



Pedigree and Provenance Model and Notation (PPMN)

Version 1.0 Beta 2

OMG Document Number: dtc/2024-09-07

Normative reference: <https://www.omg.org/spec/PPMN/Beta2>

Machine readable file(s): <https://www.omg.org/PPMN/20240801>

Normative:

Informative:

17 Copyright © 2021-2024 agnos.ai UK Limited
18 Copyright © 2021-2024 Auxilium Technology Group, LLC
19 Copyright © 2021-2024 BPM Advantage Consulting, Inc.
20 Copyright © 2021-2024 cébé IT & Knowledge Management LLC
21 Copyright © 2021-2024 Thematrix Partners LLC
22 Copyright © 2021-2024 Xzyos, LLC
23 Copyright © 2022-2024 Capacity Post, Inc.

24

25

26

27

USE OF SPECIFICATION - TERMS, CONDITIONS & NOTICES

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

32

33

LICENSES

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

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

49

50

PATENTS

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

56

57

GENERAL USE RESTRICTIONS

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

63

64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110

DISCLAIMER OF WARRANTY

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

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

RESTRICTED RIGHTS LEGEND

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

TRADEMARKS

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

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

COMPLIANCE

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

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

111
112
113
114
115

OMG's Issue Reporting Procedure

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

Table of Contents

116

117	1	Scope	22
118	2	Conformance	22
119	2.1	General.....	22
120	2.2	PPMN Modeling Conformance	22
121	2.3	Visual Conformance	22
122	3	References	23
123	3.1	Normative References.....	23
124	3.2	Non-normative References	23
125	4	Terms and Definitions	24
126	5	Symbols.....	25
127	6	Additional Information.....	25
128	6.1	Conventions	25
129	6.2	Typographical and Linguistic Conventions and Style	25
130	6.3	Display of Metamodel Diagrams	26
131	6.4	Use of Text, Color, Size, and Lines in a Diagram	27
132	6.5	Abbreviations.....	27
133	6.6	Structure of this Document.....	28
134	6.7	Acknowledgements.....	28
135	7	Overview	28
136	8	Pedigree and Provenance Model and Notation.....	30
137	8.1	Entities	31
138	8.1.1	Entity.....	33
139	8.1.2	EntityFormat	33
140	8.1.3	EntityRelationship.....	34
141	8.1.4	EntityRelationshipType.....	34
142	8.1.5	EntitySnapshot	35
143	8.1.6	EntitySnapshotType	35
144	8.1.7	EntityType.....	36
145	8.2	Occurrences	36
146	8.2.1	ActivityOccurrence	41
147	8.2.2	ActivityOccurrenceType	43
148	8.2.3	InterestedParty.....	43
149	8.2.4	Occurrence	43
150	8.2.5	OccurrenceBranchNode	44
151	8.2.6	OccurrenceChain.....	44

152	8.2.7	OccurrenceChainType.....	45
153	8.2.8	OccurrenceDependency	46
154	8.2.9	OccurrenceDependencyKind	47
155	8.2.10	OccurrenceDependencyType	47
156	8.2.11	OccurrenceGraphNode.....	48
157	8.2.12	OccurrenceGraphTransition	48
158	8.2.13	OccurrenceKind	49
159	8.2.14	OccurrenceRelationship	49
160	8.2.15	OccurrenceRole.....	50
161	8.2.16	OccurrenceRoleType.....	51
162	8.2.17	OccurrenceType	52
163	8.2.18	OccurrenceTypeGraph	52
164	8.2.19	OccurrenceTypeUsage Node.....	53
165	8.2.20	PPMNRelationshipKind.....	53
166	8.2.21	Rule	53
167	8.3	Pedigree	54
168	8.3.1	Pedigree Occurrences.....	54
	8.3.1.1	EntityPedigree	58
	8.3.1.2	EntityPedigreeType	59
	8.3.1.3	PedigreeKind	59
	8.3.1.4	PedigreeOccurrenceChain.....	59
	8.3.1.5	PedigreeOccurrence.....	60
	8.3.1.6	PedigreeOccurrenceChainType	60
	8.3.1.7	PedigreeOccurrenceType.....	61
	8.3.1.8	PedigreeTypeGraph.....	61
169	8.3.2	Derivations	62
	8.3.2.1	DerivationKind.....	63
	8.3.2.2	DerivationType.....	63
	8.3.2.3	DerivedFrom.....	64
	8.3.2.4	DescendantOf	64
	8.3.2.5	QuotedFrom.....	65
	8.3.2.6	RevisionOf.....	65
	8.3.2.7	SourcedFrom	66
170	8.4	Provenance.....	66
171	8.4.1	ChainOfProvenance	69
172	8.4.2	ChainOfProvenanceType	69
173	8.4.3	ProvenanceChangeKind	70

174	8.4.4	ProvenanceChangeOccurrence.....	70
175	8.4.5	ProvenanceChangeType.....	71
176	8.4.6	ProvenanceOccurrenceChain	71
177	8.4.7	ProvenanceOccurrenceChainType	72
178	8.4.8	ProvenanceTypeGraph	72
179	8.4.9	ResponsibilityRelationship.....	73
180	8.4.10	ResponsibilityRelationshipKind.....	73
181	8.4.11	ResponsibilityRelationshipType	74
182	8.4.12	Custody	74
	8.4.12.1	ChainOfCustody	77
	8.4.12.2	ChainOfCustodyType.....	77
	8.4.12.3	Custody.....	78
	8.4.12.4	CustodyChangeKind.....	78
	8.4.12.5	CustodyChangeOccurrence	78
	8.4.12.6	CustodyChangeType	79
	8.4.12.7	CustodyEndKind	80
	8.4.12.8	CustodyKind.....	80
	8.4.12.9	CustodyOccurrenceChain.....	80
	8.4.12.10	CustodyOccurrenceChainType	80
	8.4.12.11	CustodyStartKind.....	81
	8.4.12.12	CustodyTransferKind.....	81
	8.4.12.13	CustodyType	81
	8.4.12.14	CustodyTypeGraph	82
183	8.4.13	Ownership	82
	8.4.13.1	AcquisitionKind	85
	8.4.13.2	ChainOfOwnership.....	85
	8.4.13.3	ChainOfOwnershipType.....	86
	8.4.13.4	Ownership.....	86
	8.4.13.5	OwnershipChangeOccurrence	86
	8.4.13.6	OwnershipEndKind	87
	8.4.13.7	OwnershipKind.....	87
	8.4.13.8	OwnershipOccurrenceChain.....	87
	8.4.13.9	OwnershipOccurrenceChainType.....	88
	8.4.13.10	OwnershipOccurrenceKind.....	88
	8.4.13.11	OwnershipOccurrenceType	89
	8.4.13.12	OwnershipTransferKind	89
	8.4.13.13	OwnershipType.....	89

	8.4.13.14	OwnershipTypeGraph.....	90
184	8.5	Claims.....	90
185	8.5.1	ClaimPositivity.....	92
186	8.5.2	ClaimAssessment.....	92
187	8.5.3	ClaimKind.....	93
188	8.5.4	OccurrenceClaim.....	93
189	8.6	Rationale.....	94
190	8.6.1	Rationale.....	94
191	8.6.2	RationaleType.....	95
192	8.7	Annotations.....	95
	8.7.1.1	Annotation.....	96
	8.7.1.2	AnnotationAssignment.....	97
	8.7.1.3	AnnotationTemplate.....	97
	8.7.1.4	ChronicledAnnotation.....	97
	8.7.1.5	SimpleAnnotation.....	98
193	8.8	Delegation.....	98
194	8.8.1	ActedOnBehalfOf.....	99
195	8.8.2	DelegationAssignment.....	100
196	8.9	Additional Relationships.....	100
197	8.9.1	AlternateOf.....	101
198	8.9.2	AssociatedWith.....	101
199	8.9.3	AttributedTo.....	102
200	8.9.4	Informed.....	102
201	8.10	Packaging.....	102
202	8.10.1	PPMNDefinitions.....	103
203	8.10.2	PPMNInstances.....	104
204	8.10.3	PPMNModel.....	105
205	8.11	Primitives.....	106
206	8.11.1	DateTime.....	106
207	8.12	KindSets.....	106
208	8.12.1	PPMNKindSet.....	106
209	8.12.2	AcquisitionKindSet.....	107
210	8.12.3	ClaimKindSet.....	107
211	8.12.4	CustodyEndKindSet.....	108
212	8.12.5	CustodyStartKindSet.....	108
213	8.12.6	DerivationKindSet.....	108
214	8.12.7	OccurrenceDependencyKindSet.....	109

215	8.12.8	OwnershipEndKindSet.....	109
216	8.12.9	PedigreeEndKindSet	110
217	8.12.10	PPMNRelationshipKindSet.....	110
218	8.12.11	ResponsibilityRelationshipKindSet.....	110
219	9	PPMN Library	111
220	9.1	AcquisitionKinds	111
221	9.1.1	AcquisitionKindSet	113
222	9.1.2	Copied.....	113
223	9.1.3	Created	113
224	9.1.4	Gifted	113
225	9.1.5	Inherited	113
226	9.1.6	Purchased	113
227	9.2	ClaimKinds.....	113
228	9.2.1	ClaimKindSet.....	115
229	9.2.2	Fact.....	115
230	9.2.3	First Principle	115
231	9.2.4	Logical Argument	115
232	9.2.5	Postcondition.....	115
233	9.2.6	Precondition	115
234	9.2.7	Premise.....	115
235	9.2.8	Probability	115
236	9.3	CustodyEndKinds.....	115
237	9.3.1	CustodyEndKindSet.....	116
238	9.3.2	Delivered.....	116
239	9.3.3	Destroyed	117
240	9.3.4	Lost	117
241	9.3.5	Other	117
242	9.3.6	Transferred	117
243	9.4	CustodyStartKinds.....	117
244	9.4.1	CustodyStartKindSet.....	118
245	9.4.2	Acquisition	118
246	9.4.3	Created	118
247	9.4.4	Found	118
248	9.4.5	Other	118
249	9.5	DerivationKinds.....	118
250	9.5.1	DerivationKindSet.....	119
251	9.5.2	DerivedFrom	119

252	9.5.3	DescendantOf.....	119
253	9.5.4	QuotedFrom	120
254	9.5.5	RevisionOf	120
255	9.5.6	SourcedFrom.....	120
256	9.6	OccurrenceDependencyKinds	120
257	9.6.1	OccurrenceDependencyKindSet	121
258	9.6.2	By-product	121
259	9.6.3	Enabler	121
260	9.6.4	Input	121
261	9.6.5	Output	121
262	9.6.6	Product	122
263	9.6.7	Waste.....	122
264	9.7	OwnershipEndKinds.....	122
265	9.7.1	OwnershipEndKindSet.....	123
266	9.7.2	Bequeathed.....	123
267	9.7.3	Death	123
268	9.7.4	Gifted	123
269	9.7.5	Lost	123
270	9.7.6	Sold	123
271	9.7.7	Transferred.....	123
272	9.8	PPMNRelationshipKinds.....	123
273	9.8.1	PPMNRelationshipKinds	125
274	9.8.2	Transition	125
275	9.8.3	Additional Terms from SCE	125
	9.8.3.1	Reference	125
	9.8.3.2	Miscellaneous	125
	9.8.3.3	Composition	125
	9.8.3.4	Dependency	125
	9.8.3.5	Containment	126
	9.8.3.6	Correlation.....	126
	9.8.3.7	Generalization.....	126
276	9.9	ResponsibilityRelationshipKinds.....	126
277	9.9.1	ResponsibilityRelationshipKinds	127
278	9.9.2	Custody	127
279	9.9.3	Ownership	127
280	10	Parties Model.....	127
281	10.1	Core	127

282	10.1.1	Instances.....	127
	10.1.1.1	Delegation.....	130
	10.1.1.2	NonHumanAgent.....	130
	10.1.1.3	Organization	131
	10.1.1.4	OrganizationStructureRelationship.....	131
	10.1.1.5	Party132	
	10.1.1.6	PartyRelationship.....	133
	10.1.1.7	PartyRole	133
	10.1.1.8	Person	134
	10.1.1.9	Position.....	134
	10.1.1.10	PositionAssignment	135
283	10.1.2	Types.....	135
	10.1.2.1	DelegationType	137
	10.1.2.2	IndividualKind.....	138
	10.1.2.3	IndividualType	138
	10.1.2.4	NonHumanKind	138
	10.1.2.5	OrganizationType	139
	10.1.2.6	PartyRelationshipKind.....	139
	10.1.2.7	PartyRelationshipType	139
	10.1.2.8	PartyRoleType.....	140
	10.1.2.9	PartyType	140
	10.1.2.10	PositionAssignmentType	140
	10.1.2.11	PositionType	141
284	10.2	Locations	141
285	10.2.1	Instances.....	141
	10.2.1.1	Area 142	
	10.2.1.2	GeospatialExtent.....	142
	10.2.1.3	Location.....	143
	10.2.1.4	NetworkAddress.....	143
	10.2.1.5	Path 143	
	10.2.1.6	PhysicalAddress	144
	10.2.1.7	SpaceTime	144
286	10.2.2	Types.....	145
	10.2.2.1	AreaType	145
	10.2.2.2	LocationType.....	145
	10.2.2.3	NetworkAddressType.....	145
	10.2.2.4	PathType.....	145

	10.2.2.5	PointType	146
	10.2.2.6	SpaceTimeType	146
	10.2.2.7	VolumeType	146
287	10.3	Packages	146
288	10.3.1	Package	147
289	10.3.2	PartyDefinitions	147
290	10.3.3	PartyInstances	148
291	10.3.4	PartyModel.....	149
292	10.4	Primitives.....	149
293	10.4.1	DateTime.....	149
294	10.5	PartyKindSets	150
295	10.5.1	PartyKindSet	150
296	10.5.2	IndividualKindSet	150
297	10.5.3	PartyRelationshipKindSet	151
298	11	Parties Library	151
299	11.1	IndividualKinds	151
300	11.1.1	IndividualKinds.....	152
301	11.1.2	Machinery	152
302	11.1.3	NonHumanAgent	153
303	11.1.4	Person.....	153
304	11.1.5	Software	153
305	11.2	PartyRelationshipKinds	153
306	11.2.1	PartyRelationshipKinds.....	155
307	11.2.2	Delegation	155
308	11.2.3	Employment.....	155
309	11.2.4	General.....	155
310	11.2.5	Member	155
311	11.2.6	Part	155
312	11.2.7	PositionAssignment.....	155
313	12	PPMN and Parties Diagram Interchange (PPMN DI and Parties DI)	155
314	12.1	Scope	155
315	12.2	Diagram Definition and Interchange	156
316	12.3	Notation	156
317	12.3.1	Labels.....	156
318	12.3.2	Shape Resolution.....	157
	12.3.2.1	Depiction for PPMN Diagram Elements	157
	12.3.2.2	Depiction for Parties Diagram Elements	159

319 12.3.3 Edge Resolution 161
12.3.3.1 Depiction for PPMN Diagram Elements 161
12.3.3.2 Depiction for Parties Diagram Elements 163

320

321

Annexes

322 Annex A: PROV Traceability 165

323

324

Table of Figures

325

326	Figure 1:	PPMN Packaging Overview.....	30
327	Figure 2:	Pedigree and Provenance Packaging.....	31
328	Figure 3:	Entities and EntityTypes	32
329	Figure 4:	Entity Relationships	32
330	Figure 5:	Occurrences - Simplified.....	37
331	Figure 6:	Occurrences.....	37
332	Figure 7:	Occurrence Chains	38
333	Figure 8:	Occurrence Types.....	38
334	Figure 9:	Occurrence Type Graphs.....	39
335	Figure 10:	Occurrences Type Pattern	40
336	Figure 11:	Activity Occurrences.....	41
337	Figure 12:	Activity Occurrence	42
338	Figure 13:	OccurrencesDependencies	46
339	Figure 14:	Occurrence Dependency Types.....	47
340	Figure 15:	OccurrencesRoles.....	50
341	Figure 16:	Occurrence Role Types	51
342	Figure 17:	Pedigree Occurrence Chains - Overview	54
343	Figure 18:	Pedigree Occurrences.....	55
344	Figure 19:	Pedigree Occurrence Chains	55
345	Figure 20:	Pedigree Occurrence Chain Type.....	56
346	Figure 21:	Pedigree Occurrence Types.....	56
347	Figure 22:	Pedigree "Chains"	57
348	Figure 23:	Pedigree Chains Types.....	58
349	Figure 24:	Derivations	62
350	Figure 25:	Derivation Types.....	63
351	Figure 26:	Provenance Occurrence Chains.....	67
352	Figure 27:	Provenance Occurrence Chain Types.....	67
353	Figure 28:	Provenance "Records".....	68
354	Figure 29:	Chain of Provenance	68
355	Figure 30:	Provenance Record Types.....	68
356	Figure 31:	Chain of Provenance Types.....	69
357	Figure 32:	Custody Occurrence Chains	75
358	Figure 33:	Custody Occurrence Chain Types.....	75
359	Figure 34:	Custody Occurrence Chain Type Pattern	76
360	Figure 35:	Chain of Custody.....	76

361	Figure 36: Chain of Custody Types.....	77
362	Figure 37: Ownership Occurrence Chains.....	83
363	Figure 38: Ownership Occurrence Chain Type Pattern.....	83
364	Figure 39: Ownership Occurrence Chain Types.....	84
365	Figure 40: Chain of Ownership.....	84
366	Figure 41: Chain of Ownership Types.....	85
367	Figure 42: Claims.....	91
368	Figure 43: Claim Assessments.....	92
369	Figure 44: Rationale.....	94
370	Figure 45: Annotations.....	96
371	Figure 46: Delegation.....	99
372	Figure 47: Additional PPMN Relationships.....	101
373	Figure 48: PPMN Packaging.....	103
374	Figure 49: PPMN Primitives.....	106
375	Figure 50: PPMN KindSets.....	106
376	Figure 51: AcquisitionKinds.....	112
377	Figure 52: ClaimKinds.....	114
378	Figure 53: CustodyEndKinds.....	116
379	Figure 54: CustodyStartKinds.....	117
380	Figure 55: DerivationKinds.....	119
381	Figure 56: OccurrenceDependencyKinds.....	121
382	Figure 57: OwnershipEndKinds.....	122
383	Figure 58: PPMNRelationshipKinds.....	124
384	Figure 59: ResponsibilityRelationshipKinds.....	126
385	Figure 60: Parties.....	128
386	Figure 61: Party Relationships.....	128
387	Figure 62: Delegation.....	129
388	Figure 63: Party Role.....	129
389	Figure 64: Parties and Party Types.....	130
390	Figure 65: Party Types.....	136
391	Figure 66: Party Role Type.....	136
392	Figure 67: Delegation Types.....	137
393	Figure 68: Locations.....	142
394	Figure 69: Party Packages.....	147
395	Figure 70: Primitives.....	149
396	Figure 71: PartyKindSets.....	150
397	Figure 72: IndividualKinds.....	152

398	Figure 73: PartyRelationshipKinds.....	154
399	Figure 74: PPMN Trace to PROV - Primary PROV Elements.....	165
400	Figure 75: PPMN Trace to PROV - Agents, Responsibility, and Influence.....	166
401	Figure 76: PPMN Trace to PROV - Derivations.....	167
402	Figure 77: PPMN Trace to PROV - Entities and Activities.....	168
403	Figure 78: PPMN Trace to PROV - Influence.....	169
404	Figure 79: PPMN Trace to PROV - PROV Core Structures.....	170
405		
406		

Table of Tables

407

408	Table 1.	Glossary	24
409	Table 2.	PPMN Metamodel Color-Coding.....	26
410	Table 3.	Acronyms.....	27
411	Table 4.	Entity Attributes and/or Associations.....	33
412	Table 5.	EntityFormat Attributes and/or Associations	34
413	Table 6.	EntityRelationship Attributes and/or Associations	34
414	Table 7.	EntityRelationshipType Attributes and/or Associations	35
415	Table 8.	EntitySnapshot Attributes and/or Associations	35
416	Table 9.	EntitySnapshotType Attributes and/or Associations	35
417	Table 10.	EntityType Attributes and/or Associations	36
418	Table 11.	ActivityOccurrence Attributes and/or Associations	42
419	Table 12.	ActivityOccurrenceType Attributes and/or Associations.....	43
420	Table 13.	InterestedParty Attributes and/or Associations	43
421	Table 14.	Occurrence Attributes and/or Associations	44
422	Table 15.	OccurrenceChain Attributes and/or Associations	45
423	Table 16.	OccurrenceChainType Attributes and/or Associations	45
424	Table 17.	OccurrenceDependency Attributes and/or Associations	46
425	Table 18.	OccurrenceDependencyType Attributes and/or Associations	48
426	Table 19.	OccurrenceGraphTransition Attributes and/or Associations.....	49
427	Table 20.	OccurrenceRelationship Attributes and/or Associations.....	49
428	Table 21.	OccurrenceRole Attributes and/or Associations	50
429	Table 22.	OccurrenceRoleType Attributes and/or Associations	51
430	Table 23.	OccurrenceType Attributes and/or Associations.....	52
431	Table 24.	OccurrenceTypeGraph Attributes and/or Associations.....	53
432	Table 25.	OccurrenceTypeUsage Node Attributes and/or Associations	53
433	Table 26.	EntityPedigree Attributes and/or Associations.....	58
434	Table 27.	EntityPedigreeType Attributes and/or Associations	59
435	Table 28.	PedigreeOccurenceChain Attributes and/or Associations.....	59
436	Table 29.	PedigreeOccurrence Attributes and/or Associations	60
437	Table 30.	PedigreeOccurrenceChainType Attributes and/or Associations	61
438	Table 31.	PedigreeOccurrenceType Attributes and/or Associations.....	61
439	Table 32.	PedigreeTypeGraph Attributes and/or Associations	62
440	Table 33.	DerivationType Attributes and/or Associations	64
441	Table 34.	DerivedFrom Attributes and/or Associations.....	64
442	Table 35.	DescendantOf Attributes and/or Associations.....	65

