>
Owning Slot	owningSlot	Slot	ValueSpecification	<code>\$LiteralNull[i].owningSlot</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$LiteralNull[i].owningTemplateParameter</code>
Owning Upper	owningUpper	MultiplicityElement	ValueSpecification	<code>\$LiteralNull[i].owningUpper</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$LiteralNull[i].qualifiedName</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$LiteralNull[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$LiteralNull[i].syncElement</code>
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$LiteralNull[i].templateParameter</code>
Type	type	Type	TypedElement	<code>\$LiteralNull[i].type</code>

Visibility	visibility	VisibilityKind	PackageableElement	\$LiteralNull[i].visibility

137. LiteralReal

Generalization

- [LiteralSpecification](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$LiteralReal[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$LiteralReal[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$LiteralReal[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$LiteralReal[i].clientDependency
isEditable	editable	boolean	BaseElement	\$LiteralReal[i].editable
Expression	expression	Expression	ValueSpecification	\$LiteralReal[i].expression
humanName	humanName	String	BaseElement	\$LiteralReal[i].humanName
humanType	humanType	String	BaseElement	\$LiteralReal[i].humanType
isInvalid	invalid	boolean	ModelObject	\$LiteralReal[i].invalid
localID	localID	String	ModelObject	\$LiteralReal[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$LiteralReal[i].mdExtensions
Name	name	String	NamedElement	\$LiteralReal[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$LiteralReal[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$LiteralReal[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$LiteralReal[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$LiteralReal[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$LiteralReal[i].ownedElement
Owner	owner	Element	Element	\$LiteralReal[i].owner
Owning Constraint	owningConstraint	Constraint	ValueSpecification	\$LiteralReal[i].owningConstraint
Owning Instance Spec	owningInstanceSpec	InstanceSpecification	ValueSpecification	\$LiteralReal[i].owningInstanceSpec
Owning Lower	owningLower	MultiplicityElement	ValueSpecification	\$LiteralReal[i].owningLower
Owning Package	owningPackage	Package	PackageableElement	\$LiteralReal[i].owningPackage
Owning Parameter	owningParameter	Parameter	ValueSpecification	\$LiteralReal[i].owningParameter
Owning Property	owningProperty	Property	ValueSpecification	\$LiteralReal[i].owningProperty
Owning Slot	owningSlot	Slot	ValueSpecification	\$LiteralReal[i].owningSlot
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$LiteralReal[i].owningTemplateParameter
Owning Upper	owningUpper	MultiplicityElement	ValueSpecification	\$LiteralReal[i].owningUpper
Qualified Name	qualifiedName	String	NamedElement	\$LiteralReal[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$LiteralReal[i].supplierDependency
Sync Element	syncElement	Element	Element	\$LiteralReal[i].syncElement

Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$LiteralReal[i].templateParameter</code>
Type	type	Type	TypedElement	<code>\$LiteralReal[i].type</code>
Value	value	double	LiteralReal	<code>\$LiteralReal[i].value</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$LiteralReal[i].visibility</code>

138. LiteralSpecification

Generalization

- [ValueSpecification](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$LiteralSpecification[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$LiteralSpecification[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$LiteralSpecification[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$LiteralSpecification[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$LiteralSpecification[i].editable</code>
Expression	expression	Expression	ValueSpecification	<code>\$LiteralSpecification[i].expression</code>
humanName	humanName	String	BaseElement	<code>\$LiteralSpecification[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$LiteralSpecification[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$LiteralSpecification[i].invalid</code>
localID	localID	String	ModelObject	<code>\$LiteralSpecification[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$LiteralSpecification[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$LiteralSpecification[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$LiteralSpecification[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$LiteralSpecification[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$LiteralSpecification[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$LiteralSpecification[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$LiteralSpecification[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$LiteralSpecification[i].owner</code>
Owning Constraint	owningConstraint	Constraint	ValueSpecification	<code>\$LiteralSpecification[i].owningConstraint</code>
Owning Instance Spec	owningInstanceSpec	InstanceSpecification	ValueSpecification	<code>\$LiteralSpecification[i].owningInstanceSpec</code>
Owning Lower	owningLower	MultiplicityElement	ValueSpecification	<code>\$LiteralSpecification[i].owningLower</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$LiteralSpecification[i].owningPackage</code>
Owning Parameter	owningParameter	Parameter	ValueSpecification	<code>\$LiteralSpecification[i].owningParameter</code>
Owning Property	owningProperty	Property	ValueSpecification	<code>\$LiteralSpecification[i].owningProperty</code>
Owning Slot	owningSlot	Slot	ValueSpecification	<code>\$LiteralSpecification[i].owningSlot</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$LiteralSpecification[i].owningTemplatePara</code>

				meter
Owning Upper	owningUpper	MultiplicityElement	ValueSpecification	\$LiteralSpecification[i].owningUpper
Qualified Name	qualifiedName	String	NamedElement	\$LiteralSpecification[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$LiteralSpecification[i].supplierDependency
Sync Element	syncElement	Element	Element	\$LiteralSpecification[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$LiteralSpecification[i].templateParameter
Type	type	Type	TypedElement	\$LiteralSpecification[i].type
Visibility	visibility	VisibilityKind	PackageableElement	\$LiteralSpecification[i].visibility

139. LiteralString

Generalization

- [LiteralSpecification](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$LiteralString[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$LiteralString[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$LiteralString[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$LiteralString[i].clientDependency
isEditable	editable	boolean	BaseElement	\$LiteralString[i].editable
Expression	expression	Expression	ValueSpecification	\$LiteralString[i].expression
humanName	humanName	String	BaseElement	\$LiteralString[i].humanName
humanType	humanType	String	BaseElement	\$LiteralString[i].humanType
isInvalid	invalid	boolean	ModelObject	\$LiteralString[i].invalid
localID	localID	String	ModelObject	\$LiteralString[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$LiteralString[i].mdExtensions
Name	name	String	NamedElement	\$LiteralString[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$LiteralString[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$LiteralString[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$LiteralString[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$LiteralString[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$LiteralString[i].ownedElement
Owner	owner	Element	Element	\$LiteralString[i].owner
Owning Constraint	owningConstraint	Constraint	ValueSpecification	\$LiteralString[i].owningConstraint
Owning Instance Spec	owningInstanceSpec	InstanceSpecification	ValueSpecification	\$LiteralString[i].owningInstanceSpec
Owning Lower	owningLower	MultiplicityElement	ValueSpecification	\$LiteralString[i].owningLower
Owning Package	owningPackage	Package	PackageableElement	\$LiteralString[i].owningPackage
Owning Parameter	owningParameter	Parameter	ValueSpecification	\$LiteralString[i].owningParameter

Owning Property	owningProperty	Property	ValueSpecification	<code>\$LiteralString[i].owningProperty</code>
Owning Slot	owningSlot	Slot	ValueSpecification	<code>\$LiteralString[i].owningSlot</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$LiteralString[i].owningTemplateParameter</code>
Owning Upper	owningUpper	MultiplicityElement	ValueSpecification	<code>\$LiteralString[i].owningUpper</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$LiteralString[i].qualifiedName</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$LiteralString[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$LiteralString[i].syncElement</code>
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$LiteralString[i].templateParameter</code>
Type	type	Type	TypedElement	<code>\$LiteralString[i].type</code>
Value	value	String	LiteralString	<code>\$LiteralString[i].value</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$LiteralString[i].visibility</code>

140. LiteralUnlimitedNatural

Generalization

- [LiteralSpecification](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$LiteralUnlimitedNatural[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$LiteralUnlimitedNatural[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$LiteralUnlimitedNatural[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$LiteralUnlimitedNatural[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$LiteralUnlimitedNatural[i].editable</code>
Expression	expression	Expression	ValueSpecification	<code>\$LiteralUnlimitedNatural[i].expression</code>
humanName	humanName	String	BaseElement	<code>\$LiteralUnlimitedNatural[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$LiteralUnlimitedNatural[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$LiteralUnlimitedNatural[i].invalid</code>
localID	localID	String	ModelObject	<code>\$LiteralUnlimitedNatural[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$LiteralUnlimitedNatural[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$LiteralUnlimitedNatural[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$LiteralUnlimitedNatural[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$LiteralUnlimitedNatural[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$LiteralUnlimitedNatural[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$LiteralUnlimitedNatural[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$LiteralUnlimitedNatural[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$LiteralUnlimitedNatural[i].owner</code>

Owning Constraint	owningConstraint	Constraint	ValueSpecification	\$LiteralUnlimitedNatural[i].owningConstraint
Owning Instance Spec	owningInstanceSpec	InstanceSpecification	ValueSpecification	\$LiteralUnlimitedNatural[i].owningInstanceSpec
Owning Lower	owningLower	MultiplicityElement	ValueSpecification	\$LiteralUnlimitedNatural[i].owningLower
Owning Package	owningPackage	Package	PackageableElement	\$LiteralUnlimitedNatural[i].owningPackage
Owning Parameter	owningParameter	Parameter	ValueSpecification	\$LiteralUnlimitedNatural[i].owningParameter
Owning Property	owningProperty	Property	ValueSpecification	\$LiteralUnlimitedNatural[i].owningProperty
Owning Slot	owningSlot	Slot	ValueSpecification	\$LiteralUnlimitedNatural[i].owningSlot
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$LiteralUnlimitedNatural[i].owningTemplateParameter
Owning Upper	owningUpper	MultiplicityElement	ValueSpecification	\$LiteralUnlimitedNatural[i].owningUpper
Qualified Name	qualifiedName	String	NamedElement	\$LiteralUnlimitedNatural[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$LiteralUnlimitedNatural[i].supplierDependency
Sync Element	syncElement	Element	Element	\$LiteralUnlimitedNatural[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$LiteralUnlimitedNatural[i].templateParameter
Type	type	Type	TypedElement	\$LiteralUnlimitedNatural[i].type
Value	value	int	LiteralUnlimitedNatural	\$LiteralUnlimitedNatural[i].value
Visibility	visibility	VisibilityKind	PackageableElement	\$LiteralUnlimitedNatural[i].visibility

141. LoopNode

Generalization

- [StructuredActivityNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$LoopNode[i].ID
Activity	activity	Activity	StructuredActivityNode	\$LoopNode[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$LoopNode[i].appliedStereotypeInstance
Body Output	bodyOutput	OutputPin[]	LoopNode	\$LoopNode[i].bodyOutput
Body Part	bodyPart	ExecutableNode[]	LoopNode	\$LoopNode[i].bodyPart
classType	classType	Class	BaseElement	\$LoopNode[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$LoopNode[i].clientDependency
Contained Edge	containedEdge	ActivityEdge[]	ActivityGroup	\$LoopNode[i].containedEdge
Contained Node	containedNode	ActivityNode[]	ActivityGroup	\$LoopNode[i].containedNode
Context	context	Classifier	Action	\$LoopNode[i].context
Decider	decider	OutputPin	LoopNode	\$LoopNode[i].decider

Edge	edge	ActivityEdge[]	StructuredActivityNode	\$LoopNode[i].edge
isEditable	editable	boolean	BaseElement	\$LoopNode[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$LoopNode[i].elementImport
Handler	handler	ExceptionHandler[]	ExecutableNode	\$LoopNode[i].handler
humanName	humanName	String	BaseElement	\$LoopNode[i].humanName
humanType	humanType	String	BaseElement	\$LoopNode[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$LoopNode[i].importedMember
In Activity	inActivity	Activity	ActivityGroup	\$LoopNode[i].inActivity
In Group	inGroup	ActivityGroup[]	ActivityNode	\$LoopNode[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$LoopNode[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$LoopNode[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$LoopNode[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$LoopNode[i].incoming
Input	input	InputPin[]	Action	\$LoopNode[i].input
isInvalid	invalid	boolean	ModelObject	\$LoopNode[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$LoopNode[i].leaf
localID	localID	String	ModelObject	\$LoopNode[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$LoopNode[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$LoopNode[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$LoopNode[i].locallyReentrant
Loop Variable	loopVariable	OutputPin[]	LoopNode	\$LoopNode[i].loopVariable
Loop Variable Input	loopVariableInput	InputPin[]	LoopNode	\$LoopNode[i].loopVariableInput
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$LoopNode[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$LoopNode[i].member
isMustIsolate	mustIsolate	boolean	StructuredActivityNode	\$LoopNode[i].mustIsolate
Name	name	String	NamedElement	\$LoopNode[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$LoopNode[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$LoopNode[i].namespace
Node	node	ActivityNode[]	StructuredActivityNode	\$LoopNode[i].node
objectParent	objectParent	BaseElement	BaseElement	\$LoopNode[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$LoopNode[i].outgoing
Output	output	OutputPin[]	Action	\$LoopNode[i].output
Owned Comment	ownedComment	Comment[]	Element	\$LoopNode[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$LoopNode[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$LoopNode[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$LoopNode[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$LoopNode[i].ownedRule
Owner	owner	Element	Element	\$LoopNode[i].owner
Package Import	packageImport	PackageImport[]	Namespace	\$LoopNode[i].packageImport

Qualified Name	qualifiedName	String	NamedElement	\$LoopNode[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$LoopNode[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$LoopNode[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$LoopNode[i].redefinitionContext
Result	result	OutputPin[]	LoopNode	\$LoopNode[i].result
Setup Part	setupPart	ExecutableNode[]	LoopNode	\$LoopNode[i].setupPart
Structured Node Input	structuredNodeInput	InputPin[]	StructuredActivityNode	\$LoopNode[i].structuredNodeInput
Structured Node Output	structuredNodeOutput	OutputPin[]	StructuredActivityNode	\$LoopNode[i].structuredNodeOutput
Subgroup	subgroup	ActivityGroup[]	ActivityGroup	\$LoopNode[i].subgroup
Super Group	superGroup	ActivityGroup	ActivityGroup	\$LoopNode[i].superGroup
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$LoopNode[i].supplierDependency
Sync Element	syncElement	Element	Element	\$LoopNode[i].syncElement
Test	test	ExecutableNode[]	LoopNode	\$LoopNode[i].test
Is Tested First	testedFirst	boolean	LoopNode	\$LoopNode[i].testedFirst
Variable	variable	Variable[]	StructuredActivityNode	\$LoopNode[i].variable
Visibility	visibility	VisibilityKind	NamedElement	\$LoopNode[i].visibility

142. Manifestation

Generalization

- [Abstraction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Manifestation[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Manifestation[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Manifestation[i].classType
Client	client	NamedElement[]	Dependency	\$Manifestation[i].client
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Manifestation[i].clientDependency
isEditable	editable	boolean	BaseElement	\$Manifestation[i].editable
humanName	humanName	String	BaseElement	\$Manifestation[i].humanName
humanType	humanType	String	BaseElement	\$Manifestation[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Manifestation[i].invalid
localID	localID	String	ModelObject	\$Manifestation[i].localID
Mapping	mapping	OpaqueExpression	Abstraction	\$Manifestation[i].mapping
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Manifestation[i].mdExtensions
Name	name	String	NamedElement	\$Manifestation[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Manifestation[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Manifestation[i].namespace

objectParent	objectParent	BaseElement	BaseElement	\$Manifestation[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Manifestation[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Manifestation[i].ownedElement
Owner	owner	Element	Element	\$Manifestation[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Manifestation[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Manifestation[i].owningTemplateParameter
Qualified Name	qualifiedName	String	NamedElement	\$Manifestation[i].qualifiedName
Related Element	relatedElement	Element[]	Relationship	\$Manifestation[i].relatedElement
Source	source	Element[]	DirectedRelationship	\$Manifestation[i].source
Supplier	supplier	NamedElement[]	Dependency	\$Manifestation[i].supplier
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Manifestation[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Manifestation[i].syncElement
Target	target	Element[]	DirectedRelationship	\$Manifestation[i].target
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$Manifestation[i].templateParameter
Utilized Element	utilizedElement	PackageableElement	Manifestation	\$Manifestation[i].utilizedElement
Visibility	visibility	VisibilityKind	PackageableElement	\$Manifestation[i].visibility

143. MergeNode

Generalization

- [ControlNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$MergeNode[i].ID
Activity	activity	Activity	ActivityNode	\$MergeNode[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$MergeNode[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$MergeNode[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$MergeNode[i].clientDependency
isEditable	editable	boolean	BaseElement	\$MergeNode[i].editable
humanName	humanName	String	BaseElement	\$MergeNode[i].humanName
humanType	humanType	String	BaseElement	\$MergeNode[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$MergeNode[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$MergeNode[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$MergeNode[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$MergeNode[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$MergeNode[i].incoming
isInvalid	invalid	boolean	ModelObject	\$MergeNode[i].invalid

Is Leaf	leaf	boolean	RedefinableElement	<code>\$MergeNode[i].leaf</code>
localID	localID	String	ModelObject	<code>\$MergeNode[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$MergeNode[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$MergeNode[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$MergeNode[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$MergeNode[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$MergeNode[i].objectParent</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$MergeNode[i].outgoing</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$MergeNode[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$MergeNode[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$MergeNode[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$MergeNode[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$MergeNode[i].redefinedElement</code>
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	<code>\$MergeNode[i].redefinedNode</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$MergeNode[i].redefinitionContext</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$MergeNode[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$MergeNode[i].syncElement</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$MergeNode[i].visibility</code>

144. Message

Generalization

- [NamedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$Message[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Message[i].appliedStereotypeInstance</code>
Argument	argument	ValueSpecification[]	Message	<code>\$Message[i].argument</code>
classType	classType	Class	BaseElement	<code>\$Message[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Message[i].clientDependency</code>
Connector	connector	Connector	Message	<code>\$Message[i].connector</code>
isEditable	editable	boolean	BaseElement	<code>\$Message[i].editable</code>
Guard	guard	Constraint	Message	<code>\$Message[i].guard</code>
humanName	humanName	String	BaseElement	<code>\$Message[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Message[i].humanType</code>
Interaction	interaction	Interaction	Message	<code>\$Message[i].interaction</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Message[i].invalid</code>
localID	localID	String	ModelObject	<code>\$Message[i].localID</code>

mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$Message[i].mdExtensions</code>
Message Kind	messageKind	MessageKind	Message	<code>\$Message[i].messageKind</code>
Message Sort	messageSort	MessageSort	Message	<code>\$Message[i].messageSort</code>
Name	name	String	NamedElement	<code>\$Message[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Message[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Message[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Message[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Message[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Message[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$Message[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Message[i].qualifiedName</code>
Receive Event	receiveEvent	MessageEnd	Message	<code>\$Message[i].receiveEvent</code>
Reply Message	replyMessage	Message	Message	<code>\$Message[i].replyMessage</code>
Send Event	sendEvent	MessageEnd	Message	<code>\$Message[i].sendEvent</code>
Signature	signature	NamedElement	Message	<code>\$Message[i].signature</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Message[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Message[i].syncElement</code>
Target	target	ValueSpecification	Message	<code>\$Message[i].target</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$Message[i].visibility</code>

145. MessageEnd

Generalization

- [NamedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$MessageEnd[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$MessageEnd[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$MessageEnd[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$MessageEnd[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$MessageEnd[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$MessageEnd[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$MessageEnd[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$MessageEnd[i].invalid</code>
localID	localID	String	ModelObject	<code>\$MessageEnd[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$MessageEnd[i].mdExtensions</code>
Message	message	Message	MessageEnd	<code>\$MessageEnd[i].message</code>
Name	name	String	NamedElement	<code>\$MessageEnd[i].name</code>

Name Expression	nameExpression	StringExpression	NamedElement	<code>\$MessageEnd[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$MessageEnd[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$MessageEnd[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$MessageEnd[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$MessageEnd[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$MessageEnd[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$MessageEnd[i].qualifiedName</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$MessageEnd[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$MessageEnd[i].syncElement</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$MessageEnd[i].visibility</code>

146. MessageEvent

Generalization

- [Event](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$MessageEvent[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$MessageEvent[i].appliedStereotypeInstance</code>
Behavior	behavior	Behavior	Event	<code>\$MessageEvent[i].behavior</code>
classType	classType	Class	BaseElement	<code>\$MessageEvent[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$MessageEvent[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$MessageEvent[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$MessageEvent[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$MessageEvent[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$MessageEvent[i].invalid</code>
localID	localID	String	ModelObject	<code>\$MessageEvent[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$MessageEvent[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$MessageEvent[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$MessageEvent[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$MessageEvent[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$MessageEvent[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$MessageEvent[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$MessageEvent[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$MessageEvent[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$MessageEvent[i].owningPackage</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$MessageEvent[i].owningTemplateParameter</code>

				er
Qualified Name	qualifiedName	String	NamedElement	<code>\$MessageEvent[i].qualifiedName</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$MessageEvent[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$MessageEvent[i].syncElement</code>
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$MessageEvent[i].templateParameter</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$MessageEvent[i].visibility</code>

147. MessageOccurrenceSpecification

Generalization

- [MessageEnd](#)
- [OccurrenceSpecification](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$MessageOccurrenceSpecification[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$MessageOccurrenceSpecification[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$MessageOccurrenceSpecification[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$MessageOccurrenceSpecification[i].clientDependency</code>
Covered	covered	Lifeline[]	OccurrenceSpecification	<code>\$MessageOccurrenceSpecification[i].covered</code>
isEditable	editable	boolean	BaseElement	<code>\$MessageOccurrenceSpecification[i].editable</code>
Enclosing Interaction	enclosingInteraction	Interaction	InteractionFragment	<code>\$MessageOccurrenceSpecification[i].enclosingInteraction</code>
Enclosing Operand	enclosingOperand	InteractionOperand	InteractionFragment	<code>\$MessageOccurrenceSpecification[i].enclosingOperand</code>
General Ordering	generalOrdering	GeneralOrdering[]	InteractionFragment	<code>\$MessageOccurrenceSpecification[i].generalOrdering</code>
humanName	humanName	String	BaseElement	<code>\$MessageOccurrenceSpecification[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$MessageOccurrenceSpecification[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$MessageOccurrenceSpecification[i].invalid</code>
localID	localID	String	ModelObject	<code>\$MessageOccurrenceSpecification[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$MessageOccurrenceSpecification[i].mdExt</code>

				ensions
Message	message	Message	MessageEnd	<code>\$MessageOccurrenceSpecification[i].message</code>
Name	name	String	NamedElement	<code>\$MessageOccurrenceSpecification[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$MessageOccurrenceSpecification[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$MessageOccurrenceSpecification[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$MessageOccurrenceSpecification[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$MessageOccurrenceSpecification[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$MessageOccurrenceSpecification[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$MessageOccurrenceSpecification[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$MessageOccurrenceSpecification[i].qualifiedName</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$MessageOccurrenceSpecification[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$MessageOccurrenceSpecification[i].syncElement</code>
To After	toAfter	GeneralOrdering[]	OccurrenceSpecification	<code>\$MessageOccurrenceSpecification[i].toAfter</code>
To Before	toBefore	GeneralOrdering[]	OccurrenceSpecification	<code>\$MessageOccurrenceSpecification[i].toBefore</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$MessageOccurrenceSpecification[i].visibility</code>

148. Model

Generalization

- [Package](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$Model[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Model[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$Model[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Model[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$Model[i].editable</code>
Element Import	elementImport	ElementImport[]	Namespace	<code>\$Model[i].elementImport</code>

humanName	humanName	String	BaseElement	\$Model[i].humanName
humanType	humanType	String	BaseElement	\$Model[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Model[i].importedMember
isInvalid	invalid	boolean	ModelObject	\$Model[i].invalid
localID	localID	String	ModelObject	\$Model[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Model[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Model[i].member
Name	name	String	NamedElement	\$Model[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Model[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Model[i].namespace
Nested Package	nestedPackage	Package[]	Package	\$Model[i].nestedPackage
Nesting Package	nestingPackage	Package	Package	\$Model[i].nestingPackage
objectParent	objectParent	BaseElement	BaseElement	\$Model[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Model[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Model[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Model[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$Model[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$Model[i].ownedRule
Owned Stereotype	ownedStereotype	Stereotype[]	Package	\$Model[i].ownedStereotype
Owned Template Signature	ownedTemplateSignature	TemplateSignature	TemplateableElement	\$Model[i].ownedTemplateSignature
Owned Type	ownedType	Type[]	Package	\$Model[i].ownedType
Owner	owner	Element	Element	\$Model[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Model[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Model[i].owningTemplateParameter
Package Import	packageImport	PackageImport[]	Namespace	\$Model[i].packageImport
Package Merge	packageMerge	PackageMerge[]	Package	\$Model[i].packageMerge
Packaged Element	packagedElement	PackageableElement[]	Package	\$Model[i].packagedElement
Profile Application	profileApplication	ProfileApplication[]	Package	\$Model[i].profileApplication
Qualified Name	qualifiedName	String	NamedElement	\$Model[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Model[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Model[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$Model[i].templateBinding
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$Model[i].templateParameter
uRI	uRI	String	Package	\$Model[i].uRI
Viewpoint	viewpoint	String	Model	\$Model[i].viewpoint
Visibility	visibility	VisibilityKind	PackageableElement	\$Model[i].visibility

149. MultiplicityElement

Generalization

- [Element](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$MultiplicityElement[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$MultiplicityElement[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$MultiplicityElement[i].classType
isEditable	editable	boolean	BaseElement	\$MultiplicityElement[i].editable
humanName	humanName	String	BaseElement	\$MultiplicityElement[i].humanName
humanType	humanType	String	BaseElement	\$MultiplicityElement[i].humanType
isInvalid	invalid	boolean	ModelObject	\$MultiplicityElement[i].invalid
localID	localID	String	ModelObject	\$MultiplicityElement[i].localID
Lower	lower	int	MultiplicityElement	\$MultiplicityElement[i].lower
Lower Value	lowerValue	ValueSpecification	MultiplicityElement	\$MultiplicityElement[i].lowerValue
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	\$MultiplicityElement[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$MultiplicityElement[i].objectParent
Is Ordered	ordered	boolean	MultiplicityElement	\$MultiplicityElement[i].ordered
Owned Comment	ownedComment	COMMENT[]	Element	\$MultiplicityElement[i].ownedComment
Owned Element	ownedElement	ELEMENT[]	Element	\$MultiplicityElement[i].ownedElement
Owner	owner	Element	Element	\$MultiplicityElement[i].owner
Sync Element	syncElement	Element	Element	\$MultiplicityElement[i].syncElement
Is Unique	unique	boolean	MultiplicityElement	\$MultiplicityElement[i].unique
Upper	upper	int	MultiplicityElement	\$MultiplicityElement[i].upper
Upper Value	upperValue	ValueSpecification	MultiplicityElement	\$MultiplicityElement[i].upperValue

150. NamedElement

Generalization

- [Element](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$NamedElement[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$NamedElement[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$NamedElement[i].classType
Client Dependency	clientDependency	DEPENDENCY[]	NamedElement	\$NamedElement[i].clientDependency

isEditable	editable	boolean	BaseElement	\$NamedElement[i].editable
humanName	humanName	String	BaseElement	\$NamedElement[i].humanName
humanType	humanType	String	BaseElement	\$NamedElement[i].humanType
isInvalid	invalid	boolean	ModelObject	\$NamedElement[i].invalid
localID	localID	String	ModelObject	\$NamedElement[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$NamedElement[i].mdExtensions
Name	name	String	NamedElement	\$NamedElement[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$NamedElement[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$NamedElement[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$NamedElement[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$NamedElement[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$NamedElement[i].ownedElement
Owner	owner	Element	Element	\$NamedElement[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$NamedElement[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$NamedElement[i].supplierDependency
Sync Element	syncElement	Element	Element	\$NamedElement[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$NamedElement[i].visibility

151. Namespace

Generalization

- [NamedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Namespace[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Namespace[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Namespace[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Namespace[i].clientDependency
isEditable	editable	boolean	BaseElement	\$Namespace[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$Namespace[i].elementImport
humanName	humanName	String	BaseElement	\$Namespace[i].humanName
humanType	humanType	String	BaseElement	\$Namespace[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Namespace[i].importedMember
isInvalid	invalid	boolean	ModelObject	\$Namespace[i].invalid
localID	localID	String	ModelObject	\$Namespace[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Namespace[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Namespace[i].member
Name	name	String	NamedElement	\$Namespace[i].name

Name Expression	nameExpression	StringExpression	NamedElement	\$Namespace[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Namespace[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Namespace[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Namespace[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Namespace[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Namespace[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$Namespace[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$Namespace[i].ownedRule
Owner	owner	Element	Element	\$Namespace[i].owner
Package Import	packageImport	PackageImport[]	Namespace	\$Namespace[i].packageImport
Qualified Name	qualifiedName	String	NamedElement	\$Namespace[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Namespace[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Namespace[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$Namespace[i].visibility

152. Node

Generalization

- [Class](#)
- [DeploymentTarget](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Node[i].ID
Class	UMLClass	Class	Classifier	\$Node[i].UMLClass
Is Abstract	abstract	boolean	Class	\$Node[i].abstract
Is Active	active	boolean	Class	\$Node[i].active
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Node[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$Node[i].attribute
classType	classType	Class	BaseElement	\$Node[i].classType
Classifier Behavior	classifierBehavior	Behavior	BehavoredClassifier	\$Node[i].classifierBehavior
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Node[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$Node[i].collaborationUse
Deployed Element	deployedElement	PackageableElement[]	DeploymentTarget	\$Node[i].deployedElement
Deployment	deployment	Deployment[]	DeploymentTarget	\$Node[i].deployment
isEditable	editable	boolean	BaseElement	\$Node[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$Node[i].elementImport
Extension	extension	Extension[]	Class	\$Node[i].extension
Feature	feature	Feature[]	Classifier	\$Node[i].feature

Is Final Specialization	finalSpecialization	boolean	Classifier	\$Node[i].finalSpecialization
General	general	Classifier[]	Classifier	\$Node[i].general
Generalization	generalization	Generalization[]	Classifier	\$Node[i].generalization
humanName	humanName	String	BaseElement	\$Node[i].humanName
humanType	humanType	String	BaseElement	\$Node[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Node[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$Node[i].inheritedMember
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavioredClassifier	\$Node[i].interfaceRealization
isInvalid	invalid	boolean	ModelObject	\$Node[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$Node[i].leaf
localID	localID	String	ModelObject	\$Node[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$Node[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Node[i].member
Name	name	String	NamedElement	\$Node[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Node[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Node[i].namespace
Nested Classifier	nestedClassifier	Classifier[]	Class	\$Node[i].nestedClassifier
Nested Node	nestedNode	Node[]	Node	\$Node[i].nestedNode
objectParent	objectParent	BaseElement	BaseElement	\$Node[i].objectParent
Owned Attribute	ownedAttribute	Property[]	Class	\$Node[i].ownedAttribute
Owned Behavior	ownedBehavior	Behavior[]	BehavioredClassifier	\$Node[i].ownedBehavior
Owned Comment	ownedComment	Comment[]	Element	\$Node[i].ownedComment
Owned Connector	ownedConnector	Connector[]	StructuredClassifier	\$Node[i].ownedConnector
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Node[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Node[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$Node[i].ownedMember
Owned Operation	ownedOperation	Operation[]	Class	\$Node[i].ownedOperation
Owned Port	ownedPort	Port[]	EncapsulatedClassifier	\$Node[i].ownedPort
Owned Reception	ownedReception	Reception[]	Class	\$Node[i].ownedReception
Owned Rule	ownedRule	Constraint[]	Namespace	\$Node[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$Node[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$Node[i].ownedUseCase
Owner	owner	Element	Element	\$Node[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Node[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Node[i].owningTemplateParameter
Package	package	Package	Type	\$Node[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$Node[i].packageImport
Part	part	Property[]	StructuredClassifier	\$Node[i].part
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$Node[i].powertypeExtent