443	Table 36.	QuotedFrom Attributes and/or Associations	65
444	Table 37.	RevisionOf Attributes and/or Associations	66
445	Table 38.	SourcedFrom Attributes and/or Associations.....	66
446	Table 39.	ChainOfProvenance Attributes and/or Associations	69
447	Table 40.	ChainOfProvenanceType Attributes and/or Associations.....	70
448	Table 41.	ProvenanceChangeOccurrence Attributes and/or Associations	70
449	Table 42.	ProvenanceChangeType Attributes and/or Associations	71
450	Table 43.	ProvenanceOccurrenceChain Attributes and/or Associations.....	72
451	Table 44.	ProvenanceOccurrenceChainType Attributes and/or Associations.....	72
452	Table 45.	ResponsibilityRelationship Attributes and/or Associations	73
453	Table 46.	ResponsibilityRelationshipType Attributes and/or Associations	74
454	Table 47.	ChainOfCustody Attributes and/or Associations	77
455	Table 48.	ChainOfCustodyType Attributes and/or Associations	78
456	Table 49.	Custody Attributes and/or Associations	78
457	Table 50.	CustodyChangeOccurrence Attributes and/or Associations.....	79
458	Table 51.	CustodyChangeType Attributes and/or Associations.....	79
459	Table 52.	CustodyOccurrenceChain Attributes and/or Associations	80
460	Table 53.	CustodyOccurrenceChainType Attributes and/or Associations	81
461	Table 54.	CustodyType Attributes and/or Associations.....	82
462	Table 55.	ChainOfOwnership Attributes and/or Associations	85
463	Table 56.	ChainOfOwnershipType Attributes and/or Associations	86
464	Table 57.	Ownership Attributes and/or Associations.....	86
465	Table 58.	OwnershipChangeOccurrence Attributes and/or Associations	87
466	Table 59.	OwnershipOccurrenceChain Attributes and/or Associations	88
467	Table 60.	OwnershipOccurrenceChainType Attributes and/or Associations.....	88
468	Table 61.	OwnershipOccurrenceType Attributes and/or Associations	89
469	Table 62.	OwnershipType Attributes and/or Associations.....	90
470	Table 63.	ClaimPositivity Literals	92
471	Table 64.	ClaimAssessment Attributes and/or Associations.....	93
472	Table 65.	Evidence Attributes and/or Associations.....	93
473	Table 66.	OccurrenceClaim Attributes and/or Associations	94
474	Table 67.	Rationale Attributes and/or Associations	95
475	Table 68.	RationaleType Attributes and/or Associations.....	95
476	Table 69.	Annotation Attributes and/or Associations	96
477	Table 70.	AnnotationAssignment Attributes and/or Associations	97
478	Table 71.	AnnotationTemplate Attributes and/or Associations	97
479	Table 72.	ChronicledAnnotation Attributes and/or Associations.....	98

480	Table 73.	SimpleAnnotation Attributes and/or Associations	98
481	Table 74.	ActedOnBehalfOf Attributes and/or Associations	99
482	Table 75.	DelegationAssignment Attributes and/or Associations	100
483	Table 76.	AttributedTo Attributes and/or Associations.....	101
484	Table 77.	Informed Attributes and/or Associations.....	102
485	Table 78.	PPMNDefinitions Attributes and/or Associations.....	104
486	Table 79.	PPMNInstances Attributes and/or Associations.....	104
487	Table 80.	PPMNModel Attributes and/or Associations	105
488	Table 81.	AcquisitionKindSet Attributes and/or Associations.....	107
489	Table 82.	ClaimKindSet Attributes and/or Associations.....	107
490	Table 83.	CustodyEndKindSet Attributes and/or Associations.....	108
491	Table 84.	CustodyStartKindSet Attributes and/or Associations.....	108
492	Table 85.	DerivationKindSet Attributes and/or Associations	109
493	Table 86.	OccurrenceDependencyKindSet Attributes and/or Associations	109
494	Table 87.	OwnershipEndKindSet Attributes and/or Associations	110
495	Table 88.	PPMNRelationshipKindSet Attributes and/or Associations	110
496	Table 89.	ResponsibilityRelationshipKindSet Attributes and/or Associations	111
497	Table 90.	AcquisitionKinds KindSet.....	112
498	Table 91.	ClaimKinds KindSet	114
499	Table 92.	CustodyEndKinds KindSet	116
500	Table 93.	CustodyStartKinds KindSet	118
501	Table 94.	DerivationKinds KindSet	118
502	Table 95.	OccurrenceDependencyKinds KindSet.....	121
503	Table 96.	OwnershipEndKinds KindSet.....	123
504	Table 97.	PPMNRelationshipKinds KindSet.....	125
505	Table 98.	ResponsibilityRelationshipKinds KindSet.....	126
506	Table 99.	Delegation Attributes and/or Associations.....	130
507	Table 100.	NonHumanAgent Attributes and/or Associations.....	131
508	Table 101.	Organization Attributes and/or Associations.....	131
509	Table 102.	OrganizationStructureRelationship Attributes and/or Associations.....	131
510	Table 103.	Party Attributes and/or Associations.....	132
511	Table 104.	PartyRelationship Attributes and/or Associations.....	133
512	Table 105.	PartyRole Attributes and/or Associations	134
513	Table 106.	Person Attributes and/or Associations	134
514	Table 107.	Position Attributes and/or Associations	135
515	Table 108.	PositionAssignment Attributes and/or Associations	135
516	Table 109.	DelegationType Attributes and/or Associations.....	137

517	Table 110.	IndividualType Attributes and/or Associations.....	138
518	Table 111.	PartyRelationshipType Attributes and/or Associations.....	139
519	Table 112.	PartyRoleType Attributes and/or Associations	140
520	Table 113.	PartyType Attributes and/or Associations.....	140
521	Table 114.	PositionAssignmentType Attributes and/or Associations	141
522	Table 115.	PositionType Attributes and/or Associations	141
523	Table 116.	Area Attributes and/or Associations.....	142
524	Table 117.	GeospatialExtent Attributes and/or Associations	142
525	Table 118.	Location Attributes and/or Associations	143
526	Table 119.	NetworkAddress Attributes and/or Associations	143
527	Table 120.	Path Attributes and/or Associations	144
528	Table 121.	PhysicalAddress Attributes and/or Associations.....	144
529	Table 122.	SpaceTime Attributes and/or Associations	144
530	Table 123.	PartyDefinitions Attributes and/or Associations	148
531	Table 124.	PartyInstances Attributes and/or Associations	148
532	Table 125.	PartyModel Attributes and/or Associations.....	149
533	Table 126.	IndividualKindSet Attributes and/or Associations.....	151
534	Table 127.	PartyRelationshipKindSet Attributes and/or Associations.....	151
535	Table 128.	IndividualKinds KindSet.....	152
536	Table 129.	PartyRelationshipKinds KindSet.....	154
537	Table 130.	Depiction Resolution of PPMN Shapes	157
538	Table 131.	Depiction Resolution of Parties Shapes	160
539	Table 132.	Depiction Resolution of PPMN Edges.....	161
540	Table 133.	Depiction Resolution of Parties Edges.....	163
541			
542			

543 **Preface**

544 **OMG**

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

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

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

556 **OMG Specifications**

557 As noted, OMG specifications address middleware, modeling and vertical domain frameworks. All OMG
558 Specifications are available from the OMG website at:

559 <https://www.omg.org/spec>

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

564

565 OMG Headquarters
566 109 Highland Avenue
567 Needham, MA 02494
568 USA
569 Tel: +1-781-444-0404
570 Fax: +1-781-444-0320
571 Email: pubs@omg.org

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

573 **Issues**

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

577

578

579

580

581

582

583

584 1 Scope

585 A Pedigree and Provenance Model and Notation (**PPMN**) model is a repository of elements capturing the lineage,
586 custody and/or ownership of entities of interest. PPMN models may include elements representing the history of the
587 entities of interest as well as specifications of expected events and processes (herein referred to generally as
588 “occurrences”) related to types of entities of interest.

589 Following the approach of BPM+ Shared Data Model and Notation (**SDMN**), **PPMN** is structured to be dependent
590 on the elements defined in Specification Common Elements (**SCE** [OMG doc number bmi-2021-12-09]). Other
591 Business Modeling and Integration (BMI) Task Force and Healthcare Domain Task Force (HDTF) specifications
592 may also utilize the elements of **SCE** as those specifications are updated in the future.

593

594 2 Conformance

595 2.1 General

596 Software can claim compliance or conformance with **PPMN 1.0** if, and only if, the software fully matches the
597 applicable compliance points as stated in the specification. In addition, the structural elements provided by
598 Specification Common Elements (**SCE**) **1.0** [OMG doc number bmi-2021-12-09]) are also required in a compliant
599 or conformant software solution. Software developed only partially matching the applicable compliance points can
600 claim only that the software was based on this specification but cannot claim compliance or conformance with this
601 specification.

602 2.2 PPMN Modeling Conformance

603 The implementation claiming conformance to the Pedigree and Provenance Model and Notation SHALL comply
604 with all of the requirements set forth in Clauses 8, 9, 10, 11, 12, 13, and 14; and it SHALL be conformant with the
605 Visual Conformance in Clause 2.3.

606 This compliance point is intended to be used by **PPMN** modeling tools.

607 2.3 Visual Conformance

608 An implementation that creates and displays **PPMN** models SHALL conform to the specifications and restrictions
609 with respect to diagrammatic relationships between graphical elements, as described in Clause 14. A key element of
610 **PPMN** is the choice of shapes and icons used for the graphical elements identified in this specification. The intent is
611 to create a standard visual language that all PPMN modelers will recognize and understand. An implementation that
612 creates and displays **PPMN** models SHALL use the graphical elements, shapes, markers and decorators illustrated
613 in this specification.

614 There is flexibility in the size, color, line style, and text positions of the defined graphical elements, except where
615 otherwise specified. In particular:

- 616 • **PPMN** elements MAY have labels (e.g., its name and/or other attributes) placed inside the shape, or above
617 or below the shape, in any direction or location, depending on the preference of the modeler or modeling
618 tool vendor.
- 619 • The fills that are used for the graphical elements MAY be white or clear. The notation MAY be extended to
620 use other fill colors to suit the purpose of the modeler or tool (e.g., to highlight the value of an object
621 attribute).
- 622 • Graphical elements, shapes, and decorators MAY be of any size that suits the purposes of the modeler or
623 modeling tool with the condition that the additional graphical elements SHALL NOT conflict with any
624 current BPM+ Standard defined graphical element.
- 625 • The lines that are used to draw the graphical elements MAY be black.

- 626 ○ The notation MAY be extended to use other line colors to suit the purpose of the modeler or tool (e.g.,
627 to highlight the value of an object attribute).
- 628 ○ The notation MAY be extended to use other line styles to suit the purpose of the modeler or tool (e.g.,
629 to highlight the value of an object attribute) with the condition that the line style SHALL NOT conflict
630 with any current BPM+ Standard defined line style.

631 The following extensions to a **PPMN** model are permitted:

- 632 • New decorators or indicators MAY be added to the specified graphical elements. These decorators or
633 indicators could be used to highlight a specific attribute of a **PPMN** element or to represent a new subtype
634 of the corresponding concept with the condition that the additional graphical elements SHALL NOT
635 conflict with any current BPM+ Standard defined decorator or indicator.
- 636 • A new shape representing a new kind of **PPMN** element MAY be added to a model with the condition that
637 the shape SHALL NOT conflict with the shape specified for any other BPM+ Standard element or
638 decorator.
- 639 • Graphical elements MAY be colored, and the coloring MAY have specified semantics that extend the
640 information conveyed by the element as specified in this standard.
- 641 • The line style of a graphical element MAY be changed, but that change SHALL NOT conflict with any
642 other line style REQUIRED by this specification or the other BPM+ Standards.
- 643 • An extension SHALL NOT change the specified shape of a defined graphical element or decorator. (e.g.,
644 changing a square into a triangle, or changing rounded corners into squared corners, etc.).

645 This compliance point is intended to be used by entry-level **PPMN** tools.

646

647 **3 References**

648 **3.1 Normative References**

649 The following normative documents contain provisions which, through reference in this text, constitute provisions
650 of this specification. For dated references, subsequent amendments to, or revisions of, any of these publications do
651 not apply.

- 652 • Key words for use in RFCs to Indicate Requirement Levels, S. Bradner, IETF RFC 2119, March 1997
653 <http://www.ietf.org/rfc/rfc2119.txt>
- 654 • [BPMN] OMG Business Process and Model Notation (BPMN™): <https://www.omg.org/bpmn/>
- 655 • [CMMN] OMG Case Management Model and Model Notation
656 (CMMN™): <https://www.omg.org/spec/CMMN/>
- 657 • [DD] Diagram Definition (DD™)
- 658 • [DMN] OMG Decision Model and Model Notation (DMN™): <https://www.omg.org/spec/DMN/>
- 659 • [MOF] Meta Object Facility (MOF™): <https://www.omg.org/spec/MOF/>
- 660 • [SCE] Specification Core Elements (SCE): <https://www.omg.org/spec/SDMN/>
- 661 • [UML] Unified Modeling Language™ (UML®): <https://www.omg.org/spec/UML/>
- 662 • [XMI] XML Metadata Interchange (XMI®) <https://www.omg.org/spec/XMI/>

663

664 **3.2 Non-normative References**

665 The following normative documents contain provisions which, through reference in this text, constitute exemplars or
666 influencers of this specification. For dated references, subsequent amendments to, or revisions of, any of these
667 publications do not apply.

- 668 • [MDMI] OMG Model Driven Message Interoperability (MDMI), Version 1.0:
669 <https://www.omg.org/spec/MDMI/>
- 670 • [SysML] OMG Systems Modeling Language (SysML®): <https://www.omg.org/spec/SysML/>

671 4 Terms and Definitions

672 The table below presents a glossary for this specification:

Table 1. Glossary

Term	Definition
Area	A kind of location that encompasses some region in the world.
Chain of Control	The succession of controllers of an entity of interest. Also known as Chain of Custody.
Chain of Custody	The succession of custodians of an entity of interest. Also known as Chain of Control.
Chain of Ownership	The succession of owners of an entity of interest.
Channel	The "route" by which an entity of interest was obtained.
Controller	The party that holds an entity of interest for the owner. Also known as Custodian.
Custodian	The party that holds an entity of interest for the owner. Also known as Controller.
Delegation	A kind of Position Assignment relationship that states that one Party has been assigned a set of responsibilities by some authority.
Entity	An individual concept or informational or physical artifact that is concretized in digital or other media or in a physical representation. The W3C PROV-DM defines an entity as “ <i>An entity is a physical, digital, conceptual, or other kind of thing with some fixed aspects; entities may be real or imaginary.</i> ” ¹
Entity of Interest	The Entity (e.g., artifact, document, record, collection of materials or data element) whose provenance or pedigree is being recorded.
Geospatial Extent	A location that is a volume in the world such as a container or a room.
Location	A particular place or position.
Network Address	The address of an element or node on a network.
Non-Human Agent	Some type of automated system.
Occurrence	A "happening" of importance in a domain in some context.
Organization	Organization is used to represent a group of Parties. The group may be a company, a department within a company, a club, a consortium, or some other group.
Organization Structure Relationship	A kind of Party Relationship used to indicate internal structural relationships of a Party.
Owner	The Party that owns an entity as property. Merriam-Webster: a person who owns something : one who has the legal or rightful title to something : one to whom property belongs.
Ownership	The state, relation, or fact of being an owner. (Merriam-Webster)
Party	An abstract concept representing a Person, Role, Organization, or other entity involved in some activity, interaction or endeavor.
Party Relationship	A kind of relationship that exists between two Parties.
Party Role	A role played by a Party in some context. For instance, a Buyer or a Supplier.

¹ <https://www.w3.org/TR/2013/REC-prov-dm-20130430/#term-entity>

Path	An ordered collection of Locations.
Pedigree	Pedigree captures the "lineage" of an entity of interest. In other words, the pedigree of an Entity of Interest is the lattice formed by the sequence of activities, processes, and/or derivations performed on other entities (a.k.a, its "ancestors"), the inputs to those activities, processes, and/or derivations, and their outputs that result in or produce the Entity of Interest.
Pedigree Chain	A succession of events that have occurred in the life of an entity of interest with respect to a particular interested party.
Person	An individual homo sapiens.
Physical Address	A physical location in the real world that has an identifiable address.
Position	A Position is a formally defined role in an Organization filled by some Person. Positions are often associated with a set of responsibilities in some context. Examples of Positions include Chief Executive Officer or Technical Staff Member.
Position Assignment	Position Assignment indicates a Party is assigned to a particular Position for a particular period of time.
Provenance	Provenance captures the chain of custody or chain of ownership of an entity of interest.
Space-Time	A Location at a particular point in time.

673 5 Symbols

674 6 Additional Information

675 6.1 Conventions

676 The section introduces the conventions used in this document. This includes (text) notational conventions and
677 notations for schema components. Also included are designated namespace definitions.

678 6.2 Typographical and Linguistic Conventions and Style

679 This document incorporates the following conventions:

- 680 • The keywords “MUST,” “MUST NOT,” “REQUIRED,” “SHALL,” “SHALL NOT,” “SHOULD,”
681 “SHOULD NOT,” “RECOMMENDED,” “MAY,” and “OPTIONAL” in this document are to be
682 interpreted as described in RFC-2119.
- 683 • A **term** is a word or phrase that has a special meaning. When a term is defined, the term name is
684 highlighted in **bold** typeface.
- 685 • A reference to another definition, section, or specification is highlighted with underlined typeface and
686 provides a link to the relevant location in this specification.
- 687 • A reference to a graphical element is highlighted with a bold, capitalized word (e.g., **ProcessRef**).
- 688 • A reference to a non-graphical element or **PPMN**, **Parties**, or **SCE** concept is highlighted by being
689 italicized (e.g., *Entity*).
- 690 • A reference to an attribute or model association will be presented with the Courier New font (e.g.,
691 *Expression*).
- 692 • Non-normative examples are set off in boxes and accompanied by a brief explanation.
- 693 • XML and pseudo code is highlighted with Courier New typeface. Different font colors MAY be used to
694 highlight the different components of the XML code.
- 695 • The cardinality of any content part is specified using the following operators:

- 696 ○ [1] — exactly once
- 697 ○ [0..1] — 0 or 1
- 698 ○ [0..*] — 0 or more
- 699 ○ [1..*] — 1 or more
- 700 • Attributes separated by | and grouped within { and } — alternative values
- 701 ○ <value> — default value
- 702 ○ <type> — the type of the attribute

703 6.3 Display of Metamodel Diagrams

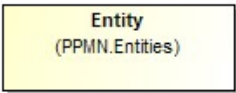

704 The metamodel presented in these sections utilizes the patterns and mechanisms that are used for the current **BPM+**
 705 specifications. **BPM+** specifications rarely display the entire metamodel of a technical specification in a single
 706 diagram. The entire metamodel would be very large, complicated, and hard to follow. Typically, a specification will
 707 present sub-sets of the overall metamodel as they apply to specific topics. For example, in the **BPMN** specification
 708 there are metamodel diagrams that show the elements relating to activities or data elements. This document will
 709 follow that pattern and present sub-sets of a larger metamodel.

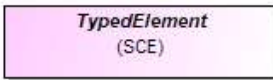
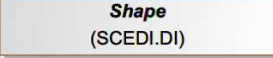
710 The metamodel diagrams are Unified Modeling Language (UML) structure diagrams. In addition to the metamodel,
 711 OMG specifications provide XML schemas which map to the metamodels. In general, it is through XML documents
 712 that **BPM+** models are stored and exchanged.

713 Further, some of the metamodel elements are references to elements from other specifications. To clarify the owner
 714 of the metamodel element, there is a parenthesized text that identifies the model owner of that element. In addition,
 715 colors are used to support the text identification of the owner-language of that element. The colors are used as an aid
 716 to distinguish the languages but does not represent a normative aspect of the metamodels nor do they add any
 717 semantic information about the metamodels.

718 The table below presents examples of elements used throughout the metamodel diagrams within this specification:

Table 2. PPMN Metamodel Color-Coding

Element	Description	Example Color
PPMN Class	These elements include the namespace in the model of the element in parentheses below the element name when that element is outside the namespace of the current diagram. These elements are color-coded light yellow and the border line color is black (see figure to the right). These make up the majority of metamodel elements shown in this specification.	
Parties Class	These elements include the namespace in the model of the element in parentheses below the element name when that element is outside the namespace of the current diagram. These elements are color-coded light green and the border line color is black (see figure to the right). These elements are primarily found in the Parties Model section of this specification but are also shown in the Pedigree and Provenance Model and Notation section of this specification.	

SCE Class	Metamodel elements from the SCE 1.0 specification [OMG doc number bmi-2021-12-09] are shown in PPMN metamodel diagrams when PPMN or Parties Model elements are dependent on a SCE element. These elements include the namespace in the metamodel in parentheses below the element name and these elements are color-coded lavender and the border line color is black (see figure to the right).	
External Class	Classes from specifications that are not specifically part of the BPM+ stack of standards can be included in metamodel diagrams and display the owner of the language in parentheses below the element name and these elements are color-coded light-gray. (see figure to the right).	

719

720 6.4 Use of Text, Color, Size, and Lines in a Diagram

- 721 • Diagram elements MAY have labels (e.g., its name and/or other attributes) placed inside the shape, or
- 722 above or below the shape, in any direction or location, depending on the preference of the modeler or
- 723 modeling tool vendor.
- 724 • The fills that are used for the graphical elements MAY be white or clear.
- 725 ○ The notation MAY be extended to use other fill colors to suit the purpose of the modeler or tool (e.g.,
- 726 to highlight the value of an object attribute).
- 727 • Diagram elements and markers MAY be of any size that suits the purposes of the modeler or modeling tool.
- 728 • The lines that are used to draw the graphical elements MAY be black.
- 729 ○ The notation MAY be extended to use other line colors to suit the purpose of the modeler or tool (e.g.,
- 730 to highlight the value of an object attribute).
- 731 ○ The notation MAY be extended to use other line styles to suit the purpose of the modeler or tool (e.g.,
- 732 to highlight the value of an object attribute) with the condition that the line style SHALL NOT conflict
- 733 with any current defined line style of the diagram.

734

735 6.5 Abbreviations

736 The table below presents a list of acronyms, and their definition, that are used in this specification:

Table 3. Acronyms

Acronym	Definition
BPM+	Business Process Management Plus
BPMN	Business Process Model and Notation
CMMN	Case Management Model and Notation
DC	Diagram Commons
DD	Diagram Definition
DI	Diagram Interchange
DMN	Decision Model and Notation
MDMI	Model Driven Message Interoperability
MOF	Meta Object Facility
OMG	Object Management Group
PPMN	Provenance and Pedigree Model and Notation

RFC	Request for Comment
SCE	Specification Common Elements
SDMN	Shared Data Model and Notation
SysML	Systems Modeling Language
URI	Uniform Resource Identifier
XMI	XML Metadata Interchange
XML	Extensible Markup Language

737

738 **6.6 Structure of this Document**

739 PPMN’s primary conceptual elements comprise *Entities*, *Occurrences*, and *Parties*, all of which are derived from
740 **SCE**. Section 7 “Overview” briefly explains concepts and depicts all relevant packages and their dependencies. It is
741 the architectural blueprint to use for all remaining sections of the document.

742 *Entities*, *Occurrences*, *Parties* and their associated packages fully describe the provenance and pedigree of entities.
743 Section 8 “Pedigree and Provenance Model and Notation” contains normative clauses defining model elements,
744 properties, associations, and packages of *Entities* and *Occurrences* and their relationships to *Parties*. Section 10
745 contains normative clauses defining model elements, properties, associations, and packages of *Parties*.