Qualified Name	qualifiedName	String	NamedElement	\$Node[i].qualifiedName
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$Node[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Node[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$Node[i].redefinitionContext
Representation	representation	CollaborationUse	Classifier	\$Node[i].representation
Role	role	ConnectableElement[]	StructuredClassifier	\$Node[i].role
Substitution	substitution	Substitution[]	Classifier	\$Node[i].substitution
Super Class	superClass	Class[]	Class	\$Node[i].superClass
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Node[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Node[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$Node[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$Node[i].templateParameter
Use Case	useCase	UseCase[]	Classifier	\$Node[i].useCase
Visibility	visibility	VisibilityKind	PackageableElement	\$Node[i].visibility

153. ObjectFlow

Generalization

- [ActivityEdge](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$ObjectFlow[i].ID
Activity	activity	Activity	ActivityEdge	\$ObjectFlow[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ObjectFlow[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ObjectFlow[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ObjectFlow[i].clientDependency
isEditable	editable	boolean	BaseElement	\$ObjectFlow[i].editable
Guard	guard	ValueSpecification	ActivityEdge	\$ObjectFlow[i].guard
humanName	humanName	String	BaseElement	\$ObjectFlow[i].humanName
humanType	humanType	String	BaseElement	\$ObjectFlow[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityEdge	\$ObjectFlow[i].inGroup
In Partition	inPartition	ActivityPartition[]	ActivityEdge	\$ObjectFlow[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityEdge	\$ObjectFlow[i].inStructuredNode
Interrupts	interrupts	InterruptibleActivityRegion	ActivityEdge	\$ObjectFlow[i].interrupts
isInvalid	invalid	boolean	ModelObject	\$ObjectFlow[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ObjectFlow[i].leaf

localID	localID	String	ModelObject	<code>\$ObjectFlow[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$ObjectFlow[i].mdExtensions</code>
Is Multicast	multicast	boolean	ObjectFlow	<code>\$ObjectFlow[i].multicast</code>
Is Multireceive	multireceive	boolean	ObjectFlow	<code>\$ObjectFlow[i].multireceive</code>
Name	name	String	NamedElement	<code>\$ObjectFlow[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$ObjectFlow[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$ObjectFlow[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$ObjectFlow[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$ObjectFlow[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$ObjectFlow[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$ObjectFlow[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$ObjectFlow[i].qualifiedName</code>
Redefined Edge	redefinedEdge	ActivityEdge[]	ActivityEdge	<code>\$ObjectFlow[i].redefinedEdge</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$ObjectFlow[i].redefinedElement</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$ObjectFlow[i].redefinitionContext</code>
Selection	selection	Behavior	ObjectFlow	<code>\$ObjectFlow[i].selection</code>
Source	source	ActivityNode	ActivityEdge	<code>\$ObjectFlow[i].source</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$ObjectFlow[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$ObjectFlow[i].syncElement</code>
Target	target	ActivityNode	ActivityEdge	<code>\$ObjectFlow[i].target</code>
Transformation	transformation	Behavior	ObjectFlow	<code>\$ObjectFlow[i].transformation</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$ObjectFlow[i].visibility</code>
Weight	weight	ValueSpecification	ActivityEdge	<code>\$ObjectFlow[i].weight</code>

154. ObjectNode

Generalization

- [ActivityNode](#)
- [TypedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$ObjectNode[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$ObjectNode[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$ObjectNode[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$ObjectNode[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$ObjectNode[i].clientDependency</code>
Is Control Type	controlType	boolean	ObjectNode	<code>\$ObjectNode[i].controlType</code>
isEditable	editable	boolean	BaseElement	<code>\$ObjectNode[i].editable</code>

humanName	humanName	String	BaseElement	\$ObjectNode[i].humanName
humanType	humanType	String	BaseElement	\$ObjectNode[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ObjectNode[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ObjectNode[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ObjectNode[i].inPartition
In State	inState	State[]	ObjectNode	\$ObjectNode[i].inState
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ObjectNode[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ObjectNode[i].incoming
isInvalid	invalid	boolean	ModelObject	\$ObjectNode[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ObjectNode[i].leaf
localID	localID	String	ModelObject	\$ObjectNode[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ObjectNode[i].mdExtensions
Name	name	String	NamedElement	\$ObjectNode[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ObjectNode[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ObjectNode[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ObjectNode[i].objectParent
Ordering	ordering	ObjectNodeOrderingKind	ObjectNode	\$ObjectNode[i].ordering
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ObjectNode[i].outgoing
Owned Comment	ownedComment	Comment[]	Element	\$ObjectNode[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ObjectNode[i].ownedElement
Owner	owner	Element	Element	\$ObjectNode[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ObjectNode[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ObjectNode[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ObjectNode[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ObjectNode[i].redefinitionContext
Selection	selection	Behavior	ObjectNode	\$ObjectNode[i].selection
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ObjectNode[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ObjectNode[i].syncElement
Type	type	Type	TypedElement	\$ObjectNode[i].type
Upper Bound	upperBound	ValueSpecification	ObjectNode	\$ObjectNode[i].upperBound
Visibility	visibility	VisibilityKind	NamedElement	\$ObjectNode[i].visibility

155. Observation

Generalization

- [PackageableElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class)	Sample Template Expression (VTL)

			Owner)	
ID	ID	String	MDOObject	\$Observation[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Observation[i].appliedStereotypeInstance
Behavior	behavior	Behavior	Observation	\$Observation[i].behavior
classType	classType	Class	BaseElement	\$Observation[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Observation[i].clientDependency
isEditable	editable	boolean	BaseElement	\$Observation[i].editable
humanName	humanName	String	BaseElement	\$Observation[i].humanName
humanType	humanType	String	BaseElement	\$Observation[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Observation[i].invalid
localID	localID	String	ModelObject	\$Observation[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Observation[i].mdExtensions
Name	name	String	NamedElement	\$Observation[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Observation[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Observation[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Observation[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Observation[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Observation[i].ownedElement
Owner	owner	Element	Element	\$Observation[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Observation[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Observation[i].owningTemplateParameter
Qualified Name	qualifiedName	String	NamedElement	\$Observation[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Observation[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Observation[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$Observation[i].templateParameter
Visibility	visibility	VisibilityKind	PackageableElement	\$Observation[i].visibility

156. OccurrenceSpecification

Generalization

- [InteractionFragment](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$OccurrenceSpecification[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$OccurrenceSpecification[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$OccurrenceSpecification[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$OccurrenceSpecification[i].clientDependen

				cy
Covered	covered	Lifeline[]	OccurrenceSpecification	\$OccurrenceSpecification[i].covered
isEditable	editable	boolean	BaseElement	\$OccurrenceSpecification[i].editable
Enclosing Interaction	enclosingInteraction	Interaction	InteractionFragment	\$OccurrenceSpecification[i].enclosingInteraction
Enclosing Operand	enclosingOperand	InteractionOperand	InteractionFragment	\$OccurrenceSpecification[i].enclosingOperand
General Ordering	generalOrdering	GeneralOrdering[]	InteractionFragment	\$OccurrenceSpecification[i].generalOrdering
humanName	humanName	String	BaseElement	\$OccurrenceSpecification[i].humanName
humanType	humanType	String	BaseElement	\$OccurrenceSpecification[i].humanType
isInvalid	invalid	boolean	ModelObject	\$OccurrenceSpecification[i].invalid
localID	localID	String	ModelObject	\$OccurrenceSpecification[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$OccurrenceSpecification[i].mdExtensions
Name	name	String	NamedElement	\$OccurrenceSpecification[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$OccurrenceSpecification[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$OccurrenceSpecification[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$OccurrenceSpecification[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$OccurrenceSpecification[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$OccurrenceSpecification[i].ownedElement
Owner	owner	Element	Element	\$OccurrenceSpecification[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$OccurrenceSpecification[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$OccurrenceSpecification[i].supplierDependency
Sync Element	syncElement	Element	Element	\$OccurrenceSpecification[i].syncElement
To After	toAfter	GeneralOrdering[]	OccurrenceSpecification	\$OccurrenceSpecification[i].toAfter
To Before	toBefore	GeneralOrdering[]	OccurrenceSpecification	\$OccurrenceSpecification[i].toBefore
Visibility	visibility	VisibilityKind	NamedElement	\$OccurrenceSpecification[i].visibility

157. OpaqueAction

Generalization

- [Action](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$OpaqueAction[i].ID

Activity	activity	Activity	ActivityNode	\$OpaqueAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$OpaqueAction[i].appliedStereotypeInstance
Body	body	String[]	OpaqueAction	\$OpaqueAction[i].body
classType	classType	Class	BaseElement	\$OpaqueAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$OpaqueAction[i].clientDependency
Context	context	Classifier	Action	\$OpaqueAction[i].context
isEditable	editable	boolean	BaseElement	\$OpaqueAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$OpaqueAction[i].handler
humanName	humanName	String	BaseElement	\$OpaqueAction[i].humanName
humanType	humanType	String	BaseElement	\$OpaqueAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$OpaqueAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$OpaqueAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$OpaqueAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$OpaqueAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$OpaqueAction[i].incoming
Input	input	InputPin[]	Action	\$OpaqueAction[i].input
Input Value	inputValue	InputPin[]	OpaqueAction	\$OpaqueAction[i].inputValue
isInvalid	invalid	boolean	ModelObject	\$OpaqueAction[i].invalid
Language	language	String[]	OpaqueAction	\$OpaqueAction[i].language
Is Leaf	leaf	boolean	RedefinableElement	\$OpaqueAction[i].leaf
localID	localID	String	ModelObject	\$OpaqueAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$OpaqueAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$OpaqueAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$OpaqueAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$OpaqueAction[i].mdExtensions
Name	name	String	NamedElement	\$OpaqueAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$OpaqueAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$OpaqueAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$OpaqueAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$OpaqueAction[i].outgoing
Output	output	OutputPin[]	Action	\$OpaqueAction[i].output
Output Value	outputValue	OutputPin[]	OpaqueAction	\$OpaqueAction[i].outputValue
Owned Comment	ownedComment	Comment[]	Element	\$OpaqueAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$OpaqueAction[i].ownedElement
Owner	owner	Element	Element	\$OpaqueAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$OpaqueAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$OpaqueAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$OpaqueAction[i].redefinedNode

Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$OpaqueAction[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$OpaqueAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$OpaqueAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$OpaqueAction[i].visibility

158. OpaqueBehavior

Generalization

- [Behavior](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$OpaqueBehavior[i].ID
Class	UMLClass	Class	Classifier	\$OpaqueBehavior[i].UMLClass
Is Abstract	abstract	boolean	Class	\$OpaqueBehavior[i].abstract
Is Active	active	boolean	Class	\$OpaqueBehavior[i].active
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$OpaqueBehavior[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$OpaqueBehavior[i].attribute
Body	body	String[]	OpaqueBehavior	\$OpaqueBehavior[i].body
classType	classType	Class	BaseElement	\$OpaqueBehavior[i].classType
Classifier Behavior	classifierBehavior	Behavior	BehavioredClassifier	\$OpaqueBehavior[i].classifierBehavior
Client Dependency	clientDependency	Dependency[]	NamedElement	\$OpaqueBehavior[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$OpaqueBehavior[i].collaborationUse
Context	context	BehavioredClassifier	Behavior	\$OpaqueBehavior[i].context
isEditable	editable	boolean	BaseElement	\$OpaqueBehavior[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$OpaqueBehavior[i].elementImport
Event	event	Event[]	Behavior	\$OpaqueBehavior[i].event
Extension	extension	Extension[]	Class	\$OpaqueBehavior[i].extension
Feature	feature	Feature[]	Classifier	\$OpaqueBehavior[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$OpaqueBehavior[i].finalSpecialization
General	general	Classifier[]	Classifier	\$OpaqueBehavior[i].general
Generalization	generalization	Generalization[]	Classifier	\$OpaqueBehavior[i].generalization
humanName	humanName	String	BaseElement	\$OpaqueBehavior[i].humanName
humanType	humanType	String	BaseElement	\$OpaqueBehavior[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$OpaqueBehavior[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$OpaqueBehavior[i].inheritedMember
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavioredClassifier	\$OpaqueBehavior[i].interfaceRealization
isInvalid	invalid	boolean	ModelObject	\$OpaqueBehavior[i].invalid

Language	language	String[]	OpaqueBehavior	\$OpaqueBehavior[i].language
Is Leaf	leaf	boolean	RedefinableElement	\$OpaqueBehavior[i].leaf
localID	localID	String	ModelObject	\$OpaqueBehavior[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$OpaqueBehavior[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$OpaqueBehavior[i].member
Name	name	String	NamedElement	\$OpaqueBehavior[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$OpaqueBehavior[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$OpaqueBehavior[i].namespace
Nested Classifier	nestedClassifier	Classifier[]	Class	\$OpaqueBehavior[i].nestedClassifier
objectParent	objectParent	BaseElement	BaseElement	\$OpaqueBehavior[i].objectParent
Observation	observation	Observation[]	Behavior	\$OpaqueBehavior[i].observation
Owned Attribute	ownedAttribute	Property[]	Class	\$OpaqueBehavior[i].ownedAttribute
Owned Behavior	ownedBehavior	Behavior[]	BehavioredClassifier	\$OpaqueBehavior[i].ownedBehavior
Owned Comment	ownedComment	Comment[]	Element	\$OpaqueBehavior[i].ownedComment
Owned Connector	ownedConnector	Connector[]	StructuredClassifier	\$OpaqueBehavior[i].ownedConnector
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$OpaqueBehavior[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$OpaqueBehavior[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$OpaqueBehavior[i].ownedMember
Owned Operation	ownedOperation	Operation[]	Class	\$OpaqueBehavior[i].ownedOperation
Owned Parameter	ownedParameter	Parameter[]	Behavior	\$OpaqueBehavior[i].ownedParameter
Owned Parameter Set	ownedParameterSet	ParameterSet[]	Behavior	\$OpaqueBehavior[i].ownedParameterSet
Owned Port	ownedPort	Port[]	EncapsulatedClassifier	\$OpaqueBehavior[i].ownedPort
Owned Reception	ownedReception	Reception[]	Class	\$OpaqueBehavior[i].ownedReception
Owned Rule	ownedRule	Constraint[]	Namespace	\$OpaqueBehavior[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$OpaqueBehavior[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$OpaqueBehavior[i].ownedUseCase
Owner	owner	Element	Element	\$OpaqueBehavior[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$OpaqueBehavior[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$OpaqueBehavior[i].owningTemplateParameter
Package	package	Package	Type	\$OpaqueBehavior[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$OpaqueBehavior[i].packageImport
Part	part	Property[]	StructuredClassifier	\$OpaqueBehavior[i].part
Postcondition	postcondition	Constraint[]	Behavior	\$OpaqueBehavior[i].postcondition
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$OpaqueBehavior[i].powertypeExtent
Precondition	precondition	Constraint[]	Behavior	\$OpaqueBehavior[i].precondition
Qualified Name	qualifiedName	String	NamedElement	\$OpaqueBehavior[i].qualifiedName
Redefined Behavior	redefinedBehavior	Behavior[]	Behavior	\$OpaqueBehavior[i].redefinedBehavior
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$OpaqueBehavior[i].redefinedClassifier

Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$OpaqueBehavior[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$OpaqueBehavior[i].redefinitionContext
Is Reentrant	reentrant	boolean	Behavior	\$OpaqueBehavior[i].reentrant
Representation	representation	CollaborationUse	Classifier	\$OpaqueBehavior[i].representation
Role	role	ConnectableElement[]	StructuredClassifier	\$OpaqueBehavior[i].role
Specification	specification	BehavioralFeature	Behavior	\$OpaqueBehavior[i].specification
Substitution	substitution	Substitution[]	Classifier	\$OpaqueBehavior[i].substitution
Super Class	superClass	Class[]	Class	\$OpaqueBehavior[i].superClass
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$OpaqueBehavior[i].supplierDependency
Sync Element	syncElement	Element	Element	\$OpaqueBehavior[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$OpaqueBehavior[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$OpaqueBehavior[i].templateParameter
Use Case	useCase	UseCase[]	Classifier	\$OpaqueBehavior[i].useCase
Visibility	visibility	VisibilityKind	PackageableElement	\$OpaqueBehavior[i].visibility

159. OpaqueExpression

Generalization

- [ValueSpecification](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$OpaqueExpression[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$OpaqueExpression[i].appliedStereotypeInstance
Behavior	behavior	Behavior	OpaqueExpression	\$OpaqueExpression[i].behavior
Body	body	String[]	OpaqueExpression	\$OpaqueExpression[i].body
classType	classType	Class	BaseElement	\$OpaqueExpression[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$OpaqueExpression[i].clientDependency
isEditable	editable	boolean	BaseElement	\$OpaqueExpression[i].editable
Expression	expression	Expression	ValueSpecification	\$OpaqueExpression[i].expression
humanName	humanName	String	BaseElement	\$OpaqueExpression[i].humanName
humanType	humanType	String	BaseElement	\$OpaqueExpression[i].humanType
isInvalid	invalid	boolean	ModelObject	\$OpaqueExpression[i].invalid
Language	language	String[]	OpaqueExpression	\$OpaqueExpression[i].language
localID	localID	String	ModelObject	\$OpaqueExpression[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$OpaqueExpression[i].mdExtensions
Name	name	String	NamedElement	\$OpaqueExpression[i].name

Name Expression	nameExpression	StringExpression	NamedElement	\$OpaqueExpression[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$OpaqueExpression[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$OpaqueExpression[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$OpaqueExpression[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$OpaqueExpression[i].ownedElement
Owner	owner	Element	Element	\$OpaqueExpression[i].owner
Owning Constraint	owningConstraint	Constraint	ValueSpecification	\$OpaqueExpression[i].owningConstraint
Owning Instance Spec	owningInstanceSpec	InstanceSpecification	ValueSpecification	\$OpaqueExpression[i].owningInstanceSpec
Owning Lower	owningLower	MultiplicityElement	ValueSpecification	\$OpaqueExpression[i].owningLower
Owning Package	owningPackage	Package	PackageableElement	\$OpaqueExpression[i].owningPackage
Owning Parameter	owningParameter	Parameter	ValueSpecification	\$OpaqueExpression[i].owningParameter
Owning Property	owningProperty	Property	ValueSpecification	\$OpaqueExpression[i].owningProperty
Owning Slot	owningSlot	Slot	ValueSpecification	\$OpaqueExpression[i].owningSlot
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$OpaqueExpression[i].owningTemplateParameter
Owning Upper	owningUpper	MultiplicityElement	ValueSpecification	\$OpaqueExpression[i].owningUpper
Qualified Name	qualifiedName	String	NamedElement	\$OpaqueExpression[i].qualifiedName
Result	result	Parameter	OpaqueExpression	\$OpaqueExpression[i].result
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$OpaqueExpression[i].supplierDependency
Sync Element	syncElement	Element	Element	\$OpaqueExpression[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$OpaqueExpression[i].templateParameter
Type	type	Type	TypedElement	\$OpaqueExpression[i].type
Visibility	visibility	VisibilityKind	PackageableElement	\$OpaqueExpression[i].visibility

160. Operation

Generalization

- [BehavioralFeature](#)
- [ParameterableElement](#)
- [TemplateableElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$Operation[i].ID
Class	UMLClass	Class	Operation	\$Operation[i].UMLClass
Is Abstract	abstract	boolean	BehavioralFeature	\$Operation[i].abstract
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Operation[i].appliedStereotypeInstance
Body Condition	bodyCondition	Constraint	Operation	\$Operation[i].bodyCondition
classType	classType	Class	BaseElement	\$Operation[i].classType

Client Dependency	clientDependency	Dependency[]	NamedElement	\$Operation[i].clientDependency
Concurrency	concurrency	CallConcurrencyKind	BehavioralFeature	\$Operation[i].concurrency
Datatype	datatype	DataType	Operation	\$Operation[i].datatype
isEditable	editable	boolean	BaseElement	\$Operation[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$Operation[i].elementImport
Featuring Classifier	featuringClassifier	Classifier	Feature	\$Operation[i].featuringClassifier
humanName	humanName	String	BaseElement	\$Operation[i].humanName
humanType	humanType	String	BaseElement	\$Operation[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Operation[i].importedMember
Interface	interface	Interface	Operation	\$Operation[i].interface
isInvalid	invalid	boolean	ModelObject	\$Operation[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$Operation[i].leaf
localID	localID	String	ModelObject	\$Operation[i].localID
Lower	lower	int	Operation	\$Operation[i].lower
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Operation[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Operation[i].member
Method	method	Behavior[]	BehavioralFeature	\$Operation[i].method
Name	name	String	NamedElement	\$Operation[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Operation[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Operation[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Operation[i].objectParent
Is Ordered	ordered	boolean	Operation	\$Operation[i].ordered
Owned Comment	ownedComment	Comment[]	Element	\$Operation[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Operation[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Operation[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$Operation[i].ownedMember
Owned Parameter	ownedParameter	Parameter[]	Operation	\$Operation[i].ownedParameter
Owned Parameter Set	ownedParameterSet	ParameterSet[]	BehavioralFeature	\$Operation[i].ownedParameterSet
Owned Rule	ownedRule	Constraint[]	Namespace	\$Operation[i].ownedRule
Owned Template Signature	ownedTemplateSignature	TemplateSignature	TemplateableElement	\$Operation[i].ownedTemplateSignature
Owner	owner	Element	Element	\$Operation[i].owner
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Operation[i].owningTemplateParameter
Package Import	packageImport	PackageImport[]	Namespace	\$Operation[i].packageImport
Postcondition	postcondition	Constraint[]	Operation	\$Operation[i].postcondition
Precondition	precondition	Constraint[]	Operation	\$Operation[i].precondition
Qualified Name	qualifiedName	String	NamedElement	\$Operation[i].qualifiedName
Is Query	query	boolean	Operation	\$Operation[i].query
Raised Exception	raisedException	Type[]	Operation	\$Operation[i].raisedException
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Operation[i].redefinedElement
Redefined Operation	redefinedOperation	Operation[]	Operation	\$Operation[i].redefinedOperation

Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$Operation[i].redefinitionContext</code>
Is Static	static	boolean	Feature	<code>\$Operation[i].static</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Operation[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Operation[i].syncElement</code>
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	<code>\$Operation[i].templateBinding</code>
Template Parameter	templateParameter	OperationTemplateParameter	Operation	<code>\$Operation[i].templateParameter</code>
Type	type	Type	Operation	<code>\$Operation[i].type</code>
Is Unique	unique	boolean	Operation	<code>\$Operation[i].unique</code>
Upper	upper	int	Operation	<code>\$Operation[i].upper</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$Operation[i].visibility</code>

161. OperationTemplateParameter

Generalization

- [TemplateParameter](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$OperationTemplateParameter[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$OperationTemplateParameter[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$OperationTemplateParameter[i].classType</code>
Default	default	ParameterableElement	TemplateParameter	<code>\$OperationTemplateParameter[i].default</code>
isEditable	editable	boolean	BaseElement	<code>\$OperationTemplateParameter[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$OperationTemplateParameter[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$OperationTemplateParameter[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$OperationTemplateParameter[i].invalid</code>
localID	localID	String	ModelObject	<code>\$OperationTemplateParameter[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$OperationTemplateParameter[i].mdExtensions</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$OperationTemplateParameter[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$OperationTemplateParameter[i].ownedComment</code>
Owned Default	ownedDefault	ParameterableElement	TemplateParameter	<code>\$OperationTemplateParameter[i].ownedDefault</code>

Owned Element	ownedElement	Element[]	Element	\$OperationTemplateParameter[i].ownedElement
Owned Parametered Element	ownedParameteredElement	ParameterableElement	TemplateParameter	\$OperationTemplateParameter[i].ownedParameteredElement
Owner	owner	Element	Element	\$OperationTemplateParameter[i].owner
Parametered Element	parameteredElement	Operation	OperationTemplateParameter	\$OperationTemplateParameter[i].parameteredElement
Signature	signature	TemplateSignature	TemplateParameter	\$OperationTemplateParameter[i].signature
Sync Element	syncElement	Element	Element	\$OperationTemplateParameter[i].syncElement

162. OutputPin

Generalization

- [Pin](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$OutputPin[i].ID
Activity	activity	Activity	ActivityNode	\$OutputPin[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$OutputPin[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$OutputPin[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$OutputPin[i].clientDependency
Is Control	control	boolean	Pin	\$OutputPin[i].control
Is Control Type	controlType	boolean	ObjectNode	\$OutputPin[i].controlType
isEditable	editable	boolean	BaseElement	\$OutputPin[i].editable
humanName	humanName	String	BaseElement	\$OutputPin[i].humanName
humanType	humanType	String	BaseElement	\$OutputPin[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$OutputPin[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$OutputPin[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$OutputPin[i].inPartition
In State	inState	State[]	ObjectNode	\$OutputPin[i].inState
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$OutputPin[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$OutputPin[i].incoming
isInvalid	invalid	boolean	ModelObject	\$OutputPin[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$OutputPin[i].leaf
localID	localID	String	ModelObject	\$OutputPin[i].localID
Lower	lower	int	MultiplicityElement	\$OutputPin[i].lower

Lower Value	lowerValue	ValueSpecification	MultiplicityElement	<code>\$OutputPin[i].lowerValue</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$OutputPin[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$OutputPin[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$OutputPin[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$OutputPin[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$OutputPin[i].objectParent</code>
Is Ordered	ordered	boolean	MultiplicityElement	<code>\$OutputPin[i].ordered</code>
Ordering	ordering	ObjectNodeOrderingKind	ObjectNode	<code>\$OutputPin[i].ordering</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$OutputPin[i].outgoing</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$OutputPin[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$OutputPin[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$OutputPin[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$OutputPin[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$OutputPin[i].redefinedElement</code>
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	<code>\$OutputPin[i].redefinedNode</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$OutputPin[i].redefinitionContext</code>
Selection	selection	Behavior	ObjectNode	<code>\$OutputPin[i].selection</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$OutputPin[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$OutputPin[i].syncElement</code>
Type	type	Type	TypedElement	<code>\$OutputPin[i].type</code>
Is Unique	unique	boolean	MultiplicityElement	<code>\$OutputPin[i].unique</code>
Upper	upper	int	MultiplicityElement	<code>\$OutputPin[i].upper</code>
Upper Bound	upperBound	ValueSpecification	ObjectNode	<code>\$OutputPin[i].upperBound</code>
Upper Value	upperValue	ValueSpecification	MultiplicityElement	<code>\$OutputPin[i].upperValue</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$OutputPin[i].visibility</code>

163. Package

Generalization

- [Namespace](#)
- [PackageableElement](#)
- [TemplateableElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$Package[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Package[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$Package[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Package[i].clientDependency</code>

isEditable	editable	boolean	BaseElement	\$Package[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$Package[i].elementImport
humanName	humanName	String	BaseElement	\$Package[i].humanName
humanType	humanType	String	BaseElement	\$Package[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Package[i].importedMember
isInvalid	invalid	boolean	ModelObject	\$Package[i].invalid
localID	localID	String	ModelObject	\$Package[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Package[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Package[i].member
Name	name	String	NamedElement	\$Package[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Package[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Package[i].namespace
Nested Package	nestedPackage	Package[]	Package	\$Package[i].nestedPackage
Nesting Package	nestingPackage	Package	Package	\$Package[i].nestingPackage
objectParent	objectParent	BaseElement	BaseElement	\$Package[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Package[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Package[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Package[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$Package[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$Package[i].ownedRule
Owned Stereotype	ownedStereotype	Stereotype[]	Package	\$Package[i].ownedStereotype
Owned Template Signature	ownedTemplateSignature	TemplateSignature	TemplateableElement	\$Package[i].ownedTemplateSignature
Owned Type	ownedType	Type[]	Package	\$Package[i].ownedType
Owner	owner	Element	Element	\$Package[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Package[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Package[i].owningTemplateParameter
Package Import	packageImport	PackageImport[]	Namespace	\$Package[i].packageImport
Package Merge	packageMerge	PackageMerge[]	Package	\$Package[i].packageMerge
Packaged Element	packagedElement	PackageableElement[]	Package	\$Package[i].packagedElement
Profile Application	profileApplication	ProfileApplication[]	Package	\$Package[i].profileApplication
Qualified Name	qualifiedName	String	NamedElement	\$Package[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Package[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Package[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$Package[i].templateBinding
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$Package[i].templateParameter
uRI	uRI	String	Package	\$Package[i].uRI
Visibility	visibility	VisibilityKind	PackageableElement	\$Package[i].visibility

164. PackageableElement

Generalization

- [NamedElement](#)
- [ParameterableElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$PackageableElement[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$PackageableElement[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$PackageableElement[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$PackageableElement[i].clientDependency
isEditable	editable	boolean	BaseElement	\$PackageableElement[i].editable
humanName	humanName	String	BaseElement	\$PackageableElement[i].humanName
humanType	humanType	String	BaseElement	\$PackageableElement[i].humanType
isInvalid	invalid	boolean	ModelObject	\$PackageableElement[i].invalid
localID	localID	String	ModelObject	\$PackageableElement[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$PackageableElement[i].mdExtensions
Name	name	String	NamedElement	\$PackageableElement[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$PackageableElement[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$PackageableElement[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$PackageableElement[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$PackageableElement[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$PackageableElement[i].ownedElement
Owner	owner	Element	Element	\$PackageableElement[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$PackageableElement[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$PackageableElement[i].owningTemplateParameter
Qualified Name	qualifiedName	String	NamedElement	\$PackageableElement[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$PackageableElement[i].supplierDependency
Sync Element	syncElement	Element	Element	\$PackageableElement[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$PackageableElement[i].templateParameter
Visibility	visibility	VisibilityKind	PackageableElement	\$PackageableElement[i].visibility

165. PackageImport

Generalization

- DirectedRelationship

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$PackageImport[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$PackageImport[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$PackageImport[i].classType</code>
isEditable	editable	boolean	BaseElement	<code>\$PackageImport[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$PackageImport[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$PackageImport[i].humanType</code>
Imported Package	importedPackage	Package	PackageImport	<code>\$PackageImport[i].importedPackage</code>
Importing Namespace	importingNamespace	Namespace	PackageImport	<code>\$PackageImport[i].importingNamespace</code>
isInvalid	invalid	boolean	ModelObject	<code>\$PackageImport[i].invalid</code>
localID	localID	String	ModelObject	<code>\$PackageImport[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$PackageImport[i].mdExtensions</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$PackageImport[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$PackageImport[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$PackageImport[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$PackageImport[i].owner</code>
Related Element	relatedElement	Element[]	Relationship	<code>\$PackageImport[i].relatedElement</code>
Source	source	Element[]	DirectedRelationship	<code>\$PackageImport[i].source</code>
Sync Element	syncElement	Element	Element	<code>\$PackageImport[i].syncElement</code>
Target	target	Element[]	DirectedRelationship	<code>\$PackageImport[i].target</code>
Visibility	visibility	VisibilityKind	PackageImport	<code>\$PackageImport[i].visibility</code>