746 Section 9 “PPMN Library” and Section 11 “Parties Library” contain libraries of terms used within sections 8 and 10,
747 respectively.

748 The last section of this document, 12, describes PPMN and Parties diagram interchange (DI) specifications making
749 it possible to serialize and interchange PPMN and Parties DI instances using XMI or XML.

750 It should be noted that the elements of PPMN and Parties build upon the elements of **SCE**, a separate specification.
751 These relationships are shown where they occur. For more detail on **SCE**, please refer to the “Specification
752 Common Elements” specification.

753 **6.7 Acknowledgements**

754 **Supporting Organizations**

755 The following organizations support this specification but are not formal submitters:

756 Department of Veterans Affairs
757 cébé IT
758 Knowledge Management LLC
759 Thematrix Partners LLC.

760

761 **Special Acknowledgements**

762 The following individuals provided major input to this specification:

763 John Butler
764 Claude Baudoïn
765 Thomas Beale
766 Elisa Kendall
767 Robert Lario
768 Pete Rivett
769 Evan Wallace
770 Steve White

771

772 **7 Overview**

773 The goal of the Pedigree and Provenance Model and Notation specification is to provide a common language for
774 expressing information about the origin, evolution, ownership, custody and potential end of life of entities of
775 interest. The primary conceptual elements in the PPMN language are Entities (the items of interest), Occurrences
776 (events that affect an Entity) and Parties (responsible actors).

777 The **PPMN** specification is organized into a number of packages that together comprise the full model for
778 expressing the pedigree and provenance of entities of interest. Starting at the bottom of the figure below, **PPMN**
779 uses elements from the **SCE** model as the basis of its elements. All elements in **PPMN** are specializations of **SCE**
780 *BaseElement* directly or *RootElement*.

781 PPMN also uses elements from the **Parties** Model as shown in the second layer from the bottom. These elements
782 support the specification of various types of parties including organizations, people, positions and roles. **Parties**
783 also defines *PartyTypes*. As described in the sections below, *PartyTypes* provide the ability to state what kind of
784 *Party* is expected to play some role within an *Occurrence*.

785 The next layer up contains the basic **PPMN** elements on which the rest of the specification is built – *Entities* and
786 *Occurrences*. Entities are the things of interest from a pedigree and provenance perspective. Occurrences are the
787 "things that happen" related to these entities and parties. The layer also includes Rationale – a set of model elements
788 supporting the capture of the basis or reason for an Occurrence or OccurrenceType.

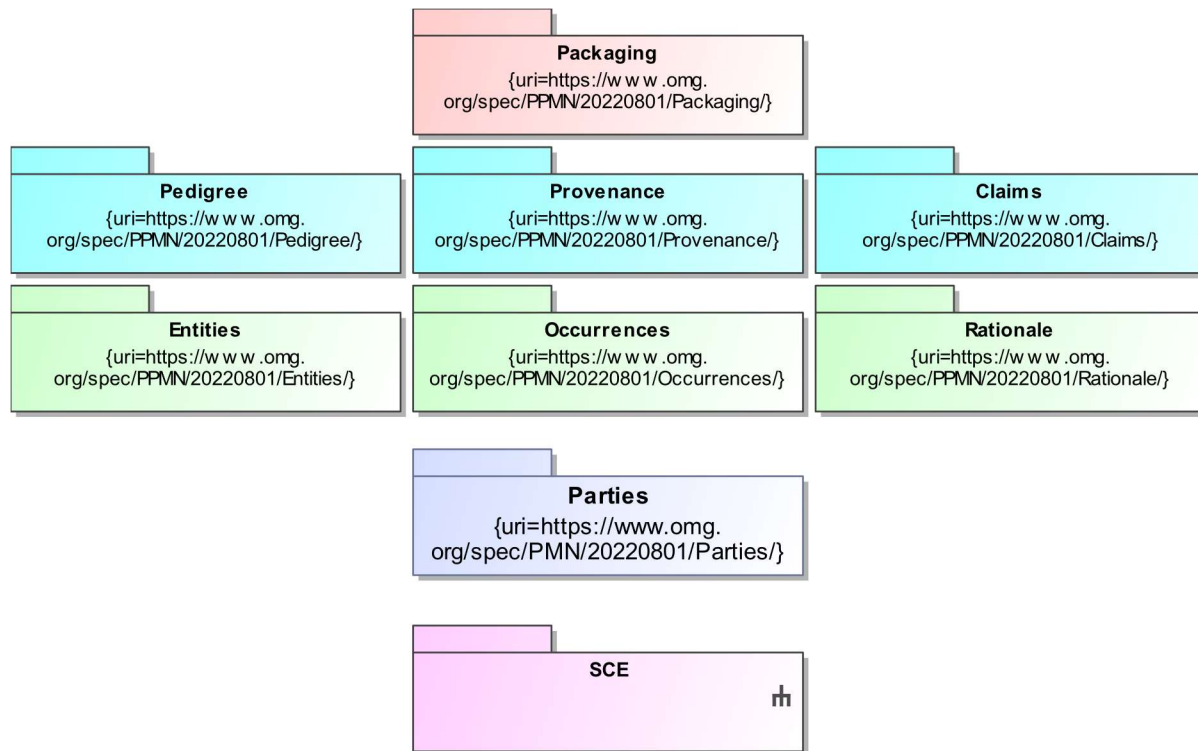
789 The fourth layer comprises a set of packages that include elements used to elaborate *Entities*, *Occurrences*, and
790 *Parties* from a pedigree and provenance perspective. *Delegation* includes elements that support the delegation of
791 responsibilities from one *Party* to another. The *Additional Relationships* package includes several specialized
792 relationships of use in capturing pedigree and provenance.

793 The fifth layer comprises the pedigree and provenance specific elements as well as mechanisms to extend the model.
794 The *Pedigree* and *Provenance* packages use elements from the lower four layers to provide the specific metadata to
795 track pedigree, the lineage of entities of interest, and provenance, the ownership and custody of those entities. The
796 *Annotations* package provides the ability to add custom metadata through annotations. *Claims* are mechanisms that
797 support the ability to capture who made a particular statement about an Occurrence and whether the statement was
798 intended to indicate that the Occurrence did in fact happen, did not happen, or may have happened.

799 Finally, the Packaging package provides elements necessary to bundle pedigree and provenance occurrence
800 instances and types into coherent sets either for storage or for exchange.

801

802



803
804 **Figure 1: PPMN Packaging Overview**

805 **8 Pedigree and Provenance Model and Notation**

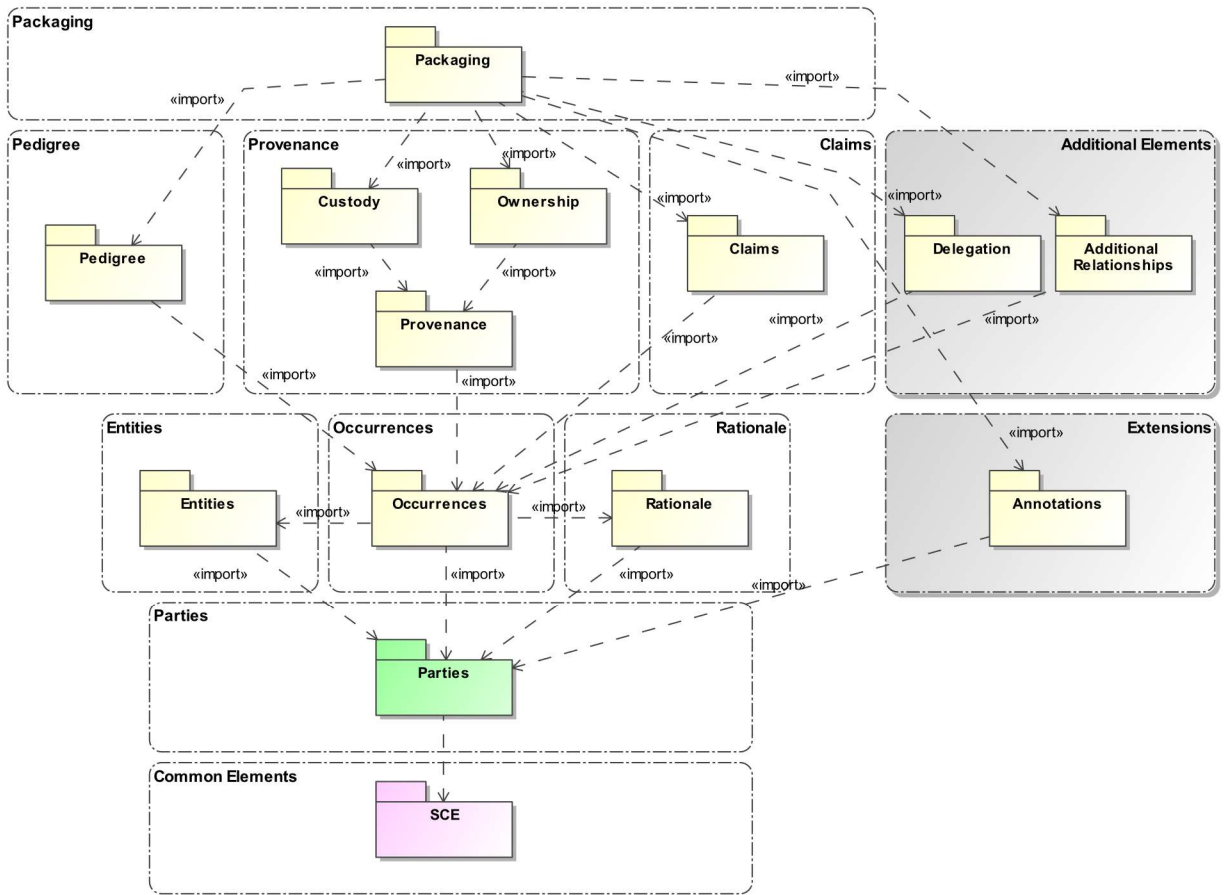
806 **PPMN** is comprised of a number of packages that group closely related elements in particular subdomains. The
807 core packages shown in the diagram below as the groupings “Pedigree”, “Provenance”, “Entities”, and
808 “Occurrences”. Pedigree describes the lineage of entities whereas Provenance describes the ownership and custody
809 of entities. Both Pedigree and Provenance build upon a general “occurrence” or “event” model contained in the
810 “Occurrences” package. The elements within these groups, along with those in the “Parties” package form the
811 essential metamodel for PPMN.

812 **PPMN** includes other packages that provide useful additions to the core model. These include “Packages”,
813 “Claims”, “Rationale”, “Delegation”, “Additional relationships”, and “Annotations”. “Packages” provides a
814 mechanism for effectively grouping elements of a PPMN model. “Claims” provide elements that allow users to
815 stipulate that assertions captured in a PPMN model are only claims and may or may not be true. “Rationale”
816 provides elements to substantiate those claims. “Delegation” comprises several elements that specify when one
817 party has acted on behalf of another or been assigned responsibilities of another. “Additional Relationships”
818 includes other less frequently used, but none the less important, relationships in the pedigree and provenance
819 domain. Finally, “Annotations” provides elements that enable the addition of various types of documentation to
820 elements of a PPMN model.

821 As shown below, **PPMN** uses the elements in both “Parties” and **SCE**. “Parties” describes people, organizations,
822 roles and their interrelationships. **SCE** comprises common metamodel elements used in PPMN and other **BPM+**
823 languages. See the **SCE** specification for more information.

824 The sections below describe the elements of PPMN in more detail.

825



826

827 **Figure 2: Pedigree and Provenance Packaging**

828 **8.1 Entities**

829 **PPMN** is concerned with recording relevant information about things of interest to stakeholders. The **Entities**
 830 package contains elements that represent those (potentially complex) things that are of interest from a pedigree
 831 and/or provenance perspective.

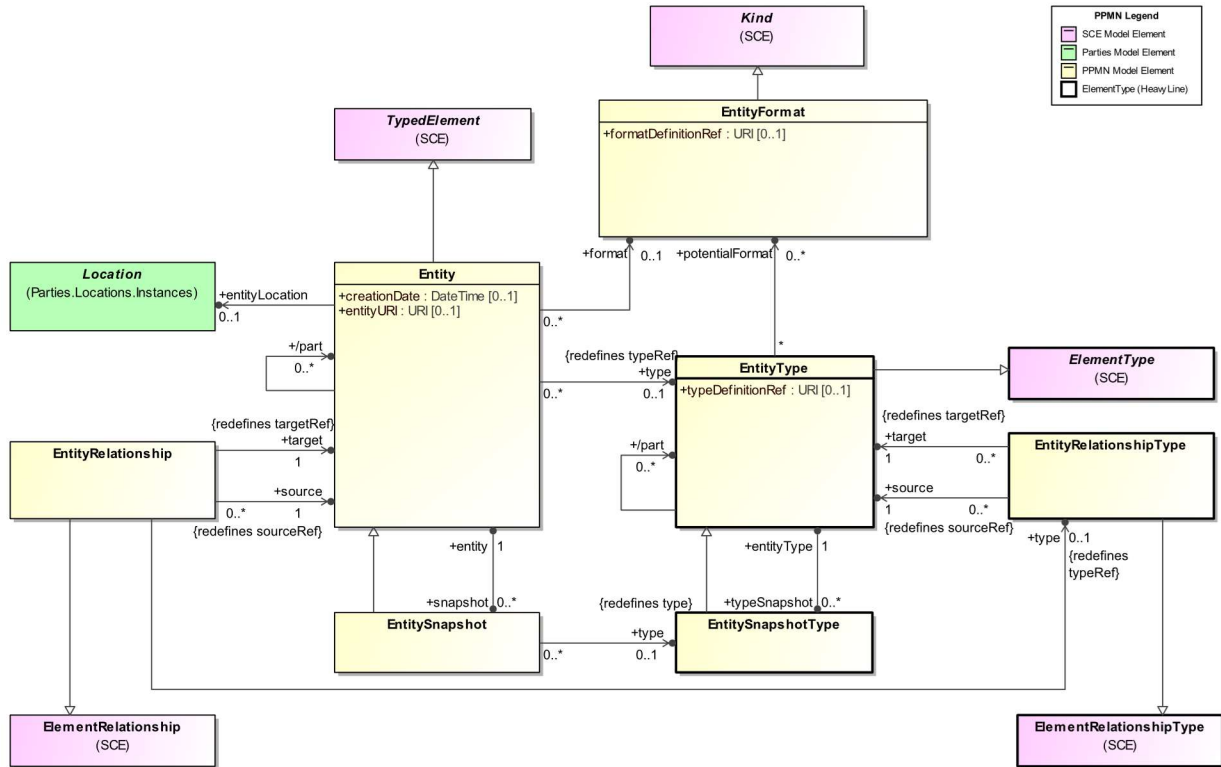
832 *Entities* are concepts or objects that may have a physical or digital embodiment. Entities may be of some defined
 833 type, *EntityType*, with a defined format and reside at some location. Entities may represent some other thing of
 834 interest through the `entityURI` property. All *Entity*-related classes are ultimately *BaseElements* and as such have
 835 a name, id, and `conceptReference`. Entities may also comprise other Entities using the *EntityRelationship*
 836 relationship.

837 *EntitySnapshots* represent some entity at a particular point in time. Like *Entities*, they may comprise other Entities
 838 using the *EntityComposition* relationship.

839 *EntityTypes*, as with *Entities*, have snapshots (*EntityTypeSnapshots*) and can comprise other *EntityTypes* through
 840 *EntityRelationshipType*. As with *EntityComposition*, *EntityRelationshipType* is also a specialization of
 841 *ElementRelationship*.

842

843



844

845 **Figure 3: Entities and EntityTypes**

846 *Entities* and *EntityTypes* are related by *EntityRelationships* and *EntityTypeRelationships*, respectively. (See the

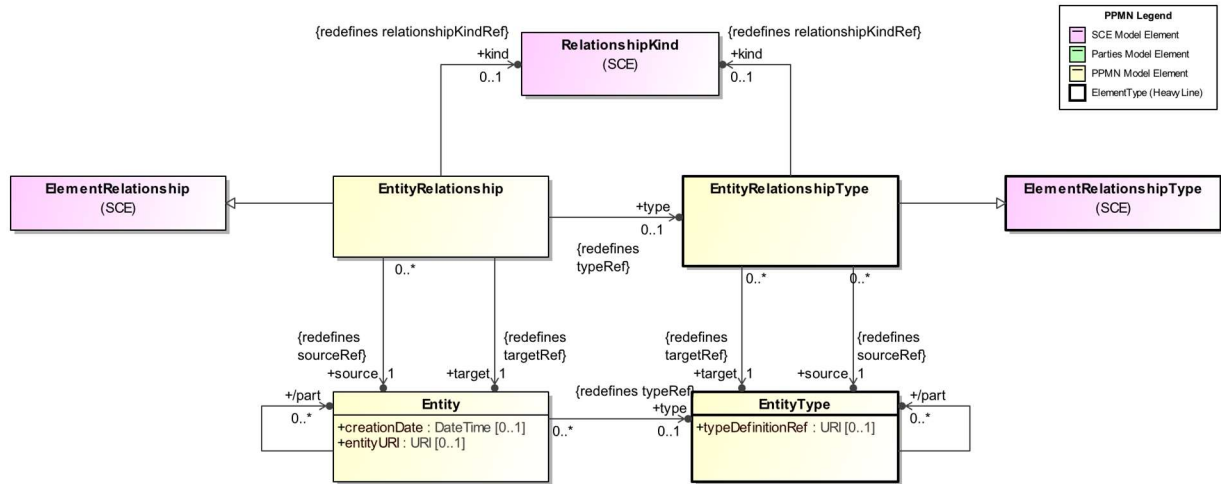
847 figure below). These relationships are used to show how *Entities* and *EntityTypes* are inter-related.

848 *EntityRelationship* is a specialization of **SCE** *ElementRelationship* whose *type* is *ElementRelationshipType* and

849 whose *kind* is a **SCE** *RelationshipKind*. *EntityRelationshipType* is a specialization of **SCE**

850 *ElementRelationshipType* and whose *kind* is also **SCE** *RelationshipKind*.

851



852

853 **Figure 4: Entity Relationships**

854

855 8.1.1 Entity

856 An individual concept, informational or physical artifact, or other kind of thing that is concretized in digital or other
857 form. The W3C PROV-DM defines an entity as “*An entity is a physical, digital, conceptual, or other kind of thing*
858 *with some fixed aspects; entities may be real or imaginary.*”² Entities may have a type and format, captured through
859 the *EntityType* and *EntityFormat*, respectively. These two classes are used together to support specifying generally
860 what kind of thing an *Entity* is and the form it may take. For example, the *EntityType* might be a "building permit"
861 and the *EntityFormat* might be ".gif". Additionally, Entities may have a location as captured by the
862 *entityLocation* property.

863 Generalizations

864 The *Entity* element inherits the attributes and/or associations of:

- 865 • *SCE TypedElement* (see the section **SCE** specification for more information).

866 Properties

867 The following table presents the additional attributes and/or associations for *Entity*:

Table 4. Entity Attributes and/or Associations

Property/Association	Description
creationDate : DateTime [0..1]	The date the <i>Entity</i> was created.
entityLocation : Location [0..1]	The location of the <i>Entity</i> .
entityURI : URI [0..1]	A URI to the <i>Entity</i> .
format : EntityFormat [0..1]	The format of the <i>Entity</i> .
part : Entity [0..*]	A derived property that indicates the <i>Entity</i> or <i>Entities</i> that comprise the <i>Entity</i> . This is determined by <i>EntityRelationships</i> whose source is the <i>Entity</i> and whose <i>kind</i> is “Composition”. (See the SCE specification for more information.)
snapshot : EntitySnapshot [0..*]	The snapshots of the <i>Entity</i> that represent the <i>Entity</i> at some particular point in time, particular <i>Location</i> , or both.
type : EntityType [0..1]	The type of the <i>Entity</i> .

868

869 8.1.2 EntityFormat

870 A kind of *Kind* that represents the format of an *Entity*. It can be something as simple as "mime types" or the
871 specification of a format documented in a formal format registry.

872 Generalizations

873 The *EntityFormat* element inherits the attributes and/or associations of:

- 874 • *Kind* (see the SCE Specification for more information).

875 Properties

876 The following table presents the additional attributes and/or associations for *EntityFormat*:

² <https://www.w3.org/TR/2013/REC-prov-dm-20130430/#term-entity>

Table 5. EntityFormat Attributes and/or Associations

Property/Association	Description
formatDefinitionRef : URI [0..1]	The identifier of the format within the specified format registry. For example "dicom" if the registry is that of W3C mime types. This is not the usual "id" found commonly in this specification. This is a "stringified" (if necessary) unique id in the context of the .formatRegistry.

877

8.1.3 EntityRelationship

879 A kind of *ElementRelationship* that represents an expected relationship between two *Entities*. The kind of
880 *EntityRelationship* is specified by the `type` property inherited from *ElementRelationship*.

881 Generalizations

882 The *EntityRelationship* element inherits the attributes and/or associations of:

- 883 • *ElementRelationship* (see the SCE Specification for more information).

884 Properties

885 The following table presents the additional attributes and/or associations for *EntityRelationship*:

Table 6. EntityRelationship Attributes and/or Associations

Property/Association	Description
occurrence : ActivityOccurrence [0..1]	The <i>Occurrence</i> that resulted in the relationship.
source : Entity [1]	The source <i>Entity</i> of the relationship.
target : Entity [1]	The target <i>Entity</i> of the relationship.
type : EntityRelationshipType [0..1]	A specification of the type of <i>EntityRelationship</i> .
kind : RelationshipKind [0..1]	The kind of <i>EntityRelationship</i> .

886

8.1.4 EntityRelationshipType

888 A kind of *ElementRelationshipType* that represents an expected relationship between two *EntityType*s. The kind of
889 *EntityRelationshipType* is specified by the `kind` property inherited from *ElementRelationshipType*.

890 Generalizations

891 The *EntityRelationshipType* element inherits the attributes and/or associations of:

- 892 • *ElementRelationshipType* (see the SCE Specification for more information).

893 Properties

894 The following table presents the additional attributes and/or associations for *EntityRelationshipType*:

Table 7. EntityRelationshipType Attributes and/or Associations

Property/Association	Description
source : EntityType [1]	The source <i>EntityType</i> of the relationship.
target : EntityType [1]	The target <i>EntityType</i> of the relationship.
kind : RelationshipKind [0..1]	The kind of <i>EntityRelationshipType</i> .

895

896 **8.1.5 EntitySnapshot**

897 A kind of *Entity* that represents a snapshot of another *Entity* at a particular point in time, a particular *Location*, or
 898 both. Additionally, *EntitySnapshots* may contain other *Entities* as specified by the *parts* that are captured through
 899 the *EntityComposition* relationship.