166. PackageMerge

Generalization

- DirectedRelationship

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$PackageMerge[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$PackageMerge[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$PackageMerge[i].classType</code>
isEditable	editable	boolean	BaseElement	<code>\$PackageMerge[i].editable</code>

humanName	humanName	String	BaseElement	\$PackageMerge[i].humanName
humanType	humanType	String	BaseElement	\$PackageMerge[i].humanType
isInvalid	invalid	boolean	ModelObject	\$PackageMerge[i].invalid
localID	localID	String	ModelObject	\$PackageMerge[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$PackageMerge[i].mdExtensions
Merged Package	mergedPackage	Package	PackageMerge	\$PackageMerge[i].mergedPackage
objectParent	objectParent	BaseElement	BaseElement	\$PackageMerge[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$PackageMerge[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$PackageMerge[i].ownedElement
Owner	owner	Element	Element	\$PackageMerge[i].owner
Receiving Package	receivingPackage	Package	PackageMerge	\$PackageMerge[i].receivingPackage
Related Element	relatedElement	Element[]	Relationship	\$PackageMerge[i].relatedElement
Source	source	Element[]	DirectedRelationship	\$PackageMerge[i].source
Sync Element	syncElement	Element	Element	\$PackageMerge[i].syncElement
Target	target	Element[]	DirectedRelationship	\$PackageMerge[i].target

167. Parameter

Generalization

- [ConnectableElement](#)
- [MultiplicityElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Parameter[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Parameter[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Parameter[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Parameter[i].clientDependency
Default	default	String	Parameter	\$Parameter[i].default
Default Value	defaultValue	ValueSpecification	Parameter	\$Parameter[i].defaultValue
Direction	direction	ParameterDirectionKind	Parameter	\$Parameter[i].direction
isEditable	editable	boolean	BaseElement	\$Parameter[i].editable
Effect	effect	ParameterEffectKind	Parameter	\$Parameter[i].effect
End	end	ConnectorEnd[]	ConnectableElement	\$Parameter[i].end
Is Exception	exception	boolean	Parameter	\$Parameter[i].exception
humanName	humanName	String	BaseElement	\$Parameter[i].humanName
humanType	humanType	String	BaseElement	\$Parameter[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Parameter[i].invalid
localID	localID	String	ModelObject	\$Parameter[i].localID

Lower	lower	int	MultiplicityElement	<code>\$Parameter[i].lower</code>
Lower Value	lowerValue	ValueSpecification	MultiplicityElement	<code>\$Parameter[i].lowerValue</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$Parameter[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$Parameter[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Parameter[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Parameter[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Parameter[i].objectParent</code>
Operation	operation	Operation	Parameter	<code>\$Parameter[i].operation</code>
Is Ordered	ordered	boolean	MultiplicityElement	<code>\$Parameter[i].ordered</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Parameter[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Parameter[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$Parameter[i].owner</code>
Owner Formal Param	ownerFormalParam	BehavioralFeature	Parameter	<code>\$Parameter[i].ownerFormalParam</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$Parameter[i].owningTemplateParameter</code>
Parameter Set	parameterSet	ParameterSet[]	Parameter	<code>\$Parameter[i].parameterSet</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Parameter[i].qualifiedName</code>
Is Stream	stream	boolean	Parameter	<code>\$Parameter[i].stream</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Parameter[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Parameter[i].syncElement</code>
Template Parameter	templateParameter	ConnectableElementTemplateParameter	ConnectableElement	<code>\$Parameter[i].templateParameter</code>
Type	type	Type	TypedElement	<code>\$Parameter[i].type</code>
Is Unique	unique	boolean	MultiplicityElement	<code>\$Parameter[i].unique</code>
Upper	upper	int	MultiplicityElement	<code>\$Parameter[i].upper</code>
Upper Value	upperValue	ValueSpecification	MultiplicityElement	<code>\$Parameter[i].upperValue</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$Parameter[i].visibility</code>

168. ParameterableElement

Generalization

- [Element](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$ParameterableElement[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$ParameterableElement[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$ParameterableElement[i].classType</code>
isEditable	editable	boolean	BaseElement	<code>\$ParameterableElement[i].editable</code>

humanName	humanName	String	BaseElement	\$ParameterableElement[i].humanName
humanType	humanType	String	BaseElement	\$ParameterableElement[i].humanType
isInvalid	invalid	boolean	ModelObject	\$ParameterableElement[i].invalid
localID	localID	String	ModelObject	\$ParameterableElement[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ParameterableElement[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$ParameterableElement[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$ParameterableElement[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ParameterableElement[i].ownedElement
Owner	owner	Element	Element	\$ParameterableElement[i].owner
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$ParameterableElement[i].owningTemplateParameter
Sync Element	syncElement	Element	Element	\$ParameterableElement[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$ParameterableElement[i].templateParameter

169. ParameterSet

Generalization

- [NamedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ParameterSet[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ParameterSet[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ParameterSet[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ParameterSet[i].clientDependency
Condition	condition	Constraint[]	ParameterSet	\$ParameterSet[i].condition
isEditable	editable	boolean	BaseElement	\$ParameterSet[i].editable
humanName	humanName	String	BaseElement	\$ParameterSet[i].humanName
humanType	humanType	String	BaseElement	\$ParameterSet[i].humanType
isInvalid	invalid	boolean	ModelObject	\$ParameterSet[i].invalid
localID	localID	String	ModelObject	\$ParameterSet[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ParameterSet[i].mdExtensions
Name	name	String	NamedElement	\$ParameterSet[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ParameterSet[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ParameterSet[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ParameterSet[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$ParameterSet[i].ownedComment

Owned Element	ownedElement	Element[]	Element	\$ParameterSet[i].ownedElement
Owner	owner	Element	Element	\$ParameterSet[i].owner
Parameter	parameter	Parameter[]	ParameterSet	\$ParameterSet[i].parameter
Qualified Name	qualifiedName	String	NamedElement	\$ParameterSet[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ParameterSet[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ParameterSet[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$ParameterSet[i].visibility

170. PartDecomposition

Generalization

- [InteractionUse](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$PartDecomposition[i].ID
Actual Gate	actualGate	Gate[]	InteractionUse	\$PartDecomposition[i].actualGate
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$PartDecomposition[i].appliedStereotypeInstance
Argument	argument	ValueSpecification[]	InteractionUse	\$PartDecomposition[i].argument
classType	classType	Class	BaseElement	\$PartDecomposition[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$PartDecomposition[i].clientDependency
Covered	covered	Lifeline[]	InteractionFragment	\$PartDecomposition[i].covered
isEditable	editable	boolean	BaseElement	\$PartDecomposition[i].editable
Enclosing Interaction	enclosingInteraction	Interaction	InteractionFragment	\$PartDecomposition[i].enclosingInteraction
Enclosing Operand	enclosingOperand	InteractionOperand	InteractionFragment	\$PartDecomposition[i].enclosingOperand
General Ordering	generalOrdering	GeneralOrdering[]	InteractionFragment	\$PartDecomposition[i].generalOrdering
humanName	humanName	String	BaseElement	\$PartDecomposition[i].humanName
humanType	humanType	String	BaseElement	\$PartDecomposition[i].humanType
isInvalid	invalid	boolean	ModelObject	\$PartDecomposition[i].invalid
localID	localID	String	ModelObject	\$PartDecomposition[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$PartDecomposition[i].mdExtensions
Name	name	String	NamedElement	\$PartDecomposition[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$PartDecomposition[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$PartDecomposition[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$PartDecomposition[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$PartDecomposition[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$PartDecomposition[i].ownedElement
Owner	owner	Element	Element	\$PartDecomposition[i].owner

Qualified Name	qualifiedName	String	NamedElement	\$PartDecomposition[i].qualifiedName
Refers To	refersTo	Interaction	InteractionUse	\$PartDecomposition[i].refersTo
Return Value	returnValue	ValueSpecification	InteractionUse	\$PartDecomposition[i].returnValue
Return Value Recipient	returnValueRecipient	Property	InteractionUse	\$PartDecomposition[i].returnValueRecipient
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$PartDecomposition[i].supplierDependency
Sync Element	syncElement	Element	Element	\$PartDecomposition[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$PartDecomposition[i].visibility

171. Pin

Generalization

- [MultiplicityElement](#)
- [ObjectNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$Pin[i].ID
Activity	activity	Activity	ActivityNode	\$Pin[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Pin[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Pin[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Pin[i].clientDependency
Is Control	control	boolean	Pin	\$Pin[i].control
Is Control Type	controlType	boolean	ObjectNode	\$Pin[i].controlType
isEditable	editable	boolean	BaseElement	\$Pin[i].editable
humanName	humanName	String	BaseElement	\$Pin[i].humanName
humanType	humanType	String	BaseElement	\$Pin[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$Pin[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$Pin[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$Pin[i].inPartition
In State	inState	State[]	ObjectNode	\$Pin[i].inState
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$Pin[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$Pin[i].incoming
isInvalid	invalid	boolean	ModelObject	\$Pin[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$Pin[i].leaf
localID	localID	String	ModelObject	\$Pin[i].localID
Lower	lower	int	MultiplicityElement	\$Pin[i].lower
Lower Value	lowerValue	ValueSpecification	MultiplicityElement	\$Pin[i].lowerValue
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	\$Pin[i].mdExtensions

Name	name	String	NamedElement	\$Pin[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Pin[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Pin[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Pin[i].objectParent
Is Ordered	ordered	boolean	MultiplicityElement	\$Pin[i].ordered
Ordering	ordering	ObjectNodeOrderingKind	ObjectNode	\$Pin[i].ordering
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$Pin[i].outgoing
Owned Comment	ownedComment	Comment[]	Element	\$Pin[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Pin[i].ownedElement
Owner	owner	Element	Element	\$Pin[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$Pin[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Pin[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$Pin[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$Pin[i].redefinitionContext
Selection	selection	Behavior	ObjectNode	\$Pin[i].selection
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Pin[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Pin[i].syncElement
Type	type	Type	TypedElement	\$Pin[i].type
Is Unique	unique	boolean	MultiplicityElement	\$Pin[i].unique
Upper	upper	int	MultiplicityElement	\$Pin[i].upper
Upper Bound	upperBound	ValueSpecification	ObjectNode	\$Pin[i].upperBound
Upper Value	upperValue	ValueSpecification	MultiplicityElement	\$Pin[i].upperValue
Visibility	visibility	VisibilityKind	NamedElement	\$Pin[i].visibility

172. Port

Generalization

- [Property](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Port[i].ID
Class	UMLClass	Class	Property	\$Port[i].UMLClass
Aggregation	aggregation	AggregationKind	Property	\$Port[i].aggregation
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Port[i].appliedStereotypeInstance
Association	association	Association	Property	\$Port[i].association
Association End	associationEnd	Property	Property	\$Port[i].associationEnd
Is Behavior	behavior	boolean	Port	\$Port[i].behavior
classType	classType	Class	BaseElement	\$Port[i].classType

Classifier	classifier	Classifier	Property	<code>\$Port[i].classifier</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Port[i].clientDependency</code>
Is Composite	composite	boolean	Property	<code>\$Port[i].composite</code>
Is Conjugated	conjugated	boolean	Port	<code>\$Port[i].conjugated</code>
Datatype	datatype	DataType	Property	<code>\$Port[i].datatype</code>
Default Value	defaultValue	ValueSpecification	Property	<code>\$Port[i].defaultValue</code>
Deployed Element	deployedElement	PackageableElement[]	DeploymentTarget	<code>\$Port[i].deployedElement</code>
Deployment	deployment	Deployment[]	DeploymentTarget	<code>\$Port[i].deployment</code>
Is Derived	derived	boolean	Property	<code>\$Port[i].derived</code>
Is Derived Union	derivedUnion	boolean	Property	<code>\$Port[i].derivedUnion</code>
isEditable	editable	boolean	BaseElement	<code>\$Port[i].editable</code>
End	end	ConnectorEnd[]	ConnectableElement	<code>\$Port[i].end</code>
Featuring Classifier	featuringClassifier	Classifier	Feature	<code>\$Port[i].featuringClassifier</code>
humanName	humanName	String	BaseElement	<code>\$Port[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Port[i].humanType</code>
Interface	interface	Interface	Property	<code>\$Port[i].interface</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Port[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$Port[i].leaf</code>
localID	localID	String	ModelObject	<code>\$Port[i].localID</code>
Lower	lower	int	MultiplicityElement	<code>\$Port[i].lower</code>
Lower Value	lowerValue	ValueSpecification	MultiplicityElement	<code>\$Port[i].lowerValue</code>
mdExtensions	mdExtensions	MDExtension[]	MDObject	<code>\$Port[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$Port[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Port[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Port[i].namespace</code>
isNavigable	navigable	boolean	Property	<code>\$Port[i].navigable</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Port[i].objectParent</code>
Opposite	opposite	Property	Property	<code>\$Port[i].opposite</code>
Is Ordered	ordered	boolean	MultiplicityElement	<code>\$Port[i].ordered</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Port[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Port[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$Port[i].owner</code>
Owning Association	owningAssociation	Association	Property	<code>\$Port[i].owningAssociation</code>
Owning Signal	owningSignal	Signal	Property	<code>\$Port[i].owningSignal</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$Port[i].owningTemplateParameter</code>
Protocol	protocol	ProtocolStateMachine	Port	<code>\$Port[i].protocol</code>
Provided	provided	Interface[]	Port	<code>\$Port[i].provided</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Port[i].qualifiedName</code>
Qualifier	qualifier	Property[]	Property	<code>\$Port[i].qualifier</code>
Is Read Only	readOnly	boolean	StructuralFeature	<code>\$Port[i].readOnly</code>

Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Port[i].redefinedElement
Redefined Port	redefinedPort	Port[]	Port	\$Port[i].redefinedPort
Redefined Property	redefinedProperty	Property[]	Property	\$Port[i].redefinedProperty
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$Port[i].redefinitionContext
Required	required	Interface[]	Port	\$Port[i].required
Is Service	service	boolean	Port	\$Port[i].service
Is Static	static	boolean	Feature	\$Port[i].static
Subsetted Property	subsettedProperty	Property[]	Property	\$Port[i].subsettedProperty
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Port[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Port[i].syncElement
Template Parameter	templateParameter	ConnectableElementTemplateParameter	ConnectableElement	\$Port[i].templateParameter
Type	type	Type	TypedElement	\$Port[i].type
Is Unique	unique	boolean	MultiplicityElement	\$Port[i].unique
Upper	upper	int	MultiplicityElement	\$Port[i].upper
Upper Value	upperValue	ValueSpecification	MultiplicityElement	\$Port[i].upperValue
Visibility	visibility	VisibilityKind	NamedElement	\$Port[i].visibility

173. PrimitiveType

Generalization

- [DataType](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$PrimitiveType[i].ID
Class	UMLClass	Class	Classifier	\$PrimitiveType[i].UMLClass
Is Abstract	abstract	boolean	Classifier	\$PrimitiveType[i].abstract
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$PrimitiveType[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$PrimitiveType[i].attribute
classType	classType	Class	BaseElement	\$PrimitiveType[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$PrimitiveType[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$PrimitiveType[i].collaborationUse
isEditable	editable	boolean	BaseElement	\$PrimitiveType[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$PrimitiveType[i].elementImport
Feature	feature	Feature[]	Classifier	\$PrimitiveType[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$PrimitiveType[i].finalSpecialization
General	general	Classifier[]	Classifier	\$PrimitiveType[i].general
Generalization	generalization	Generalization[]	Classifier	\$PrimitiveType[i].generalization

humanName	humanName	String	BaseElement	\$PrimitiveType[i].humanName
humanType	humanType	String	BaseElement	\$PrimitiveType[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$PrimitiveType[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$PrimitiveType[i].inheritedMember
isInvalid	invalid	boolean	ModelObject	\$PrimitiveType[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$PrimitiveType[i].leaf
localID	localID	String	ModelObject	\$PrimitiveType[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$PrimitiveType[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$PrimitiveType[i].member
Name	name	String	NamedElement	\$PrimitiveType[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$PrimitiveType[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$PrimitiveType[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$PrimitiveType[i].objectParent
Owned Attribute	ownedAttribute	Property[]	DataType	\$PrimitiveType[i].ownedAttribute
Owned Comment	ownedComment	Comment[]	Element	\$PrimitiveType[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$PrimitiveType[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$PrimitiveType[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$PrimitiveType[i].ownedMember
Owned Operation	ownedOperation	Operation[]	DataType	\$PrimitiveType[i].ownedOperation
Owned Rule	ownedRule	Constraint[]	Namespace	\$PrimitiveType[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$PrimitiveType[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$PrimitiveType[i].ownedUseCase
Owner	owner	Element	Element	\$PrimitiveType[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$PrimitiveType[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$PrimitiveType[i].owningTemplateParameter
Package	package	Package	Type	\$PrimitiveType[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$PrimitiveType[i].packageImport
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$PrimitiveType[i].powertypeExtent
Qualified Name	qualifiedName	String	NamedElement	\$PrimitiveType[i].qualifiedName
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$PrimitiveType[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$PrimitiveType[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$PrimitiveType[i].redefinitionContext
Representation	representation	CollaborationUse	Classifier	\$PrimitiveType[i].representation
Substitution	substitution	Substitution[]	Classifier	\$PrimitiveType[i].substitution
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$PrimitiveType[i].supplierDependency
Sync Element	syncElement	Element	Element	\$PrimitiveType[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$PrimitiveType[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParam	Classifier	\$PrimitiveType[i].templateParameter

		eter		
Use Case	useCase	UseCase[]	Classifier	\$PrimitiveType[i].useCase
Visibility	visibility	VisibilityKind	PackageableElement	\$PrimitiveType[i].visibility

174. Profile

Generalization

- [Package](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$Profile[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Profile[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Profile[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Profile[i].clientDependency
isEditable	editable	boolean	BaseElement	\$Profile[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$Profile[i].elementImport
humanName	humanName	String	BaseElement	\$Profile[i].humanName
humanType	humanType	String	BaseElement	\$Profile[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Profile[i].importedMember
isInvalid	invalid	boolean	ModelObject	\$Profile[i].invalid
localID	localID	String	ModelObject	\$Profile[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$Profile[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Profile[i].member
Metaclass Reference	metaclassReference	ElementImport[]	Profile	\$Profile[i].metaclassReference
Metamodel Reference	metamodelReference	PackageImport[]	Profile	\$Profile[i].metamodelReference
Name	name	String	NamedElement	\$Profile[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Profile[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Profile[i].namespace
Nested Package	nestedPackage	Package[]	Package	\$Profile[i].nestedPackage
Nesting Package	nestingPackage	Package	Package	\$Profile[i].nestingPackage
objectParent	objectParent	BaseElement	BaseElement	\$Profile[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Profile[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Profile[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Profile[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$Profile[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$Profile[i].ownedRule
Owned Stereotype	ownedStereotype	Stereotype[]	Package	\$Profile[i].ownedStereotype
Owned Template Signature	ownedTemplateSignature	TemplateSignature	TemplateableElement	\$Profile[i].ownedTemplateSignature

Owned Type	ownedType	Type[]	Package	\$Profile[i].ownedType
Owner	owner	Element	Element	\$Profile[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Profile[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Profile[i].owningTemplateParameter
Package Import	packageImport	PackageImport[]	Namespace	\$Profile[i].packageImport
Package Merge	packageMerge	PackageMerge[]	Package	\$Profile[i].packageMerge
Packaged Element	packagedElement	PackageableElement[]	Package	\$Profile[i].packagedElement
Profile Application	profileApplication	ProfileApplication[]	Package	\$Profile[i].profileApplication
Qualified Name	qualifiedName	String	NamedElement	\$Profile[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Profile[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Profile[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$Profile[i].templateBinding
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$Profile[i].templateParameter
uRI	uRI	String	Package	\$Profile[i].uRI
Visibility	visibility	VisibilityKind	PackageableElement	\$Profile[i].visibility

175. ProfileApplication

Generalization

- DirectedRelationship

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ProfileApplication[i].ID
Applied Profile	appliedProfile	Profile	ProfileApplication	\$ProfileApplication[i].appliedProfile
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ProfileApplication[i].appliedStereotypeInstance
Applying Package	applyingPackage	Package	ProfileApplication	\$ProfileApplication[i].applyingPackage
classType	classType	Class	BaseElement	\$ProfileApplication[i].classType
isEditable	editable	boolean	BaseElement	\$ProfileApplication[i].editable
humanName	humanName	String	BaseElement	\$ProfileApplication[i].humanName
humanType	humanType	String	BaseElement	\$ProfileApplication[i].humanType
isInvalid	invalid	boolean	ModelObject	\$ProfileApplication[i].invalid
localID	localID	String	ModelObject	\$ProfileApplication[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ProfileApplication[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$ProfileApplication[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$ProfileApplication[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ProfileApplication[i].ownedElement
Owner	owner	Element	Element	\$ProfileApplication[i].owner

Related Element	relatedElement	Element[]	Relationship	<code>\$ProfileApplication[i].relatedElement</code>
Source	source	Element[]	DirectedRelationship	<code>\$ProfileApplication[i].source</code>
Is Strict	strict	boolean	ProfileApplication	<code>\$ProfileApplication[i].strict</code>
Sync Element	syncElement	Element	Element	<code>\$ProfileApplication[i].syncElement</code>
Target	target	Element[]	DirectedRelationship	<code>\$ProfileApplication[i].target</code>

176. Property

Generalization

- [ConnectableElement](#)
- [DeploymentTarget](#)
- [StructuralFeature](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$Property[i].ID</code>
Class	UMLClass	Class	Property	<code>\$Property[i].UMLClass</code>
Aggregation	aggregation	AggregationKind	Property	<code>\$Property[i].aggregation</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Property[i].appliedStereotypeInstance</code>
Association	association	Association	Property	<code>\$Property[i].association</code>
Association End	associationEnd	Property	Property	<code>\$Property[i].associationEnd</code>
classType	classType	Class	BaseElement	<code>\$Property[i].classType</code>
Classifier	classifier	Classifier	Property	<code>\$Property[i].classifier</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Property[i].clientDependency</code>
Is Composite	composite	boolean	Property	<code>\$Property[i].composite</code>
Datatype	datatype	DataType	Property	<code>\$Property[i].datatype</code>
Default Value	defaultValue	ValueSpecification	Property	<code>\$Property[i].defaultValue</code>
Deployed Element	deployedElement	PackageableElement[]	DeploymentTarget	<code>\$Property[i].deployedElement</code>
Deployment	deployment	Deployment[]	DeploymentTarget	<code>\$Property[i].deployment</code>
Is Derived	derived	boolean	Property	<code>\$Property[i].derived</code>
Is Derived Union	derivedUnion	boolean	Property	<code>\$Property[i].derivedUnion</code>
isEditable	editable	boolean	BaseElement	<code>\$Property[i].editable</code>
End	end	ConnectorEnd[]	ConnectableElement	<code>\$Property[i].end</code>
Featuring Classifier	featuringClassifier	Classifier	Feature	<code>\$Property[i].featuringClassifier</code>
humanName	humanName	String	BaseElement	<code>\$Property[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Property[i].humanType</code>
Interface	interface	Interface	Property	<code>\$Property[i].interface</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Property[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$Property[i].leaf</code>

localID	localID	String	ModelObject	\$Property[i].localID
Lower	lower	int	MultiplicityElement	\$Property[i].lower
Lower Value	lowerValue	ValueSpecification	MultiplicityElement	\$Property[i].lowerValue
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Property[i].mdExtensions
Name	name	String	NamedElement	\$Property[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Property[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Property[i].namespace
isNavigable	navigable	boolean	Property	\$Property[i].navigable
objectParent	objectParent	BaseElement	BaseElement	\$Property[i].objectParent
Opposite	opposite	Property	Property	\$Property[i].opposite
Is Ordered	ordered	boolean	MultiplicityElement	\$Property[i].ordered
Owned Comment	ownedComment	Comment[]	Element	\$Property[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Property[i].ownedElement
Owner	owner	Element	Element	\$Property[i].owner
Owning Association	owningAssociation	Association	Property	\$Property[i].owningAssociation
Owning Signal	owningSignal	Signal	Property	\$Property[i].owningSignal
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Property[i].owningTemplateParameter
Qualified Name	qualifiedName	String	NamedElement	\$Property[i].qualifiedName
Qualifier	qualifier	Property[]	Property	\$Property[i].qualifier
Is Read Only	readOnly	boolean	StructuralFeature	\$Property[i].readOnly
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Property[i].redefinedElement
Redefined Property	redefinedProperty	Property[]	Property	\$Property[i].redefinedProperty
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$Property[i].redefinitionContext
Is Static	static	boolean	Feature	\$Property[i].static
Subsetted Property	subsettedProperty	Property[]	Property	\$Property[i].subsettedProperty
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Property[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Property[i].syncElement
Template Parameter	templateParameter	ConnectableElementTemplateParameter	ConnectableElement	\$Property[i].templateParameter
Type	type	Type	TypedElement	\$Property[i].type
Is Unique	unique	boolean	MultiplicityElement	\$Property[i].unique
Upper	upper	int	MultiplicityElement	\$Property[i].upper
Upper Value	upperValue	ValueSpecification	MultiplicityElement	\$Property[i].upperValue
Visibility	visibility	VisibilityKind	NamedElement	\$Property[i].visibility

177. ProtocolConformance

Generalization

- [DirectedRelationship](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$ProtocolConformance[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ProtocolConformance[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ProtocolConformance[i].classType
isEditable	editable	boolean	BaseElement	\$ProtocolConformance[i].editable
General Machine	generalMachine	ProtocolStateMachine	ProtocolConformance	\$ProtocolConformance[i].generalMachine
humanName	humanName	String	BaseElement	\$ProtocolConformance[i].humanName
humanType	humanType	String	BaseElement	\$ProtocolConformance[i].humanType
isInvalid	invalid	boolean	ModelObject	\$ProtocolConformance[i].invalid
localID	localID	String	ModelObject	\$ProtocolConformance[i].localID
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	\$ProtocolConformance[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$ProtocolConformance[i].objectParent
Owned Comment	ownedComment	COMMENT[]	Element	\$ProtocolConformance[i].ownedComment
Owned Element	ownedElement	ELEMENT[]	Element	\$ProtocolConformance[i].ownedElement
Owner	owner	Element	Element	\$ProtocolConformance[i].owner
Related Element	relatedElement	ELEMENT[]	Relationship	\$ProtocolConformance[i].relatedElement
Source	source	ELEMENT[]	DirectedRelationship	\$ProtocolConformance[i].source
Specific Machine	specificMachine	ProtocolStateMachine	ProtocolConformance	\$ProtocolConformance[i].specificMachine
Sync Element	syncElement	Element	Element	\$ProtocolConformance[i].syncElement
Target	target	ELEMENT[]	DirectedRelationship	\$ProtocolConformance[i].target

178. ProtocolStateMachine

Generalization

- [StateMachine](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$ProtocolStateMachine[i].ID
Class	UMLClass	Class	Classifier	\$ProtocolStateMachine[i].UMLClass
Is Abstract	abstract	boolean	Class	\$ProtocolStateMachine[i].abstract
Is Active	active	boolean	Class	\$ProtocolStateMachine[i].active
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ProtocolStateMachine[i].appliedStereotypeInstance
Attribute	attribute	PROPERTY[]	Classifier	\$ProtocolStateMachine[i].attribute
classType	classType	Class	BaseElement	\$ProtocolStateMachine[i].classType

Classifier Behavior	classifierBehavior	Behavior	BehavioredClassifier	\$ProtocolStateMachine[i].classifierBehavior
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ProtocolStateMachine[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$ProtocolStateMachine[i].collaborationUse
Conformance	conformance	ProtocolConformance[]	ProtocolStateMachine	\$ProtocolStateMachine[i].conformance
Connection Point	connectionPoint	Pseudostate[]	StateMachine	\$ProtocolStateMachine[i].connectionPoint
Context	context	BehavioredClassifier	Behavior	\$ProtocolStateMachine[i].context
isEditable	editable	boolean	BaseElement	\$ProtocolStateMachine[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$ProtocolStateMachine[i].elementImport
Event	event	Event[]	Behavior	\$ProtocolStateMachine[i].event
Extended State Machine	extendedStateMachine	StateMachine[]	StateMachine	\$ProtocolStateMachine[i].extendedStateMa chine
Extension	extension	Extension[]	Class	\$ProtocolStateMachine[i].extension
Feature	feature	Feature[]	Classifier	\$ProtocolStateMachine[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$ProtocolStateMachine[i].finalSpecialization
General	general	Classifier[]	Classifier	\$ProtocolStateMachine[i].general
Generalization	generalization	Generalization[]	Classifier	\$ProtocolStateMachine[i].generalization
humanName	humanName	String	BaseElement	\$ProtocolStateMachine[i].humanName
humanType	humanType	String	BaseElement	\$ProtocolStateMachine[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$ProtocolStateMachine[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$ProtocolStateMachine[i].inheritedMember
Interface	interface	Interface	ProtocolStateMachine	\$ProtocolStateMachine[i].interface
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavioredClassifier	\$ProtocolStateMachine[i].interfaceRealizati on
isInvalid	invalid	boolean	ModelObject	\$ProtocolStateMachine[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ProtocolStateMachine[i].leaf
localID	localID	String	ModelObject	\$ProtocolStateMachine[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$ProtocolStateMachine[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$ProtocolStateMachine[i].member
Name	name	String	NamedElement	\$ProtocolStateMachine[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ProtocolStateMachine[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ProtocolStateMachine[i].namespace
Nested Classifier	nestedClassifier	Classifier[]	Class	\$ProtocolStateMachine[i].nestedClassifier
objectParent	objectParent	BaseElement	BaseElement	\$ProtocolStateMachine[i].objectParent
Observation	observation	Observation[]	Behavior	\$ProtocolStateMachine[i].observation
Owned Attribute	ownedAttribute	Property[]	Class	\$ProtocolStateMachine[i].ownedAttribute
Owned Behavior	ownedBehavior	Behavior[]	BehavioredClassifier	\$ProtocolStateMachine[i].ownedBehavior
Owned Comment	ownedComment	Comment[]	Element	\$ProtocolStateMachine[i].ownedComment
Owned Connector	ownedConnector	Connector[]	StructuredClassifier	\$ProtocolStateMachine[i].ownedConnector
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$ProtocolStateMachine[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$ProtocolStateMachine[i].ownedElement