900 **Generalizations**

901 The *EntitySnapshot* element inherits the attributes and/or associations of:

- 902 • *Entity* (see the section entitled “[Entity](#)” for more information).

903 **Properties**

904 The following table presents the additional attributes and/or associations for *EntitySnapshot*:

Table 8. EntitySnapshot Attributes and/or Associations

Property/Association	Description
entity : Entity [1]	The <i>Entity</i> that the <i>EntitySnapshot</i> represents at some particular point in time and potentially some <i>Location</i> .
type : EntitySnapshotType [0..1]	The type of the <i>Entity</i> .

905

906 **8.1.6 EntitySnapshotType**

907 A kind of *EntityType* that represents a expected snapshot of an *EntityType* at a particular point in time, a particular
 908 *Location*, or both. Additionally, *EntityTypeSnapshots* may contain other *EntityTypes* as specified by the *part*
 909 property that are captured through the *EntityTypeComposition* relationship.

910 **Generalizations**

911 The *EntitySnapshotType* element inherits the attributes and/or associations of:

- 912 • *EntityType* (see the section entitled “[EntityType](#)” for more information).

913 **Properties**

914 The following table presents the additional attributes and/or associations for *EntitySnapshotType*:

Table 9. EntitySnapshotType Attributes and/or Associations

Property/Association	Description
entityType : EntityType [1]	The <i>EntityType</i> that the <i>EntityTypeSnapshot</i> represents at some particular point in time, particular <i>Location</i> , or both.

915

916 8.1.7 EntityType

917 *EntityType* is a designation defined for the convenience of an organization and can be used to define any concept
918 concerning an *Entity* that serves the organization. *EntityType* has 1..* potential formats specified through the
919 `potentialFormat` property to *EntityFormat*. E.g., an *EntityType* might be "Building Layout" and the possible
920 formats may be .gif, .jpeg, or paper.

921 Generalizations

922 The *EntityType* element inherits the attributes and/or associations of:

- 923 • **SCE *ElementType*** (see the section **SCE** specification for more information).

924 Properties

925 The following table presents the additional attributes and/or associations for *EntityType*:

Table 10. EntityType Attributes and/or Associations

Property/Association	Description
part : EntityType [0..*]	A derived property indicating that the <i>EntityType</i> or <i>EntityTypes</i> that comprise the <i>EntityType</i> . This is determined by <i>EntityRelationshipTypes</i> whose source is the <i>EntityType</i> and whose <code>kind</code> is "Composition". (See the SCE specification for more information.)
potentialFormat : EntityFormat [0..*]	Formats in which <i>Entities</i> of type <i>EntityType</i> may exist.
snapshotType : EntitySnapshotType [0..*]	The snapshots of the <i>EntityType</i> that represent the <i>EntityType</i> at some particular point in time, particular <i>Location</i> , or both.
typeDefinitionRef : URI [0..1]	An external definition of the <i>EntityType</i> .

926

927 8.2 Occurrences

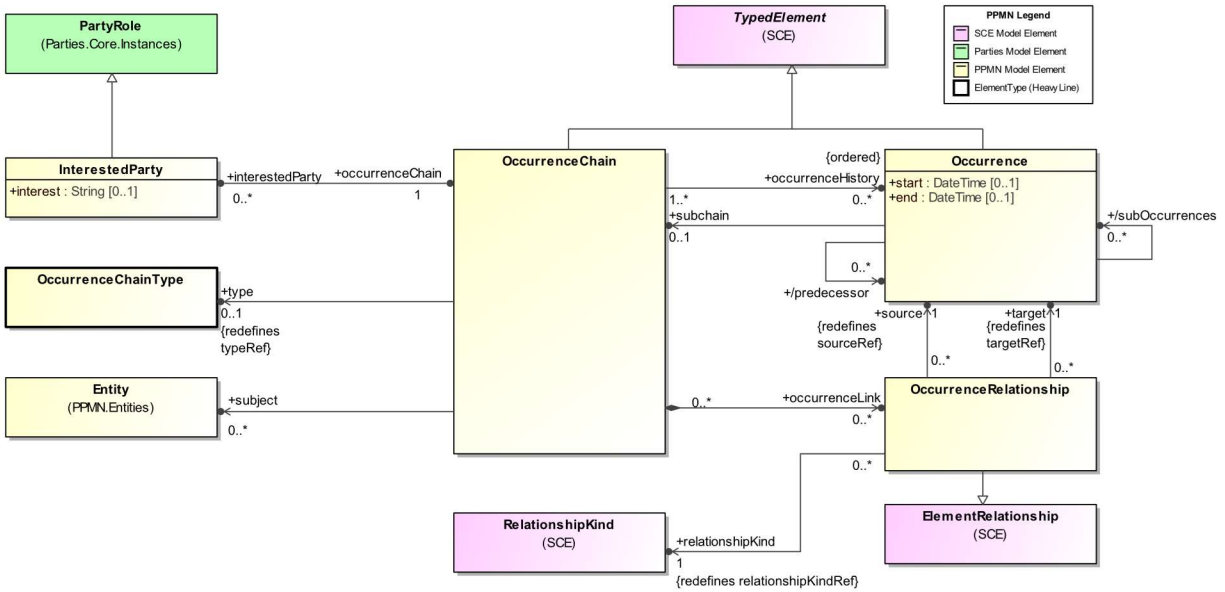
928 The Occurrences package contains general elements related to the "happenings" or events that occur over the
929 lifetime of an entity of interest. These happenings might signify anything of interest to some *Party* but are intended
930 capture common properties of pedigree- and provenance-related events.

931 **PPMN Occurrences** are "happenings" related to one or more *Entities* that have to do with the pedigree or
932 provenance of the *Entity* or *Entities*. *Occurrences* are *TypedElements* whose type is an *OccurrenceType*.
933 *Occurrences* have a `start` and `end` Date/Time and may occur at some particular `location`. *OccurrenceChains*
934 track some series of *Occurrences* related to some set of *Entities* that are the subject of the *Occurrences*.

935 Occurrences may have a number of different kinds of relationships with other types of elements. These elements
936 include *OccurrenceRelationships*, *OccurrenceDependencies*, and *OccurrenceRoles*. These are all kinds of
937 *ElementRelationship*. *OccurrenceRelationships* track the predecessor *Occurrences* of a particular *Occurrence*.
938 *OccurrenceDependencies* track the *Entities* related to a particular *Occurrence* as well as the role that the *Entity*
939 played in the *Occurrence*. *OccurrenceRoles* capture the role played by *Parties* in the *Occurrence*.

940

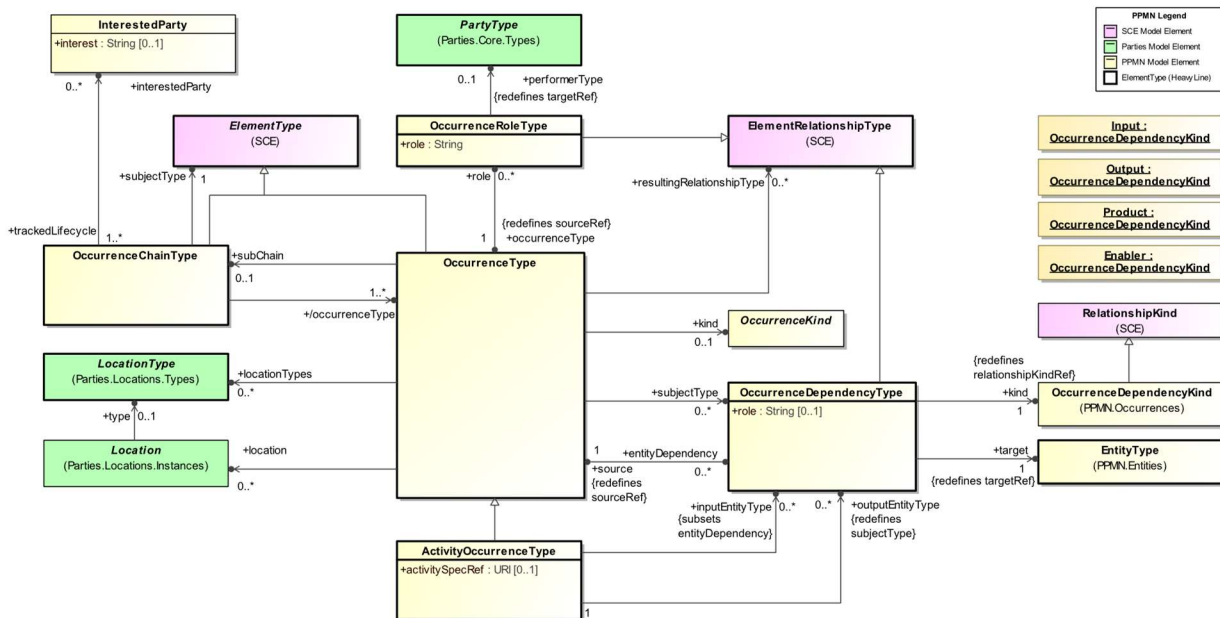
941



953
954
955 **Figure 7: Occurrence Chains**

956 *OccurrenceTypes* support the definition of expected Occurrences in an *OccurrenceChain*. Essentially,
957 *OccurrenceTypes* represent *Occurrence* instances that are expected to happen to entities of a particular type from the
958 perspective of the *InterestedParties*. *OccurrenceTypes* can be organized into graphs, *OccurrenceTypeGraphs*, that
959 show an expected sequence or "chain" of those types of *Occurrences*. Further, *OccurrenceTypes* can optionally
960 have sub-chain types so that *OccurrenceTypeGraphs* can be nested within one another. *OccurrenceTypeRole*
961 captures roles expected to be played by *Parties* in those *Occurrences*.

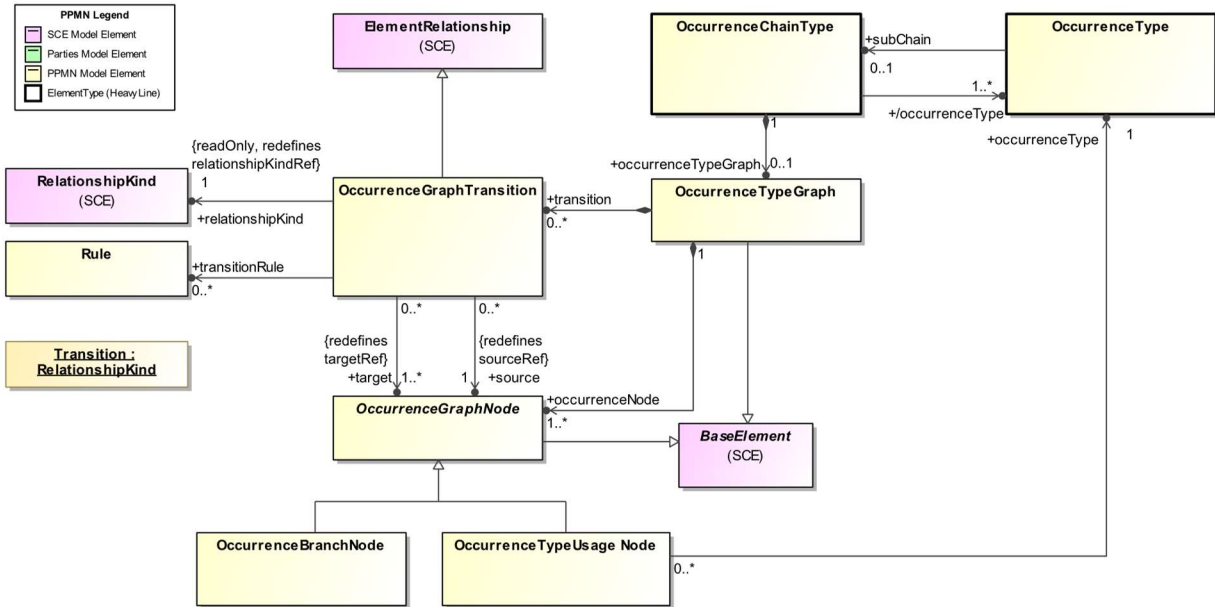
962
963



964
965 **Figure 8: Occurrence Types**

966 Expected *OccurrenceTypes* can be organized into graphs, *OccurrenceTypeGraphs*, that show an expected sequence
 967 or "chain" of those types of *Occurrences*. Further, *OccurrenceTypes* can optionally have sub-chain types so that
 968 *OccurrenceTypeGraphs* can be nested within one another. *OccurrenceTypeRole* captures roles expected to be
 969 played by *Parties*.

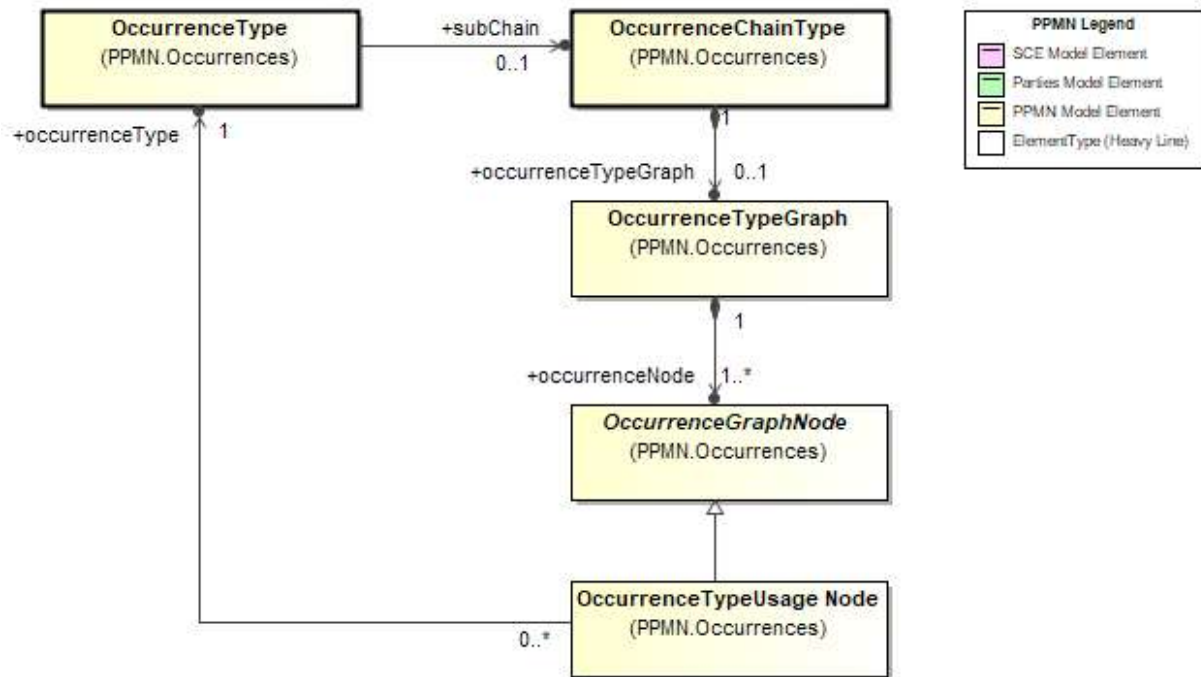
970
 971



972
 973 **Figure 9: Occurrence Type Graphs**

974 PPMN establishes a pattern of elements that supports the "nesting" of *OccurrenceChains* within an *Occurrence*.
 975 This pattern allows for encapsulation of parts of a chain where the details of the *Occurrences* of that part of a larger
 976 chain are either not known initially or are not deemed important in some context. The figure below illustrates this
 977 pattern at the "type" level.

978

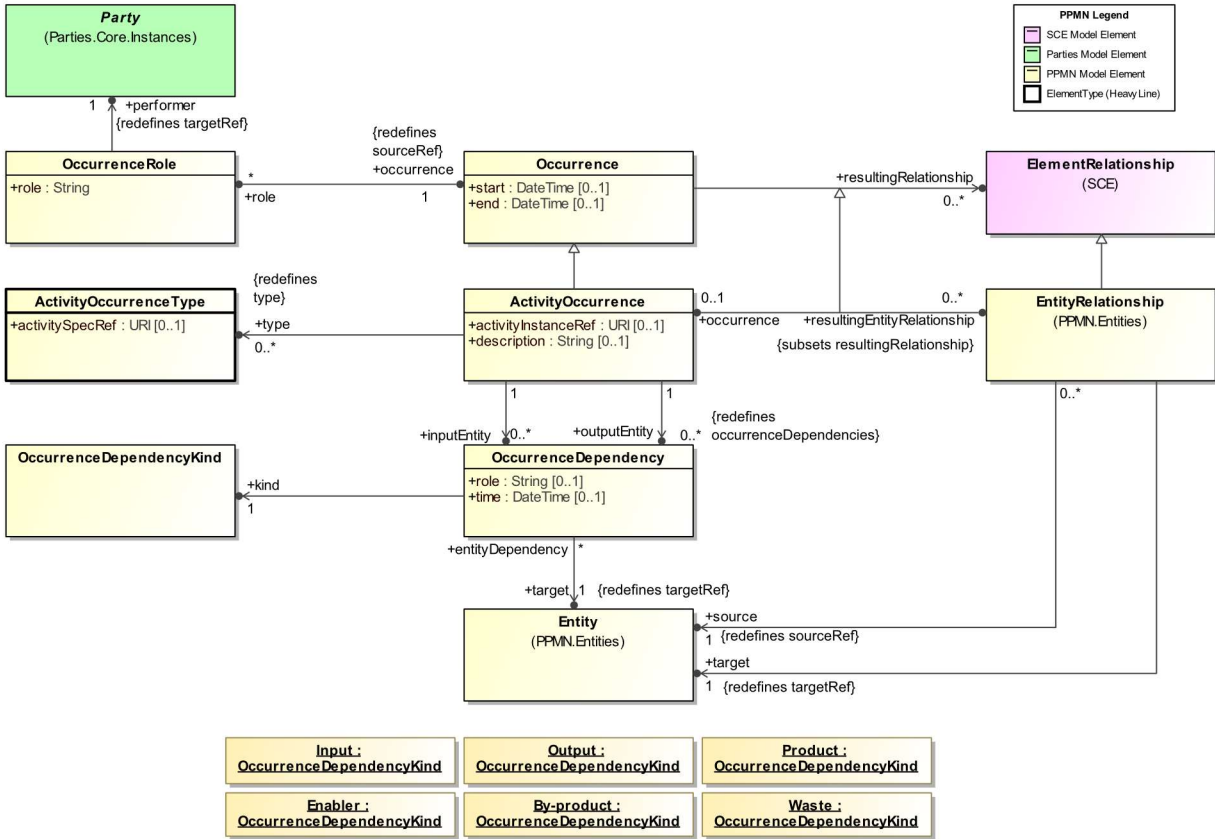


979
 980 **Figure 10: Occurrences Type Pattern**

981
 982
 983
 984
 985
 986
 987
 988
 989
 990
 991
 992

An *ActivityOccurrence* is a kind of *Occurrence* that represents some activity that produces or modifies one or more entities. The *ActivityOccurrenceType* specifies the type of activity of the *ActivityOccurrence* providing a URI for a specification of the *ActivityOccurrenceType*.

ActivityOccurrence has a name (inherited), a URI reference to a specification of the instance if one exists, and a description. *ActivityOccurrence* includes zero or more references to *Parties* that play a part in the activity through the inherited *OccurrenceRole* property and references to the entities used in the activity through the *inputEntity* property which holds a collection of *OccurrenceDependencies*. Output entities of the activity are captured through the *outputEntity* property.



993
994 **Figure 11: Activity Occurrences**

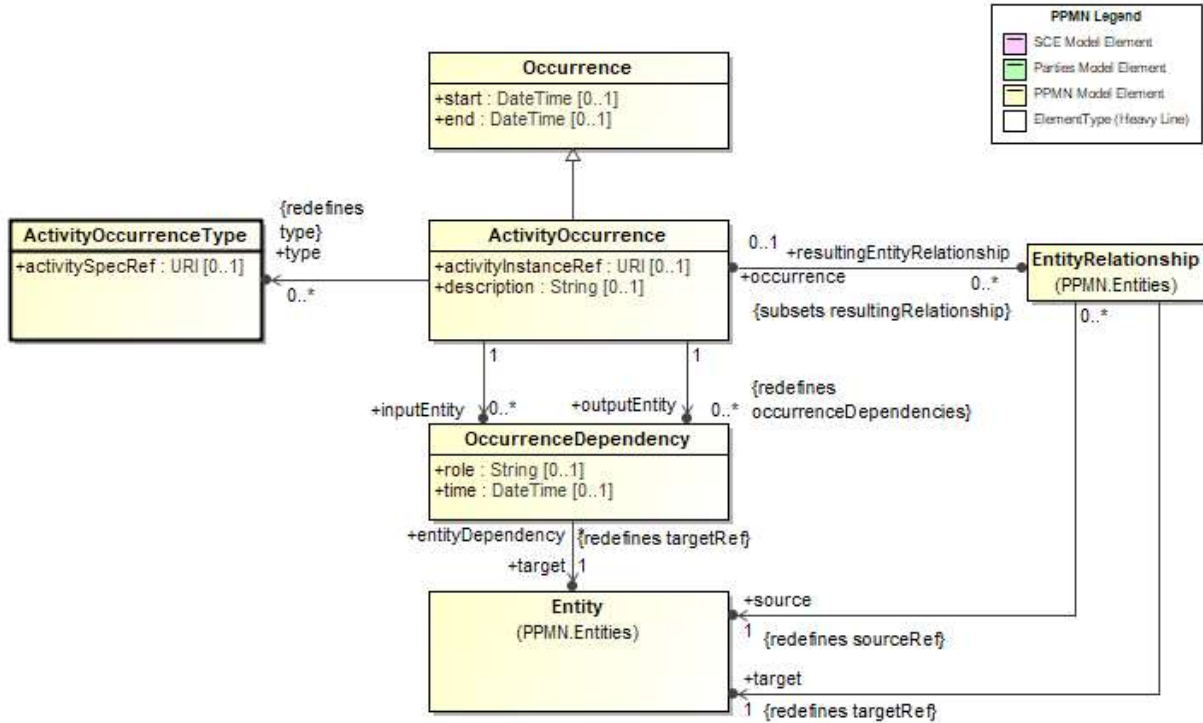
995 **8.2.1 ActivityOccurrence**

996 A kind of *Occurrence* that records the input and output entities of interest as the result of some activity or derivation.

997 An *ActivityOccurrence* is a kind of *Occurrence* that represents some activity that produces or modifies one or more
998 entities. The *ActivityOccurrenceType* specifies the type of activity of the *ActivityOccurrence* providing a URI for a
999 specification.