Owned Member	ownedMember	NamedElement[]	Namespace	\$ProtocolStateMachine[i].ownedMember
Owned Operation	ownedOperation	Operation[]	Class	\$ProtocolStateMachine[i].ownedOperation
Owned Parameter	ownedParameter	Parameter[]	Behavior	\$ProtocolStateMachine[i].ownedParameter
Owned Parameter Set	ownedParameterSet	ParameterSet[]	Behavior	\$ProtocolStateMachine[i].ownedParameterSet
Owned Port	ownedPort	Port[]	EncapsulatedClassifier	\$ProtocolStateMachine[i].ownedPort
Owned Reception	ownedReception	Reception[]	Class	\$ProtocolStateMachine[i].ownedReception
Owned Rule	ownedRule	Constraint[]	Namespace	\$ProtocolStateMachine[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$ProtocolStateMachine[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$ProtocolStateMachine[i].ownedUseCase
Owner	owner	Element	Element	\$ProtocolStateMachine[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$ProtocolStateMachine[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$ProtocolStateMachine[i].owningTemplateParameter
Package	package	Package	Type	\$ProtocolStateMachine[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$ProtocolStateMachine[i].packageImport
Part	part	Property[]	StructuredClassifier	\$ProtocolStateMachine[i].part
Postcondition	postcondition	Constraint[]	Behavior	\$ProtocolStateMachine[i].postcondition
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$ProtocolStateMachine[i].powertypeExtent
Precondition	precondition	Constraint[]	Behavior	\$ProtocolStateMachine[i].precondition
Qualified Name	qualifiedName	String	NamedElement	\$ProtocolStateMachine[i].qualifiedName
Redefined Behavior	redefinedBehavior	Behavior[]	Behavior	\$ProtocolStateMachine[i].redefinedBehavior
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$ProtocolStateMachine[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ProtocolStateMachine[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ProtocolStateMachine[i].redefinitionContext
Is Reentrant	reentrant	boolean	Behavior	\$ProtocolStateMachine[i].reentrant
Region	region	Region[]	StateMachine	\$ProtocolStateMachine[i].region
Representation	representation	CollaborationUse	Classifier	\$ProtocolStateMachine[i].representation
Role	role	ConnectableElement[]	StructuredClassifier	\$ProtocolStateMachine[i].role
Specification	specification	BehavioralFeature	Behavior	\$ProtocolStateMachine[i].specification
Submachine State	submachineState	State[]	StateMachine	\$ProtocolStateMachine[i].submachineState
Substitution	substitution	Substitution[]	Classifier	\$ProtocolStateMachine[i].substitution
Super Class	superClass	Class[]	Class	\$ProtocolStateMachine[i].superClass
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ProtocolStateMachine[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ProtocolStateMachine[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$ProtocolStateMachine[i].templateBinding

Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$ProtocolStateMachine[i].templateParameter
Use Case	useCase	UseCase[]	Classifier	\$ProtocolStateMachine[i].useCase
Visibility	visibility	VisibilityKind	PackageableElement	\$ProtocolStateMachine[i].visibility

179. ProtocolTransition

Generalization

- [Transition](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$ProtocolTransition[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ProtocolTransition[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ProtocolTransition[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ProtocolTransition[i].clientDependency
Container	container	Region	Transition	\$ProtocolTransition[i].container
isEditable	editable	boolean	BaseElement	\$ProtocolTransition[i].editable
Effect	effect	Behavior	Transition	\$ProtocolTransition[i].effect
Element Import	elementImport	ElementImport[]	Namespace	\$ProtocolTransition[i].elementImport
Guard	guard	Constraint	Transition	\$ProtocolTransition[i].guard
humanName	humanName	String	BaseElement	\$ProtocolTransition[i].humanName
humanType	humanType	String	BaseElement	\$ProtocolTransition[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$ProtocolTransition[i].importedMember
isInvalid	invalid	boolean	ModelObject	\$ProtocolTransition[i].invalid
Kind	kind	TransitionKind	Transition	\$ProtocolTransition[i].kind
Is Leaf	leaf	boolean	RedefinableElement	\$ProtocolTransition[i].leaf
localID	localID	String	ModelObject	\$ProtocolTransition[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$ProtocolTransition[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$ProtocolTransition[i].member
Name	name	String	NamedElement	\$ProtocolTransition[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ProtocolTransition[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ProtocolTransition[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ProtocolTransition[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$ProtocolTransition[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$ProtocolTransition[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$ProtocolTransition[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$ProtocolTransition[i].ownedMember

Owned Rule	ownedRule	Constraint[]	Namespace	\$ProtocolTransition[i].ownedRule
Owner	owner	Element	Element	\$ProtocolTransition[i].owner
Package Import	packageImport	PackageImport[]	Namespace	\$ProtocolTransition[i].packageImport
Post Condition	postCondition	Constraint	ProtocolTransition	\$ProtocolTransition[i].postCondition
Pre Condition	preCondition	Constraint	ProtocolTransition	\$ProtocolTransition[i].preCondition
Qualified Name	qualifiedName	String	NamedElement	\$ProtocolTransition[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ProtocolTransition[i].redefinedElement
Redefined Transition	redefinedTransition	Transition	Transition	\$ProtocolTransition[i].redefinedTransition
Redefinition Context	redefinitionContext	Classifier[]	Transition	\$ProtocolTransition[i].redefinitionContext
Referred	referred	Operation[]	ProtocolTransition	\$ProtocolTransition[i].referred
Source	source	Vertex	Transition	\$ProtocolTransition[i].source
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ProtocolTransition[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ProtocolTransition[i].syncElement
Target	target	Vertex	Transition	\$ProtocolTransition[i].target
Trigger	trigger	Trigger[]	Transition	\$ProtocolTransition[i].trigger
Visibility	visibility	VisibilityKind	NamedElement	\$ProtocolTransition[i].visibility

180. Pseudostate

Generalization

- [Vertex](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Pseudostate[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Pseudostate[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Pseudostate[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Pseudostate[i].clientDependency
Container	container	Region	Vertex	\$Pseudostate[i].container
isEditable	editable	boolean	BaseElement	\$Pseudostate[i].editable
humanName	humanName	String	BaseElement	\$Pseudostate[i].humanName
humanType	humanType	String	BaseElement	\$Pseudostate[i].humanType
Incoming	incoming	Transition[]	Vertex	\$Pseudostate[i].incoming
isInvalid	invalid	boolean	ModelObject	\$Pseudostate[i].invalid
Kind	kind	PseudostateKind	Pseudostate	\$Pseudostate[i].kind
Is Leaf	leaf	boolean	RedefinableElement	\$Pseudostate[i].leaf
localID	localID	String	ModelObject	\$Pseudostate[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Pseudostate[i].mdExtensions
Name	name	String	NamedElement	\$Pseudostate[i].name

Name Expression	nameExpression	StringExpression	NamedElement	\$Pseudostate[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Pseudostate[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Pseudostate[i].objectParent
Outgoing	outgoing	Transition[]	Vertex	\$Pseudostate[i].outgoing
Owned Comment	ownedComment	Comment[]	Element	\$Pseudostate[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Pseudostate[i].ownedElement
Owner	owner	Element	Element	\$Pseudostate[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$Pseudostate[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Pseudostate[i].redefinedElement
Redefined Vertex	redefinedVertex	Vertex	Vertex	\$Pseudostate[i].redefinedVertex
Redefinition Context	redefinitionContext	Classifier[]	Vertex	\$Pseudostate[i].redefinitionContext
State	state	State	Pseudostate	\$Pseudostate[i].state
State Machine	stateMachine	StateMachine	Pseudostate	\$Pseudostate[i].stateMachine
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Pseudostate[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Pseudostate[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$Pseudostate[i].visibility

181. QualifierValue

Generalization

- [Element](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$QualifierValue[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$QualifierValue[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$QualifierValue[i].classType
isEditable	editable	boolean	BaseElement	\$QualifierValue[i].editable
humanName	humanName	String	BaseElement	\$QualifierValue[i].humanName
humanType	humanType	String	BaseElement	\$QualifierValue[i].humanType
isInvalid	invalid	boolean	ModelObject	\$QualifierValue[i].invalid
localID	localID	String	ModelObject	\$QualifierValue[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$QualifierValue[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$QualifierValue[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$QualifierValue[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$QualifierValue[i].ownedElement
Owner	owner	Element	Element	\$QualifierValue[i].owner
Qualifier	qualifier	Property	QualifierValue	\$QualifierValue[i].qualifier

Sync Element	syncElement	Element	Element	\$QualifierValue[i].syncElement
Value	value	InputPin	QualifierValue	\$QualifierValue[i].value

182. RaiseExceptionAction

Generalization

- [Action](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$RaiseExceptionAction[i].ID
Activity	activity	Activity	ActivityNode	\$RaiseExceptionAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$RaiseExceptionAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$RaiseExceptionAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$RaiseExceptionAction[i].clientDependency
Context	context	Classifier	Action	\$RaiseExceptionAction[i].context
isEditable	editable	boolean	BaseElement	\$RaiseExceptionAction[i].editable
Exception	exception	InputPin	RaiseExceptionAction	\$RaiseExceptionAction[i].exception
Handler	handler	ExceptionHandler[]	ExecutableNode	\$RaiseExceptionAction[i].handler
humanName	humanName	String	BaseElement	\$RaiseExceptionAction[i].humanName
humanType	humanType	String	BaseElement	\$RaiseExceptionAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$RaiseExceptionAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$RaiseExceptionAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$RaiseExceptionAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$RaiseExceptionAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$RaiseExceptionAction[i].incoming
Input	input	InputPin[]	Action	\$RaiseExceptionAction[i].input
isInvalid	invalid	boolean	ModelObject	\$RaiseExceptionAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$RaiseExceptionAction[i].leaf
localID	localID	String	ModelObject	\$RaiseExceptionAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$RaiseExceptionAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$RaiseExceptionAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$RaiseExceptionAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$RaiseExceptionAction[i].mdExtensions
Name	name	String	NamedElement	\$RaiseExceptionAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$RaiseExceptionAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$RaiseExceptionAction[i].namespace

objectParent	objectParent	BaseElement	BaseElement	\$RaiseExceptionAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$RaiseExceptionAction[i].outgoing
Output	output	OutputPin[]	Action	\$RaiseExceptionAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$RaiseExceptionAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$RaiseExceptionAction[i].ownedElement
Owner	owner	Element	Element	\$RaiseExceptionAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$RaiseExceptionAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$RaiseExceptionAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$RaiseExceptionAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$RaiseExceptionAction[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$RaiseExceptionAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$RaiseExceptionAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$RaiseExceptionAction[i].visibility

183. ReadExtentAction

Generalization

- [Action](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ReadExtentAction[i].ID
Activity	activity	Activity	ActivityNode	\$ReadExtentAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ReadExtentAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ReadExtentAction[i].classType
Classifier	classifier	Classifier	ReadExtentAction	\$ReadExtentAction[i].classifier
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ReadExtentAction[i].clientDependency
Context	context	Classifier	Action	\$ReadExtentAction[i].context
isEditable	editable	boolean	BaseElement	\$ReadExtentAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$ReadExtentAction[i].handler
humanName	humanName	String	BaseElement	\$ReadExtentAction[i].humanName
humanType	humanType	String	BaseElement	\$ReadExtentAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ReadExtentAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ReadExtentAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ReadExtentAction[i].inPartition

In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ReadExtentAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ReadExtentAction[i].incoming
Input	input	InputPin[]	Action	\$ReadExtentAction[i].input
isInvalid	invalid	boolean	ModelObject	\$ReadExtentAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ReadExtentAction[i].leaf
localID	localID	String	ModelObject	\$ReadExtentAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$ReadExtentAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$ReadExtentAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$ReadExtentAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$ReadExtentAction[i].mdExtensions
Name	name	String	NamedElement	\$ReadExtentAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ReadExtentAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ReadExtentAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ReadExtentAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ReadExtentAction[i].outgoing
Output	output	OutputPin[]	Action	\$ReadExtentAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$ReadExtentAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ReadExtentAction[i].ownedElement
Owner	owner	Element	Element	\$ReadExtentAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ReadExtentAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ReadExtentAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ReadExtentAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ReadExtentAction[i].redefinitionContext
Result	result	OutputPin	ReadExtentAction	\$ReadExtentAction[i].result
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ReadExtentAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ReadExtentAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$ReadExtentAction[i].visibility

184. ReadIsClassifiedObjectAction

Generalization

- [Action](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDObject	\$ReadIsClassifiedObjectAction[i].ID
Activity	activity	Activity	ActivityNode	\$ReadIsClassifiedObjectAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ReadIsClassifiedObjectAction[i].appliedStereotypeInstance

classType	classType	Class	BaseElement	\$ReadIsClassifiedObjectAction[i].classType
Classifier	classifier	Classifier	ReadIsClassifiedObjectAction	\$ReadIsClassifiedObjectAction[i].classifier
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ReadIsClassifiedObjectAction[i].clientDependency
Context	context	Classifier	Action	\$ReadIsClassifiedObjectAction[i].context
Is Direct	direct	boolean	ReadIsClassifiedObjectAction	\$ReadIsClassifiedObjectAction[i].direct
isEditable	editable	boolean	BaseElement	\$ReadIsClassifiedObjectAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$ReadIsClassifiedObjectAction[i].handler
humanName	humanName	String	BaseElement	\$ReadIsClassifiedObjectAction[i].humanName
humanType	humanType	String	BaseElement	\$ReadIsClassifiedObjectAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ReadIsClassifiedObjectAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ReadIsClassifiedObjectAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ReadIsClassifiedObjectAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ReadIsClassifiedObjectAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ReadIsClassifiedObjectAction[i].incoming
Input	input	InputPin[]	Action	\$ReadIsClassifiedObjectAction[i].input
isInvalid	invalid	boolean	ModelObject	\$ReadIsClassifiedObjectAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ReadIsClassifiedObjectAction[i].leaf
localID	localID	String	ModelObject	\$ReadIsClassifiedObjectAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$ReadIsClassifiedObjectAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$ReadIsClassifiedObjectAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$ReadIsClassifiedObjectAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$ReadIsClassifiedObjectAction[i].mdExtensions
Name	name	String	NamedElement	\$ReadIsClassifiedObjectAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ReadIsClassifiedObjectAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ReadIsClassifiedObjectAction[i].namespace
Object	object	InputPin	ReadIsClassifiedObjectAction	\$ReadIsClassifiedObjectAction[i].object
objectParent	objectParent	BaseElement	BaseElement	\$ReadIsClassifiedObjectAction[i].objectParent

				nt
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$ReadIsClassifiedObjectAction[i].outgoing</code>
Output	output	OutputPin[]	Action	<code>\$ReadIsClassifiedObjectAction[i].output</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$ReadIsClassifiedObjectAction[i].ownedCo mment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$ReadIsClassifiedObjectAction[i].ownedEle ment</code>
Owner	owner	Element	Element	<code>\$ReadIsClassifiedObjectAction[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$ReadIsClassifiedObjectAction[i].qualifiedN ame</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$ReadIsClassifiedObjectAction[i].redefined Element</code>
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	<code>\$ReadIsClassifiedObjectAction[i].redefined Node</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$ReadIsClassifiedObjectAction[i].redefinitio nContext</code>
Result	result	OutputPin	ReadIsClassifiedObjectA ction	<code>\$ReadIsClassifiedObjectAction[i].result</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$ReadIsClassifiedObjectAction[i].supplierDe pendency</code>
Sync Element	syncElement	Element	Element	<code>\$ReadIsClassifiedObjectAction[i].syncElem ent</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$ReadIsClassifiedObjectAction[i].visibility</code>

185. ReadLinkAction

Generalization

- [LinkAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$ReadLinkAction[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$ReadLinkAction[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$ReadLinkAction[i].appliedStereotypeInstan ce</code>
classType	classType	Class	BaseElement	<code>\$ReadLinkAction[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$ReadLinkAction[i].clientDependency</code>
Context	context	Classifier	Action	<code>\$ReadLinkAction[i].context</code>
isEditable	editable	boolean	BaseElement	<code>\$ReadLinkAction[i].editable</code>
End Data	endData	LinkEndData[]	LinkAction	<code>\$ReadLinkAction[i].endData</code>

Handler	handler	ExceptionHandler[]	ExecutableNode	\$ReadLinkAction[i].handler
humanName	humanName	String	BaseElement	\$ReadLinkAction[i].humanName
humanType	humanType	String	BaseElement	\$ReadLinkAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ReadLinkAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ReadLinkAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ReadLinkAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ReadLinkAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ReadLinkAction[i].incoming
Input	input	InputPin[]	Action	\$ReadLinkAction[i].input
Input Value	inputValue	InputPin[]	LinkAction	\$ReadLinkAction[i].inputValue
isInvalid	invalid	boolean	ModelObject	\$ReadLinkAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ReadLinkAction[i].leaf
localID	localID	String	ModelObject	\$ReadLinkAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$ReadLinkAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$ReadLinkAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$ReadLinkAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$ReadLinkAction[i].mdExtensions
Name	name	String	NamedElement	\$ReadLinkAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ReadLinkAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ReadLinkAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ReadLinkAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ReadLinkAction[i].outgoing
Output	output	OutputPin[]	Action	\$ReadLinkAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$ReadLinkAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ReadLinkAction[i].ownedElement
Owner	owner	Element	Element	\$ReadLinkAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ReadLinkAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ReadLinkAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ReadLinkAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ReadLinkAction[i].redefinitionContext
Result	result	OutputPin	ReadLinkAction	\$ReadLinkAction[i].result
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ReadLinkAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ReadLinkAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$ReadLinkAction[i].visibility

186. ReadLinkObjectEndAction

Generalization

- Action

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$ReadLinkObjectEndAction[i].ID
Activity	activity	Activity	ActivityNode	\$ReadLinkObjectEndAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ReadLinkObjectEndAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ReadLinkObjectEndAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ReadLinkObjectEndAction[i].clientDependency
Context	context	Classifier	Action	\$ReadLinkObjectEndAction[i].context
isEditable	editable	boolean	BaseElement	\$ReadLinkObjectEndAction[i].editable
End	end	Property	ReadLinkObjectEndAction	\$ReadLinkObjectEndAction[i].end
Handler	handler	ExceptionHandler[]	ExecutableNode	\$ReadLinkObjectEndAction[i].handler
humanName	humanName	String	BaseElement	\$ReadLinkObjectEndAction[i].humanName
humanType	humanType	String	BaseElement	\$ReadLinkObjectEndAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ReadLinkObjectEndAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ReadLinkObjectEndAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ReadLinkObjectEndAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ReadLinkObjectEndAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ReadLinkObjectEndAction[i].incoming
Input	input	InputPin[]	Action	\$ReadLinkObjectEndAction[i].input
isInvalid	invalid	boolean	ModelObject	\$ReadLinkObjectEndAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ReadLinkObjectEndAction[i].leaf
localID	localID	String	ModelObject	\$ReadLinkObjectEndAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$ReadLinkObjectEndAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$ReadLinkObjectEndAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$ReadLinkObjectEndAction[i].locallyReentrant
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	\$ReadLinkObjectEndAction[i].mdExtensions
Name	name	String	NamedElement	\$ReadLinkObjectEndAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ReadLinkObjectEndAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ReadLinkObjectEndAction[i].namespace

Object	object	InputPin	ReadLinkObjectEndAction	<code>\$ReadLinkObjectEndAction[i].object</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$ReadLinkObjectEndAction[i].objectParent</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$ReadLinkObjectEndAction[i].outgoing</code>
Output	output	OutputPin[]	Action	<code>\$ReadLinkObjectEndAction[i].output</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$ReadLinkObjectEndAction[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$ReadLinkObjectEndAction[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$ReadLinkObjectEndAction[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$ReadLinkObjectEndAction[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$ReadLinkObjectEndAction[i].redefinedElement</code>
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	<code>\$ReadLinkObjectEndAction[i].redefinedNode</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$ReadLinkObjectEndAction[i].redefinitionContext</code>
Result	result	OutputPin	ReadLinkObjectEndAction	<code>\$ReadLinkObjectEndAction[i].result</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$ReadLinkObjectEndAction[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$ReadLinkObjectEndAction[i].syncElement</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$ReadLinkObjectEndAction[i].visibility</code>

187. ReadLinkObjectEndQualifierAction

Generalization

- [Action](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$ReadLinkObjectEndQualifierAction[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$ReadLinkObjectEndQualifierAction[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$ReadLinkObjectEndQualifierAction[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$ReadLinkObjectEndQualifierAction[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$ReadLinkObjectEndQualifierAction[i].client</code>

				Dependency
Context	context	Classifier	Action	\$ReadLinkObjectEndQualifierAction[i].context
isEditable	editable	boolean	BaseElement	\$ReadLinkObjectEndQualifierAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$ReadLinkObjectEndQualifierAction[i].handler
humanName	humanName	String	BaseElement	\$ReadLinkObjectEndQualifierAction[i].humanName
humanType	humanType	String	BaseElement	\$ReadLinkObjectEndQualifierAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ReadLinkObjectEndQualifierAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ReadLinkObjectEndQualifierAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ReadLinkObjectEndQualifierAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ReadLinkObjectEndQualifierAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ReadLinkObjectEndQualifierAction[i].incoming
Input	input	InputPin[]	Action	\$ReadLinkObjectEndQualifierAction[i].input
isInvalid	invalid	boolean	ModelObject	\$ReadLinkObjectEndQualifierAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ReadLinkObjectEndQualifierAction[i].leaf
localID	localID	String	ModelObject	\$ReadLinkObjectEndQualifierAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$ReadLinkObjectEndQualifierAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$ReadLinkObjectEndQualifierAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$ReadLinkObjectEndQualifierAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$ReadLinkObjectEndQualifierAction[i].mdExtensions
Name	name	String	NamedElement	\$ReadLinkObjectEndQualifierAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ReadLinkObjectEndQualifierAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ReadLinkObjectEndQualifierAction[i].namespace
Object	object	InputPin	ReadLinkObjectEndQualif	\$ReadLinkObjectEndQualifierAction[i].objec

			ierAction	t
objectParent	objectParent	BaseElement	BaseElement	\$ReadLinkObjectEndQualifierAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ReadLinkObjectEndQualifierAction[i].outgoing
Output	output	OutputPin[]	Action	\$ReadLinkObjectEndQualifierAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$ReadLinkObjectEndQualifierAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ReadLinkObjectEndQualifierAction[i].ownedElement
Owner	owner	Element	Element	\$ReadLinkObjectEndQualifierAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ReadLinkObjectEndQualifierAction[i].qualifiedName
Qualifier	qualifier	Property	ReadLinkObjectEndQualifierAction	\$ReadLinkObjectEndQualifierAction[i].qualifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ReadLinkObjectEndQualifierAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ReadLinkObjectEndQualifierAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ReadLinkObjectEndQualifierAction[i].redefinitionContext
Result	result	OutputPin	ReadLinkObjectEndQualifierAction	\$ReadLinkObjectEndQualifierAction[i].result
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ReadLinkObjectEndQualifierAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ReadLinkObjectEndQualifierAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$ReadLinkObjectEndQualifierAction[i].visibility

188. ReadSelfAction

Generalization

- [Action](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOject	\$ReadSelfAction[i].ID

Activity	activity	Activity	ActivityNode	\$ReadSelfAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ReadSelfAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ReadSelfAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ReadSelfAction[i].clientDependency
Context	context	Classifier	Action	\$ReadSelfAction[i].context
isEditable	editable	boolean	BaseElement	\$ReadSelfAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$ReadSelfAction[i].handler
humanName	humanName	String	BaseElement	\$ReadSelfAction[i].humanName
humanType	humanType	String	BaseElement	\$ReadSelfAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ReadSelfAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ReadSelfAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ReadSelfAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ReadSelfAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ReadSelfAction[i].incoming
Input	input	InputPin[]	Action	\$ReadSelfAction[i].input
isInvalid	invalid	boolean	ModelObject	\$ReadSelfAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ReadSelfAction[i].leaf
localID	localID	String	ModelObject	\$ReadSelfAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$ReadSelfAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$ReadSelfAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$ReadSelfAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ReadSelfAction[i].mdExtensions
Name	name	String	NamedElement	\$ReadSelfAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ReadSelfAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ReadSelfAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ReadSelfAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ReadSelfAction[i].outgoing
Output	output	OutputPin[]	Action	\$ReadSelfAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$ReadSelfAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ReadSelfAction[i].ownedElement
Owner	owner	Element	Element	\$ReadSelfAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ReadSelfAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ReadSelfAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ReadSelfAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ReadSelfAction[i].redefinitionContext
Result	result	OutputPin	ReadSelfAction	\$ReadSelfAction[i].result
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ReadSelfAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ReadSelfAction[i].syncElement

Visibility	visibility	VisibilityKind	NamedElement	\$ReadSelfAction[i].visibility

189. ReadStructuralFeatureAction

Generalization

- [StructuralFeatureAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$ReadStructuralFeatureAction[i].ID
Activity	activity	Activity	ActivityNode	\$ReadStructuralFeatureAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ReadStructuralFeatureAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ReadStructuralFeatureAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ReadStructuralFeatureAction[i].clientDependency
Context	context	Classifier	Action	\$ReadStructuralFeatureAction[i].context
isEditable	editable	boolean	BaseElement	\$ReadStructuralFeatureAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$ReadStructuralFeatureAction[i].handler
humanName	humanName	String	BaseElement	\$ReadStructuralFeatureAction[i].humanName
humanType	humanType	String	BaseElement	\$ReadStructuralFeatureAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ReadStructuralFeatureAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ReadStructuralFeatureAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ReadStructuralFeatureAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ReadStructuralFeatureAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ReadStructuralFeatureAction[i].incoming
Input	input	InputPin[]	Action	\$ReadStructuralFeatureAction[i].input
isInvalid	invalid	boolean	ModelObject	\$ReadStructuralFeatureAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ReadStructuralFeatureAction[i].leaf
localID	localID	String	ModelObject	\$ReadStructuralFeatureAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$ReadStructuralFeatureAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$ReadStructuralFeatureAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$ReadStructuralFeatureAction[i].locallyReentrant

mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ReadStructuralFeatureAction[i].mdExtensions
Name	name	String	NamedElement	\$ReadStructuralFeatureAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ReadStructuralFeatureAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ReadStructuralFeatureAction[i].namespace
Object	object	InputPin	StructuralFeatureAction	\$ReadStructuralFeatureAction[i].object
objectParent	objectParent	BaseElement	BaseElement	\$ReadStructuralFeatureAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ReadStructuralFeatureAction[i].outgoing
Output	output	OutputPin[]	Action	\$ReadStructuralFeatureAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$ReadStructuralFeatureAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ReadStructuralFeatureAction[i].ownedElement
Owner	owner	Element	Element	\$ReadStructuralFeatureAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ReadStructuralFeatureAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ReadStructuralFeatureAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ReadStructuralFeatureAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ReadStructuralFeatureAction[i].redefinitionContext
Result	result	OutputPin	ReadStructuralFeatureAction	\$ReadStructuralFeatureAction[i].result
Structural Feature	structuralFeature	StructuralFeature	StructuralFeatureAction	\$ReadStructuralFeatureAction[i].structuralFeature
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ReadStructuralFeatureAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ReadStructuralFeatureAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$ReadStructuralFeatureAction[i].visibility

190. ReadVariableAction

Generalization

- [VariableAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$ReadVariableAction[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$ReadVariableAction[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$ReadVariableAction[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$ReadVariableAction[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$ReadVariableAction[i].clientDependency</code>
Context	context	Classifier	Action	<code>\$ReadVariableAction[i].context</code>
isEditable	editable	boolean	BaseElement	<code>\$ReadVariableAction[i].editable</code>
Handler	handler	ExceptionHandler[]	ExecutableNode	<code>\$ReadVariableAction[i].handler</code>
humanName	humanName	String	BaseElement	<code>\$ReadVariableAction[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$ReadVariableAction[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$ReadVariableAction[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$ReadVariableAction[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$ReadVariableAction[i].inPartition</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$ReadVariableAction[i].inStructuredNode</code>
Incoming	incoming	ActivityEdge[]	ActivityNode	<code>\$ReadVariableAction[i].incoming</code>
Input	input	InputPin[]	Action	<code>\$ReadVariableAction[i].input</code>
isInvalid	invalid	boolean	ModelObject	<code>\$ReadVariableAction[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$ReadVariableAction[i].leaf</code>
localID	localID	String	ModelObject	<code>\$ReadVariableAction[i].localID</code>
Local Postcondition	localPostcondition	Constraint[]	Action	<code>\$ReadVariableAction[i].localPostcondition</code>
Local Precondition	localPrecondition	Constraint[]	Action	<code>\$ReadVariableAction[i].localPrecondition</code>
Is Locally Reentrant	locallyReentrant	boolean	Action	<code>\$ReadVariableAction[i].locallyReentrant</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$ReadVariableAction[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$ReadVariableAction[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$ReadVariableAction[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$ReadVariableAction[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$ReadVariableAction[i].objectParent</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$ReadVariableAction[i].outgoing</code>
Output	output	OutputPin[]	Action	<code>\$ReadVariableAction[i].output</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$ReadVariableAction[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$ReadVariableAction[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$ReadVariableAction[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$ReadVariableAction[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$ReadVariableAction[i].redefinedElement</code>
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	<code>\$ReadVariableAction[i].redefinedNode</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$ReadVariableAction[i].redefinitionContext</code>
Result	result	OutputPin	ReadVariableAction	<code>\$ReadVariableAction[i].result</code>

Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$ReadVariableAction[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$ReadVariableAction[i].syncElement</code>
Variable	variable	Variable	VariableAction	<code>\$ReadVariableAction[i].variable</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$ReadVariableAction[i].visibility</code>

191. Realization

Generalization

- [Abstraction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$Realization[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Realization[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$Realization[i].classType</code>
Client	client	NamedElement[]	Dependency	<code>\$Realization[i].client</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Realization[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$Realization[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$Realization[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Realization[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Realization[i].invalid</code>
localID	localID	String	ModelObject	<code>\$Realization[i].localID</code>
Mapping	mapping	OpaqueExpression	Abstraction	<code>\$Realization[i].mapping</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$Realization[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$Realization[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Realization[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Realization[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Realization[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Realization[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Realization[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$Realization[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$Realization[i].owningPackage</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$Realization[i].owningTemplateParameter</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Realization[i].qualifiedName</code>
Related Element	relatedElement	Element[]	Relationship	<code>\$Realization[i].relatedElement</code>
Source	source	Element[]	DirectedRelationship	<code>\$Realization[i].source</code>
Supplier	supplier	NamedElement[]	Dependency	<code>\$Realization[i].supplier</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Realization[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Realization[i].syncElement</code>

Target	target	Element[]	DirectedRelationship	<code>\$Realization[i].target</code>
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$Realization[i].templateParameter</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$Realization[i].visibility</code>

192. Reception

Generalization

- [BehavioralFeature](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOwner	<code>\$Reception[i].ID</code>
Is Abstract	abstract	boolean	BehavioralFeature	<code>\$Reception[i].abstract</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Reception[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$Reception[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Reception[i].clientDependency</code>
Concurrency	concurrency	CallConcurrencyKind	BehavioralFeature	<code>\$Reception[i].concurrency</code>
isEditable	editable	boolean	BaseElement	<code>\$Reception[i].editable</code>
Element Import	elementImport	ElementImport[]	Namespace	<code>\$Reception[i].elementImport</code>
Featuring Classifier	featuringClassifier	Classifier	Feature	<code>\$Reception[i].featuringClassifier</code>
humanName	humanName	String	BaseElement	<code>\$Reception[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Reception[i].humanType</code>
Imported Member	importedMember	PackageableElement[]	Namespace	<code>\$Reception[i].importedMember</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Reception[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$Reception[i].leaf</code>
localID	localID	String	ModelObject	<code>\$Reception[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOwner	<code>\$Reception[i].mdExtensions</code>
Member	member	NamedElement[]	Namespace	<code>\$Reception[i].member</code>
Method	method	Behavior[]	BehavioralFeature	<code>\$Reception[i].method</code>
Name	name	String	NamedElement	<code>\$Reception[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Reception[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Reception[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Reception[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Reception[i].ownedComment</code>
Owned Diagram	ownedDiagram	Diagram[]	Namespace	<code>\$Reception[i].ownedDiagram</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Reception[i].ownedElement</code>
Owned Member	ownedMember	NamedElement[]	Namespace	<code>\$Reception[i].ownedMember</code>
Owned Parameter	ownedParameter	Parameter[]	BehavioralFeature	<code>\$Reception[i].ownedParameter</code>
Owned Parameter Set	ownedParameterSet	ParameterSet[]	BehavioralFeature	<code>\$Reception[i].ownedParameterSet</code>