1000 *ActivityOccurrences* have a name (inherited), a URI reference to an instance if one exists, and a description.
1001 *ActivityOccurrences* include references to *Parties* that play a part in the activity through the inherited
1002 *OccurrenceRole* property and references to the entities used in the activity through the *inputEntity* property
1003 which holds a collection of *OccurrenceDependencies*. Output entities of the activity are captured through the
1004 *outputEntity* property of *PedigreeOccurrence*.

1005



1006
 1007 **Figure 12: Activity Occurrence**
 1008 **Generalizations**

1009 The *ActivityOccurrence* element inherits the attributes and/or associations of:

- 1010 • *Occurrence* (see the section entitled “[Occurrence](#)” for more information).

1011 **Properties**

1012 The following table presents the additional attributes and/or associations for *ActivityOccurrence*:

Table 11. ActivityOccurrence Attributes and/or Associations

Property/Association	Description
activityInstanceRef : URI [0..1]	A reference to an instance that the <i>ActivityOccurrence</i> represents. This could be an instance running in a business process execution engine or some other tool.
description : String [0..1]	A textual description of the activity.
inputEntity : OccurrenceDependency [0..*]	A set of dependencies to the entities that were inputs to the <i>ActivityOccurrence</i> .
outputEntity : OccurrenceDependency [0..*]	A set of dependencies on entities that were outputs or results of the <i>ActivityOccurrence</i> .
resultingEntityRelationship : EntityRelationship [0..*]	EntityRelationships created as a result of the Occurrence.
type : ActivityOccurrenceType [0..*]	The type of the <i>ActivityOccurrence</i> .

1013

1014 **8.2.2 ActivityOccurrenceType**

1015 A potentially complex *OccurrenceType* that identifies an expected activity that may have input and output entities of
1016 interest.

1017 **Generalizations**

1018 The *ActivityOccurrenceType* element inherits the attributes and/or associations of:

- 1019 • *OccurrenceType* (see the section entitled "[OccurrenceType](#)" for more information).

1020 **Properties**

1021 The following table presents the additional attributes and/or associations for *ActivityOccurrenceType*:

Table 12. ActivityOccurrenceType Attributes and/or Associations

Property/Association	Description
activitySpecRef : URI [0..1]	A reference to a specification for the activity.
inputEntityType : OccurrenceDependencyType [0..*]	A set of dependencies that point to the types of entities that are expected to be consumed or used by instances of the <i>OccurrenceType</i> .
outputEntityType : OccurrenceDependencyType [0..*]	A set of dependencies that point to the types of entities that are expected to be produced by instances of the <i>OccurrenceType</i> .

1022

1023 **8.2.3 InterestedParty**

1024 A kind of *PartyRole* that captures the fact that a *Party* has some interest in a particular occurrence chain as specified
1025 by its *occurrenceChain* property or so some set of *OccurrenceChains* as defined by an *OccurrenceChainType*.

1026 **Generalizations**

1027 The *InterestedParty* element inherits the attributes and/or associations of:

- 1028 • *PartyRole* (see the section entitled "[PartyRole](#)" for more information).

1029 **Properties**

1030 The following table presents the additional attributes and/or associations for *InterestedParty*:

Table 13. InterestedParty Attributes and/or Associations

Property/Association	Description
interest : String [0..1]	A textual description of the interest the associated <i>Party</i> has in the <i>Occurrences</i> .
occurrenceChain : OccurrenceChain [1]	The <i>OccurrenceChains</i> of interest to some <i>Party</i> .

1031

1032 **8.2.4 Occurrence**

1033 A *Occurrence* or "happening" of importance in a domain in some context.

1034 **Generalizations**

1035 The *Occurrence* element inherits the attributes and/or associations of:

- 1036 • *SCE TypedElement* (see the section **SCE** specification for more information).

1037 **Properties**

1038 The following table presents the additional attributes and/or associations for *Occurrence*:

Table 14. Occurrence Attributes and/or Associations

Property/Association	Description
end : DateTime [0..1]	The <i>DateTime</i> of the end of the <i>Occurrence</i> .
kind : OccurrenceKind [0..1]	A reference to a definition of the specific kind of <i>Occurrence</i> .
location : Location [0..1]	The location at which an <i>Occurrence</i> took place.
occurrenceDependencies : OccurrenceDependency [0..*]	A dependency on the subject(s) of the <i>Occurrence</i> .
predecessor : Occurrence [0..*]	A derived property indicating a dependency on one or more preceding <i>Occurrences</i> .
rationale : Rationale [0..*]	The <i>Rationale</i> given for the <i>Occurrence</i> .
resultingRelationship : ElementRelationship [0..*]	The relationships generated by the <i>Occurrence</i> .
role : OccurrenceRole [*]	A role played by some <i>Party</i> in an <i>Occurrence</i> .
start : DateTime [0..1]	The <i>DateTime</i> of the start of the <i>Occurrence</i> .
subchain : OccurrenceChain [0..1]	An <i>OccurrenceChain</i> that is encapsulated by the <i>Occurrence</i> - essentially a "sub-chain".
subOccurrences : Occurrence [0..*]	A set of <i>Occurrences</i> that happen as part of the parent <i>Occurrence</i> . These <i>Occurrences</i> are normally part of a "sub-chain".
type : OccurrenceType [0..1]	The type of an <i>Occurrence</i> .

1039

1040 **8.2.5 OccurrenceBranchNode**

1041 A kind of *OccurrenceGraphNode* that allows for branching or other kinds of connections between other
1042 *OccurrenceGraphNode*s.

1043 **Generalizations**

1044 The *OccurrenceBranchNode* element inherits the attributes and/or associations of:

- 1045 • *OccurrenceGraphNode* (see the section entitled "[OccurrenceGraphNode](#)" for more information).

1046 **Properties**

1047 The *OccurrenceBranchNode* element does not have any additional attributes and/or associations.

1048 **8.2.6 OccurrenceChain**

1049 A succession of *Occurrences* (events or activities) that have happened in the life of some *RootElement* that are of
1050 interest to some *Party*.

1051 **Generalizations**

1052 The *OccurrenceChain* element inherits the attributes and/or associations of:

- 1053 • **SCE TypedElement** (see the section **SCE** specification for more information).

1054 **Properties**

1055 The following table presents the additional attributes and/or associations for *OccurrenceChain*:

Table 15. OccurrenceChain Attributes and/or Associations

Property/Association	Description
interestedParty : InterestedParty [0..*]	The <i>Parties</i> that have some interest in <i>Occurrences</i> related to the subject elements.
occurrenceHistory : Occurrence [0..*]	A set of <i>Occurrences</i> that comprise the chain.
occurrenceLink : OccurrenceRelationship [0..*]	The <i>OccurrenceRelationship(s)</i> that show(s) the relationship(s) between <i>Occurrences</i> in the chain.
subject : Entity [0..*]	The element(s) that is(are) the result of the <i>Occurrences</i> in the chain.
type : OccurrenceChainType [0..1]	The type of the <i>OccurrenceChain</i> .

1056

1057 **8.2.7 OccurrenceChainType**

1058 An *OccurrenceChainType* is a kind of *ElementType* that captures a specification for a series potential *Occurrences*
 1059 that are expected in a particular context. An *OccurrenceChainType* captures this specification through the
 1060 occurrenceTypeGraph property - a graph of *OccurrenceGraphNodes* and *OccurrenceTransitionTypes*.

1061 **Generalizations**

1062 The *OccurrenceChainType* element inherits the attributes and/or associations of:

- 1063 • **SCE ElementType** (see the section **SCE** specification for more information).

1064 **Properties**

1065 The following table presents the additional attributes and/or associations for *OccurrenceChainType*:

Table 16. OccurrenceChainType Attributes and/or Associations

Property/Association	Description
interestedParty : InterestedParty [0..*]	The parties that are interested in the "lifecycle" specified by the <i>OccurrenceChainType</i> .
occurrenceType : OccurrenceType [1..*]	The occurrenceType derived property is based on the series of relationships between from <i>OccurrenceChainType</i> through other classes to <i>OccurrenceType</i> : OccurrenceChainType.occurrenceTypeGraph.occurrenceNode.occurrenceType.
occurrenceTypeGraph : OccurrenceTypeGraph [0..1]	A graph of <i>OccurrenceTypes</i> that specifies the sequencing of expected <i>Occurrences</i> in the lifecycle of an entity of interest to one or more <i>InterestedParties</i> .

subjectType : ElementType [1]	The subject of the <i>OccurrenceChainType</i> .
--------------------------------------	---

1066

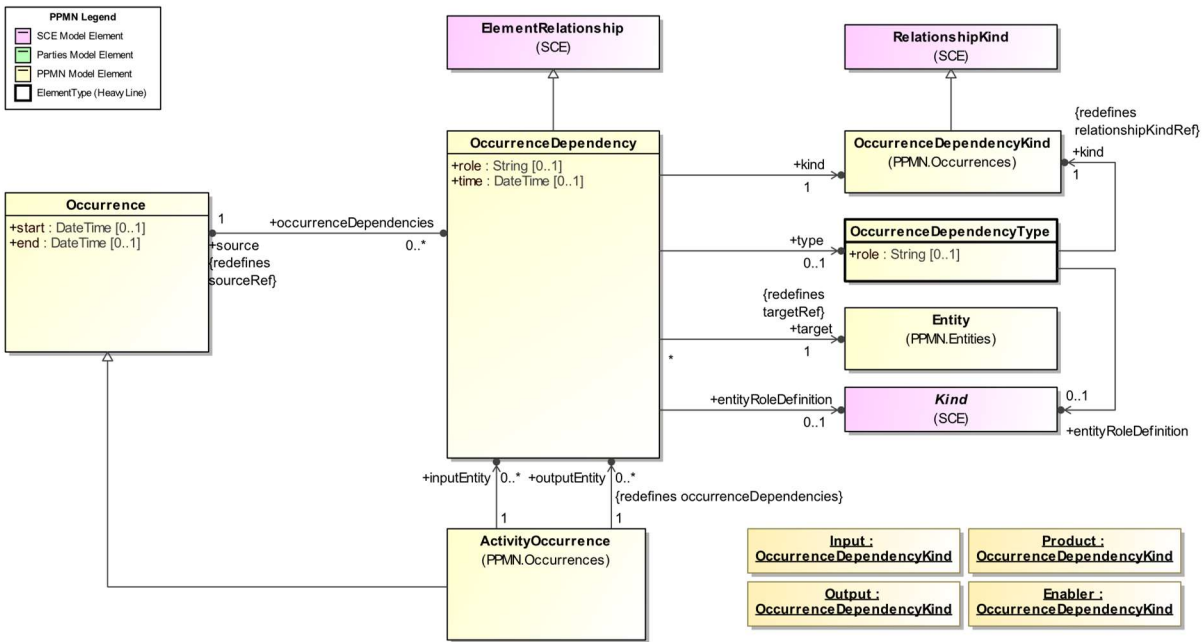
1067 **8.2.8 OccurrenceDependency**

1068 A type of relationship that records the dependence on an entity of interest for some particular purpose. That purpose
 1069 is captured as the *role*.

1070 *OccurrenceDependencies* indicate how *Entities* are used within an *Occurrence*.

1071

1072



1073

1074 **Figure 13: OccurrencesDependencies**

1075 **Generalizations**

1076 The *OccurrenceDependency* element inherits the attributes and/or associations of:

- 1077 • *ElementRelationship* (see the SCE Specification for more information).

1078 **Properties**

1079 The following table presents the additional attributes and/or associations for *OccurrenceDependency*:

Table 17. OccurrenceDependency Attributes and/or Associations

Property/Association	Description
entityRoleDefinition : Kind [0..1]	A <i>Kind</i> that provides a definition of the way the <i>Entity</i> was used in the <i>Occurrence</i> .
kind : OccurrenceDependencyKind [1] default: Output	A description of the type of dependency an <i>OccurrenceType</i> has on an <i>EntityType</i> . See <i>RelationshipKind</i> , below, for more details.

relationshipKind : RelationshipKind [1] default: Dependency	A description of the type of the relationship. See <i>RelationshipKind</i> , below, for more details. This property is read only and set to Dependency.
role : String [0..1]	The role of the target element in the source <i>Occurrence</i> .
source : Occurrence [1]	The <i>Occurrence</i> that has some dependency on the target <i>Occurrence</i> .
target : Entity [1]	The <i>Entity</i> on which some <i>Occurrence</i> depends.
time : DateTime [0..1]	The time that the <i>Occurrence</i> had the dependency on the <i>Entity</i> .
type : OccurrenceDependencyType [0..1]	The type of the <i>EntityDependency</i> .

1080

1081 8.2.9 OccurrenceDependencyKind

1082 A class indicating the kind of dependency an *Occurrence* has on an *Entity*.

1083 Generalizations

1084 The *OccurrenceDependencyKind* element inherits the attributes and/or associations of:

- 1085 • *RelationshipKind* (see the SCE Specification for more information).

1086 Properties

1087 The *OccurrenceDependencyKind* element does not have any additional attributes and/or associations.

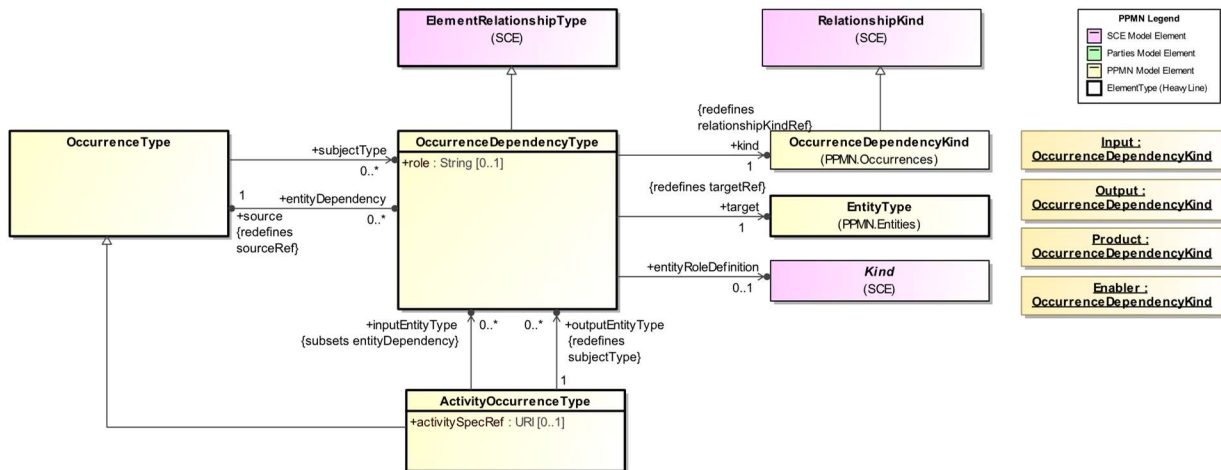
1088 8.2.10 OccurrenceDependencyType

1089 A kind of *ElementRelationship* that captures a dependency of a type of *Occurrence* on a particular type of entity and
1090 the role the entity plays in that type of *Occurrence*.

1091 *OccurrenceRoleTypes* indicate how *Parties* are expected to participate in an *Occurrence*.

1092

1093



1094

1095 **Figure 14: Occurrence Dependency Types**

1096 **Generalizations**

1097 The *OccurrenceDependencyType* element inherits the attributes and/or associations of:

- 1098
 - *ElementRelationshipType* (see the SCE Specification for more information).

1099 **Properties**

1100 The following table presents the additional attributes and/or associations for *OccurrenceDependencyType*:

Table 18. OccurrenceDependencyType Attributes and/or Associations

Property/Association	Description
entityRoleDefinition : Kind [0..1]	A <i>Kind</i> that provides a definition of the way the <i>EntityType</i> is expected to be used in the <i>OccurrenceType</i> .
kind : OccurrenceDependencyKind [1] default: Output	A description of the type of dependency an <i>OccurrenceType</i> has on an <i>EntityType</i> . See <i>EntityDependencyKind</i> for more details.
role : String [0..1]	The role of the <i>ElementType</i> in the <i>OccurrenceType</i> .
source : OccurrenceType [1]	The <i>OccurrenceType</i> whose instances are the source of instances of the <i>ElementType</i> .
target : EntityType [1]	The <i>ElementType</i> on which the <i>OccurrenceType</i> depends.

1101

1102 **8.2.11 OccurrenceGraphNode**

1103 A type of graph *Node* that is particular to an *OccurrenceTypeGraph*.

1104 **Generalizations**

1105 The *OccurrenceGraphNode* element inherits the attributes and/or associations of:

- 1106
 - *SCE BaseElement* (see the section SCE specification for more information).

1107 **Properties**

1108 The *OccurrenceGraphNode* element does not have any additional attributes and/or associations.

1109 **8.2.12 OccurrenceGraphTransition**

1110 A type of *Link* in a *OccurrenceTypeGraph* definition from one *OccurrenceType* to another.

1111 **Generalizations**

1112 The *OccurrenceGraphTransition* element inherits the attributes and/or associations of:

- 1113
 - *ElementRelationship* (see the SCE Specification for more information).

1114 **Properties**

1115 The following table presents the additional attributes and/or associations for *OccurrenceGraphTransition*:

Table 19. OccurrenceGraphTransition Attributes and/or Associations

Property/Association	Description
relationshipKind : RelationshipKind [1] default: Transition	A description of the type of the relationship. See <i>RelationshipKind</i> in the SCE specification for more details. This property is read only and set to Transition.
source : OccurrenceGraphNode [1]	The <i>OccurrenceGraphNode</i> from which the transition leaves.
target : OccurrenceGraphNode [1..*]	The <i>OccurrenceGraphNode</i> to which the transition leads.
transitionRule : Rule [0..*]	The <i>Rules</i> that constrain the <i>OccurrenceTransitionType</i> .

1116

1117 8.2.13 OccurrenceKind

1118 A class indicating the specific kind of *Occurrence*.

1119 Generalizations

1120 The *OccurrenceKind* element inherits the attributes and/or associations of:

- 1121 • *Kind* (see the SCE specification for more information).

1122 Properties

1123 The *OccurrenceKind* element does not have any additional attributes and/or associations.

1124 8.2.14 OccurrenceRelationship

1125 A kind of *ElementRelationship* that captures the fact that one *Occurrence* has a relationship to another for some
 1126 reason. Examples include an *Occurrence* using an Entity created by another *Occurrence*. This usage implies that
 1127 the first *Occurrence* depended on the second *Occurrence* for that *Entity*. For these types of “flow” relationships the
 1128 *relationshipKind* would be set to “Transition”.

1129 Another example is the aggregation of several *Occurrences* into one containing *Occurrence*. In this case, the
 1130 *relationshipKind* would be set to “Composition”.

1131 In this way, an *OccurrenceChain* can be built by capturing and analyzing the relationships and generating the
 1132 implied chain.

1133 Generalizations

1134 The *OccurrenceRelationship* element inherits the attributes and/or associations of:

- 1135 • *ElementRelationship* (see the SCE specification for more information).

1136 Properties

1137 The following table presents the additional attributes and/or associations for *OccurrenceRelationship*:

Table 20. OccurrenceRelationship Attributes and/or Associations

Property/Association	Description
source : Occurrence [1]	The dependent <i>Occurrence</i> .
target : Occurrence [1]	The <i>Occurrence</i> on which the source <i>Occurrence</i> depends.

relationshipKind : RelationshipKind [1]	A description of the kind of the relationship between the two <i>Occurrences</i> . See <i>RelationshipKind</i> in the SCE specification for more details. This property is read only and set to Transition.
--	---

1138

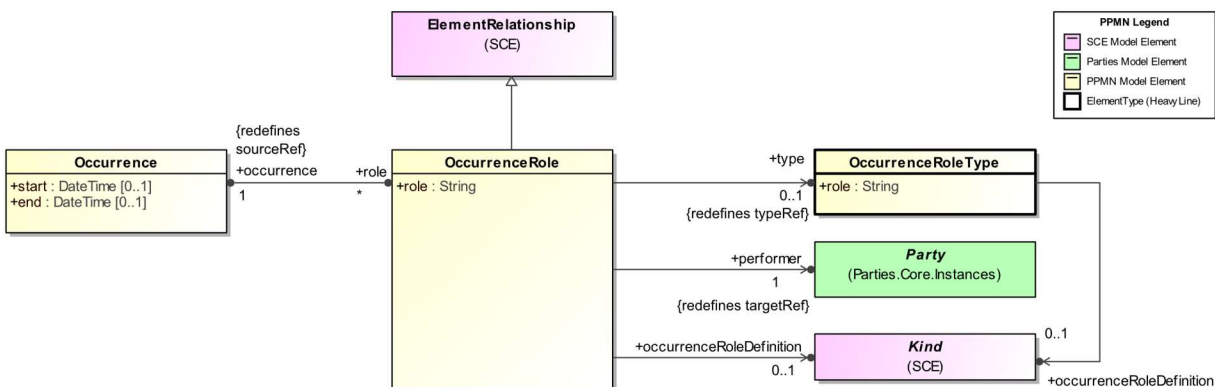
1139 8.2.15 OccurrenceRole

1140 A role played by some *Party* in an *Occurrence*.

1141 *OccurrenceRoles* indicate how a *Party* participated in an *Occurrence*.

1142

1143



1144

1145 **Figure 15: OccurrencesRoles**

1146 Generalizations

1147 The *OccurrenceRole* element inherits the attributes and/or associations of:

- 1148 • *ElementRelationship* (see the SCE specification for more information).

1149 Properties

1150 The following table presents the additional attributes and/or associations for *OccurrenceRole*:

Table 21. OccurrenceRole Attributes and/or Associations

Property/Association	Description
occurrence : Occurrence [1]	The <i>Occurrence</i> in which the <i>Party</i> plays the role.
occurrenceRoleDefinition : Kind [0..1]	A <i>Kind</i> that provides a definition of the role the <i>Party</i> played in the <i>Occurrence</i> .
performer : Party [1]	The <i>Party</i> that plays the role in an <i>Occurrence</i> specified by the <i>OccurrenceRole</i> .
role : String []	A textual description of the actual role played by the performer in the activity.
type : OccurrenceRoleType [0..1]	The type of the role played by the performer <i>Party</i> in the <i>Occurrence</i> .

1151

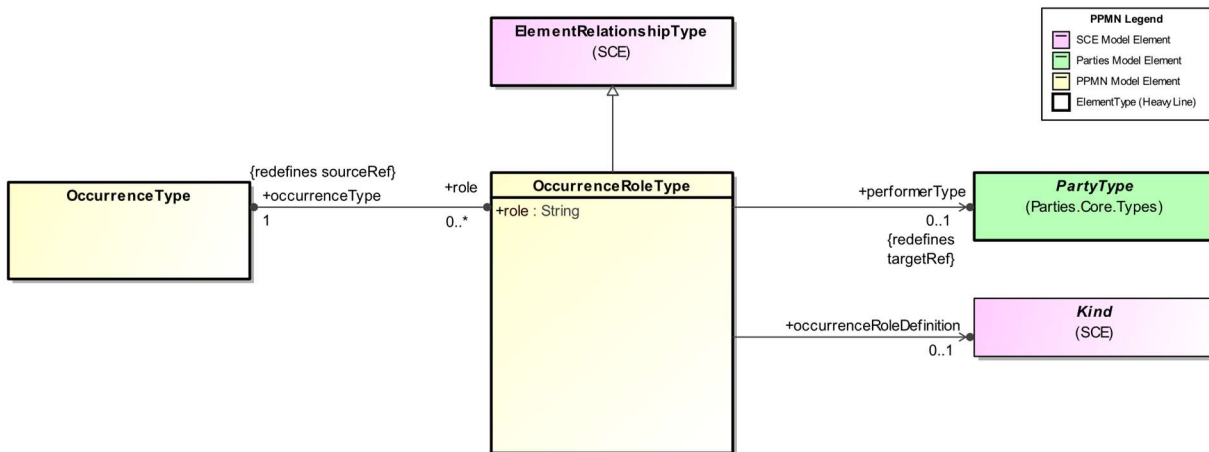
1152 **8.2.16 OccurrenceRoleType**

1153 A specification of the type of party expected to play a role an *OccurrenceType*.

1154 *OccurrenceTypes* support the definition of expected Occurrences in a Pedigree or Provenance Chain. Essentially,
 1155 *OccurrenceTypes* represent *Occurrence* instances that are expected to with respect to entities of a particular type
 1156 from the perspective of the *InterestedParties*. These expected *OccurrenceTypes* can be organized into graphs,
 1157 *OccurrenceTypeGraphs*, that show an expected sequence or "chain" of those types of *Occurrences*. Further,
 1158 *OccurrenceTypes* can optionally have sub-chain types so that *OccurrenceTypeGraphs* can be nested within one
 1159 another. *OccurrenceTypeRole* captures roles expected to be played by *Parties*.

1160

1161



1162

1163 **Figure 16: Occurrence Role Types**

1164 **Generalizations**

1165 The *OccurrenceRoleType* element inherits the attributes and/or associations of:

- 1166 • *ElementRelationshipType* (see the SCE specification for more information).

1167 **Properties**

1168 The following table presents the additional attributes and/or associations for *OccurrenceRoleType*:

Table 22. OccurrenceRoleType Attributes and/or Associations

Property/Association	Description
occurrenceRoleDefinition : Kind [0..1]	A <i>Kind</i> that provides a definition of the role the <i>PartyType</i> is expected to play in the <i>OccurrenceType</i> .
occurrenceType : OccurrenceType [1]	The type of <i>Occurrence</i> in which the <i>expectedPerformer</i> to perform in the role.
performerType : PartyType [0..1]	The <i>Party</i> that is expected to perform in a particular role in an <i>Occurrence</i> .
role : String []	A textual description of the role in the <i>Occurrence</i> .

1169

1170 **8.2.17 OccurrenceType**

1171 The type or specification of an *Occurrence* that may happen or be of interest. An *OccurrenceType* may have a
 1172 *subChainType* enabling nesting of *OccurrenceChainTypes*.

1173 **Generalizations**

1174 The *OccurrenceType* element inherits the attributes and/or associations of:

- 1175 • **SCE ElementType** (see the **SCE** specification for more information).

1176 **Properties**

1177 The following table presents the additional attributes and/or associations for *OccurrenceType*:

Table 23. OccurrenceType Attributes and/or Associations

Property/Association	Description
entityDependency : OccurrenceDependencyType [0..*]	A dependency on the <i>ElementTypes</i> that are involved in this <i>OccurrenceType</i> .
kind : OccurrenceKind [0..1]	A reference to a definition of the specific kind of Occurrence.
location : Location [0..*]	The location at which <i>Occurrences</i> of type <i>OccurrenceType</i> are planned or expected to happen.
locationTypes : LocationType [0..*]	The types of <i>Locations</i> at which <i>Occurrences</i> of type <i>OccurrenceType</i> are planned or expected to happen.
rationale : Rationale [0..1]	The <i>Rationale</i> given for the <i>OccurrenceType</i> .
resultingRelationshipType : ElementRelationshipType [0..*]	The <i>ElementRelationshipTypes</i> that exist as a result of <i>Occurrences</i> of type <i>OccurrenceType</i> .
role : OccurrenceRoleType [0..*]	A set of <i>OccurrenceTypeRoles</i> that specify the role a <i>Party</i> is expected to play in an <i>Occurrence</i> .
subChain : OccurrenceChainType [0..1]	An <i>OccurrenceChainType</i> that is encapsulated within the <i>OccurrenceType</i> to create a "subchain".
subjectType : OccurrenceDependencyType [0..*]	A dependency on the <i>ElementTypes</i> that are the subject of this <i>OccurrenceType</i> .

1178

1179 **8.2.18 OccurrenceTypeGraph**

1180 A type of Graph that captures the *OccurrenceTypes* that are expected in the lifecycle of one or more *EntityTypes*.

1181 **Generalizations**

1182 The *OccurrenceTypeGraph* element inherits the attributes and/or associations of:

- 1183 • **SCE SCEElement** (see the **SCE** specification for more information).

1184 **Properties**

1185 The following table presents the additional attributes and/or associations for *OccurrenceTypeGraph*:

Table 24. OccurrenceTypeGraph Attributes and/or Associations

Property/Association	Description
occurrenceNode : OccurrenceGraphNode [1..*]	The <i>OccurrenceGraphNode</i> s included in the <i>OccurrenceTypeGraph</i> .
transition : OccurrenceGraphTransition [0..*]	The <i>OccurrenceTypeTransitions</i> included in the <i>OccurrenceTypeGraph</i> .

1186

1187 **8.2.19 OccurrenceTypeUsage Node**

1188 A kind of *OccurrenceGraphNode* that identifies the usage of an *OccurrenceType* in an *OccurrenceTypeGraph*.

1189 **Generalizations**

1190 The *OccurrenceTypeUsage Node* element inherits the attributes and/or associations of:

- 1191 • *OccurrenceGraphNode* (see the section entitled “[OccurrenceGraphNode](#)” for more information).

1192 **Properties**

1193 The following table presents the additional attributes and/or associations for *OccurrenceTypeUsage Node*:

Table 25. OccurrenceTypeUsage Node Attributes and/or Associations

Property/Association	Description
occurrenceType : OccurrenceType [1]	The <i>OccurrenceType</i> that the node represents.

1194

1195 **8.2.20 PPMNRelationshipKind**

1196 A class indicating the kind of relationship between two **PPMN** elements.

1197 **Generalizations**

1198 The *PPMNRelationshipKind* element inherits the attributes and/or associations of:

- 1199 • *RelationshipKind* (see the SCE specification for more information).

1200 **Properties**

1201 The *PPMNRelationshipKind* element does not have any additional attributes and/or associations.

1202 **8.2.21 Rule**

1203 A condition that can be evaluated in some context as being either True or False.

1204 **Generalizations**

1205 The *Rule* element does not inherit any attributes or associations of from another element.

1206 **Properties**

1207 The *Rule* element does not have any additional attributes and/or associations.

1208 **8.3 Pedigree**

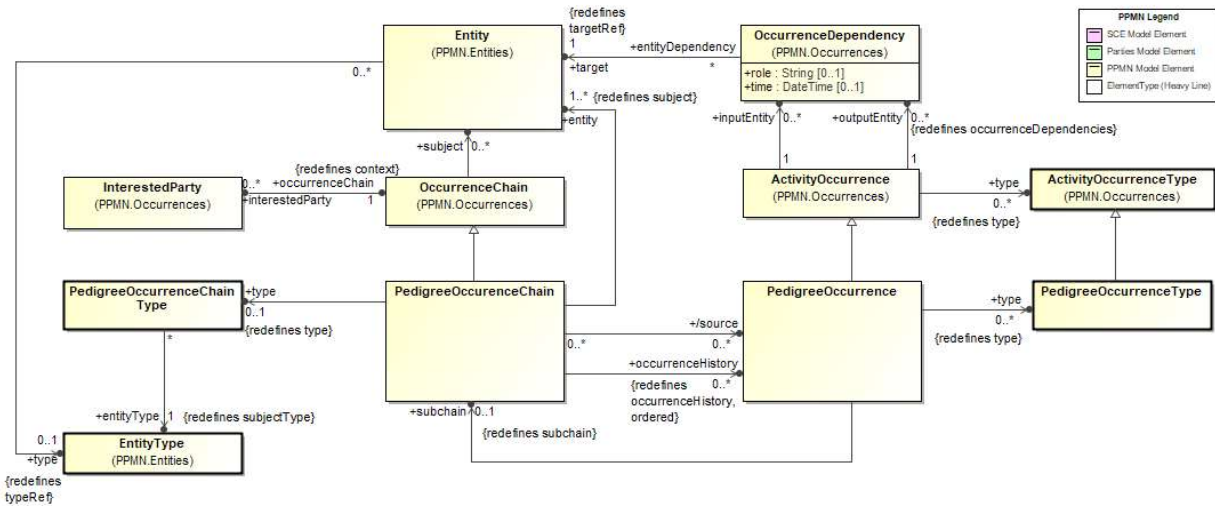
1209 The *Pedigree* package contains elements necessary to capture the lineage or pedigree of *Entities* along with the
 1210 *Occurrences* that resulted in that lineage.

1211 **8.3.1 Pedigree Occurrences**

1212 The *Pedigree Occurrences* package contains elements necessary to capture the events or activities, i.e. the
 1213 *Occurrences*, that affect the lifecycle of *Entities*.

1214 *PedigreeChains* record the actual events or processes that happen as part of the history of an entity of interest.
 1215 *PedigreeChains* also record a reference to the entity to which the *Occurrences* relate through the `entity` property.
 1216 Conceptually, *PedigreeChains* are “instances” of *PedigreeChainTypes* and as such may be governed by the relations
 1217 established in the *PedigreeChainType*. These occurrences represent actual events or activities in the history of one or
 1218 more *Entities* that are of interest to some *Party*.

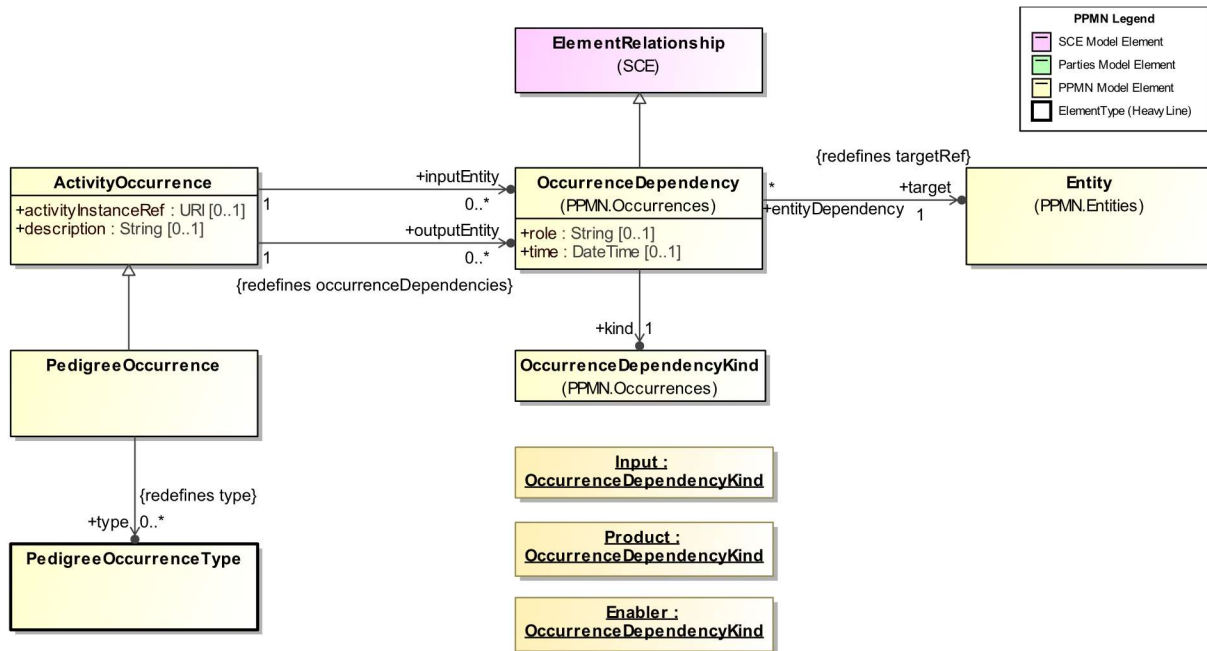
1219
 1220



1221
 1222 **Figure 17: Pedigree Occurrence Chains - Overview**

1223 *PedigreeOccurrence* is a kind of *ActivityOccurrence* that affects the lifecycle of one or more *Entities*.

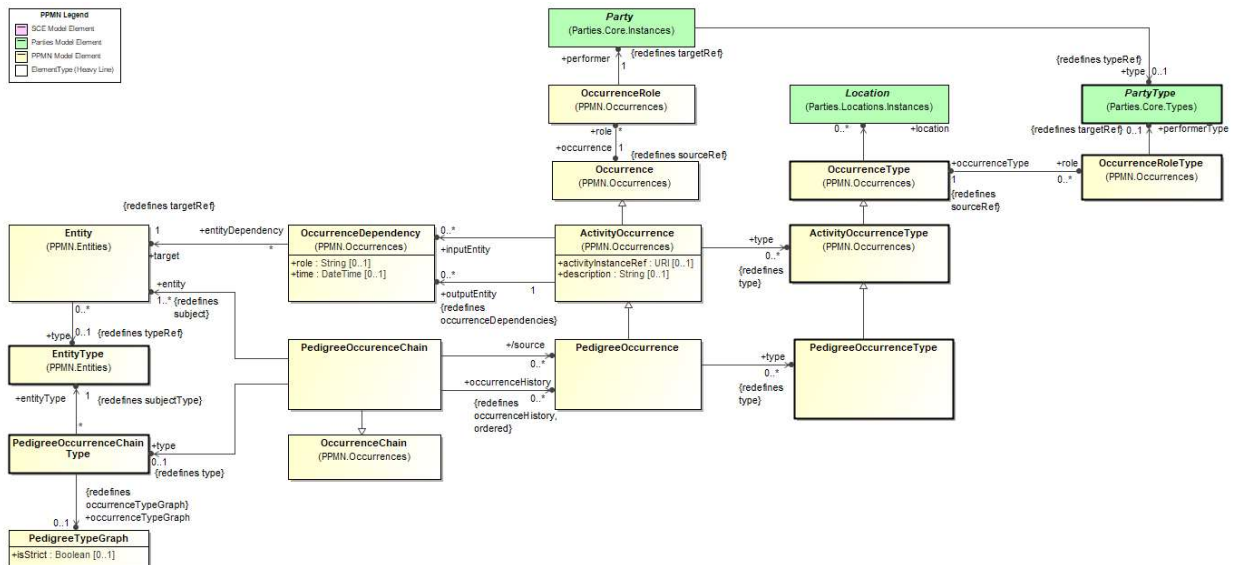
1224
 1225
 1226



1227
1228 **Figure 18: Pedigree Occurrences**

1229 *PedigreeOccurrenceChains* record the actual *PedigreeOccurrences* that happen as part of the
1230 *occurrenceHistory* property, an ordered list. *PedigreeOccurrenceChains* include a reference to the *Entity* or
1231 *Entities* to which the *Occurrences* relate through the *entity* property. *PedigreeOccurrenceChains* are essentially
1232 instances of *PedigreeChainTypes* and as such are governed by the relations established in the *PedigreeChainType*.
1233 *PedigreeOccurrences* are instances of *PedigreeOccurrenceTypes*. These occurrences represent actual events or
1234 activities in the history of an *Entity* that is of interest to some *Party*.

1235
1236



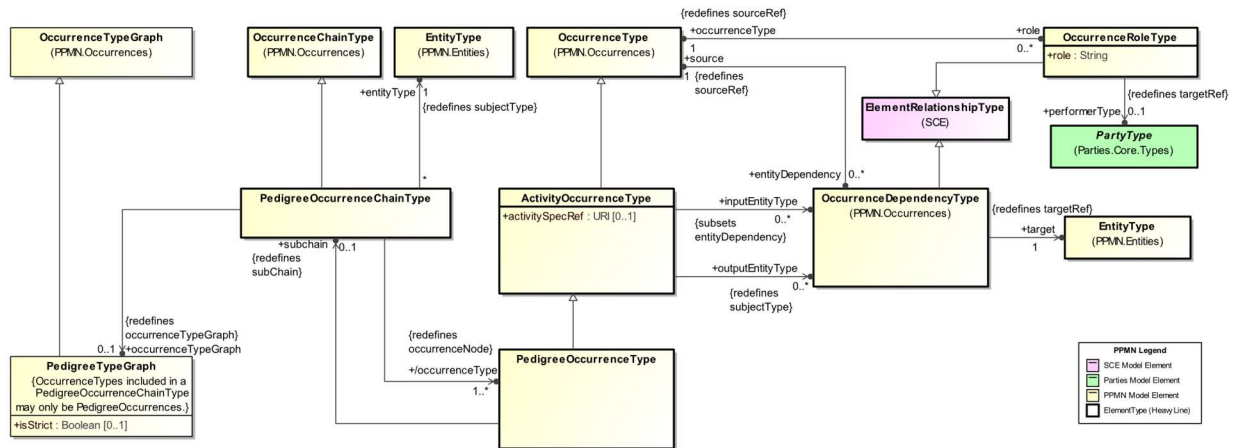
1237
1238 **Figure 19: Pedigree Occurrence Chains**

1239 *PedigreeChainType* supports the definition of types of occurrences expected in *PedigreeChains* related to an

1240 *EntityType* in which some *Party* is interested. *PedigreeChainTypes* are modeled as simple graphs so that rich
 1241 definitions of entity lifecycles can be created (though they are not required). The model also supports simple
 1242 definitions of valid *PedigreeOccurrenceTypes* or no lifecycle definitions at all.

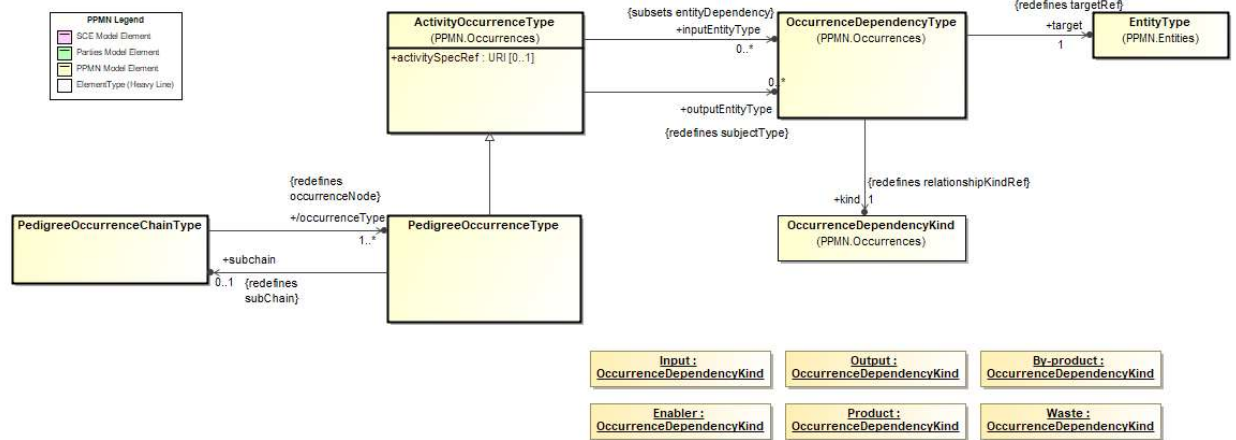
1243 *InterestedParty* is a kind of *PartyRole* that indicates that a *Party* has some interest in the with respect to an entity.
 1244 *PedigreeChainTypes* are specific to one or more *InterestedParties*. As an example, an automobile manufacturer
 1245 may be interested a set of occurrences related to the building of a car such as StartAssembly, InstallEngine,
 1246 PaintCar, TestCar, and ShipCar. A dealership on the other hand would likely be interested in tracking other events
 1247 such as BuildCar, ShipCar, ReceiveCar, and SellCar.

1248
 1249



1250
 1251 **Figure 20: Pedigree Occurrence Chain Type**

1252 *PedigreeChainType* supports the definition of types of occurrences expected in *PedigreeChains* related to an entity
 1253 type in which some *Party* is interested. *PedigreeChainTypes* are modeled as simple graphs so that rich definitions
 1254 of entity lifecycles can be created (though they are not required). The model also supports simple definitions of
 1255 valid *PedigreeOccurrenceTypes* or no lifecycle definitions at all.

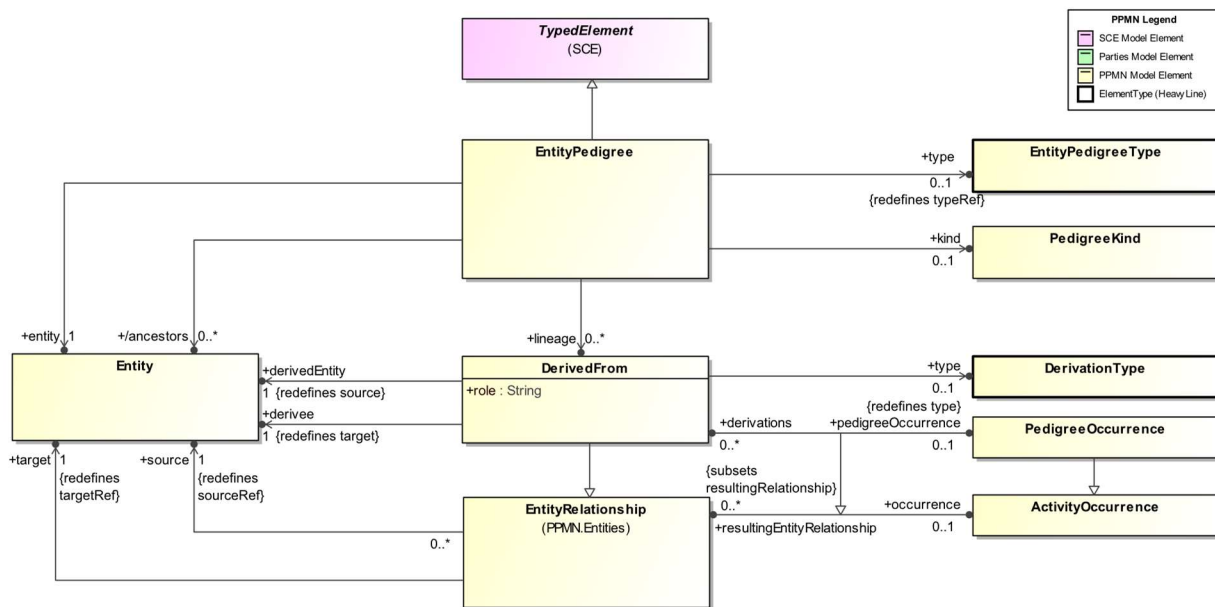


1256
 1257 **Figure 21: Pedigree Occurrence Types**

1258 The lineage of an *Entity*, herein referred to as its "pedigree" or "pedigree chain", is a lattice comprising *Entities* as
 1259 nodes and derivations (*DerivedFrom* relationships) as edges. Pedigree chains are created by *Occurrences* that result
 1260 in some number of *Entities* being used to create one or more new *Entities* or evolve one or more existing *Entities*.
 1261 These *Occurrences* result one or more derivations between "input" *Entities* and the "output" *Entities*.