Owned Rule	ownedRule	Constraint[]	Namespace	\$Reception[i].ownedRule
Owner	owner	Element	Element	\$Reception[i].owner
Package Import	packageImport	PackageImport[]	Namespace	\$Reception[i].packageImport
Qualified Name	qualifiedName	String	NamedElement	\$Reception[i].qualifiedName
Raised Exception	raisedException	Type[]	BehavioralFeature	\$Reception[i].raisedException
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$Reception[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$Reception[i].redefinitionContext
Signal	signal	Signal	Reception	\$Reception[i].signal
Is Static	static	boolean	Feature	\$Reception[i].static
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Reception[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Reception[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$Reception[i].visibility

193. ReclassifyObjectAction

Generalization

- [Action](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ReclassifyObjectAction[i].ID
Activity	activity	Activity	ActivityNode	\$ReclassifyObjectAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ReclassifyObjectAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ReclassifyObjectAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ReclassifyObjectAction[i].clientDependency
Context	context	Classifier	Action	\$ReclassifyObjectAction[i].context
isEditable	editable	boolean	BaseElement	\$ReclassifyObjectAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$ReclassifyObjectAction[i].handler
humanName	humanName	String	BaseElement	\$ReclassifyObjectAction[i].humanName
humanType	humanType	String	BaseElement	\$ReclassifyObjectAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ReclassifyObjectAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ReclassifyObjectAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ReclassifyObjectAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ReclassifyObjectAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ReclassifyObjectAction[i].incoming

Input	input	InputPin[]	Action	\$ReclassifyObjectAction[i].input
isInvalid	invalid	boolean	ModelObject	\$ReclassifyObjectAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ReclassifyObjectAction[i].leaf
localID	localID	String	ModelObject	\$ReclassifyObjectAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$ReclassifyObjectAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$ReclassifyObjectAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$ReclassifyObjectAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ReclassifyObjectAction[i].mdExtensions
Name	name	String	NamedElement	\$ReclassifyObjectAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ReclassifyObjectAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ReclassifyObjectAction[i].namespace
New Classifier	newClassifier	Classifier[]	ReclassifyObjectAction	\$ReclassifyObjectAction[i].newClassifier
Object	object	InputPin	ReclassifyObjectAction	\$ReclassifyObjectAction[i].object
objectParent	objectParent	BaseElement	BaseElement	\$ReclassifyObjectAction[i].objectParent
Old Classifier	oldClassifier	Classifier[]	ReclassifyObjectAction	\$ReclassifyObjectAction[i].oldClassifier
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ReclassifyObjectAction[i].outgoing
Output	output	OutputPin[]	Action	\$ReclassifyObjectAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$ReclassifyObjectAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ReclassifyObjectAction[i].ownedElement
Owner	owner	Element	Element	\$ReclassifyObjectAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ReclassifyObjectAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ReclassifyObjectAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ReclassifyObjectAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ReclassifyObjectAction[i].redefinitionContext
Is Replace All	replaceAll	boolean	ReclassifyObjectAction	\$ReclassifyObjectAction[i].replaceAll
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ReclassifyObjectAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ReclassifyObjectAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$ReclassifyObjectAction[i].visibility

194. [RedefinableElement](#)

Generalization

- [NamedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class)	Sample Template Expression (VTL)

			Owner)	
ID	ID	String	MDOObject	\$RedefinableElement[j].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$RedefinableElement[j].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$RedefinableElement[j].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$RedefinableElement[j].clientDependency
isEditable	editable	boolean	BaseElement	\$RedefinableElement[j].editable
humanName	humanName	String	BaseElement	\$RedefinableElement[j].humanName
humanType	humanType	String	BaseElement	\$RedefinableElement[j].humanType
isInvalid	invalid	boolean	ModelObject	\$RedefinableElement[j].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$RedefinableElement[j].leaf
localID	localID	String	ModelObject	\$RedefinableElement[j].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$RedefinableElement[j].mdExtensions
Name	name	String	NamedElement	\$RedefinableElement[j].name
Name Expression	nameExpression	StringExpression	NamedElement	\$RedefinableElement[j].nameExpression
Namespace	namespace	Namespace	NamedElement	\$RedefinableElement[j].namespace
objectParent	objectParent	BaseElement	BaseElement	\$RedefinableElement[j].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$RedefinableElement[j].ownedComment
Owned Element	ownedElement	Element[]	Element	\$RedefinableElement[j].ownedElement
Owner	owner	Element	Element	\$RedefinableElement[j].owner
Qualified Name	qualifiedName	String	NamedElement	\$RedefinableElement[j].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$RedefinableElement[j].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$RedefinableElement[j].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$RedefinableElement[j].supplierDependency
Sync Element	syncElement	Element	Element	\$RedefinableElement[j].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$RedefinableElement[j].visibility

195. RedefinableTemplateSignature

Generalization

- [RedefinableElement](#)
- [TemplateSignature](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$RedefinableTemplateSignature[j].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$RedefinableTemplateSignature[j].appliedStereotypeInstance

classType	classType	Class	BaseElement	\$RedefinableTemplateSignature[i].classType
Classifier	classifier	Classifier	RedefinableTemplateSignature	\$RedefinableTemplateSignature[i].classifier
Client Dependency	clientDependency	Dependency[]	NamedElement	\$RedefinableTemplateSignature[i].clientDependency
isEditable	editable	boolean	BaseElement	\$RedefinableTemplateSignature[i].editable
Extended Signature	extendedSignature	RedefinableTemplateSignature[]	RedefinableTemplateSignature	\$RedefinableTemplateSignature[i].extendedSignature
humanName	humanName	String	BaseElement	\$RedefinableTemplateSignature[i].humanName
humanType	humanType	String	BaseElement	\$RedefinableTemplateSignature[i].humanType
Inherited Parameter	inheritedParameter	TemplateParameter[]	RedefinableTemplateSignature	\$RedefinableTemplateSignature[i].inheritedParameter
isInvalid	invalid	boolean	ModelObject	\$RedefinableTemplateSignature[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$RedefinableTemplateSignature[i].leaf
localID	localID	String	ModelObject	\$RedefinableTemplateSignature[i].localID
mdExtensions	mdExtensions	MDEExtension[]	MDOObject	\$RedefinableTemplateSignature[i].mdExtensions
Name	name	String	NamedElement	\$RedefinableTemplateSignature[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$RedefinableTemplateSignature[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$RedefinableTemplateSignature[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$RedefinableTemplateSignature[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$RedefinableTemplateSignature[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$RedefinableTemplateSignature[i].ownedElement
Owned Parameter	ownedParameter	TemplateParameter[]	TemplateSignature	\$RedefinableTemplateSignature[i].ownedParameter
Owner	owner	Element	Element	\$RedefinableTemplateSignature[i].owner
Parameter	parameter	TemplateParameter[]	TemplateSignature	\$RedefinableTemplateSignature[i].parameter
Qualified Name	qualifiedName	String	NamedElement	\$RedefinableTemplateSignature[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$RedefinableTemplateSignature[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$RedefinableTemplateSignature[i].redefinitionContext

				onContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$RedefinableTemplateSignature[i].supplierDependency
Sync Element	syncElement	Element	Element	\$RedefinableTemplateSignature[i].syncElement
Template	template	TemplateableElement	TemplateSignature	\$RedefinableTemplateSignature[i].template
Visibility	visibility	VisibilityKind	NamedElement	\$RedefinableTemplateSignature[i].visibility

196. ReduceAction

Generalization

- [Action](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ReduceAction[i].ID
Activity	activity	Activity	ActivityNode	\$ReduceAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ReduceAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ReduceAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ReduceAction[i].clientDependency
Collection	collection	InputPin	ReduceAction	\$ReduceAction[i].collection
Context	context	Classifier	Action	\$ReduceAction[i].context
isEditable	editable	boolean	BaseElement	\$ReduceAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$ReduceAction[i].handler
humanName	humanName	String	BaseElement	\$ReduceAction[i].humanName
humanType	humanType	String	BaseElement	\$ReduceAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ReduceAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ReduceAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ReduceAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ReduceAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ReduceAction[i].incoming
Input	input	InputPin[]	Action	\$ReduceAction[i].input
isInvalid	invalid	boolean	ModelObject	\$ReduceAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ReduceAction[i].leaf
localID	localID	String	ModelObject	\$ReduceAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$ReduceAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$ReduceAction[i].localPrecondition

Is Locally Reentrant	locallyReentrant	boolean	Action	\$ReduceAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ReduceAction[i].mdExtensions
Name	name	String	NamedElement	\$ReduceAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ReduceAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ReduceAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ReduceAction[i].objectParent
Is Ordered	ordered	boolean	ReduceAction	\$ReduceAction[i].ordered
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ReduceAction[i].outgoing
Output	output	OutputPin[]	Action	\$ReduceAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$ReduceAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ReduceAction[i].ownedElement
Owner	owner	Element	Element	\$ReduceAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ReduceAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ReduceAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ReduceAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ReduceAction[i].redefinitionContext
Reducer	reducer	Behavior	ReduceAction	\$ReduceAction[i].reducer
Result	result	OutputPin	ReduceAction	\$ReduceAction[i].result
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ReduceAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ReduceAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$ReduceAction[i].visibility

197. Region

Generalization

- [Namespace](#)
- [RedefinableElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Region[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Region[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Region[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Region[i].clientDependency
isEditable	editable	boolean	BaseElement	\$Region[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$Region[i].elementImport
Extended Region	extendedRegion	Region	Region	\$Region[i].extendedRegion
humanName	humanName	String	BaseElement	\$Region[i].humanName
humanType	humanType	String	BaseElement	\$Region[i].humanType

Imported Member	importedMember	PackageableElement[]	Namespace	<code>\$Region[i].importedMember</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Region[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$Region[i].leaf</code>
localID	localID	String	ModelObject	<code>\$Region[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$Region[i].mdExtensions</code>
Member	member	NamedElement[]	Namespace	<code>\$Region[i].member</code>
Name	name	String	NamedElement	<code>\$Region[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Region[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Region[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Region[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Region[i].ownedComment</code>
Owned Diagram	ownedDiagram	Diagram[]	Namespace	<code>\$Region[i].ownedDiagram</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Region[i].ownedElement</code>
Owned Member	ownedMember	NamedElement[]	Namespace	<code>\$Region[i].ownedMember</code>
Owned Rule	ownedRule	Constraint[]	Namespace	<code>\$Region[i].ownedRule</code>
Owner	owner	Element	Element	<code>\$Region[i].owner</code>
Package Import	packageImport	PackageImport[]	Namespace	<code>\$Region[i].packageImport</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Region[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$Region[i].redefinedElement</code>
Redefinition Context	redefinitionContext	Classifier[]	Region	<code>\$Region[i].redefinitionContext</code>
State	state	State	Region	<code>\$Region[i].state</code>
State Machine	stateMachine	StateMachine	Region	<code>\$Region[i].stateMachine</code>
Subvertex	subvertex	Vertex[]	Region	<code>\$Region[i].subvertex</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Region[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Region[i].syncElement</code>
Transition	transition	Transition[]	Region	<code>\$Region[i].transition</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$Region[i].visibility</code>

198. Relationship

Generalization

- [Element](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$Relationship[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Relationship[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$Relationship[i].classType</code>
isEditable	editable	boolean	BaseElement	<code>\$Relationship[i].editable</code>

humanName	humanName	String	BaseElement	\$Relationship[i].humanName
humanType	humanType	String	BaseElement	\$Relationship[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Relationship[i].invalid
localID	localID	String	ModelObject	\$Relationship[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$Relationship[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$Relationship[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Relationship[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Relationship[i].ownedElement
Owner	owner	Element	Element	\$Relationship[i].owner
Related Element	relatedElement	Element[]	Relationship	\$Relationship[i].relatedElement
Sync Element	syncElement	Element	Element	\$Relationship[i].syncElement

199. RemoveStructuralFeatureValueAction

Generalization

- WriteStructuralFeatureAction

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$RemoveStructuralFeatureValueAction[i].ID
Activity	activity	Activity	ActivityNode	\$RemoveStructuralFeatureValueAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$RemoveStructuralFeatureValueAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$RemoveStructuralFeatureValueAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$RemoveStructuralFeatureValueAction[i].clientDependency
Context	context	Classifier	Action	\$RemoveStructuralFeatureValueAction[i].context
isEditable	editable	boolean	BaseElement	\$RemoveStructuralFeatureValueAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$RemoveStructuralFeatureValueAction[i].handler
humanName	humanName	String	BaseElement	\$RemoveStructuralFeatureValueAction[i].humanName
humanType	humanType	String	BaseElement	\$RemoveStructuralFeatureValueAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$RemoveStructuralFeatureValueAction[i].inGroup

In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$RemoveStructuralFeatureValueAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$RemoveStructuralFeatureValueAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$RemoveStructuralFeatureValueAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$RemoveStructuralFeatureValueAction[i].incoming
Input	input	InputPin[]	Action	\$RemoveStructuralFeatureValueAction[i].input
isInvalid	invalid	boolean	ModelObject	\$RemoveStructuralFeatureValueAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$RemoveStructuralFeatureValueAction[i].leaf
localID	localID	String	ModelObject	\$RemoveStructuralFeatureValueAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$RemoveStructuralFeatureValueAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$RemoveStructuralFeatureValueAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$RemoveStructuralFeatureValueAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$RemoveStructuralFeatureValueAction[i].mdExtensions
Name	name	String	NamedElement	\$RemoveStructuralFeatureValueAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$RemoveStructuralFeatureValueAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$RemoveStructuralFeatureValueAction[i].namespace
Object	object	InputPin	StructuralFeatureAction	\$RemoveStructuralFeatureValueAction[i].object
objectParent	objectParent	BaseElement	BaseElement	\$RemoveStructuralFeatureValueAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$RemoveStructuralFeatureValueAction[i].outgoing
Output	output	OutputPin[]	Action	\$RemoveStructuralFeatureValueAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$RemoveStructuralFeatureValueAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$RemoveStructuralFeatureValueAction[i].o

				wnedElement
Owner	owner	Element	Element	\$RemoveStructuralFeatureValueAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$RemoveStructuralFeatureValueAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$RemoveStructuralFeatureValueAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$RemoveStructuralFeatureValueAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$RemoveStructuralFeatureValueAction[i].redefinitionContext
Remove At	removeAt	InputPin	RemoveStructuralFeatureValueAction	\$RemoveStructuralFeatureValueAction[i].removeAt
Is Remove Duplicates	removeDuplicates	boolean	RemoveStructuralFeatureValueAction	\$RemoveStructuralFeatureValueAction[i].removeDuplicates
Result	result	OutputPin	WriteStructuralFeatureAction	\$RemoveStructuralFeatureValueAction[i].result
Structural Feature	structuralFeature	StructuralFeature	StructuralFeatureAction	\$RemoveStructuralFeatureValueAction[i].structuralFeature
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$RemoveStructuralFeatureValueAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$RemoveStructuralFeatureValueAction[i].syncElement
Value	value	InputPin	WriteStructuralFeatureAction	\$RemoveStructuralFeatureValueAction[i].value
Visibility	visibility	VisibilityKind	NamedElement	\$RemoveStructuralFeatureValueAction[i].visibility

200. RemoveVariableValueAction

Generalization

- [WriteVariableAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$RemoveVariableValueAction[i].ID
Activity	activity	Activity	ActivityNode	\$RemoveVariableValueAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$RemoveVariableValueAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$RemoveVariableValueAction[i].classType

Client Dependency	clientDependency	Dependency[]	NamedElement	\$RemoveVariableValueAction[i].clientDependency
Context	context	Classifier	Action	\$RemoveVariableValueAction[i].context
isEditable	editable	boolean	BaseElement	\$RemoveVariableValueAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$RemoveVariableValueAction[i].handler
humanName	humanName	String	BaseElement	\$RemoveVariableValueAction[i].humanName
humanType	humanType	String	BaseElement	\$RemoveVariableValueAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$RemoveVariableValueAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$RemoveVariableValueAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$RemoveVariableValueAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$RemoveVariableValueAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$RemoveVariableValueAction[i].incoming
Input	input	InputPin[]	Action	\$RemoveVariableValueAction[i].input
isInvalid	invalid	boolean	ModelObject	\$RemoveVariableValueAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$RemoveVariableValueAction[i].leaf
localID	localID	String	ModelObject	\$RemoveVariableValueAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$RemoveVariableValueAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$RemoveVariableValueAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$RemoveVariableValueAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$RemoveVariableValueAction[i].mdExtensions
Name	name	String	NamedElement	\$RemoveVariableValueAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$RemoveVariableValueAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$RemoveVariableValueAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$RemoveVariableValueAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$RemoveVariableValueAction[i].outgoing
Output	output	OutputPin[]	Action	\$RemoveVariableValueAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$RemoveVariableValueAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$RemoveVariableValueAction[i].ownedElement
Owner	owner	Element	Element	\$RemoveVariableValueAction[i].owner

Qualified Name	qualifiedName	String	NamedElement	\$RemoveVariableValueAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$RemoveVariableValueAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$RemoveVariableValueAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$RemoveVariableValueAction[i].redefinitionContext
Remove At	removeAt	InputPin	RemoveVariableValueAction	\$RemoveVariableValueAction[i].removeAt
Is Remove Duplicates	removeDuplicates	boolean	RemoveVariableValueAction	\$RemoveVariableValueAction[i].removeDuplicates
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$RemoveVariableValueAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$RemoveVariableValueAction[i].syncElement
Value	value	InputPin	WriteVariableAction	\$RemoveVariableValueAction[i].value
Variable	variable	Variable	VariableAction	\$RemoveVariableValueAction[i].variable
Visibility	visibility	VisibilityKind	NamedElement	\$RemoveVariableValueAction[i].visibility

201. ReplyAction

Generalization

- Action

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ReplyAction[i].ID
Activity	activity	Activity	ActivityNode	\$ReplyAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ReplyAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ReplyAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ReplyAction[i].clientDependency
Context	context	Classifier	Action	\$ReplyAction[i].context
isEditable	editable	boolean	BaseElement	\$ReplyAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$ReplyAction[i].handler
humanName	humanName	String	BaseElement	\$ReplyAction[i].humanName
humanType	humanType	String	BaseElement	\$ReplyAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ReplyAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ReplyAction[i].inInterruptibleRegion

In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ReplyAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ReplyAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ReplyAction[i].incoming
Input	input	InputPin[]	Action	\$ReplyAction[i].input
isInvalid	invalid	boolean	ModelObject	\$ReplyAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ReplyAction[i].leaf
localID	localID	String	ModelObject	\$ReplyAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$ReplyAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$ReplyAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$ReplyAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$ReplyAction[i].mdExtensions
Name	name	String	NamedElement	\$ReplyAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ReplyAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ReplyAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ReplyAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ReplyAction[i].outgoing
Output	output	OutputPin[]	Action	\$ReplyAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$ReplyAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ReplyAction[i].ownedElement
Owner	owner	Element	Element	\$ReplyAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ReplyAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ReplyAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ReplyAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ReplyAction[i].redefinitionContext
Reply To Call	replyToCall	Trigger	ReplyAction	\$ReplyAction[i].replyToCall
Reply Value	replyValue	InputPin[]	ReplyAction	\$ReplyAction[i].replyValue
Return Information	returnInformation	InputPin	ReplyAction	\$ReplyAction[i].returnInformation
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ReplyAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ReplyAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$ReplyAction[i].visibility

202. SendObjectAction

Generalization

- [InvocationAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDObject	\$SendObjectAction[i].ID

Activity	activity	Activity	ActivityNode	\$SendObjectAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$SendObjectAction[i].appliedStereotypeInstance
Argument	argument	InputPin[]	InvocationAction	\$SendObjectAction[i].argument
classType	classType	Class	BaseElement	\$SendObjectAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$SendObjectAction[i].clientDependency
Context	context	Classifier	Action	\$SendObjectAction[i].context
isEditable	editable	boolean	BaseElement	\$SendObjectAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$SendObjectAction[i].handler
humanName	humanName	String	BaseElement	\$SendObjectAction[i].humanName
humanType	humanType	String	BaseElement	\$SendObjectAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$SendObjectAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$SendObjectAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$SendObjectAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$SendObjectAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$SendObjectAction[i].incoming
Input	input	InputPin[]	Action	\$SendObjectAction[i].input
isInvalid	invalid	boolean	ModelObject	\$SendObjectAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$SendObjectAction[i].leaf
localID	localID	String	ModelObject	\$SendObjectAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$SendObjectAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$SendObjectAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$SendObjectAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$SendObjectAction[i].mdExtensions
Name	name	String	NamedElement	\$SendObjectAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$SendObjectAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$SendObjectAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$SendObjectAction[i].objectParent
On Port	onPort	Port	InvocationAction	\$SendObjectAction[i].onPort
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$SendObjectAction[i].outgoing
Output	output	OutputPin[]	Action	\$SendObjectAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$SendObjectAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$SendObjectAction[i].ownedElement
Owner	owner	Element	Element	\$SendObjectAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$SendObjectAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$SendObjectAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$SendObjectAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$SendObjectAction[i].redefinitionContext
Request	request	InputPin	SendObjectAction	\$SendObjectAction[i].request

Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$SendObjectAction[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$SendObjectAction[i].syncElement</code>
Target	target	InputPin	SendObjectAction	<code>\$SendObjectAction[i].target</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$SendObjectAction[i].visibility</code>

203. SendSignalAction

Generalization

- [InvocationAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$SendSignalAction[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$SendSignalAction[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$SendSignalAction[i].appliedStereotypeInstance</code>
Argument	argument	InputPin[]	InvocationAction	<code>\$SendSignalAction[i].argument</code>
classType	classType	Class	BaseElement	<code>\$SendSignalAction[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$SendSignalAction[i].clientDependency</code>
Context	context	Classifier	Action	<code>\$SendSignalAction[i].context</code>
isEditable	editable	boolean	BaseElement	<code>\$SendSignalAction[i].editable</code>
Handler	handler	ExceptionHandler[]	ExecutableNode	<code>\$SendSignalAction[i].handler</code>
humanName	humanName	String	BaseElement	<code>\$SendSignalAction[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$SendSignalAction[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$SendSignalAction[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$SendSignalAction[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$SendSignalAction[i].inPartition</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$SendSignalAction[i].inStructuredNode</code>
Incoming	incoming	ActivityEdge[]	ActivityNode	<code>\$SendSignalAction[i].incoming</code>
Input	input	InputPin[]	Action	<code>\$SendSignalAction[i].input</code>
isInvalid	invalid	boolean	ModelObject	<code>\$SendSignalAction[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$SendSignalAction[i].leaf</code>
localID	localID	String	ModelObject	<code>\$SendSignalAction[i].localID</code>
Local Postcondition	localPostcondition	Constraint[]	Action	<code>\$SendSignalAction[i].localPostcondition</code>
Local Precondition	localPrecondition	Constraint[]	Action	<code>\$SendSignalAction[i].localPrecondition</code>
Is Locally Reentrant	locallyReentrant	boolean	Action	<code>\$SendSignalAction[i].locallyReentrant</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$SendSignalAction[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$SendSignalAction[i].name</code>

Name Expression	nameExpression	StringExpression	NamedElement	<code>\$SendSignalAction[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$SendSignalAction[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$SendSignalAction[i].objectParent</code>
On Port	onPort	Port	InvocationAction	<code>\$SendSignalAction[i].onPort</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$SendSignalAction[i].outgoing</code>
Output	output	OutputPin[]	Action	<code>\$SendSignalAction[i].output</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$SendSignalAction[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$SendSignalAction[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$SendSignalAction[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$SendSignalAction[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$SendSignalAction[i].redefinedElement</code>
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	<code>\$SendSignalAction[i].redefinedNode</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$SendSignalAction[i].redefinitionContext</code>
Signal	signal	Signal	SendSignalAction	<code>\$SendSignalAction[i].signal</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$SendSignalAction[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$SendSignalAction[i].syncElement</code>
Target	target	InputPin	SendSignalAction	<code>\$SendSignalAction[i].target</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$SendSignalAction[i].visibility</code>

204. SequenceNode

Generalization

- [StructuredActivityNode](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$SequenceNode[i].ID</code>
Activity	activity	Activity	StructuredActivityNode	<code>\$SequenceNode[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$SequenceNode[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$SequenceNode[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$SequenceNode[i].clientDependency</code>
Contained Edge	containedEdge	ActivityEdge[]	ActivityGroup	<code>\$SequenceNode[i].containedEdge</code>
Contained Node	containedNode	ActivityNode[]	ActivityGroup	<code>\$SequenceNode[i].containedNode</code>
Context	context	Classifier	Action	<code>\$SequenceNode[i].context</code>
Edge	edge	ActivityEdge[]	StructuredActivityNode	<code>\$SequenceNode[i].edge</code>
isEditable	editable	boolean	BaseElement	<code>\$SequenceNode[i].editable</code>
Element Import	elementImport	ElementImport[]	Namespace	<code>\$SequenceNode[i].elementImport</code>
Executable Node	executableNode	ExecutableNode[]	SequenceNode	<code>\$SequenceNode[i].executableNode</code>

Handler	handler	ExceptionHandler[]	ExecutableNode	\$SequenceNode[i].handler
humanName	humanName	String	BaseElement	\$SequenceNode[i].humanName
humanType	humanType	String	BaseElement	\$SequenceNode[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$SequenceNode[i].importedMember
In Activity	inActivity	Activity	ActivityGroup	\$SequenceNode[i].inActivity
In Group	inGroup	ActivityGroup[]	ActivityNode	\$SequenceNode[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$SequenceNode[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$SequenceNode[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$SequenceNode[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$SequenceNode[i].incoming
Input	input	InputPin[]	Action	\$SequenceNode[i].input
isInvalid	invalid	boolean	ModelObject	\$SequenceNode[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$SequenceNode[i].leaf
localID	localID	String	ModelObject	\$SequenceNode[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$SequenceNode[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$SequenceNode[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$SequenceNode[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$SequenceNode[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$SequenceNode[i].member
isMustIsolate	mustIsolate	boolean	StructuredActivityNode	\$SequenceNode[i].mustIsolate
Name	name	String	NamedElement	\$SequenceNode[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$SequenceNode[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$SequenceNode[i].namespace
Node	node	ActivityNode[]	StructuredActivityNode	\$SequenceNode[i].node
objectParent	objectParent	BaseElement	BaseElement	\$SequenceNode[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$SequenceNode[i].outgoing
Output	output	OutputPin[]	Action	\$SequenceNode[i].output
Owned Comment	ownedComment	Comment[]	Element	\$SequenceNode[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$SequenceNode[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$SequenceNode[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$SequenceNode[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$SequenceNode[i].ownedRule
Owner	owner	Element	Element	\$SequenceNode[i].owner
Package Import	packageImport	PackageImport[]	Namespace	\$SequenceNode[i].packageImport
Qualified Name	qualifiedName	String	NamedElement	\$SequenceNode[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$SequenceNode[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$SequenceNode[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$SequenceNode[i].redefinitionContext
Structured Node Input	structuredNodeInput	InputPin[]	StructuredActivityNode	\$SequenceNode[i].structuredNodeInput

Structured Node Output	structuredNodeOutput	OutputPin[]	StructuredActivityNode	\$SequenceNode[i].structuredNodeOutput
Subgroup	subgroup	ActivityGroup[]	ActivityGroup	\$SequenceNode[i].subgroup
Super Group	superGroup	ActivityGroup	ActivityGroup	\$SequenceNode[i].superGroup
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$SequenceNode[i].supplierDependency
Sync Element	syncElement	Element	Element	\$SequenceNode[i].syncElement
Variable	variable	Variable[]	StructuredActivityNode	\$SequenceNode[i].variable
Visibility	visibility	VisibilityKind	NamedElement	\$SequenceNode[i].visibility

205. Signal

Generalization

- [Classifier](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Signal[i].ID
Class	UMLClass	Class	Classifier	\$Signal[i].UMLClass
Is Abstract	abstract	boolean	Classifier	\$Signal[i].abstract
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Signal[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$Signal[i].attribute
classType	classType	Class	BaseElement	\$Signal[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Signal[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$Signal[i].collaborationUse
isEditable	editable	boolean	BaseElement	\$Signal[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$Signal[i].elementImport
Feature	feature	Feature[]	Classifier	\$Signal[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$Signal[i].finalSpecialization
General	general	Classifier[]	Classifier	\$Signal[i].general
Generalization	generalization	Generalization[]	Classifier	\$Signal[i].generalization
humanName	humanName	String	BaseElement	\$Signal[i].humanName
humanType	humanType	String	BaseElement	\$Signal[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Signal[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$Signal[i].inheritedMember
isInvalid	invalid	boolean	ModelObject	\$Signal[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$Signal[i].leaf
localID	localID	String	ModelObject	\$Signal[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Signal[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Signal[i].member
Name	name	String	NamedElement	\$Signal[i].name

Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Signal[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Signal[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Signal[i].objectParent</code>
Owned Attribute	ownedAttribute	Property[]	Signal	<code>\$Signal[i].ownedAttribute</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Signal[i].ownedComment</code>
Owned Diagram	ownedDiagram	Diagram[]	Namespace	<code>\$Signal[i].ownedDiagram</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Signal[i].ownedElement</code>
Owned Member	ownedMember	NamedElement[]	Namespace	<code>\$Signal[i].ownedMember</code>
Owned Rule	ownedRule	Constraint[]	Namespace	<code>\$Signal[i].ownedRule</code>
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	<code>\$Signal[i].ownedTemplateSignature</code>
Owned Use Case	ownedUseCase	UseCase[]	Classifier	<code>\$Signal[i].ownedUseCase</code>
Owner	owner	Element	Element	<code>\$Signal[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$Signal[i].owningPackage</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$Signal[i].owningTemplateParameter</code>
Package	package	Package	Type	<code>\$Signal[i].package</code>
Package Import	packageImport	PackageImport[]	Namespace	<code>\$Signal[i].packageImport</code>
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	<code>\$Signal[i].powertypeExtent</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Signal[i].qualifiedName</code>
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	<code>\$Signal[i].redefinedClassifier</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$Signal[i].redefinedElement</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$Signal[i].redefinitionContext</code>
Representation	representation	CollaborationUse	Classifier	<code>\$Signal[i].representation</code>
Substitution	substitution	Substitution[]	Classifier	<code>\$Signal[i].substitution</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Signal[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Signal[i].syncElement</code>
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	<code>\$Signal[i].templateBinding</code>
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	<code>\$Signal[i].templateParameter</code>
Use Case	useCase	UseCase[]	Classifier	<code>\$Signal[i].useCase</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$Signal[i].visibility</code>

206. SignalEvent

Generalization

- [MessageEvent](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)

ID	ID	String	MDOBJECT	<code>\$SignalEvent[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$SignalEvent[i].appliedStereotypeInstance</code>
Behavior	behavior	Behavior	Event	<code>\$SignalEvent[i].behavior</code>
classType	classType	Class	BaseElement	<code>\$SignalEvent[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$SignalEvent[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$SignalEvent[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$SignalEvent[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$SignalEvent[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$SignalEvent[i].invalid</code>
localID	localID	String	ModelObject	<code>\$SignalEvent[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$SignalEvent[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$SignalEvent[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$SignalEvent[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$SignalEvent[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$SignalEvent[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$SignalEvent[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$SignalEvent[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$SignalEvent[i].owner</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$SignalEvent[i].owningPackage</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$SignalEvent[i].owningTemplateParameter</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$SignalEvent[i].qualifiedName</code>
Signal	signal	Signal	SignalEvent	<code>\$SignalEvent[i].signal</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$SignalEvent[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$SignalEvent[i].syncElement</code>
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$SignalEvent[i].templateParameter</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$SignalEvent[i].visibility</code>

207. Slot

Generalization

- [Element](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$Slot[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Slot[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$Slot[i].classType</code>
Defining Feature	definingFeature	StructuralFeature	Slot	<code>\$Slot[i].definingFeature</code>
isEditable	editable	boolean	BaseElement	<code>\$Slot[i].editable</code>

humanName	humanName	String	BaseElement	\$Slot[i].humanName
humanType	humanType	String	BaseElement	\$Slot[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Slot[i].invalid
localID	localID	String	ModelObject	\$Slot[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Slot[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$Slot[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Slot[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Slot[i].ownedElement
Owner	owner	Element	Element	\$Slot[i].owner
Owning Instance	owningInstance	InstanceSpecification	Slot	\$Slot[i].owningInstance
Sync Element	syncElement	Element	Element	\$Slot[i].syncElement
Value	value	ValueSpecification[]	Slot	\$Slot[i].value

208. StartClassifierBehaviorAction

Generalization

- Action

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$StartClassifierBehaviorAction[i].ID
Activity	activity	Activity	ActivityNode	\$StartClassifierBehaviorAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$StartClassifierBehaviorAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$StartClassifierBehaviorAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$StartClassifierBehaviorAction[i].clientDependency
Context	context	Classifier	Action	\$StartClassifierBehaviorAction[i].context
isEditable	editable	boolean	BaseElement	\$StartClassifierBehaviorAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$StartClassifierBehaviorAction[i].handler
humanName	humanName	String	BaseElement	\$StartClassifierBehaviorAction[i].humanName
humanType	humanType	String	BaseElement	\$StartClassifierBehaviorAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$StartClassifierBehaviorAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$StartClassifierBehaviorAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$StartClassifierBehaviorAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$StartClassifierBehaviorAction[i].inStructure

				dNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$StartClassifierBehaviorAction[i].incoming
Input	input	InputPin[]	Action	\$StartClassifierBehaviorAction[i].input
isInvalid	invalid	boolean	ModelObject	\$StartClassifierBehaviorAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$StartClassifierBehaviorAction[i].leaf
localID	localID	String	ModelObject	\$StartClassifierBehaviorAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$StartClassifierBehaviorAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$StartClassifierBehaviorAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$StartClassifierBehaviorAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$StartClassifierBehaviorAction[i].mdExtensions
Name	name	String	NamedElement	\$StartClassifierBehaviorAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$StartClassifierBehaviorAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$StartClassifierBehaviorAction[i].namespace
Object	object	InputPin	StartClassifierBehaviorAction	\$StartClassifierBehaviorAction[i].object
objectParent	objectParent	BaseElement	BaseElement	\$StartClassifierBehaviorAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$StartClassifierBehaviorAction[i].outgoing
Output	output	OutputPin[]	Action	\$StartClassifierBehaviorAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$StartClassifierBehaviorAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$StartClassifierBehaviorAction[i].ownedElement
Owner	owner	Element	Element	\$StartClassifierBehaviorAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$StartClassifierBehaviorAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$StartClassifierBehaviorAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$StartClassifierBehaviorAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$StartClassifierBehaviorAction[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$StartClassifierBehaviorAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$StartClassifierBehaviorAction[i].syncElement

				nt
Visibility	visibility	VisibilityKind	NamedElement	\$StartClassifierBehaviorAction[i].visibility

209. StartObjectBehaviorAction

Generalization

- [CallAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$StartObjectBehaviorAction[i].ID
Activity	activity	Activity	ActivityNode	\$StartObjectBehaviorAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$StartObjectBehaviorAction[i].appliedStereotypeInstance
Argument	argument	InputPin[]	InvocationAction	\$StartObjectBehaviorAction[i].argument
classType	classType	Class	BaseElement	\$StartObjectBehaviorAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$StartObjectBehaviorAction[i].clientDependency
Context	context	Classifier	Action	\$StartObjectBehaviorAction[i].context
isEditable	editable	boolean	BaseElement	\$StartObjectBehaviorAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$StartObjectBehaviorAction[i].handler
humanName	humanName	String	BaseElement	\$StartObjectBehaviorAction[i].humanName
humanType	humanType	String	BaseElement	\$StartObjectBehaviorAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$StartObjectBehaviorAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$StartObjectBehaviorAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$StartObjectBehaviorAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$StartObjectBehaviorAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$StartObjectBehaviorAction[i].incoming
Input	input	InputPin[]	Action	\$StartObjectBehaviorAction[i].input
isInvalid	invalid	boolean	ModelObject	\$StartObjectBehaviorAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$StartObjectBehaviorAction[i].leaf
localID	localID	String	ModelObject	\$StartObjectBehaviorAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$StartObjectBehaviorAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$StartObjectBehaviorAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$StartObjectBehaviorAction[i].locallyReentrant

				ant
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$StartObjectBehaviorAction[i].mdExtensions
Name	name	String	NamedElement	\$StartObjectBehaviorAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$StartObjectBehaviorAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$StartObjectBehaviorAction[i].namespace
Object	object	InputPin	StartObjectBehaviorAction	\$StartObjectBehaviorAction[i].object
objectParent	objectParent	BaseElement	BaseElement	\$StartObjectBehaviorAction[i].objectParent
On Port	onPort	Port	InvocationAction	\$StartObjectBehaviorAction[i].onPort
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$StartObjectBehaviorAction[i].outgoing
Output	output	OutputPin[]	Action	\$StartObjectBehaviorAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$StartObjectBehaviorAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$StartObjectBehaviorAction[i].ownedElement
Owner	owner	Element	Element	\$StartObjectBehaviorAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$StartObjectBehaviorAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$StartObjectBehaviorAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$StartObjectBehaviorAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$StartObjectBehaviorAction[i].redefinitionContext
Result	result	OutputPin[]	CallAction	\$StartObjectBehaviorAction[i].result
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$StartObjectBehaviorAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$StartObjectBehaviorAction[i].syncElement
Is Synchronous	synchronous	boolean	CallAction	\$StartObjectBehaviorAction[i].synchronous
Visibility	visibility	VisibilityKind	NamedElement	\$StartObjectBehaviorAction[i].visibility

210. State

Generalization

- [Namespace](#)
- [Vertex](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$State[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$State[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$State[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$State[i].clientDependency</code>
Is Composite	composite	boolean	State	<code>\$State[i].composite</code>
Connection	connection	ConnectionPointReference[]	State	<code>\$State[i].connection</code>
Connection Point	connectionPoint	Pseudostate[]	State	<code>\$State[i].connectionPoint</code>
Container	container	Region	Vertex	<code>\$State[i].container</code>
Deferrable Trigger	deferrableTrigger	Trigger[]	State	<code>\$State[i].deferrableTrigger</code>
Do Activity	doActivity	Behavior	State	<code>\$State[i].doActivity</code>
isEditable	editable	boolean	BaseElement	<code>\$State[i].editable</code>
Element Import	elementImport	ElementImport[]	Namespace	<code>\$State[i].elementImport</code>
Entry	entry	Behavior	State	<code>\$State[i].entry</code>
Exit	exit	Behavior	State	<code>\$State[i].exit</code>
humanName	humanName	String	BaseElement	<code>\$State[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$State[i].humanType</code>
Imported Member	importedMember	PackageableElement[]	Namespace	<code>\$State[i].importedMember</code>
Incoming	incoming	Transition[]	Vertex	<code>\$State[i].incoming</code>
isInvalid	invalid	boolean	ModelObject	<code>\$State[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$State[i].leaf</code>
localID	localID	String	ModelObject	<code>\$State[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$State[i].mdExtensions</code>
Member	member	NamedElement[]	Namespace	<code>\$State[i].member</code>
Name	name	String	NamedElement	<code>\$State[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$State[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$State[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$State[i].objectParent</code>
Is Orthogonal	orthogonal	boolean	State	<code>\$State[i].orthogonal</code>
Outgoing	outgoing	Transition[]	Vertex	<code>\$State[i].outgoing</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$State[i].ownedComment</code>
Owned Diagram	ownedDiagram	Diagram[]	Namespace	<code>\$State[i].ownedDiagram</code>
Owned Element	ownedElement	Element[]	Element	<code>\$State[i].ownedElement</code>
Owned Member	ownedMember	NamedElement[]	Namespace	<code>\$State[i].ownedMember</code>
Owned Rule	ownedRule	Constraint[]	Namespace	<code>\$State[i].ownedRule</code>
Owner	owner	Element	Element	<code>\$State[i].owner</code>
Package Import	packageImport	PackageImport[]	Namespace	<code>\$State[i].packageImport</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$State[i].qualifiedName</code>

Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$State[i].redefinedElement
Redefined Vertex	redefinedVertex	Vertex	Vertex	\$State[i].redefinedVertex
Redefinition Context	redefinitionContext	Classifier[]	Vertex	\$State[i].redefinitionContext
Region	region	Region[]	State	\$State[i].region
Is Simple	simple	boolean	State	\$State[i].simple
State Invariant	stateInvariant	Constraint	State	\$State[i].stateInvariant
Submachine	submachine	StateMachine	State	\$State[i].submachine
Is Submachine State	submachineState	boolean	State	\$State[i].submachineState
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$State[i].supplierDependency
Sync Element	syncElement	Element	Element	\$State[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$State[i].visibility

211. StateInvariant

Generalization

- [InteractionFragment](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$StateInvariant[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$StateInvariant[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$StateInvariant[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$StateInvariant[i].clientDependency
Covered	covered	Lifeline[]	StateInvariant	\$StateInvariant[i].covered
isEditable	editable	boolean	BaseElement	\$StateInvariant[i].editable
Enclosing Interaction	enclosingInteraction	Interaction	InteractionFragment	\$StateInvariant[i].enclosingInteraction
Enclosing Operand	enclosingOperand	InteractionOperand	InteractionFragment	\$StateInvariant[i].enclosingOperand
General Ordering	generalOrdering	GeneralOrdering[]	InteractionFragment	\$StateInvariant[i].generalOrdering
humanName	humanName	String	BaseElement	\$StateInvariant[i].humanName
humanType	humanType	String	BaseElement	\$StateInvariant[i].humanType
isInvalid	invalid	boolean	ModelObject	\$StateInvariant[i].invalid
Invariant	invariant	Constraint	StateInvariant	\$StateInvariant[i].invariant
localID	localID	String	ModelObject	\$StateInvariant[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$StateInvariant[i].mdExtensions
Name	name	String	NamedElement	\$StateInvariant[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$StateInvariant[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$StateInvariant[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$StateInvariant[i].objectParent

Owned Comment	ownedComment	Comment[]	Element	\$StateInvariant[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$StateInvariant[i].ownedElement
Owner	owner	Element	Element	\$StateInvariant[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$StateInvariant[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$StateInvariant[i].supplierDependency
Sync Element	syncElement	Element	Element	\$StateInvariant[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$StateInvariant[i].visibility

212. StateMachine

Generalization

- [Behavior](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$StateMachine[i].ID
Class	UMLClass	Class	Classifier	\$StateMachine[i].UMLClass
Is Abstract	abstract	boolean	Class	\$StateMachine[i].abstract
Is Active	active	boolean	Class	\$StateMachine[i].active
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$StateMachine[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$StateMachine[i].attribute
classType	classType	Class	BaseElement	\$StateMachine[i].classType
Classifier Behavior	classifierBehavior	Behavior	BehavioredClassifier	\$StateMachine[i].classifierBehavior
Client Dependency	clientDependency	Dependency[]	NamedElement	\$StateMachine[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$StateMachine[i].collaborationUse
Connection Point	connectionPoint	Pseudostate[]	StateMachine	\$StateMachine[i].connectionPoint
Context	context	BehavioredClassifier	Behavior	\$StateMachine[i].context
isEditable	editable	boolean	BaseElement	\$StateMachine[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$StateMachine[i].elementImport
Event	event	Event[]	Behavior	\$StateMachine[i].event
Extended State Machine	extendedStateMachine	StateMachine[]	StateMachine	\$StateMachine[i].extendedStateMachine
Extension	extension	Extension[]	Class	\$StateMachine[i].extension
Feature	feature	Feature[]	Classifier	\$StateMachine[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$StateMachine[i].finalSpecialization
General	general	Classifier[]	Classifier	\$StateMachine[i].general
Generalization	generalization	Generalization[]	Classifier	\$StateMachine[i].generalization
humanName	humanName	String	BaseElement	\$StateMachine[i].humanName
humanType	humanType	String	BaseElement	\$StateMachine[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$StateMachine[i].importedMember

Inherited Member	inheritedMember	NamedElement[]	Classifier	\$StateMachine[i].inheritedMember
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavioredClassifier	\$StateMachine[i].interfaceRealization
isInvalid	invalid	boolean	ModelObject	\$StateMachine[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$StateMachine[i].leaf
localID	localID	String	ModelObject	\$StateMachine[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$StateMachine[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$StateMachine[i].member
Name	name	String	NamedElement	\$StateMachine[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$StateMachine[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$StateMachine[i].namespace
Nested Classifier	nestedClassifier	Classifier[]	Class	\$StateMachine[i].nestedClassifier
objectParent	objectParent	BaseElement	BaseElement	\$StateMachine[i].objectParent
Observation	observation	Observation[]	Behavior	\$StateMachine[i].observation
Owned Attribute	ownedAttribute	Property[]	Class	\$StateMachine[i].ownedAttribute
Owned Behavior	ownedBehavior	Behavior[]	BehavioredClassifier	\$StateMachine[i].ownedBehavior
Owned Comment	ownedComment	Comment[]	Element	\$StateMachine[i].ownedComment
Owned Connector	ownedConnector	Connector[]	StructuredClassifier	\$StateMachine[i].ownedConnector
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$StateMachine[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$StateMachine[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$StateMachine[i].ownedMember
Owned Operation	ownedOperation	Operation[]	Class	\$StateMachine[i].ownedOperation
Owned Parameter	ownedParameter	Parameter[]	Behavior	\$StateMachine[i].ownedParameter
Owned Parameter Set	ownedParameterSet	ParameterSet[]	Behavior	\$StateMachine[i].ownedParameterSet
Owned Port	ownedPort	Port[]	EncapsulatedClassifier	\$StateMachine[i].ownedPort
Owned Reception	ownedReception	Reception[]	Class	\$StateMachine[i].ownedReception
Owned Rule	ownedRule	Constraint[]	Namespace	\$StateMachine[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$StateMachine[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$StateMachine[i].ownedUseCase
Owner	owner	Element	Element	\$StateMachine[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$StateMachine[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$StateMachine[i].owningTemplateParameter
Package	package	Package	Type	\$StateMachine[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$StateMachine[i].packageImport
Part	part	Property[]	StructuredClassifier	\$StateMachine[i].part
Postcondition	postcondition	Constraint[]	Behavior	\$StateMachine[i].postcondition
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$StateMachine[i].powertypeExtent
Precondition	precondition	Constraint[]	Behavior	\$StateMachine[i].precondition
Qualified Name	qualifiedName	String	NamedElement	\$StateMachine[i].qualifiedName

Redefined Behavior	redefinedBehavior	Behavior[]	Behavior	<code>\$StateMachine[i].redefinedBehavior</code>
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	<code>\$StateMachine[i].redefinedClassifier</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$StateMachine[i].redefinedElement</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$StateMachine[i].redefinitionContext</code>
Is Reentrant	reentrant	boolean	Behavior	<code>\$StateMachine[i].reentrant</code>
Region	region	Region[]	StateMachine	<code>\$StateMachine[i].region</code>
Representation	representation	CollaborationUse	Classifier	<code>\$StateMachine[i].representation</code>
Role	role	ConnectableElement[]	StructuredClassifier	<code>\$StateMachine[i].role</code>
Specification	specification	BehavioralFeature	Behavior	<code>\$StateMachine[i].specification</code>
Submachine State	submachineState	State[]	StateMachine	<code>\$StateMachine[i].submachineState</code>
Substitution	substitution	Substitution[]	Classifier	<code>\$StateMachine[i].substitution</code>
Super Class	superClass	Class[]	Class	<code>\$StateMachine[i].superClass</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$StateMachine[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$StateMachine[i].syncElement</code>
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	<code>\$StateMachine[i].templateBinding</code>
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	<code>\$StateMachine[i].templateParameter</code>
Use Case	useCase	UseCase[]	Classifier	<code>\$StateMachine[i].useCase</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$StateMachine[i].visibility</code>

213. Stereotype

Generalization

- [Class](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$Stereotype[i].ID</code>
Class	UMLClass	Class	Classifier	<code>\$Stereotype[i].UMLClass</code>
Is Abstract	abstract	boolean	Class	<code>\$Stereotype[i].abstract</code>
Is Active	active	boolean	Class	<code>\$Stereotype[i].active</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Stereotype[i].appliedStereotypeInstance</code>
Attribute	attribute	Property[]	Classifier	<code>\$Stereotype[i].attribute</code>
classType	classType	Class	BaseElement	<code>\$Stereotype[i].classType</code>
Classifier Behavior	classifierBehavior	Behavior	BehavoredClassifier	<code>\$Stereotype[i].classifierBehavior</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Stereotype[i].clientDependency</code>
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	<code>\$Stereotype[i].collaborationUse</code>
isEditable	editable	boolean	BaseElement	<code>\$Stereotype[i].editable</code>
Element Import	elementImport	ElementImport[]	Namespace	<code>\$Stereotype[i].elementImport</code>

Extension	extension	Extension[]	Class	\$Stereotype[i].extension
Feature	feature	Feature[]	Classifier	\$Stereotype[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$Stereotype[i].finalSpecialization
General	general	Classifier[]	Classifier	\$Stereotype[i].general
Generalization	generalization	Generalization[]	Classifier	\$Stereotype[i].generalization
humanName	humanName	String	BaseElement	\$Stereotype[i].humanName
humanType	humanType	String	BaseElement	\$Stereotype[i].humanType
Icon	icon	Image[]	Stereotype	\$Stereotype[i].icon
Imported Member	importedMember	PackageableElement[]	Namespace	\$Stereotype[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$Stereotype[i].inheritedMember
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavioredClassifier	\$Stereotype[i].interfaceRealization
isInvalid	invalid	boolean	ModelObject	\$Stereotype[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$Stereotype[i].leaf
localID	localID	String	ModelObject	\$Stereotype[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Stereotype[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$Stereotype[i].member
Name	name	String	NamedElement	\$Stereotype[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Stereotype[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Stereotype[i].namespace
Nested Classifier	nestedClassifier	Classifier[]	Class	\$Stereotype[i].nestedClassifier
objectParent	objectParent	BaseElement	BaseElement	\$Stereotype[i].objectParent
Owned Attribute	ownedAttribute	Property[]	Class	\$Stereotype[i].ownedAttribute
Owned Behavior	ownedBehavior	Behavior[]	BehavioredClassifier	\$Stereotype[i].ownedBehavior
Owned Comment	ownedComment	Comment[]	Element	\$Stereotype[i].ownedComment
Owned Connector	ownedConnector	Connector[]	StructuredClassifier	\$Stereotype[i].ownedConnector
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$Stereotype[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$Stereotype[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$Stereotype[i].ownedMember
Owned Operation	ownedOperation	Operation[]	Class	\$Stereotype[i].ownedOperation
Owned Port	ownedPort	Port[]	EncapsulatedClassifier	\$Stereotype[i].ownedPort
Owned Reception	ownedReception	Reception[]	Class	\$Stereotype[i].ownedReception
Owned Rule	ownedRule	Constraint[]	Namespace	\$Stereotype[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$Stereotype[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$Stereotype[i].ownedUseCase
Owner	owner	Element	Element	\$Stereotype[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Stereotype[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Stereotype[i].owningTemplateParameter
Package	package	Package	Type	\$Stereotype[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$Stereotype[i].packageImport

Part	part	Property[]	StructuredClassifier	<code>\$Stereotype[i].part</code>
PowerType Extent	powertypeExtent	GeneralizationSet[]	Classifier	<code>\$Stereotype[i].powertypeExtent</code>
Profile	profile	Profile	Stereotype	<code>\$Stereotype[i].profile</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Stereotype[i].qualifiedName</code>
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	<code>\$Stereotype[i].redefinedClassifier</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$Stereotype[i].redefinedElement</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$Stereotype[i].redefinitionContext</code>
Representation	representation	CollaborationUse	Classifier	<code>\$Stereotype[i].representation</code>
Role	role	ConnectableElement[]	StructuredClassifier	<code>\$Stereotype[i].role</code>
Substitution	substitution	Substitution[]	Classifier	<code>\$Stereotype[i].substitution</code>
Super Class	superClass	Class[]	Class	<code>\$Stereotype[i].superClass</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Stereotype[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Stereotype[i].syncElement</code>
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	<code>\$Stereotype[i].templateBinding</code>
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	<code>\$Stereotype[i].templateParameter</code>
Use Case	useCase	UseCase[]	Classifier	<code>\$Stereotype[i].useCase</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$Stereotype[i].visibility</code>

214. StringExpression

Generalization

- [Expression](#)
- [TemplateableElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$StringExpression[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$StringExpression[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$StringExpression[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$StringExpression[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$StringExpression[i].editable</code>
Expression	expression	Expression	ValueSpecification	<code>\$StringExpression[i].expression</code>
humanName	humanName	String	BaseElement	<code>\$StringExpression[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$StringExpression[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$StringExpression[i].invalid</code>
localID	localID	String	ModelObject	<code>\$StringExpression[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$StringExpression[i].mdExtensions</code>

Name	name	String	NamedElement	\$StringExpression[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$StringExpression[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$StringExpression[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$StringExpression[i].objectParent
Operand	operand	ValueSpecification[]	Expression	\$StringExpression[i].operand
Owned Comment	ownedComment	Comment[]	Element	\$StringExpression[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$StringExpression[i].ownedElement
Owned Template Signature	ownedTemplateSignature	TemplateSignature	TemplateableElement	\$StringExpression[i].ownedTemplateSignature
Owner	owner	Element	Element	\$StringExpression[i].owner
Owning Constraint	owningConstraint	Constraint	ValueSpecification	\$StringExpression[i].owningConstraint
Owning Expression	owningExpression	StringExpression	StringExpression	\$StringExpression[i].owningExpression
Owning Instance Spec	owningInstanceSpec	InstanceSpecification	ValueSpecification	\$StringExpression[i].owningInstanceSpec
Owning Lower	owningLower	MultiplicityElement	ValueSpecification	\$StringExpression[i].owningLower
Owning Package	owningPackage	Package	PackageableElement	\$StringExpression[i].owningPackage
Owning Parameter	owningParameter	Parameter	ValueSpecification	\$StringExpression[i].owningParameter
Owning Property	owningProperty	Property	ValueSpecification	\$StringExpression[i].owningProperty
Owning Slot	owningSlot	Slot	ValueSpecification	\$StringExpression[i].owningSlot
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$StringExpression[i].owningTemplateParameter
Owning Upper	owningUpper	MultiplicityElement	ValueSpecification	\$StringExpression[i].owningUpper
Qualified Name	qualifiedName	String	NamedElement	\$StringExpression[i].qualifiedName
Sub Expression	subExpression	StringExpression[]	StringExpression	\$StringExpression[i].subExpression
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$StringExpression[i].supplierDependency
Symbol	symbol	String	Expression	\$StringExpression[i].symbol
Sync Element	syncElement	Element	Element	\$StringExpression[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$StringExpression[i].templateBinding
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$StringExpression[i].templateParameter
Type	type	Type	TypedElement	\$StringExpression[i].type
Visibility	visibility	VisibilityKind	PackageableElement	\$StringExpression[i].visibility

215. StructuralFeature

Generalization

- Feature
- MultiplicityElement
- TypedElement

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class)	Sample Template Expression (VTL)

			Owner)	
ID	ID	String	MDOBJECT	\$StructuralFeature[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$StructuralFeature[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$StructuralFeature[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$StructuralFeature[i].clientDependency
isEditable	editable	boolean	BaseElement	\$StructuralFeature[i].editable
Featuring Classifier	featuringClassifier	Classifier	Feature	\$StructuralFeature[i].featuringClassifier
humanName	humanName	String	BaseElement	\$StructuralFeature[i].humanName
humanType	humanType	String	BaseElement	\$StructuralFeature[i].humanType
isInvalid	invalid	boolean	ModelObject	\$StructuralFeature[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$StructuralFeature[i].leaf
localID	localID	String	ModelObject	\$StructuralFeature[i].localID
Lower	lower	int	MultiplicityElement	\$StructuralFeature[i].lower
Lower Value	lowerValue	ValueSpecification	MultiplicityElement	\$StructuralFeature[i].lowerValue
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	\$StructuralFeature[i].mdExtensions
Name	name	String	NamedElement	\$StructuralFeature[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$StructuralFeature[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$StructuralFeature[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$StructuralFeature[i].objectParent
Is Ordered	ordered	boolean	MultiplicityElement	\$StructuralFeature[i].ordered
Owned Comment	ownedComment	Comment[]	Element	\$StructuralFeature[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$StructuralFeature[i].ownedElement
Owner	owner	Element	Element	\$StructuralFeature[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$StructuralFeature[i].qualifiedName
Is Read Only	readOnly	boolean	StructuralFeature	\$StructuralFeature[i].readOnly
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$StructuralFeature[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$StructuralFeature[i].redefinitionContext
Is Static	static	boolean	Feature	\$StructuralFeature[i].static
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$StructuralFeature[i].supplierDependency
Sync Element	syncElement	Element	Element	\$StructuralFeature[i].syncElement
Type	type	Type	TypedElement	\$StructuralFeature[i].type
Is Unique	unique	boolean	MultiplicityElement	\$StructuralFeature[i].unique
Upper	upper	int	MultiplicityElement	\$StructuralFeature[i].upper
Upper Value	upperValue	ValueSpecification	MultiplicityElement	\$StructuralFeature[i].upperValue
Visibility	visibility	VisibilityKind	NamedElement	\$StructuralFeature[i].visibility

216. StructuralFeatureAction

Generalization

- Action

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$StructuralFeatureAction[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$StructuralFeatureAction[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$StructuralFeatureAction[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$StructuralFeatureAction[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$StructuralFeatureAction[i].clientDependency</code>
Context	context	Classifier	Action	<code>\$StructuralFeatureAction[i].context</code>
isEditable	editable	boolean	BaseElement	<code>\$StructuralFeatureAction[i].editable</code>
Handler	handler	ExceptionHandler[]	ExecutableNode	<code>\$StructuralFeatureAction[i].handler</code>
humanName	humanName	String	BaseElement	<code>\$StructuralFeatureAction[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$StructuralFeatureAction[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$StructuralFeatureAction[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$StructuralFeatureAction[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$StructuralFeatureAction[i].inPartition</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$StructuralFeatureAction[i].inStructuredNode</code>
Incoming	incoming	ActivityEdge[]	ActivityNode	<code>\$StructuralFeatureAction[i].incoming</code>
Input	input	InputPin[]	Action	<code>\$StructuralFeatureAction[i].input</code>
isInvalid	invalid	boolean	ModelObject	<code>\$StructuralFeatureAction[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$StructuralFeatureAction[i].leaf</code>
localID	localID	String	ModelObject	<code>\$StructuralFeatureAction[i].localID</code>
Local Postcondition	localPostcondition	Constraint[]	Action	<code>\$StructuralFeatureAction[i].localPostcondition</code>
Local Precondition	localPrecondition	Constraint[]	Action	<code>\$StructuralFeatureAction[i].localPrecondition</code>
Is Locally Reentrant	locallyReentrant	boolean	Action	<code>\$StructuralFeatureAction[i].locallyReentrant</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$StructuralFeatureAction[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$StructuralFeatureAction[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$StructuralFeatureAction[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$StructuralFeatureAction[i].namespace</code>
Object	object	InputPin	StructuralFeatureAction	<code>\$StructuralFeatureAction[i].object</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$StructuralFeatureAction[i].objectParent</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$StructuralFeatureAction[i].outgoing</code>
Output	output	OutputPin[]	Action	<code>\$StructuralFeatureAction[i].output</code>

Owned Comment	ownedComment	Comment[]	Element	\$StructuralFeatureAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$StructuralFeatureAction[i].ownedElement
Owner	owner	Element	Element	\$StructuralFeatureAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$StructuralFeatureAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$StructuralFeatureAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$StructuralFeatureAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$StructuralFeatureAction[i].redefinitionContext
Structural Feature	structuralFeature	StructuralFeature	StructuralFeatureAction	\$StructuralFeatureAction[i].structuralFeature
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$StructuralFeatureAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$StructuralFeatureAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$StructuralFeatureAction[i].visibility