1262 Given that a particular *Occurrence* may encapsulate a sub-chain of *Occurrences*, derivations may involve a series of
 1263 one or more *Occurrences* that create or evolve an entity of interest into another. In these cases, the *Occurrences* that
 1264 comprise the sub-chain would also potentially result in derivations that would combine to result in the derivations of
 1265 the containing *Occurrence*. As stated above, derivations are noted in the form of a *DerivedFrom* relationship
 1266 between one *Entity* that is the *derivationSource* and another that is the *derivedEntity*. The derivation
 1267 may be related to an *ActivityOccurrence* that caused the transformation. This may specifically be a
 1268 *PedigreeOccurrence* but may also be a more general *ActivityOccurrence*. Often, the activities that result in
 1269 derivations are not easily tracked or quantified and so just noting the *Entity* or *Entities* from which the entity of
 1270 interest is derived is all that is necessary or in some cases even possible.

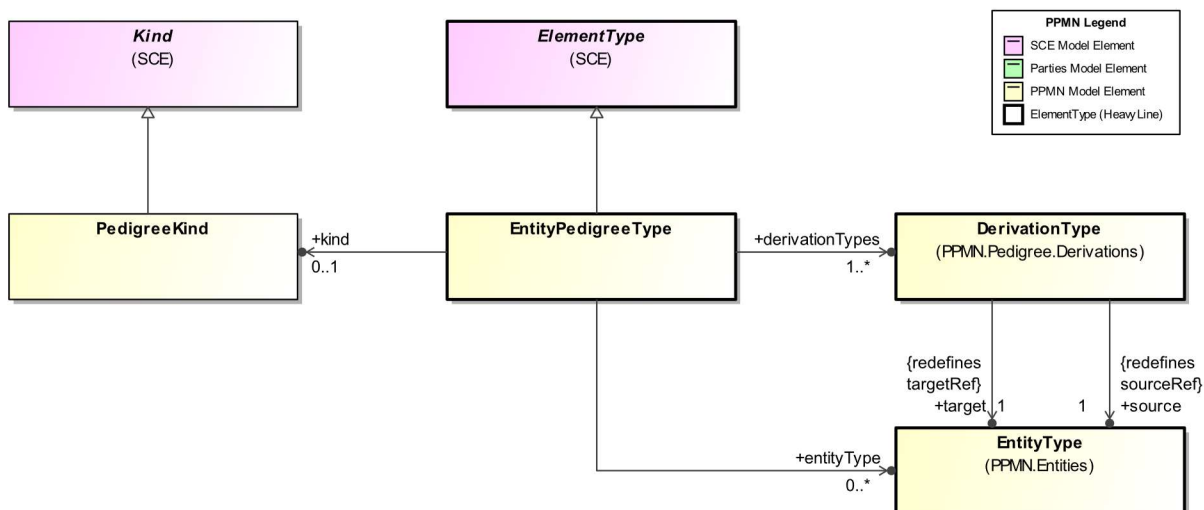
1271
 1272



1273
 1274 **Figure 22: Pedigree “Chains”**

1275 *EntityPedigreeTypes* support the ability to define different kinds of pedigree or lineage of particular kinds of
 1276 *Entities*. This is accomplished by specifying the *EntityTypes* and the types of derivations between them. Derivations
 1277 involve a series of one or more *Occurrences* that create or evolve an entity of interest into another. Derivations are
 1278 noted in the form of a *DerivedFrom* relationship between one *Entity* that is the *derivationSource* and another
 1279 that is the *derivedEntity*. To specify the expected type of derivation between two *Entities* PPMN provides the
 1280 *DerivationType* element. In addition, PPMN specifies three types of derivation: revision, quotation, and sourcing.
 1281 (See section 8.3.2, below, for further explanation.)

1282



1283
1284 **Figure 23: Pedigree Chains Types**

1285 **8.3.1.1 EntityPedigree**

1286 The class representing the pedigree or lineage of an *Entity*.

1287 **Generalizations**

1288 The *EntityPedigree* element inherits the attributes and/or associations of:

- 1289 • **SCE TypedElement** (see the **SCE** specification for more information).

1290 **Properties**

1291 The following table presents the additional attributes and/or associations for *EntityPedigree*:

Table 26. EntityPedigree Attributes and/or Associations

Property/Association	Description
ancestors : Entity [0..*]	The set of <i>Entities</i> from which the <i>entity</i> was derived. This is a derived property determined by walking the set of <i>DerivedFrom</i> relationships from <i>Entity</i> to <i>Entity</i> until the end of each path of the directed acyclic graph (DAG).
entity : Entity [1]	The <i>Entity</i> to which the pedigree applies.
kind : PedigreeKind [0..1]	A specification of the kind of pedigree the <i>EntityPedigree</i> captures.
lineage : <i>DerivedFrom</i> [0..*]	The set of <i>DerivedFrom</i> relationships that led to the creation and/or evolution of the <i>entity</i> . The combination of the <i>DerivedFrom</i> relationships and the <i>Entities</i> at their ends must form a directed acyclic graph (DAG) starting with the <i>entity</i> and ending with <i>Entities</i> that were created by some <i>Occurrence</i> or whose origin is unknown.
type : EntityPedigreeType [0..1]	

1292

1293 **8.3.1.2 EntityPedigreeType**

1294 The type of pedigree or lineage between *Entities* of type *entityType*.

1295 **Generalizations**

1296 The *EntityPedigreeType* element inherits the attributes and/or associations of:

- 1297 • *SCE ElementType* (see the **SCE** specification for more information).

1298 **Properties**

1299 The following table presents the additional attributes and/or associations for *EntityPedigreeType*:

Table 27. EntityPedigreeType Attributes and/or Associations

Property/Association	Description
derivationTypes : DerivationType [1..*]	The types of derivations that are captured by the EntityPedigreeType.
entityType : EntityType [0..*]	The <i>EntityType(s)</i> to which the EntityPedigreeType applies.
kind : PedigreeKind [0..1]	The kind of entity pedigree or lineage the EntityPedigreeType represents.

1300

1301 **8.3.1.3 PedigreeKind**

1302 A class that indicates the kind of pedigree or lineage between *Entities*.

1303 **Generalizations**

1304 The *PedigreeKind* element inherits the attributes and/or associations of:

- 1305 • *Kind* (see the SCE specification for more information).

1306 **Properties**

1307 The *PedigreeKind* element does not have any additional attributes and/or associations.

1308 **8.3.1.4 PedigreeOccurrenceChain**

1309 A succession of *PedigreeOccurrences* that have happened in the life of an entity that is of interest to some *Party*.

1310 **Generalizations**

1311 The *PedigreeOccurrenceChain* element inherits the attributes and/or associations of:

- 1312 • *OccurrenceChain* (see the section entitled "[OccurrenceChain](#)" for more information).

1313 **Properties**

1314 The following table presents the additional attributes and/or associations for *PedigreeOccurrenceChain*:

Table 28. PedigreeOccurrenceChain Attributes and/or Associations

Property/Association	Description
entity : Entity [1]	The <i>Entity</i> or <i>Entities</i> for which the <i>PedigreeChain</i> represents the history of <i>PedigreeOccurrences</i> .

occurrenceHistory : PedigreeOccurrence [0..*]	A sequence of <i>PedigreeOccurrences</i> that represent the history of <i>PedigreeOccurrences</i> that took place with respect to a particular entity.
source : PedigreeOccurrence [0..*]	The <i>PedigreeOccurrences</i> that were the original sources for ancestor entities of the subject entity.
type : PedigreeOccurrenceChainType [0..1]	The type of the <i>PedigreeChain</i> .

1315

1316 8.3.1.5 PedigreeOccurrence

1317 An *ActivityOccurrence* in the lifecycle of an entity related to the source or evolution of that entity that is of interest
1318 to some *Party*.

1319 Generalizations

1320 The *PedigreeOccurrence* element inherits the attributes and/or associations of:

- 1321 • *ActivityOccurrence* (see the section entitled “[ActivityOccurrence](#)” for more information).

1322 Properties

1323 The following table presents the additional attributes and/or associations for *PedigreeOccurrence*:

Table 29. PedigreeOccurrence Attributes and/or Associations

Property/Association	Description
derivations : DerivedFrom [0..*]	Derivations created as a result of the <i>PedigreeOccurrence</i> .
subchain : PedigreeOccurrenceChain [0..1]	A sequence of <i>PedigreeOccurrences</i> that take the <i>inputEntity</i> of the <i>PedigreeOccurrence</i> and transform them into the <i>outputEntity</i> of the <i>PedigreeOccurrence</i> and are encapsulated by the <i>PedigreeOccurrence</i> .
type : PedigreeOccurrenceType [0..*]	The type of the <i>PedigreeOccurrence</i> .

1324

1325 8.3.1.6 PedigreeOccurrenceChainType

1326 A kind of *OccurrenceChainType* that captures the expected *OccurrenceTypes*, *PedigreeOccurrenceTypes*, that result
1327 in the creation or evolution of particular types of entities.

1328 Generalizations

1329 The *PedigreeOccurrenceChainType* element inherits the attributes and/or associations of:

- 1330 • *OccurrenceChainType* (see the section entitled “[OccurrenceChainType](#)” for more information).

1331 Properties

1332 The following table presents the additional attributes and/or associations for *PedigreeOccurrenceChainType*:

Table 30. PedigreeOccurrenceChainType Attributes and/or Associations

Property/Association	Description
entityType : EntityType [1]	The type of entity expected as a result of the chain.
occurrenceType : PedigreeOccurrenceType [1..*]	The <code>occurrenceType</code> derived property is based on the series of relationships between from <i>PedigreeChainType</i> through other classes to <i>PedigreeOccurrenceType</i> : <code>OccurrenceChainType.occurrenceTypeGraph.occurrenceNode.occurrenceType</code> .
occurrenceTypeGraph : PedigreeTypeGraph [0..1]	A graph of <i>PedigreeOccurrenceTypes</i> that are expected in the lifecycle of a particular type of entity.

1333

1334 8.3.1.7 PedigreeOccurrenceType

1335 An expected type of *PedigreeOccurrence* in the lifecycle of an entity that is of interest to some *Party*.

1336 Generalizations

1337 The *PedigreeOccurrenceType* element inherits the attributes and/or associations of:

- 1338 • *ActivityOccurrenceType* (see the section entitled “[ActivityOccurrenceType](#)” for more information).

1339 Properties

1340 The following table presents the additional attributes and/or associations for *PedigreeOccurrenceType*:

Table 31. PedigreeOccurrenceType Attributes and/or Associations

Property/Association	Description
subchain : PedigreeOccurrenceChainType [0..1]	A <i>PedigreeChainType</i> that is encapsulated within the <i>PedigreeOccurrenceType</i> to create a "subchain".

1341

1342 8.3.1.8 PedigreeTypeGraph

1343 A *PedigreeChainType* is a specification for the types of *Occurrences* that happen with respect to an entity that are of
 1344 interest to a particular *Party*. If the property `isStrict=True`, then only the *Occurrences* of type
 1345 *PedigreeOccurrenceType* will be included in related *PedigreeChains*. If the property is `False` then *Occurrences* of
 1346 other types may be included in related *PedigreeChains*.

1347 Generalizations

1348 The *PedigreeTypeGraph* element inherits the attributes and/or associations of:

- 1349 • *OccurrenceTypeGraph* (see the section entitled “[OccurrenceTypeGraph](#)” for more information).

1350 Properties

1351 The following table presents the additional attributes and/or associations for *PedigreeTypeGraph*:

Table 32. PedigreeTypeGraph Attributes and/or Associations

Property/Association	Description
isStrict : Boolean [0..1]	A boolean that specifies whether or not adherence to the <i>PedigreeTypeGraph</i> is strict or not. If the value is True, then only the <i>Occurrences</i> of type <i>PedigreeOccurrenceType</i> will be included in related <i>PedigreeChains</i> . If the value is False then <i>Occurrences</i> of other types may be included in related <i>PedigreeChains</i> .

1352

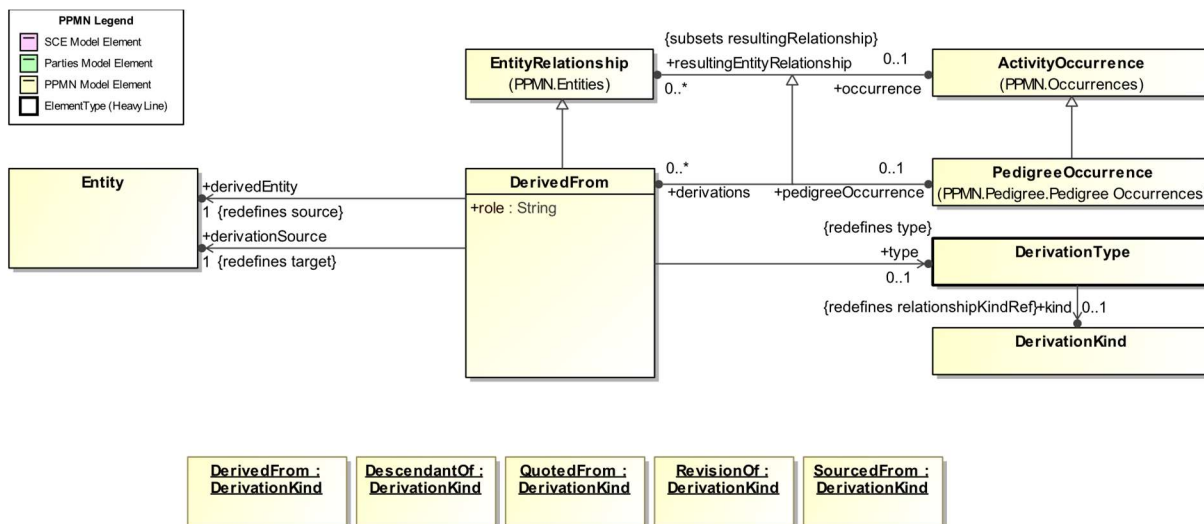
1353 8.3.2 Derivations

1354 The Derivations package contains elements that capture the derivation relationships between Entities. These
 1355 elements, in conjunction with Entities, capture the lineage or pedigree of Entities.

1356 Derivations capture the lineal relationships between Entities or Entity Snapshots. Derivations are noted in the form
 1357 of a *DerivedFrom* relationship, or one of its specializations, between one *Entity* that is the *derivationSource*
 1358 and another that is the *derivedEntity*. A derivation may be the result of a general *ActivityOccurrence* or
 1359 specifically a *PedigreeOccurrence*. Please note that the activities that result in derivations are not always easily
 1360 tracked or quantified and so just noting the entity from which the entity of interest is derived is all that is possible.

1361 **PPMN** specifies four types of derivation: revision, quotation, sourcing, and descendant. Revision is captured by
 1362 setting the *kind* attribute of the *DerivationType* to *RevisionOf*. *RevisionOf* is an instance of *DerivedKind* and is
 1363 used in situations where one entity is a revision of another as in a report or publication. Quotation is captured by
 1364 setting the *kind* attribute of the *DerivationType* to *QuotedFrom* and specifies that part of all of one entity is a
 1365 repeat of part or all of another entity, presumably some textual report or publication. The quotation may or may not
 1366 be by the original author of the quoted entity. *SourcedFrom* is captured by setting the *kind* attribute of the
 1367 *DerivationType* to *SourcedFrom* and specifies that the entity of interest came from another entity which was in turn
 1368 produced by some party potentially with some special experience or knowledge. Finally, *DescendantOf* is captured
 1369 by setting the *kind* attribute of the *DerivationType* to *DescendantOf* and indicates that the entity of interest is a
 1370 descendant of the ancestor *Entity*.

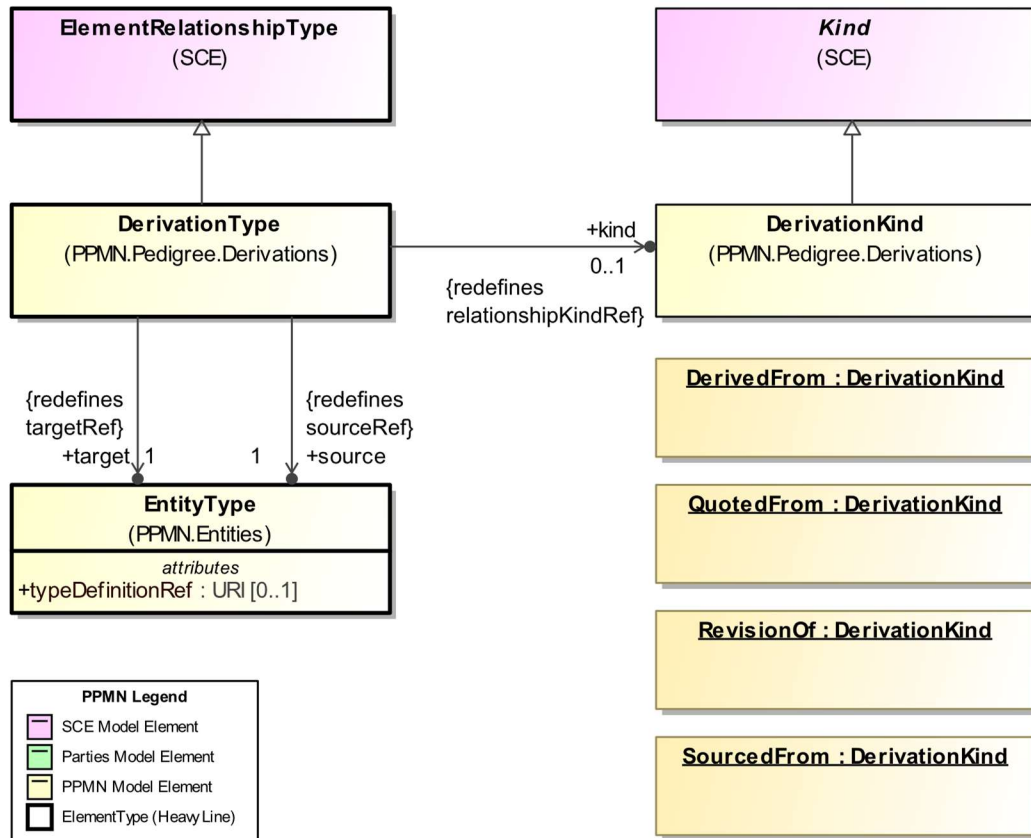
1371



1372

1373 **Figure 24: Derivations**

1374 DerivationTypes support the definition of the expected kinds of derivations that might result in the generation of one
 1375 *EntityType* from or more others.
 1376
 1377



1378
 1379 **Figure 25: Derivation Types**

1380 **8.3.2.1 DerivationKind**

1381 A class indicating the kind of derivation that exists between two *Entities*.

1382 **Generalizations**

1383 The *DerivationKind* element inherits the attributes and/or associations of:

- 1384 • *Kind* (see the **SCE** specification for more information).

1385 **Properties**

1386 The *DerivationKind* element does not have any additional attributes and/or associations.

1387 **8.3.2.2 DerivationType**

1388 A kind of *ElementRelationship* that captures the type of derivation between one particular *EntityType* and another.

1389 **Generalizations**

1390 The *DerivationType* element inherits the attributes and/or associations of:

- 1391 • *ElementRelationshipType* (see the **SCE** specification for more information).

1392 **Properties**

1393 The following table presents the additional attributes and/or associations for *DerivationType*:

Table 33. DerivationType Attributes and/or Associations

Property/Association	Description
kind : DerivationKind [0..1]	A description of the kind of derivation that produced one <i>Entity</i> from another. See <i>DeivationKind</i> for more details.
source : EntityType [1]	The <i>EntityType</i> that was derived.
target : EntityType [1]	The <i>EntityType</i> from which the <i>source EntityType</i> was derived.

1394

1395 **8.3.2.3 DerivedFrom**

1396 Derivations are noted in the form of a *DerivedFrom* relationship between one *Entity* that is the
1397 *derivationSource* and another that is the *derivedEntity*. The derivation may be related to an
1398 *ActivityOccurrence* that specifies the particular *Occurrence* that caused the transformation. Often, the activities that
1399 result in derivations are not easily tracked or quantified and so just noting the entity from which the entity of interest
1400 is derived is all that is necessary.

1401 **Generalizations**

1402 The *DerivedFrom* element inherits the attributes and/or associations of:

- 1403 • *EntityRelationship* (see the section entitled “[EntityRelationship](#)” for more information).

1404 **Properties**

1405 The following table presents the additional attributes and/or associations for *DerivedFrom*:

Table 34. DerivedFrom Attributes and/or Associations

Property/Association	Description
derivedEntity : Entity [1]	The <i>Entity</i> that was derived.
derivationSource : Entity [1]	The <i>Entity</i> from which the <i>derivedEntity</i> was derived.
pedigreeOccurrence : PedigreeOccurrence [0..1]	The <i>PedigreeOccurrence</i> that resulted in the derivation.
role : String []	A string that captures the role in the <i>derivationOccurrence</i> that produced the element.
type : DerivationType [0..1]	The type of derivation.

1406

1407 **8.3.2.4 DescendantOf**

1408 *DescendantOf* is a specialization of *DerivedFrom* that identifies that the entity of interest is a descendant of another
1409 *Entity*.

1410 **Generalizations**

1411 The *DescendantOf* element inherits the attributes and/or associations of:

- 1412 • *DerivedFrom* (see the section entitled “[DerivedFrom](#)” for more information).

1413 **Properties**

1414 The following table presents the additional attributes and/or associations for *DescendantOf*:

Table 35. DescendantOf Attributes and/or Associations

Property/Association	Description
ancestor : Entity [1]	The ancestor <i>Entity</i> .
descendant : Entity [1]	The descendant <i>Entity</i> .

1415

1416 **8.3.2.5 QuotedFrom**

1417 Quotation is captured by the *QuotedFrom* specialization of *DerivedFrom* and specifies that part of all of one entity
1418 is a repeat of part or all of another entity, presumably some textual report or publication. The quotation may or may
1419 not be by the original author of the quoted entity.

1420 **Generalizations**

1421 The *QuotedFrom* element inherits the attributes and/or associations of:

- 1422 • *DerivedFrom* (see the section entitled “[DerivedFrom](#)” for more information).