217. StructuredActivityNode

Generalization

- [Action](#)
- [ActivityGroup](#)
- [Namespace](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$StructuredActivityNode[i].ID
Activity	activity	Activity	StructuredActivityNode	\$StructuredActivityNode[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$StructuredActivityNode[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$StructuredActivityNode[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$StructuredActivityNode[i].clientDependency
Contained Edge	containedEdge	ActivityEdge[]	ActivityGroup	\$StructuredActivityNode[i].containedEdge
Contained Node	containedNode	ActivityNode[]	ActivityGroup	\$StructuredActivityNode[i].containedNode
Context	context	Classifier	Action	\$StructuredActivityNode[i].context
Edge	edge	ActivityEdge[]	StructuredActivityNode	\$StructuredActivityNode[i].edge
isEditable	editable	boolean	BaseElement	\$StructuredActivityNode[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$StructuredActivityNode[i].elementImport
Handler	handler	ExceptionHandler[]	ExecutableNode	\$StructuredActivityNode[i].handler

humanName	humanName	String	BaseElement	\$StructuredActivityNode[i].humanName
humanType	humanType	String	BaseElement	\$StructuredActivityNode[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$StructuredActivityNode[i].importedMember
In Activity	inActivity	Activity	ActivityGroup	\$StructuredActivityNode[i].inActivity
In Group	inGroup	ActivityGroup[]	ActivityNode	\$StructuredActivityNode[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$StructuredActivityNode[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$StructuredActivityNode[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$StructuredActivityNode[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$StructuredActivityNode[i].incoming
Input	input	InputPin[]	Action	\$StructuredActivityNode[i].input
isInvalid	invalid	boolean	ModelObject	\$StructuredActivityNode[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$StructuredActivityNode[i].leaf
localID	localID	String	ModelObject	\$StructuredActivityNode[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$StructuredActivityNode[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$StructuredActivityNode[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$StructuredActivityNode[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$StructuredActivityNode[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$StructuredActivityNode[i].member
isMustIsolate	mustIsolate	boolean	StructuredActivityNode	\$StructuredActivityNode[i].mustIsolate
Name	name	String	NamedElement	\$StructuredActivityNode[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$StructuredActivityNode[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$StructuredActivityNode[i].namespace
Node	node	ActivityNode[]	StructuredActivityNode	\$StructuredActivityNode[i].node
objectParent	objectParent	BaseElement	BaseElement	\$StructuredActivityNode[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$StructuredActivityNode[i].outgoing
Output	output	OutputPin[]	Action	\$StructuredActivityNode[i].output
Owned Comment	ownedComment	Comment[]	Element	\$StructuredActivityNode[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$StructuredActivityNode[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$StructuredActivityNode[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$StructuredActivityNode[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$StructuredActivityNode[i].ownedRule
Owner	owner	Element	Element	\$StructuredActivityNode[i].owner
Package Import	packageImport	PackageImport[]	Namespace	\$StructuredActivityNode[i].packageImport
Qualified Name	qualifiedName	String	NamedElement	\$StructuredActivityNode[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$StructuredActivityNode[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$StructuredActivityNode[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$StructuredActivityNode[i].redefinitionContext

				xt
Structured Node Input	structuredNodeInput	InputPin[]	StructuredActivityNode	\$StructuredActivityNode[i].structuredNodeInput
Structured Node Output	structuredNodeOutput	OutputPin[]	StructuredActivityNode	\$StructuredActivityNode[i].structuredNodeOutput
Subgroup	subgroup	ActivityGroup[]	ActivityGroup	\$StructuredActivityNode[i].subgroup
Super Group	superGroup	ActivityGroup	ActivityGroup	\$StructuredActivityNode[i].superGroup
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$StructuredActivityNode[i].supplierDependency
Sync Element	syncElement	Element	Element	\$StructuredActivityNode[i].syncElement
Variable	variable	Variable[]	StructuredActivityNode	\$StructuredActivityNode[i].variable
Visibility	visibility	VisibilityKind	NamedElement	\$StructuredActivityNode[i].visibility

218. StructuredClassifier

Generalization

- [Classifier](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$StructuredClassifier[i].ID
Class	UMLClass	Class	Classifier	\$StructuredClassifier[i].UMLClass
Is Abstract	abstract	boolean	Classifier	\$StructuredClassifier[i].abstract
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$StructuredClassifier[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$StructuredClassifier[i].attribute
classType	classType	Class	BaseElement	\$StructuredClassifier[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$StructuredClassifier[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$StructuredClassifier[i].collaborationUse
isEditable	editable	boolean	BaseElement	\$StructuredClassifier[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$StructuredClassifier[i].elementImport
Feature	feature	Feature[]	Classifier	\$StructuredClassifier[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$StructuredClassifier[i].finalSpecialization
General	general	Classifier[]	Classifier	\$StructuredClassifier[i].general
Generalization	generalization	Generalization[]	Classifier	\$StructuredClassifier[i].generalization
humanName	humanName	String	BaseElement	\$StructuredClassifier[i].humanName
humanType	humanType	String	BaseElement	\$StructuredClassifier[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$StructuredClassifier[i].importedMember
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$StructuredClassifier[i].inheritedMember

isInvalid	invalid	boolean	ModelObject	\$StructuredClassifier[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$StructuredClassifier[i].leaf
localID	localID	String	ModelObject	\$StructuredClassifier[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$StructuredClassifier[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$StructuredClassifier[i].member
Name	name	String	NamedElement	\$StructuredClassifier[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$StructuredClassifier[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$StructuredClassifier[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$StructuredClassifier[i].objectParent
Owned Attribute	ownedAttribute	Property[]	StructuredClassifier	\$StructuredClassifier[i].ownedAttribute
Owned Comment	ownedComment	Comment[]	Element	\$StructuredClassifier[i].ownedComment
Owned Connector	ownedConnector	Connector[]	StructuredClassifier	\$StructuredClassifier[i].ownedConnector
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$StructuredClassifier[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$StructuredClassifier[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$StructuredClassifier[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$StructuredClassifier[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$StructuredClassifier[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$StructuredClassifier[i].ownedUseCase
Owner	owner	Element	Element	\$StructuredClassifier[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$StructuredClassifier[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$StructuredClassifier[i].owningTemplateParameter
Package	package	Package	Type	\$StructuredClassifier[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$StructuredClassifier[i].packageImport
Part	part	Property[]	StructuredClassifier	\$StructuredClassifier[i].part
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$StructuredClassifier[i].powertypeExtent
Qualified Name	qualifiedName	String	NamedElement	\$StructuredClassifier[i].qualifiedName
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$StructuredClassifier[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$StructuredClassifier[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$StructuredClassifier[i].redefinitionContext
Representation	representation	CollaborationUse	Classifier	\$StructuredClassifier[i].representation
Role	role	ConnectableElement[]	StructuredClassifier	\$StructuredClassifier[i].role
Substitution	substitution	Substitution[]	Classifier	\$StructuredClassifier[i].substitution
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$StructuredClassifier[i].supplierDependency
Sync Element	syncElement	Element	Element	\$StructuredClassifier[i].syncElement
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	\$StructuredClassifier[i].templateBinding
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	\$StructuredClassifier[i].templateParameter
Use Case	useCase	UseCase[]	Classifier	\$StructuredClassifier[i].useCase

Visibility	visibility	VisibilityKind	PackageableElement	\$StructuredClassifier[i].visibility

219. Substitution

Generalization

- Realization

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$Substitution[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Substitution[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Substitution[i].classType
Client	client	NamedElement[]	Dependency	\$Substitution[i].client
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Substitution[i].clientDependency
Contract	contract	Classifier	Substitution	\$Substitution[i].contract
isEditable	editable	boolean	BaseElement	\$Substitution[i].editable
humanName	humanName	String	BaseElement	\$Substitution[i].humanName
humanType	humanType	String	BaseElement	\$Substitution[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Substitution[i].invalid
localID	localID	String	ModelObject	\$Substitution[i].localID
Mapping	mapping	OpaqueExpression	Abstraction	\$Substitution[i].mapping
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	\$Substitution[i].mdExtensions
Name	name	String	NamedElement	\$Substitution[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Substitution[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Substitution[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Substitution[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Substitution[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Substitution[i].ownedElement
Owner	owner	Element	Element	\$Substitution[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Substitution[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Substitution[i].owningTemplateParameter
Qualified Name	qualifiedName	String	NamedElement	\$Substitution[i].qualifiedName
Related Element	relatedElement	Element[]	Relationship	\$Substitution[i].relatedElement
Source	source	Element[]	DirectedRelationship	\$Substitution[i].source
Substituting Classifier	substitutingClassifier	Classifier	Substitution	\$Substitution[i].substitutingClassifier
Supplier	supplier	NamedElement[]	Dependency	\$Substitution[i].supplier
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Substitution[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Substitution[i].syncElement
Target	target	Element[]	DirectedRelationship	\$Substitution[i].target

Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$Substitution[i].templateParameter</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$Substitution[i].visibility</code>

220. TemplateableElement

Generalization

- [Element](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$TemplateableElement[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$TemplateableElement[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$TemplateableElement[i].classType</code>
isEditable	editable	boolean	BaseElement	<code>\$TemplateableElement[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$TemplateableElement[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$TemplateableElement[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$TemplateableElement[i].invalid</code>
localID	localID	String	ModelObject	<code>\$TemplateableElement[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	<code>\$TemplateableElement[i].mdExtensions</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$TemplateableElement[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$TemplateableElement[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$TemplateableElement[i].ownedElement</code>
Owned Template Signature	ownedTemplateSignature	TemplateSignature	TemplateableElement	<code>\$TemplateableElement[i].ownedTemplateSignature</code>
Owner	owner	Element	Element	<code>\$TemplateableElement[i].owner</code>
Sync Element	syncElement	Element	Element	<code>\$TemplateableElement[i].syncElement</code>
Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	<code>\$TemplateableElement[i].templateBinding</code>

221. TemplateBinding

Generalization

- [DirectedRelationship](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$TemplateBinding[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$TemplateBinding[i].appliedStereotypeInsta</code>

				nce
Bound Element	boundElement	TemplateableElement	TemplateBinding	\$TemplateBinding[i].boundElement
classType	classType	Class	BaseElement	\$TemplateBinding[i].classType
isEditable	editable	boolean	BaseElement	\$TemplateBinding[i].editable
humanName	humanName	String	BaseElement	\$TemplateBinding[i].humanName
humanType	humanType	String	BaseElement	\$TemplateBinding[i].humanType
isInvalid	invalid	boolean	ModelObject	\$TemplateBinding[i].invalid
localID	localID	String	ModelObject	\$TemplateBinding[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$TemplateBinding[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$TemplateBinding[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$TemplateBinding[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$TemplateBinding[i].ownedElement
Owner	owner	Element	Element	\$TemplateBinding[i].owner
Parameter Substitution	parameterSubstitution	TemplateParameterSubstitution[]	TemplateBinding	\$TemplateBinding[i].parameterSubstitution
Related Element	relatedElement	Element[]	Relationship	\$TemplateBinding[i].relatedElement
Signature	signature	TemplateSignature	TemplateBinding	\$TemplateBinding[i].signature
Source	source	Element[]	DirectedRelationship	\$TemplateBinding[i].source
Sync Element	syncElement	Element	Element	\$TemplateBinding[i].syncElement
Target	target	Element[]	DirectedRelationship	\$TemplateBinding[i].target

222. TemplateParameter

Generalization

- [Element](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$TemplateParameter[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$TemplateParameter[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$TemplateParameter[i].classType
Default	default	ParameterableElement	TemplateParameter	\$TemplateParameter[i].default
isEditable	editable	boolean	BaseElement	\$TemplateParameter[i].editable
humanName	humanName	String	BaseElement	\$TemplateParameter[i].humanName
humanType	humanType	String	BaseElement	\$TemplateParameter[i].humanType
isInvalid	invalid	boolean	ModelObject	\$TemplateParameter[i].invalid
localID	localID	String	ModelObject	\$TemplateParameter[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$TemplateParameter[i].mdExtensions

objectParent	objectParent	BaseElement	BaseElement	\$TemplateParameter[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$TemplateParameter[i].ownedComment
Owned Default	ownedDefault	ParameterableElement	TemplateParameter	\$TemplateParameter[i].ownedDefault
Owned Element	ownedElement	Element[]	Element	\$TemplateParameter[i].ownedElement
Owned Parametered Element	ownedParameteredElement	ParameterableElement	TemplateParameter	\$TemplateParameter[i].ownedParameteredElement
Owner	owner	Element	Element	\$TemplateParameter[i].owner
Parametered Element	parameteredElement	ParameterableElement	TemplateParameter	\$TemplateParameter[i].parameteredElement
Signature	signature	TemplateSignature	TemplateParameter	\$TemplateParameter[i].signature
Sync Element	syncElement	Element	Element	\$TemplateParameter[i].syncElement

223. TemplateParameterSubstitution

Generalization

- [Element](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$TemplateParameterSubstitution[i].ID
Actual	actual	ParameterableElement	TemplateParameterSubstitution	\$TemplateParameterSubstitution[i].actual
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$TemplateParameterSubstitution[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$TemplateParameterSubstitution[i].classType
isEditable	editable	boolean	BaseElement	\$TemplateParameterSubstitution[i].editable
Formal	formal	TemplateParameter	TemplateParameterSubstitution	\$TemplateParameterSubstitution[i].formal
humanName	humanName	String	BaseElement	\$TemplateParameterSubstitution[i].humanName
humanType	humanType	String	BaseElement	\$TemplateParameterSubstitution[i].humanType
isInvalid	invalid	boolean	ModelObject	\$TemplateParameterSubstitution[i].invalid
localID	localID	String	ModelObject	\$TemplateParameterSubstitution[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$TemplateParameterSubstitution[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$TemplateParameterSubstitution[i].objectParent
Owned Actual	ownedActual	ParameterableElement	TemplateParameterSubstitution	\$TemplateParameterSubstitution[i].ownedActual

			itution	ctual
Owned Comment	ownedComment	Comment[]	Element	\$TemplateParameterSubstitution[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$TemplateParameterSubstitution[i].ownedElement
Owner	owner	Element	Element	\$TemplateParameterSubstitution[i].owner
Sync Element	syncElement	Element	Element	\$TemplateParameterSubstitution[i].syncElement
Template Binding	templateBinding	TemplateBinding	TemplateParameterSubstitution	\$TemplateParameterSubstitution[i].templateBinding

224. TemplateSignature

Generalization

- [Element](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$TemplateSignature[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$TemplateSignature[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$TemplateSignature[i].classType
isEditable	editable	boolean	BaseElement	\$TemplateSignature[i].editable
humanName	humanName	String	BaseElement	\$TemplateSignature[i].humanName
humanType	humanType	String	BaseElement	\$TemplateSignature[i].humanType
isInvalid	invalid	boolean	ModelObject	\$TemplateSignature[i].invalid
localID	localID	String	ModelObject	\$TemplateSignature[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$TemplateSignature[i].mdExtensions
objectParent	objectParent	BaseElement	BaseElement	\$TemplateSignature[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$TemplateSignature[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$TemplateSignature[i].ownedElement
Owned Parameter	ownedParameter	TemplateParameter[]	TemplateSignature	\$TemplateSignature[i].ownedParameter
Owner	owner	Element	Element	\$TemplateSignature[i].owner
Parameter	parameter	TemplateParameter[]	TemplateSignature	\$TemplateSignature[i].parameter
Sync Element	syncElement	Element	Element	\$TemplateSignature[i].syncElement
Template	template	TemplateableElement	TemplateSignature	\$TemplateSignature[i].template

225. TestIdentityAction

Generalization

- Action

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$TestIdentityAction[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$TestIdentityAction[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$TestIdentityAction[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$TestIdentityAction[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$TestIdentityAction[i].clientDependency</code>
Context	context	Classifier	Action	<code>\$TestIdentityAction[i].context</code>
isEditable	editable	boolean	BaseElement	<code>\$TestIdentityAction[i].editable</code>
First	first	InputPin	TestIdentityAction	<code>\$TestIdentityAction[i].first</code>
Handler	handler	ExceptionHandler[]	ExecutableNode	<code>\$TestIdentityAction[i].handler</code>
humanName	humanName	String	BaseElement	<code>\$TestIdentityAction[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$TestIdentityAction[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$TestIdentityAction[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$TestIdentityAction[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$TestIdentityAction[i].inPartition</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$TestIdentityAction[i].inStructuredNode</code>
Incoming	incoming	ActivityEdge[]	ActivityNode	<code>\$TestIdentityAction[i].incoming</code>
Input	input	InputPin[]	Action	<code>\$TestIdentityAction[i].input</code>
isInvalid	invalid	boolean	ModelObject	<code>\$TestIdentityAction[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$TestIdentityAction[i].leaf</code>
localID	localID	String	ModelObject	<code>\$TestIdentityAction[i].localID</code>
Local Postcondition	localPostcondition	Constraint[]	Action	<code>\$TestIdentityAction[i].localPostcondition</code>
Local Precondition	localPrecondition	Constraint[]	Action	<code>\$TestIdentityAction[i].localPrecondition</code>
Is Locally Reentrant	locallyReentrant	boolean	Action	<code>\$TestIdentityAction[i].locallyReentrant</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$TestIdentityAction[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$TestIdentityAction[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$TestIdentityAction[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$TestIdentityAction[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$TestIdentityAction[i].objectParent</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$TestIdentityAction[i].outgoing</code>
Output	output	OutputPin[]	Action	<code>\$TestIdentityAction[i].output</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$TestIdentityAction[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$TestIdentityAction[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$TestIdentityAction[i].owner</code>

Qualified Name	qualifiedName	String	NamedElement	\$TestIdentityAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$TestIdentityAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$TestIdentityAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$TestIdentityAction[i].redefinitionContext
Result	result	OutputPin	TestIdentityAction	\$TestIdentityAction[i].result
Second	second	InputPin	TestIdentityAction	\$TestIdentityAction[i].second
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$TestIdentityAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$TestIdentityAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$TestIdentityAction[i].visibility

226. TimeConstraint

Generalization

- IntervalConstraint

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$TimeConstraint[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$TimeConstraint[i].appliedStereotypeInstance
Body Context	bodyContext	Operation	Constraint	\$TimeConstraint[i].bodyContext
classType	classType	Class	BaseElement	\$TimeConstraint[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$TimeConstraint[i].clientDependency
Constrained Element	constrainedElement	Element[]	Constraint	\$TimeConstraint[i].constrainedElement
Context	context	Namespace	Constraint	\$TimeConstraint[i].context
isEditable	editable	boolean	BaseElement	\$TimeConstraint[i].editable
isFirstEvent	firstEvent	boolean	TimeConstraint	\$TimeConstraint[i].firstEvent
humanName	humanName	String	BaseElement	\$TimeConstraint[i].humanName
humanType	humanType	String	BaseElement	\$TimeConstraint[i].humanType
isInvalid	invalid	boolean	ModelObject	\$TimeConstraint[i].invalid
localID	localID	String	ModelObject	\$TimeConstraint[i].localID
mdExtensions	mdExtensions	MDEXTENSION[]	MDOBJECT	\$TimeConstraint[i].mdExtensions
Name	name	String	NamedElement	\$TimeConstraint[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$TimeConstraint[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$TimeConstraint[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$TimeConstraint[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$TimeConstraint[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$TimeConstraint[i].ownedElement
Owner	owner	Element	Element	\$TimeConstraint[i].owner

Owning Package	owningPackage	Package	PackageableElement	\$TimeConstraint[i].owningPackage
Owning State	owningState	State	Constraint	\$TimeConstraint[i].owningState
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$TimeConstraint[i].owningTemplateParameter
Owning Transition	owningTransition	ProtocolTransition	Constraint	\$TimeConstraint[i].owningTransition
Post Context	postContext	Operation	Constraint	\$TimeConstraint[i].postContext
Pre Context	preContext	Operation	Constraint	\$TimeConstraint[i].preContext
Qualified Name	qualifiedName	String	NamedElement	\$TimeConstraint[i].qualifiedName
Specification	specification	TimeInterval	TimeConstraint	\$TimeConstraint[i].specification
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$TimeConstraint[i].supplierDependency
Sync Element	syncElement	Element	Element	\$TimeConstraint[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$TimeConstraint[i].templateParameter
Visibility	visibility	VisibilityKind	PackageableElement	\$TimeConstraint[i].visibility

227. TimeEvent

Generalization

- [Event](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$TimeEvent[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$TimeEvent[i].appliedStereotypeInstance
Behavior	behavior	Behavior	Event	\$TimeEvent[i].behavior
classType	classType	Class	BaseElement	\$TimeEvent[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$TimeEvent[i].clientDependency
isEditable	editable	boolean	BaseElement	\$TimeEvent[i].editable
humanName	humanName	String	BaseElement	\$TimeEvent[i].humanName
humanType	humanType	String	BaseElement	\$TimeEvent[i].humanType
isInvalid	invalid	boolean	ModelObject	\$TimeEvent[i].invalid
localID	localID	String	ModelObject	\$TimeEvent[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$TimeEvent[i].mdExtensions
Name	name	String	NamedElement	\$TimeEvent[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$TimeEvent[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$TimeEvent[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$TimeEvent[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$TimeEvent[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$TimeEvent[i].ownedElement
Owner	owner	Element	Element	\$TimeEvent[i].owner

Owning Package	owningPackage	Package	PackageableElement	\$TimeEvent[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$TimeEvent[i].owningTemplateParameter
Qualified Name	qualifiedName	String	NamedElement	\$TimeEvent[i].qualifiedName
Is Relative	relative	boolean	TimeEvent	\$TimeEvent[i].relative
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$TimeEvent[i].supplierDependency
Sync Element	syncElement	Element	Element	\$TimeEvent[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$TimeEvent[i].templateParameter
Visibility	visibility	VisibilityKind	PackageableElement	\$TimeEvent[i].visibility
When	when	TimeExpression	TimeEvent	\$TimeEvent[i].when

228. TimeExpression

Generalization

- [ValueSpecification](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$TimeExpression[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$TimeExpression[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$TimeExpression[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$TimeExpression[i].clientDependency
isEditable	editable	boolean	BaseElement	\$TimeExpression[i].editable
Expr	expr	ValueSpecification	TimeExpression	\$TimeExpression[i].expr
Expression	expression	Expression	ValueSpecification	\$TimeExpression[i].expression
humanName	humanName	String	BaseElement	\$TimeExpression[i].humanName
humanType	humanType	String	BaseElement	\$TimeExpression[i].humanType
isInvalid	invalid	boolean	ModelObject	\$TimeExpression[i].invalid
localID	localID	String	ModelObject	\$TimeExpression[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$TimeExpression[i].mdExtensions
Name	name	String	NamedElement	\$TimeExpression[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$TimeExpression[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$TimeExpression[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$TimeExpression[i].objectParent
Observation	observation	Observation[]	TimeExpression	\$TimeExpression[i].observation
Owned Comment	ownedComment	Comment[]	Element	\$TimeExpression[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$TimeExpression[i].ownedElement
Owner	owner	Element	Element	\$TimeExpression[i].owner
Owning Constraint	owningConstraint	Constraint	ValueSpecification	\$TimeExpression[i].owningConstraint

Owning Instance Spec	owningInstanceSpec	InstanceSpecification	ValueSpecification	\$TimeExpression[i].owningInstanceSpec
Owning Lower	owningLower	MultiplicityElement	ValueSpecification	\$TimeExpression[i].owningLower
Owning Package	owningPackage	Package	PackageableElement	\$TimeExpression[i].owningPackage
Owning Parameter	owningParameter	Parameter	ValueSpecification	\$TimeExpression[i].owningParameter
Owning Property	owningProperty	Property	ValueSpecification	\$TimeExpression[i].owningProperty
Owning Slot	owningSlot	Slot	ValueSpecification	\$TimeExpression[i].owningSlot
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$TimeExpression[i].owningTemplateParameter
Owning Upper	owningUpper	MultiplicityElement	ValueSpecification	\$TimeExpression[i].owningUpper
Qualified Name	qualifiedName	String	NamedElement	\$TimeExpression[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$TimeExpression[i].supplierDependency
Sync Element	syncElement	Element	Element	\$TimeExpression[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$TimeExpression[i].templateParameter
Type	type	Type	TypedElement	\$TimeExpression[i].type
Visibility	visibility	VisibilityKind	PackageableElement	\$TimeExpression[i].visibility

229. TimeInterval

Generalization

- Interval

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$TimeInterval[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$TimeInterval[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$TimeInterval[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$TimeInterval[i].clientDependency
isEditable	editable	boolean	BaseElement	\$TimeInterval[i].editable
Expression	expression	Expression	ValueSpecification	\$TimeInterval[i].expression
humanName	humanName	String	BaseElement	\$TimeInterval[i].humanName
humanType	humanType	String	BaseElement	\$TimeInterval[i].humanType
isInvalid	invalid	boolean	ModelObject	\$TimeInterval[i].invalid
localID	localID	String	ModelObject	\$TimeInterval[i].localID
Max	max	TimeExpression	TimeInterval	\$TimeInterval[i].max
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$TimeInterval[i].mdExtensions
Min	min	TimeExpression	TimeInterval	\$TimeInterval[i].min
Name	name	String	NamedElement	\$TimeInterval[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$TimeInterval[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$TimeInterval[i].namespace

objectParent	objectParent	BaseElement	BaseElement	\$TimeInterval[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$TimeInterval[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$TimeInterval[i].ownedElement
Owner	owner	Element	Element	\$TimeInterval[i].owner
Owning Constraint	owningConstraint	Constraint	ValueSpecification	\$TimeInterval[i].owningConstraint
Owning Instance Spec	owningInstanceSpec	InstanceSpecification	ValueSpecification	\$TimeInterval[i].owningInstanceSpec
Owning Lower	owningLower	MultiplicityElement	ValueSpecification	\$TimeInterval[i].owningLower
Owning Package	owningPackage	Package	PackageableElement	\$TimeInterval[i].owningPackage
Owning Parameter	owningParameter	Parameter	ValueSpecification	\$TimeInterval[i].owningParameter
Owning Property	owningProperty	Property	ValueSpecification	\$TimeInterval[i].owningProperty
Owning Slot	owningSlot	Slot	ValueSpecification	\$TimeInterval[i].owningSlot
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$TimeInterval[i].owningTemplateParameter
Owning Upper	owningUpper	MultiplicityElement	ValueSpecification	\$TimeInterval[i].owningUpper
Qualified Name	qualifiedName	String	NamedElement	\$TimeInterval[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$TimeInterval[i].supplierDependency
Sync Element	syncElement	Element	Element	\$TimeInterval[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$TimeInterval[i].templateParameter
Type	type	Type	TypedElement	\$TimeInterval[i].type
Visibility	visibility	VisibilityKind	PackageableElement	\$TimeInterval[i].visibility

230. TimeObservation

Generalization

- [Observation](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$TimeObservation[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$TimeObservation[i].appliedStereotypeInstance
Behavior	behavior	Behavior	Observation	\$TimeObservation[i].behavior
classType	classType	Class	BaseElement	\$TimeObservation[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$TimeObservation[i].clientDependency
isEditable	editable	boolean	BaseElement	\$TimeObservation[i].editable
Event	event	NamedElement	TimeObservation	\$TimeObservation[i].event
isFirstEvent	firstEvent	boolean	TimeObservation	\$TimeObservation[i].firstEvent
humanName	humanName	String	BaseElement	\$TimeObservation[i].humanName
humanType	humanType	String	BaseElement	\$TimeObservation[i].humanType
isInvalid	invalid	boolean	ModelObject	\$TimeObservation[i].invalid

localID	localID	String	ModelObject	\$TimeObservation[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$TimeObservation[i].mdExtensions
Name	name	String	NamedElement	\$TimeObservation[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$TimeObservation[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$TimeObservation[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$TimeObservation[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$TimeObservation[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$TimeObservation[i].ownedElement
Owner	owner	Element	Element	\$TimeObservation[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$TimeObservation[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$TimeObservation[i].owningTemplateParameter
Qualified Name	qualifiedName	String	NamedElement	\$TimeObservation[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$TimeObservation[i].supplierDependency
Sync Element	syncElement	Element	Element	\$TimeObservation[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$TimeObservation[i].templateParameter
Visibility	visibility	VisibilityKind	PackageableElement	\$TimeObservation[i].visibility

231. Transition

Generalization

- [Namespace](#)
- [RedefinableElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Transition[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Transition[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Transition[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Transition[i].clientDependency
Container	container	Region	Transition	\$Transition[i].container
isEditable	editable	boolean	BaseElement	\$Transition[i].editable
Effect	effect	Behavior	Transition	\$Transition[i].effect
Element Import	elementImport	ElementImport[]	Namespace	\$Transition[i].elementImport
Guard	guard	Constraint	Transition	\$Transition[i].guard
humanName	humanName	String	BaseElement	\$Transition[i].humanName
humanType	humanType	String	BaseElement	\$Transition[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$Transition[i].importedMember
isInvalid	invalid	boolean	ModelObject	\$Transition[i].invalid

Kind	kind	TransitionKind	Transition	<code>\$Transition[i].kind</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$Transition[i].leaf</code>
localID	localID	String	ModelObject	<code>\$Transition[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$Transition[i].mdExtensions</code>
Member	member	NamedElement[]	Namespace	<code>\$Transition[i].member</code>
Name	name	String	NamedElement	<code>\$Transition[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Transition[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Transition[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Transition[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Transition[i].ownedComment</code>
Owned Diagram	ownedDiagram	Diagram[]	Namespace	<code>\$Transition[i].ownedDiagram</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Transition[i].ownedElement</code>
Owned Member	ownedMember	NamedElement[]	Namespace	<code>\$Transition[i].ownedMember</code>
Owned Rule	ownedRule	Constraint[]	Namespace	<code>\$Transition[i].ownedRule</code>
Owner	owner	Element	Element	<code>\$Transition[i].owner</code>
Package Import	packageImport	PackageImport[]	Namespace	<code>\$Transition[i].packageImport</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Transition[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$Transition[i].redefinedElement</code>
Redefined Transition	redefinedTransition	Transition	Transition	<code>\$Transition[i].redefinedTransition</code>
Redefinition Context	redefinitionContext	Classifier[]	Transition	<code>\$Transition[i].redefinitionContext</code>
Source	source	Vertex	Transition	<code>\$Transition[i].source</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Transition[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Transition[i].syncElement</code>
Target	target	Vertex	Transition	<code>\$Transition[i].target</code>
Trigger	trigger	Trigger[]	Transition	<code>\$Transition[i].trigger</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$Transition[i].visibility</code>

232. Trigger

Generalization

- [NamedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$Trigger[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Trigger[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$Trigger[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Trigger[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$Trigger[i].editable</code>

Event	event	Event	Trigger	\$Trigger[i].event
humanName	humanName	String	BaseElement	\$Trigger[i].humanName
humanType	humanType	String	BaseElement	\$Trigger[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Trigger[i].invalid
localID	localID	String	ModelObject	\$Trigger[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Trigger[i].mdExtensions
Name	name	String	NamedElement	\$Trigger[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Trigger[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Trigger[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$Trigger[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Trigger[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Trigger[i].ownedElement
Owner	owner	Element	Element	\$Trigger[i].owner
Port	port	Port[]	Trigger	\$Trigger[i].port
Qualified Name	qualifiedName	String	NamedElement	\$Trigger[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Trigger[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Trigger[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$Trigger[i].visibility

233. Type

Generalization

- [PackageableElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Type[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Type[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Type[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Type[i].clientDependency
isEditable	editable	boolean	BaseElement	\$Type[i].editable
humanName	humanName	String	BaseElement	\$Type[i].humanName
humanType	humanType	String	BaseElement	\$Type[i].humanType
isInvalid	invalid	boolean	ModelObject	\$Type[i].invalid
localID	localID	String	ModelObject	\$Type[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$Type[i].mdExtensions
Name	name	String	NamedElement	\$Type[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$Type[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$Type[i].namespace

objectParent	objectParent	BaseElement	BaseElement	\$Type[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Type[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Type[i].ownedElement
Owner	owner	Element	Element	\$Type[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Type[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Type[i].owningTemplateParameter
Package	package	Package	Type	\$Type[i].package
Qualified Name	qualifiedName	String	NamedElement	\$Type[i].qualifiedName
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Type[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Type[i].syncElement
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$Type[i].templateParameter
Visibility	visibility	VisibilityKind	PackageableElement	\$Type[i].visibility

234. TypedElement

Generalization

- [NamedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$TypedElement[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$TypedElement[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$TypedElement[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$TypedElement[i].clientDependency
isEditable	editable	boolean	BaseElement	\$TypedElement[i].editable
humanName	humanName	String	BaseElement	\$TypedElement[i].humanName
humanType	humanType	String	BaseElement	\$TypedElement[i].humanType
isInvalid	invalid	boolean	ModelObject	\$TypedElement[i].invalid
localID	localID	String	ModelObject	\$TypedElement[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$TypedElement[i].mdExtensions
Name	name	String	NamedElement	\$TypedElement[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$TypedElement[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$TypedElement[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$TypedElement[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$TypedElement[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$TypedElement[i].ownedElement
Owner	owner	Element	Element	\$TypedElement[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$TypedElement[i].qualifiedName

Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$TypedElement[i].supplierDependency
Sync Element	syncElement	Element	Element	\$TypedElement[i].syncElement
Type	type	Type	TypedElement	\$TypedElement[i].type
Visibility	visibility	VisibilityKind	NamedElement	\$TypedElement[i].visibility

235. UnmarshallAction

Generalization

- [Action](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	\$UnmarshallAction[i].ID
Activity	activity	Activity	ActivityNode	\$UnmarshallAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$UnmarshallAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$UnmarshallAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$UnmarshallAction[i].clientDependency
Context	context	Classifier	Action	\$UnmarshallAction[i].context
isEditable	editable	boolean	BaseElement	\$UnmarshallAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$UnmarshallAction[i].handler
humanName	humanName	String	BaseElement	\$UnmarshallAction[i].humanName
humanType	humanType	String	BaseElement	\$UnmarshallAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$UnmarshallAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$UnmarshallAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$UnmarshallAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$UnmarshallAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$UnmarshallAction[i].incoming
Input	input	InputPin[]	Action	\$UnmarshallAction[i].input
isInvalid	invalid	boolean	ModelObject	\$UnmarshallAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$UnmarshallAction[i].leaf
localID	localID	String	ModelObject	\$UnmarshallAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$UnmarshallAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$UnmarshallAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$UnmarshallAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$UnmarshallAction[i].mdExtensions
Name	name	String	NamedElement	\$UnmarshallAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$UnmarshallAction[i].nameExpression

Namespace	namespace	Namespace	NamedElement	<code>\$UnmarshallAction[i].namespace</code>
Object	object	InputPin	UnmarshallAction	<code>\$UnmarshallAction[i].object</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$UnmarshallAction[i].objectParent</code>
Outgoing	outgoing	ActivityEdge[]	ActivityNode	<code>\$UnmarshallAction[i].outgoing</code>
Output	output	OutputPin[]	Action	<code>\$UnmarshallAction[i].output</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$UnmarshallAction[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$UnmarshallAction[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$UnmarshallAction[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$UnmarshallAction[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$UnmarshallAction[i].redefinedElement</code>
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	<code>\$UnmarshallAction[i].redefinedNode</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$UnmarshallAction[i].redefinitionContext</code>
Result	result	OutputPin[]	UnmarshallAction	<code>\$UnmarshallAction[i].result</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$UnmarshallAction[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$UnmarshallAction[i].syncElement</code>
Unmarshall Type	unmarshallType	Classifier	UnmarshallAction	<code>\$UnmarshallAction[i].unmarshallType</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$UnmarshallAction[i].visibility</code>

236. Usage

Generalization

- [Dependency](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$Usage[i].ID</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Usage[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$Usage[i].classType</code>
Client	client	NamedElement[]	Dependency	<code>\$Usage[i].client</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Usage[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$Usage[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$Usage[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Usage[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Usage[i].invalid</code>
localID	localID	String	ModelObject	<code>\$Usage[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$Usage[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$Usage[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Usage[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Usage[i].namespace</code>

objectParent	objectParent	BaseElement	BaseElement	\$Usage[i].objectParent
Owned Comment	ownedComment	Comment[]	Element	\$Usage[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$Usage[i].ownedElement
Owner	owner	Element	Element	\$Usage[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$Usage[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$Usage[i].owningTemplateParameter
Qualified Name	qualifiedName	String	NamedElement	\$Usage[i].qualifiedName
Related Element	relatedElement	Element[]	Relationship	\$Usage[i].relatedElement
Source	source	Element[]	DirectedRelationship	\$Usage[i].source
Supplier	supplier	NamedElement[]	Dependency	\$Usage[i].supplier
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Usage[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Usage[i].syncElement
Target	target	Element[]	DirectedRelationship	\$Usage[i].target
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	\$Usage[i].templateParameter
Visibility	visibility	VisibilityKind	PackageableElement	\$Usage[i].visibility

237. UseCase

Generalization

- [BehavoredClassifier](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$UseCase[i].ID
Class	UMLClass	Class	Classifier	\$UseCase[i].UMLClass
Is Abstract	abstract	boolean	Classifier	\$UseCase[i].abstract
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$UseCase[i].appliedStereotypeInstance
Attribute	attribute	Property[]	Classifier	\$UseCase[i].attribute
classType	classType	Class	BaseElement	\$UseCase[i].classType
Classifier Behavior	classifierBehavior	Behavior	BehavoredClassifier	\$UseCase[i].classifierBehavior
Client Dependency	clientDependency	Dependency[]	NamedElement	\$UseCase[i].clientDependency
Collaboration Use	collaborationUse	CollaborationUse[]	Classifier	\$UseCase[i].collaborationUse
isEditable	editable	boolean	BaseElement	\$UseCase[i].editable
Element Import	elementImport	ElementImport[]	Namespace	\$UseCase[i].elementImport
Extend	extend	Extend[]	UseCase	\$UseCase[i].extend
Extension Point	extensionPoint	ExtensionPoint[]	UseCase	\$UseCase[i].extensionPoint
Feature	feature	Feature[]	Classifier	\$UseCase[i].feature
Is Final Specialization	finalSpecialization	boolean	Classifier	\$UseCase[i].finalSpecialization
General	general	Classifier[]	Classifier	\$UseCase[i].general

Generalization	generalization	Generalization[]	Classifier	\$UseCase[i].generalization
humanName	humanName	String	BaseElement	\$UseCase[i].humanName
humanType	humanType	String	BaseElement	\$UseCase[i].humanType
Imported Member	importedMember	PackageableElement[]	Namespace	\$UseCase[i].importedMember
Include	include	Include[]	UseCase	\$UseCase[i].include
Inherited Member	inheritedMember	NamedElement[]	Classifier	\$UseCase[i].inheritedMember
Interface Realization	interfaceRealization	InterfaceRealization[]	BehavioredClassifier	\$UseCase[i].interfaceRealization
isInvalid	invalid	boolean	ModelObject	\$UseCase[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$UseCase[i].leaf
localID	localID	String	ModelObject	\$UseCase[i].localID
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$UseCase[i].mdExtensions
Member	member	NamedElement[]	Namespace	\$UseCase[i].member
Name	name	String	NamedElement	\$UseCase[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$UseCase[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$UseCase[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$UseCase[i].objectParent
Owned Behavior	ownedBehavior	Behavior[]	BehavioredClassifier	\$UseCase[i].ownedBehavior
Owned Comment	ownedComment	Comment[]	Element	\$UseCase[i].ownedComment
Owned Diagram	ownedDiagram	Diagram[]	Namespace	\$UseCase[i].ownedDiagram
Owned Element	ownedElement	Element[]	Element	\$UseCase[i].ownedElement
Owned Member	ownedMember	NamedElement[]	Namespace	\$UseCase[i].ownedMember
Owned Rule	ownedRule	Constraint[]	Namespace	\$UseCase[i].ownedRule
Owned Template Signature	ownedTemplateSignature	RedefinableTemplateSignature	Classifier	\$UseCase[i].ownedTemplateSignature
Owned Use Case	ownedUseCase	UseCase[]	Classifier	\$UseCase[i].ownedUseCase
Owner	owner	Element	Element	\$UseCase[i].owner
Owning Package	owningPackage	Package	PackageableElement	\$UseCase[i].owningPackage
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	\$UseCase[i].owningTemplateParameter
Package	package	Package	Type	\$UseCase[i].package
Package Import	packageImport	PackageImport[]	Namespace	\$UseCase[i].packageImport
Powertype Extent	powertypeExtent	GeneralizationSet[]	Classifier	\$UseCase[i].powertypeExtent
Qualified Name	qualifiedName	String	NamedElement	\$UseCase[i].qualifiedName
Redefined Classifier	redefinedClassifier	Classifier[]	Classifier	\$UseCase[i].redefinedClassifier
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$UseCase[i].redefinedElement
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$UseCase[i].redefinitionContext
Representation	representation	CollaborationUse	Classifier	\$UseCase[i].representation
Subject	subject	Classifier[]	UseCase	\$UseCase[i].subject
Substitution	substitution	Substitution[]	Classifier	\$UseCase[i].substitution
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$UseCase[i].supplierDependency
Sync Element	syncElement	Element	Element	\$UseCase[i].syncElement

Template Binding	templateBinding	TemplateBinding[]	TemplateableElement	<code>\$UseCase[i].templateBinding</code>
Template Parameter	templateParameter	ClassifierTemplateParameter	Classifier	<code>\$UseCase[i].templateParameter</code>
Use Case	useCase	UseCase[]	Classifier	<code>\$UseCase[i].useCase</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$UseCase[i].visibility</code>

238. ValuePin

Generalization

- [InputPin](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$ValuePin[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$ValuePin[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$ValuePin[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$ValuePin[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$ValuePin[i].clientDependency</code>
Is Control	control	boolean	Pin	<code>\$ValuePin[i].control</code>
Is Control Type	controlType	boolean	ObjectNode	<code>\$ValuePin[i].controlType</code>
isEditable	editable	boolean	BaseElement	<code>\$ValuePin[i].editable</code>
humanName	humanName	String	BaseElement	<code>\$ValuePin[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$ValuePin[i].humanType</code>
In Group	inGroup	ActivityGroup[]	ActivityNode	<code>\$ValuePin[i].inGroup</code>
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	<code>\$ValuePin[i].inInterruptibleRegion</code>
In Partition	inPartition	ActivityPartition[]	ActivityNode	<code>\$ValuePin[i].inPartition</code>
In State	inState	State[]	ObjectNode	<code>\$ValuePin[i].inState</code>
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	<code>\$ValuePin[i].inStructuredNode</code>
Incoming	incoming	ActivityEdge[]	ActivityNode	<code>\$ValuePin[i].incoming</code>
isInvalid	invalid	boolean	ModelObject	<code>\$ValuePin[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$ValuePin[i].leaf</code>
localID	localID	String	ModelObject	<code>\$ValuePin[i].localID</code>
Lower	lower	int	MultiplicityElement	<code>\$ValuePin[i].lower</code>
Lower Value	lowerValue	ValueSpecification	MultiplicityElement	<code>\$ValuePin[i].lowerValue</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$ValuePin[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$ValuePin[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$ValuePin[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$ValuePin[i].namespace</code>

objectParent	objectParent	BaseElement	BaseElement	\$ValuePin[i].objectParent
Is Ordered	ordered	boolean	MultiplicityElement	\$ValuePin[i].ordered
Ordering	ordering	ObjectNodeOrderingKind	ObjectNode	\$ValuePin[i].ordering
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ValuePin[i].outgoing
Owned Comment	ownedComment	Comment[]	Element	\$ValuePin[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ValuePin[i].ownedElement
Owner	owner	Element	Element	\$ValuePin[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ValuePin[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ValuePin[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ValuePin[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ValuePin[i].redefinitionContext
Selection	selection	Behavior	ObjectNode	\$ValuePin[i].selection
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$ValuePin[i].supplierDependency
Sync Element	syncElement	Element	Element	\$ValuePin[i].syncElement
Type	type	Type	TypedElement	\$ValuePin[i].type
Is Unique	unique	boolean	MultiplicityElement	\$ValuePin[i].unique
Upper	upper	int	MultiplicityElement	\$ValuePin[i].upper
Upper Bound	upperBound	ValueSpecification	ObjectNode	\$ValuePin[i].upperBound
Upper Value	upperValue	ValueSpecification	MultiplicityElement	\$ValuePin[i].upperValue
Value	value	ValueSpecification	ValuePin	\$ValuePin[i].value
Visibility	visibility	VisibilityKind	NamedElement	\$ValuePin[i].visibility

239. ValueSpecification

Generalization

- [PackageableElement](#)
- [TypedElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$ValueSpecification[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$ValueSpecification[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$ValueSpecification[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$ValueSpecification[i].clientDependency
isEditable	editable	boolean	BaseElement	\$ValueSpecification[i].editable
Expression	expression	Expression	ValueSpecification	\$ValueSpecification[i].expression
humanName	humanName	String	BaseElement	\$ValueSpecification[i].humanName
humanType	humanType	String	BaseElement	\$ValueSpecification[i].humanType

isInvalid	invalid	boolean	ModelObject	<code>\$ValueSpecification[i].invalid</code>
localID	localID	String	ModelObject	<code>\$ValueSpecification[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$ValueSpecification[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$ValueSpecification[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$ValueSpecification[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$ValueSpecification[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$ValueSpecification[i].objectParent</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$ValueSpecification[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$ValueSpecification[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$ValueSpecification[i].owner</code>
Owning Constraint	owningConstraint	Constraint	ValueSpecification	<code>\$ValueSpecification[i].owningConstraint</code>
Owning Instance Spec	owningInstanceSpec	InstanceSpecification	ValueSpecification	<code>\$ValueSpecification[i].owningInstanceSpec</code>
Owning Lower	owningLower	MultiplicityElement	ValueSpecification	<code>\$ValueSpecification[i].owningLower</code>
Owning Package	owningPackage	Package	PackageableElement	<code>\$ValueSpecification[i].owningPackage</code>
Owning Parameter	owningParameter	Parameter	ValueSpecification	<code>\$ValueSpecification[i].owningParameter</code>
Owning Property	owningProperty	Property	ValueSpecification	<code>\$ValueSpecification[i].owningProperty</code>
Owning Slot	owningSlot	Slot	ValueSpecification	<code>\$ValueSpecification[i].owningSlot</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$ValueSpecification[i].owningTemplateParameter</code>
Owning Upper	owningUpper	MultiplicityElement	ValueSpecification	<code>\$ValueSpecification[i].owningUpper</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$ValueSpecification[i].qualifiedName</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$ValueSpecification[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$ValueSpecification[i].syncElement</code>
Template Parameter	templateParameter	TemplateParameter	ParameterableElement	<code>\$ValueSpecification[i].templateParameter</code>
Type	type	Type	TypedElement	<code>\$ValueSpecification[i].type</code>
Visibility	visibility	VisibilityKind	PackageableElement	<code>\$ValueSpecification[i].visibility</code>

240. ValueSpecificationAction

Generalization

- [Action](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$ValueSpecificationAction[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$ValueSpecificationAction[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$ValueSpecificationAction[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$ValueSpecificationAction[i].classType</code>

Client Dependency	clientDependency	Dependency[]	NamedElement	\$ValueSpecificationAction[i].clientDependency
Context	context	Classifier	Action	\$ValueSpecificationAction[i].context
isEditable	editable	boolean	BaseElement	\$ValueSpecificationAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$ValueSpecificationAction[i].handler
humanName	humanName	String	BaseElement	\$ValueSpecificationAction[i].humanName
humanType	humanType	String	BaseElement	\$ValueSpecificationAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$ValueSpecificationAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$ValueSpecificationAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$ValueSpecificationAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$ValueSpecificationAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$ValueSpecificationAction[i].incoming
Input	input	InputPin[]	Action	\$ValueSpecificationAction[i].input
isInvalid	invalid	boolean	ModelObject	\$ValueSpecificationAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$ValueSpecificationAction[i].leaf
localID	localID	String	ModelObject	\$ValueSpecificationAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$ValueSpecificationAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$ValueSpecificationAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$ValueSpecificationAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$ValueSpecificationAction[i].mdExtensions
Name	name	String	NamedElement	\$ValueSpecificationAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$ValueSpecificationAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$ValueSpecificationAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$ValueSpecificationAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$ValueSpecificationAction[i].outgoing
Output	output	OutputPin[]	Action	\$ValueSpecificationAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$ValueSpecificationAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$ValueSpecificationAction[i].ownedElement
Owner	owner	Element	Element	\$ValueSpecificationAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$ValueSpecificationAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$ValueSpecificationAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$ValueSpecificationAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$ValueSpecificationAction[i].redefinitionContext

				text
Result	result	OutputPin	ValueSpecificationAction	<code>\$ValueSpecificationAction[i].result</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$ValueSpecificationAction[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$ValueSpecificationAction[i].syncElement</code>
Value	value	ValueSpecification	ValueSpecificationAction	<code>\$ValueSpecificationAction[i].value</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$ValueSpecificationAction[i].visibility</code>

241. Variable

Generalization

- [ConnectableElement](#)
- [MultiplicityElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	id	String	MDOObject	<code>\$Variable[i].id</code>
Activity Scope	activityScope	Activity	Variable	<code>\$Variable[i].activityScope</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$Variable[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$Variable[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$Variable[i].clientDependency</code>
isEditable	editable	boolean	BaseElement	<code>\$Variable[i].editable</code>
End	end	ConnectorEnd[]	ConnectableElement	<code>\$Variable[i].end</code>
humanName	humanName	String	BaseElement	<code>\$Variable[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$Variable[i].humanType</code>
isInvalid	invalid	boolean	ModelObject	<code>\$Variable[i].invalid</code>
localID	localID	String	ModelObject	<code>\$Variable[i].localID</code>
Lower	lower	int	MultiplicityElement	<code>\$Variable[i].lower</code>
Lower Value	lowerValue	ValueSpecification	MultiplicityElement	<code>\$Variable[i].lowerValue</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$Variable[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$Variable[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Variable[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Variable[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Variable[i].objectParent</code>
Is Ordered	ordered	boolean	MultiplicityElement	<code>\$Variable[i].ordered</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Variable[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Variable[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$Variable[i].owner</code>
Owning Template Parameter	owningTemplateParameter	TemplateParameter	ParameterableElement	<code>\$Variable[i].owningTemplateParameter</code>

Qualified Name	qualifiedName	String	NamedElement	\$Variable[i].qualifiedName
Scope	scope	StructuredActivityNode	Variable	\$Variable[i].scope
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$Variable[i].supplierDependency
Sync Element	syncElement	Element	Element	\$Variable[i].syncElement
Template Parameter	templateParameter	ConnectableElementTemplateParameter	ConnectableElement	\$Variable[i].templateParameter
Type	type	Type	TypedElement	\$Variable[i].type
Is Unique	unique	boolean	MultiplicityElement	\$Variable[i].unique
Upper	upper	int	MultiplicityElement	\$Variable[i].upper
Upper Value	upperValue	ValueSpecification	MultiplicityElement	\$Variable[i].upperValue
Visibility	visibility	VisibilityKind	NamedElement	\$Variable[i].visibility

242. VariableAction

Generalization

- Action

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$VariableAction[i].ID
Activity	activity	Activity	ActivityNode	\$VariableAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$VariableAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$VariableAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$VariableAction[i].clientDependency
Context	context	Classifier	Action	\$VariableAction[i].context
isEditable	editable	boolean	BaseElement	\$VariableAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$VariableAction[i].handler
humanName	humanName	String	BaseElement	\$VariableAction[i].humanName
humanType	humanType	String	BaseElement	\$VariableAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$VariableAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$VariableAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$VariableAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$VariableAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$VariableAction[i].incoming
Input	input	InputPin[]	Action	\$VariableAction[i].input
isInvalid	invalid	boolean	ModelObject	\$VariableAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$VariableAction[i].leaf

localID	localID	String	ModelObject	\$VariableAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$VariableAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$VariableAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$VariableAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOObject	\$VariableAction[i].mdExtensions
Name	name	String	NamedElement	\$VariableAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$VariableAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$VariableAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$VariableAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$VariableAction[i].outgoing
Output	output	OutputPin[]	Action	\$VariableAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$VariableAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$VariableAction[i].ownedElement
Owner	owner	Element	Element	\$VariableAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$VariableAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$VariableAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$VariableAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$VariableAction[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$VariableAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$VariableAction[i].syncElement
Variable	variable	Variable	VariableAction	\$VariableAction[i].variable
Visibility	visibility	VisibilityKind	NamedElement	\$VariableAction[i].visibility

243. Vertex

Generalization

- [RedefinableElement](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	\$Vertex[i].ID
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$Vertex[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$Vertex[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$Vertex[i].clientDependency
Container	container	Region	Vertex	\$Vertex[i].container
isEditable	editable	boolean	BaseElement	\$Vertex[i].editable
humanName	humanName	String	BaseElement	\$Vertex[i].humanName
humanType	humanType	String	BaseElement	\$Vertex[i].humanType
Incoming	incoming	Transition[]	Vertex	\$Vertex[i].incoming

isInvalid	invalid	boolean	ModelObject	<code>\$Vertex[i].invalid</code>
Is Leaf	leaf	boolean	RedefinableElement	<code>\$Vertex[i].leaf</code>
localID	localID	String	ModelObject	<code>\$Vertex[i].localID</code>
mdExtensions	mdExtensions	MDExtension[]	MDOObject	<code>\$Vertex[i].mdExtensions</code>
Name	name	String	NamedElement	<code>\$Vertex[i].name</code>
Name Expression	nameExpression	StringExpression	NamedElement	<code>\$Vertex[i].nameExpression</code>
Namespace	namespace	Namespace	NamedElement	<code>\$Vertex[i].namespace</code>
objectParent	objectParent	BaseElement	BaseElement	<code>\$Vertex[i].objectParent</code>
Outgoing	outgoing	Transition[]	Vertex	<code>\$Vertex[i].outgoing</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$Vertex[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$Vertex[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$Vertex[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$Vertex[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$Vertex[i].redefinedElement</code>
Redefined Vertex	redefinedVertex	Vertex	Vertex	<code>\$Vertex[i].redefinedVertex</code>
Redefinition Context	redefinitionContext	Classifier[]	Vertex	<code>\$Vertex[i].redefinitionContext</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$Vertex[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$Vertex[i].syncElement</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$Vertex[i].visibility</code>

244. WriteLinkAction

Generalization

- [LinkAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOObject	<code>\$WriteLinkAction[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$WriteLinkAction[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$WriteLinkAction[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$WriteLinkAction[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$WriteLinkAction[i].clientDependency</code>
Context	context	Classifier	Action	<code>\$WriteLinkAction[i].context</code>
isEditable	editable	boolean	BaseElement	<code>\$WriteLinkAction[i].editable</code>
End Data	endData	LinkEndData[]	LinkAction	<code>\$WriteLinkAction[i].endData</code>
Handler	handler	ExceptionHandler[]	ExecutableNode	<code>\$WriteLinkAction[i].handler</code>
humanName	humanName	String	BaseElement	<code>\$WriteLinkAction[i].humanName</code>
humanType	humanType	String	BaseElement	<code>\$WriteLinkAction[i].humanType</code>

In Group	inGroup	ActivityGroup[]	ActivityNode	\$WriteLinkAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$WriteLinkAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$WriteLinkAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$WriteLinkAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$WriteLinkAction[i].incoming
Input	input	InputPin[]	Action	\$WriteLinkAction[i].input
Input Value	inputValue	InputPin[]	LinkAction	\$WriteLinkAction[i].inputValue
isInvalid	invalid	boolean	ModelObject	\$WriteLinkAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$WriteLinkAction[i].leaf
localID	localID	String	ModelObject	\$WriteLinkAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$WriteLinkAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$WriteLinkAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$WriteLinkAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$WriteLinkAction[i].mdExtensions
Name	name	String	NamedElement	\$WriteLinkAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$WriteLinkAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$WriteLinkAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$WriteLinkAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$WriteLinkAction[i].outgoing
Output	output	OutputPin[]	Action	\$WriteLinkAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$WriteLinkAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$WriteLinkAction[i].ownedElement
Owner	owner	Element	Element	\$WriteLinkAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$WriteLinkAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$WriteLinkAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$WriteLinkAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$WriteLinkAction[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$WriteLinkAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$WriteLinkAction[i].syncElement
Visibility	visibility	VisibilityKind	NamedElement	\$WriteLinkAction[i].visibility

245. WriteStructuralFeatureAction

Generalization

- [StructuralFeatureAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)

ID	ID	String	MDOBJECT	\$WriteStructuralFeatureAction[i].ID
Activity	activity	Activity	ActivityNode	\$WriteStructuralFeatureAction[i].activity
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	\$WriteStructuralFeatureAction[i].appliedStereotypeInstance
classType	classType	Class	BaseElement	\$WriteStructuralFeatureAction[i].classType
Client Dependency	clientDependency	Dependency[]	NamedElement	\$WriteStructuralFeatureAction[i].clientDependency
Context	context	Classifier	Action	\$WriteStructuralFeatureAction[i].context
isEditable	editable	boolean	BaseElement	\$WriteStructuralFeatureAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$WriteStructuralFeatureAction[i].handler
humanName	humanName	String	BaseElement	\$WriteStructuralFeatureAction[i].humanName
humanType	humanType	String	BaseElement	\$WriteStructuralFeatureAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$WriteStructuralFeatureAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$WriteStructuralFeatureAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$WriteStructuralFeatureAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$WriteStructuralFeatureAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$WriteStructuralFeatureAction[i].incoming
Input	input	InputPin[]	Action	\$WriteStructuralFeatureAction[i].input
isInvalid	invalid	boolean	ModelObject	\$WriteStructuralFeatureAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$WriteStructuralFeatureAction[i].leaf
localID	localID	String	ModelObject	\$WriteStructuralFeatureAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$WriteStructuralFeatureAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$WriteStructuralFeatureAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$WriteStructuralFeatureAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDOBJECT	\$WriteStructuralFeatureAction[i].mdExtensions
Name	name	String	NamedElement	\$WriteStructuralFeatureAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$WriteStructuralFeatureAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$WriteStructuralFeatureAction[i].namespace
Object	object	InputPin	StructuralFeatureAction	\$WriteStructuralFeatureAction[i].object
objectParent	objectParent	BaseElement	BaseElement	\$WriteStructuralFeatureAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$WriteStructuralFeatureAction[i].outgoing

Output	output	OutputPin[]	Action	<code>\$WriteStructuralFeatureAction[i].output</code>
Owned Comment	ownedComment	Comment[]	Element	<code>\$WriteStructuralFeatureAction[i].ownedComment</code>
Owned Element	ownedElement	Element[]	Element	<code>\$WriteStructuralFeatureAction[i].ownedElement</code>
Owner	owner	Element	Element	<code>\$WriteStructuralFeatureAction[i].owner</code>
Qualified Name	qualifiedName	String	NamedElement	<code>\$WriteStructuralFeatureAction[i].qualifiedName</code>
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	<code>\$WriteStructuralFeatureAction[i].redefinedElement</code>
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	<code>\$WriteStructuralFeatureAction[i].redefinedNode</code>
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	<code>\$WriteStructuralFeatureAction[i].redefinitionContext</code>
Result	result	OutputPin	WriteStructuralFeatureAction	<code>\$WriteStructuralFeatureAction[i].result</code>
Structural Feature	structuralFeature	StructuralFeature	StructuralFeatureAction	<code>\$WriteStructuralFeatureAction[i].structuralFeature</code>
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	<code>\$WriteStructuralFeatureAction[i].supplierDependency</code>
Sync Element	syncElement	Element	Element	<code>\$WriteStructuralFeatureAction[i].syncElement</code>
Value	value	InputPin	WriteStructuralFeatureAction	<code>\$WriteStructuralFeatureAction[i].value</code>
Visibility	visibility	VisibilityKind	NamedElement	<code>\$WriteStructuralFeatureAction[i].visibility</code>

246. WriteVariableAction

Generalization

- [VariableAction](#)

Attribute Label	Attribute Name	Attribute Type (UML Class)	Declaring Class (Class Owner)	Sample Template Expression (VTL)
ID	ID	String	MDOBJECT	<code>\$WriteVariableAction[i].ID</code>
Activity	activity	Activity	ActivityNode	<code>\$WriteVariableAction[i].activity</code>
Applied Stereotype Instance	appliedStereotypeInstance	InstanceSpecification	Element	<code>\$WriteVariableAction[i].appliedStereotypeInstance</code>
classType	classType	Class	BaseElement	<code>\$WriteVariableAction[i].classType</code>
Client Dependency	clientDependency	Dependency[]	NamedElement	<code>\$WriteVariableAction[i].clientDependency</code>
Context	context	Classifier	Action	<code>\$WriteVariableAction[i].context</code>

isEditable	editable	boolean	BaseElement	\$WriteVariableAction[i].editable
Handler	handler	ExceptionHandler[]	ExecutableNode	\$WriteVariableAction[i].handler
humanName	humanName	String	BaseElement	\$WriteVariableAction[i].humanName
humanType	humanType	String	BaseElement	\$WriteVariableAction[i].humanType
In Group	inGroup	ActivityGroup[]	ActivityNode	\$WriteVariableAction[i].inGroup
In Interruptible Region	inInterruptibleRegion	InterruptibleActivityRegion[]	ActivityNode	\$WriteVariableAction[i].inInterruptibleRegion
In Partition	inPartition	ActivityPartition[]	ActivityNode	\$WriteVariableAction[i].inPartition
In Structured Node	inStructuredNode	StructuredActivityNode	ActivityNode	\$WriteVariableAction[i].inStructuredNode
Incoming	incoming	ActivityEdge[]	ActivityNode	\$WriteVariableAction[i].incoming
Input	input	InputPin[]	Action	\$WriteVariableAction[i].input
isInvalid	invalid	boolean	ModelObject	\$WriteVariableAction[i].invalid
Is Leaf	leaf	boolean	RedefinableElement	\$WriteVariableAction[i].leaf
localID	localID	String	ModelObject	\$WriteVariableAction[i].localID
Local Postcondition	localPostcondition	Constraint[]	Action	\$WriteVariableAction[i].localPostcondition
Local Precondition	localPrecondition	Constraint[]	Action	\$WriteVariableAction[i].localPrecondition
Is Locally Reentrant	locallyReentrant	boolean	Action	\$WriteVariableAction[i].locallyReentrant
mdExtensions	mdExtensions	MDExtension[]	MDObject	\$WriteVariableAction[i].mdExtensions
Name	name	String	NamedElement	\$WriteVariableAction[i].name
Name Expression	nameExpression	StringExpression	NamedElement	\$WriteVariableAction[i].nameExpression
Namespace	namespace	Namespace	NamedElement	\$WriteVariableAction[i].namespace
objectParent	objectParent	BaseElement	BaseElement	\$WriteVariableAction[i].objectParent
Outgoing	outgoing	ActivityEdge[]	ActivityNode	\$WriteVariableAction[i].outgoing
Output	output	OutputPin[]	Action	\$WriteVariableAction[i].output
Owned Comment	ownedComment	Comment[]	Element	\$WriteVariableAction[i].ownedComment
Owned Element	ownedElement	Element[]	Element	\$WriteVariableAction[i].ownedElement
Owner	owner	Element	Element	\$WriteVariableAction[i].owner
Qualified Name	qualifiedName	String	NamedElement	\$WriteVariableAction[i].qualifiedName
Redefined Element	redefinedElement	RedefinableElement[]	RedefinableElement	\$WriteVariableAction[i].redefinedElement
Redefined Node	redefinedNode	ActivityNode[]	ActivityNode	\$WriteVariableAction[i].redefinedNode
Redefinition Context	redefinitionContext	Classifier[]	RedefinableElement	\$WriteVariableAction[i].redefinitionContext
Supplier Dependency	supplierDependency	Dependency[]	NamedElement	\$WriteVariableAction[i].supplierDependency
Sync Element	syncElement	Element	Element	\$WriteVariableAction[i].syncElement
Value	value	InputPin	WriteVariableAction	\$WriteVariableAction[i].value
Variable	variable	Variable	VariableAction	\$WriteVariableAction[i].variable
Visibility	visibility	VisibilityKind	NamedElement	\$WriteVariableAction[i].visibility