1423 **Properties**

1424 The following table presents the additional attributes and/or associations for *QuotedFrom*:

Table 36. QuotedFrom Attributes and/or Associations

Property/Association	Description
quotation : Entity [1]	The element that is the quotation.
quotedEntity : Entity [1]	The quoted element.

1425

1426 **8.3.2.6 RevisionOf**

1427 Revision is captured in the form of the *RevisionOf* relationship. *RevisionOf* is a specialization of *DerivedFrom* and
1428 is used in situations where one entity is a revision of another as in a report or publication.

1429 **Generalizations**

1430 The *RevisionOf* element inherits the attributes and/or associations of:

- 1431 • *DerivedFrom* (see the section entitled “[DerivedFrom](#)” for more information).

1432 **Properties**

1433 The following table presents the additional attributes and/or associations for *RevisionOf*:

Table 37. RevisionOf Attributes and/or Associations

Property/Association	Description
revisedEntity : Entity [1]	The revised element.
revision : Entity [1]	The result of the revision.

1434

1435 8.3.2.7 SourcedFrom

1436 *SourcedFrom* is a specialization of *DerivedFrom* that identifies that the entity of interest came from another entity
1437 which was in turn produced by some party potentially with some special experience or knowledge.

1438 Generalizations

1439 The *SourcedFrom* element inherits the attributes and/or associations of:

- 1440 • *DerivedFrom* (see the section entitled "[DerivedFrom](#)" for more information).

1441 Properties

1442 The following table presents the additional attributes and/or associations for *SourcedFrom*:

Table 38. SourcedFrom Attributes and/or Associations

Property/Association	Description
sourcedEntity : Entity [1]	The sourced element.
sourceEntity : Entity [1]	The entity from which the <code>sourcedEntity</code> was sourced.

1443

1444 8.4 Provenance

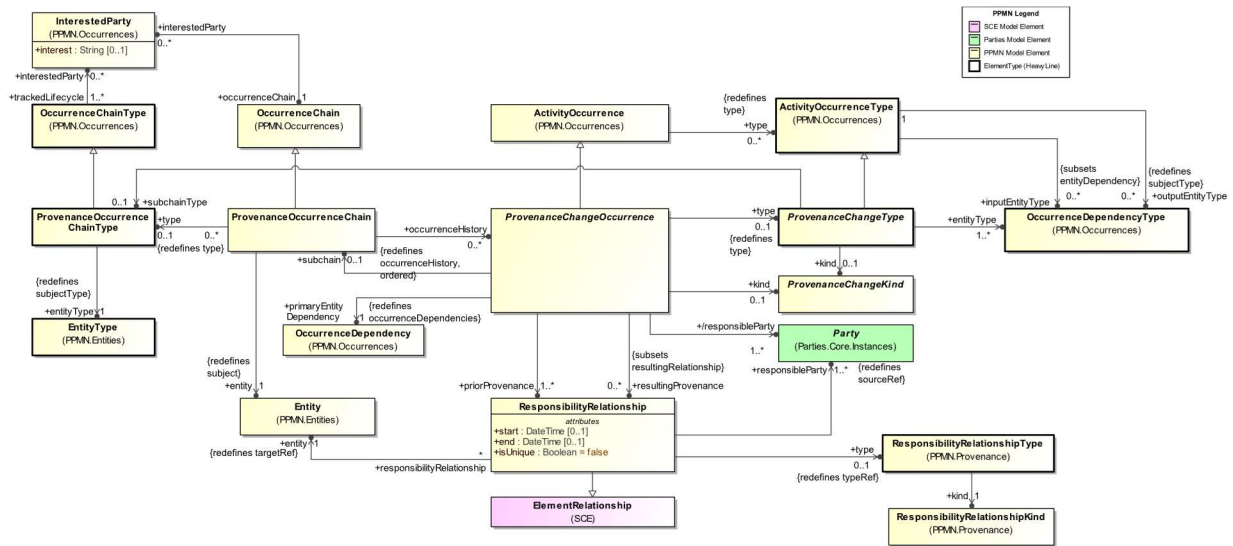
1445 The Provenance package contains elements related to the notion of the ownership and custody of entities of interest.
1446 This includes the *Occurrences* that result in changes in the ownership or custody of those entities of interest.

1447 *ProvenanceOccurrences* are specializations of *Occurrence* related to changes in ownership or custody of an entity.
1448 *ProvenanceOccurrences* are instances of *ProvenanceOccurrenceType* or one of its specializations. Similar to
1449 *OccurrenceType*, *ProvenanceOccurrenceType* is a specification of "expected" *ProvenanceOccurrences*. They
1450 capture the *Parties* expected to be involved in the instances. Expected types of entities to which the occurrences
1451 refer are noted through the `entityType` property.

1452 A *ProvenanceChain* records the provenance-related events that happen as part of the lifecycle of an entity. These
1453 events are recorded as part of the `occurrenceHistory` property, an ordered list of *ProvenanceOccurrences*. A
1454 *ProvenanceChain* also records a reference to the entity to which the *Occurrences* relate through the `entity`
1455 property. *ProvenanceChains* are essentially instances of *ProvenanceChainTypes* and as such are governed by the
1456 relations established in the *ProvenanceChainType*. If the *ProvenanceChainType* `isStrict` property is set to
1457 "True" then the types of occurrences maintained in the *ProvenanceChain* are constrained to those included in the
1458 *ProvenanceChainType*.

1459

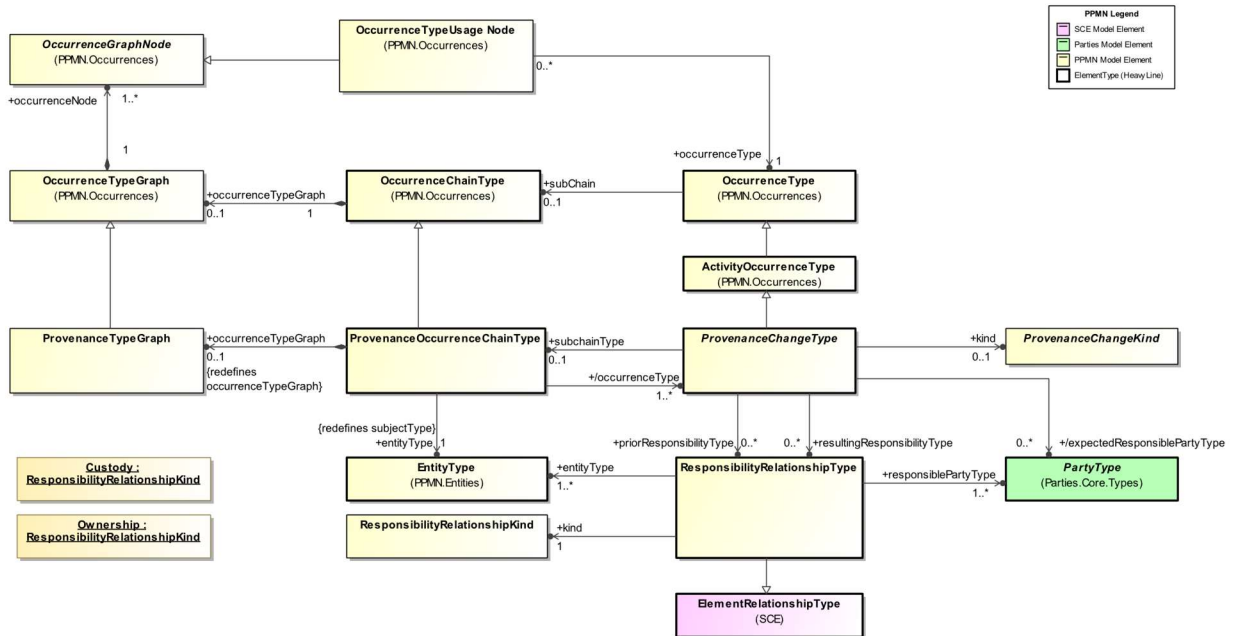
1460



1461
1462 **Figure 26: Provenance Occurrence Chains**

1463 *ProvenanceChains*, *ProvenanceChainTypes*, *ProvenanceOccurrences*, and *ProvenanceOccurrenceTypes* follow the
1464 same pattern that PPMN establishes for *Occurrences*. This pattern supports the "nesting" of *ProvenanceChains*
1465 within *ProvenanceOccurrences*. This pattern allows for encapsulation of parts of a chain where the details of the
1466 *ProvenanceOccurrences* of that part of a larger chain are either not known initially or are not deemed important in
1467 some context.

1468
1469



1470
1471 **Figure 27: Provenance Occurrence Chain Types**

1472 In addition to tracking changes in ownership or custody for an entity of interest over time, stakeholders also require
1473 the ability to make direct statements about who owns or has custody of an entity at a particular point in time. The
1474 *Ownership* and *Custody* classes provide this capability. Both *Ownership* and *Custody* specializations of

1475 *ResponsibilityRelationship* and, as such, capture the *Party* that owns or has custody of, respectively, a particular
 1476 *Entity* for a particular period of time. These provenance "records" can either be maintained in real time or generated
 1477 based on *Occurrences* that have been tracked for an entity.

1478

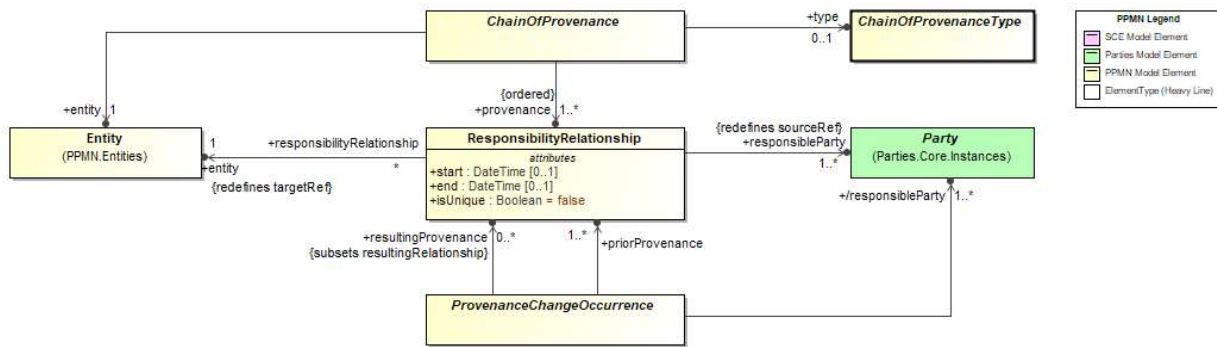
1479

1480

1481 **Figure 28: Provenance "Records"**

1482 TBD.

1483



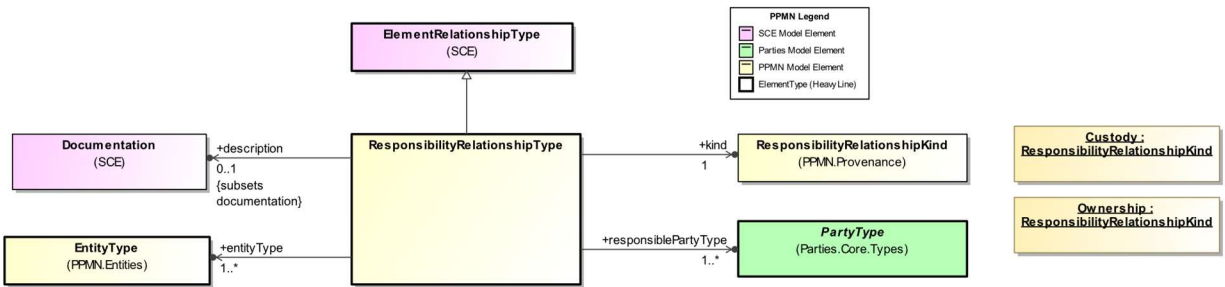
1484

1485 **Figure 29: Chain of Provenance**

1486 TBD.

1487

1488

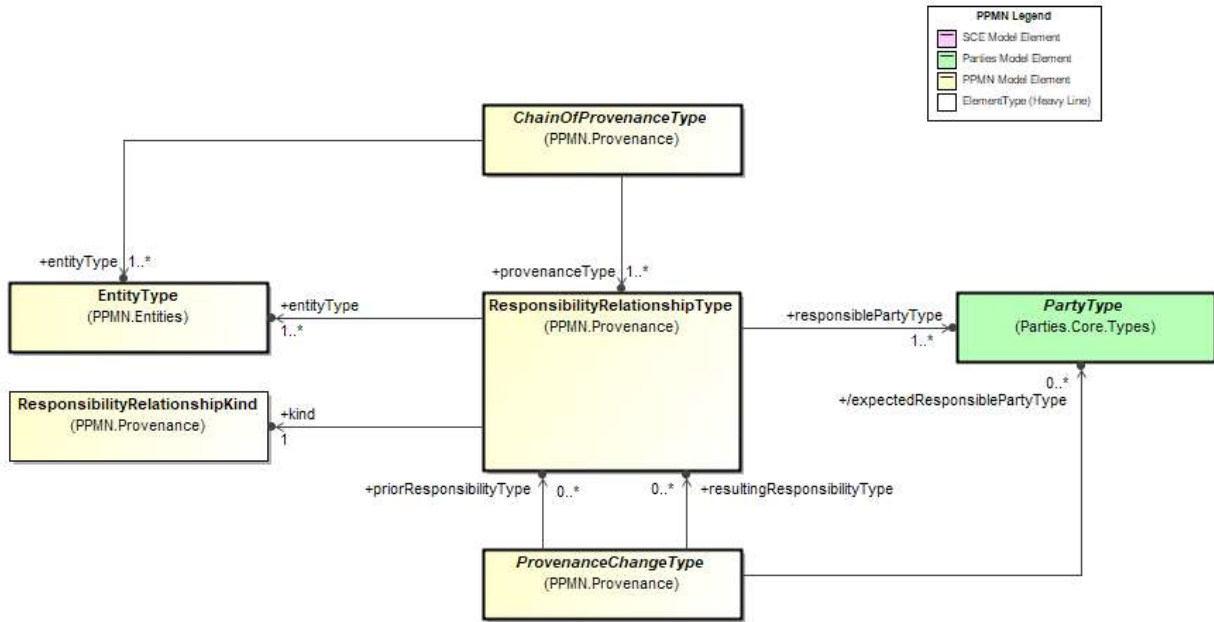


1489

1490 **Figure 30: Provenance Record Types**

1491 TBD.

1492



1493
1494 **Figure 31: Chain of Provenance Types**

1495 **8.4.1 ChainOfProvenance**

1496 An ordered set of *ResponsibilityRelationships* that captures the provenance of a particular entity over the course of
1497 its lifecycle.

1498 **Generalizations**

1499 The *ChainOfProvenance* element does not inherit any attributes or associations of from another element.

1500 **Properties**

1501 The following table presents the additional attributes and/or associations for *ChainOfProvenance*:

Table 39. ChainOfProvenance Attributes and/or Associations

Property/Association	Description
entity : Entity [1]	The entity to which the <i>ChainOfProvenance</i> refers.
provenance : ResponsibilityRelationship [1..*]	A set of ResponsibilityRelationships related to the provenance of an entity.
type : ChainOfProvenanceType [0..1]	The type of the <i>ChainOfProvenance</i> .

1502
1503 **8.4.2 ChainOfProvenanceType**
1504 An *ElementType* that specifies a set of expected provenance chains (*ChainOfProvenance*) that capture an ordered set
1505 of *ResponsibilityRelationships* of type *ResponsibilityRelationshipType*.

1506 **Generalizations**

1507 The *ChainOfProvenanceType* element does not inherit any attributes or associations of from another element.

1508 **Properties**

1509 The following table presents the additional attributes and/or associations for *ChainOfProvenanceType*:

Table 40. ChainOfProvenanceType Attributes and/or Associations

Property/Association	Description
entityType : EntityType [1..*]	The <i>EntityType</i> for which the <i>ChainOfProvenanceType</i> applies.
provenanceType : ResponsibilityRelationshipType [1..*]	The type of the responsibility relationships expected to be included in provenance chains of type <i>ChainOfProvenanceType</i> .

1510

1511 **8.4.3 ProvenanceChangeKind**

1512 A class indicating the kind of provenance change that is expected.

1513 **Generalizations**

1514 The *ProvenanceChangeKind* element does not inherit any attributes or associations of from another element.

1515 **Properties**

1516 The *ProvenanceChangeKind* element does not have any additional attributes and/or associations.

1517 **8.4.4 ProvenanceChangeOccurrence**

1518 An *Occurrence* in the lifecycle of an entity related to the custody and/or ownership of that entity.

1519 **Generalizations**

1520 The *ProvenanceChangeOccurrence* element inherits the attributes and/or associations of:

- 1521 • *ActivityOccurrence* (see the section entitled “[ActivityOccurrence](#)” for more information).

1522 **Properties**

1523 The following table presents the additional attributes and/or associations for *ProvenanceChangeOccurrence*:

Table 41. ProvenanceChangeOccurrence Attributes and/or Associations

Property/Association	Description
kind : ProvenanceChangeKind [0..1]	A reference to a definition of the specific kind of provenance change.
primaryEntityDependency : OccurrenceDependency [1]	The <i>OccurrenceDependency</i> whose <i>target</i> is the <i>Entity</i> to which the <i>ProvenanceOccurrence</i> applies.
priorProvenance : ResponsibilityRelationship [1..*]	The <i>ResponsibilityRelationships</i> prior to the <i>ProvenanceChangeOccurrence</i> .
responsibleParty : Party [1..*]	The <i>Party</i> that has responsibility for the entity as a result of the <i>ProvenanceOccurrence</i> .

resultingProvenance : ResponsibilityRelationship [0..*]	The <i>ResponsibilityRelationships</i> that result from the <i>ProvenanceChangeOccurrence</i> .
subchain : ProvenanceOccurrenceChain [0..1]	A <i>ProvenanceChain</i> that is encapsulated by the <i>ProvenanceOccurrence</i> , essentially creating a "sub-chain".
type : ProvenanceChangeType [0..1]	The type of the <i>ProvenanceOccurrence</i> .

1524

1525 8.4.5 ProvenanceChangeType

1526 The type of a *ProvenanceOccurrence* in the lifecycle of an entity that is of interest to some *Party*.

1527 Generalizations

1528 The *ProvenanceChangeType* element inherits the attributes and/or associations of:

- 1529 • *ActivityOccurrenceType* (see the section entitled "[ActivityOccurrenceType](#)" for more information).

1530 Properties

1531 The following table presents the additional attributes and/or associations for *ProvenanceChangeType*:

Table 42. ProvenanceChangeType Attributes and/or Associations

Property/Association	Description
entityType : OccurrenceDependencyType [1..*]	A relationship to the expected type of entity involved in the <i>ProvenanceChangeType</i> .
expectedResponsiblePartyType : PartyType [0..*]	The <i>Party</i> that is expected to be responsible in some way for an <i>entity</i> of a particular type.
kind : ProvenanceChangeKind [0..1]	A reference to a definition of the specific kind of provenance change.
priorResponsibilityType : ResponsibilityRelationshipType [0..*]	The <i>ResponsibilityRelationshipType</i> expected to exist prior to occurrences of type <i>ProvenanceChangeType</i> .
resultingResponsibilityType : ResponsibilityRelationshipType [0..*]	The type of <i>ResponsibilityRelationships</i> expected as a result of the <i>ProvenanceChangeType</i> .
subchainType : ProvenanceOccurrenceChainType [0..1]	A <i>ProvenanceChainType</i> that is encapsulated within the <i>ProvenanceOccurrenceType</i> to create a "subchain".

1532

1533 8.4.6 ProvenanceOccurrenceChain

1534 A succession of *ProvenanceOccurrences* that have happened in the life of an entity that is of interest to some *Party*.

1535 Generalizations

1536 The *ProvenanceOccurrenceChain* element inherits the attributes and/or associations of:

- 1537 • *OccurrenceChain* (see the section entitled “[OccurrenceChain](#)” for more information).

1538 **Properties**

1539 The following table presents the additional attributes and/or associations for *ProvenanceOccurrenceChain*:

Table 43. ProvenanceOccurrenceChain Attributes and/or Associations

Property/Association	Description
entity : Entity [1]	The entity that is the subject of the <i>ProvenanceChain</i> .
occurrenceHistory : ProvenanceChangeOccurrence [0..*]	A set of <i>ProvenanceOccurrences</i> that comprise the chain.
type : ProvenanceOccurrenceChainType [0..1]	The type of the <i>ProvenanceChain</i> .

1540

1541 **8.4.7 ProvenanceOccurrenceChainType**

1542 A kind of *OccurrenceChainType* that captures a specification for a series of potential *ProvenanceOccurrences* that
1543 are expected in a particular context. A *ProvenanceChainType* captures this specification through the
1544 *occurrenceTypeGraph* property - a graph of *OccurrenceGraphNode*s and *OccurrenceTransitionType*s.

1545 **Generalizations**

1546 The *ProvenanceOccurrenceChainType* element inherits the attributes and/or associations of:

- 1547 • *OccurrenceChainType* (see the section entitled “[OccurrenceChainType](#)” for more information).

1548 **Properties**

1549 The following table presents the additional attributes and/or associations for *ProvenanceOccurrenceChainType*:

Table 44. ProvenanceOccurrenceChainType Attributes and/or Associations

Property/Association	Description
entityType : EntityType [1]	The subject of the <i>ProvenanceChainType</i> .
occurrenceType : ProvenanceChangeType [1..*]	A derived property that holds the set of <i>ProvenanceOccurrenceTypes</i> that represent the types of <i>ProvenanceOccurrences</i> expected to occur as part of <i>ProvenanceChains</i> that the <i>ProvenanceChainType</i> specifies.
occurrenceTypeGraph : ProvenanceTypeGraph [0..1]	A graph of <i>ProvenanceOccurrenceTypes</i> that specifies the sequencing of expected <i>ProvenanceOccurrences</i> in the lifecycle of an entity of interest to zero or more <i>InterestedParties</i> .

1550

1551 **8.4.8 ProvenanceTypeGraph**

1552 A specialized type of *OccurrenceTypeGraph* that captures the *ProvenanceOccurrenceTypes* that are expected in the
1553 lifecycle of one or more types of entities.

1554 **Generalizations**

1555 The *ProvenanceTypeGraph* element inherits the attributes and/or associations of:

- 1556 • *OccurrenceTypeGraph* (see the section entitled “[OccurrenceTypeGraph](#)” for more information).

1557 **Properties**

1558 The *ProvenanceTypeGraph* element does not have any additional attributes and/or associations.

1559 **8.4.9 ResponsibilityRelationship**

1560 A *ResponsibilityRelationship* is a kind of *ElementRelationship* that specifies a *Party* has some provenance-related
1561 responsibility for an entity for a particular period of time.

1562 **Generalizations**

1563 The *ResponsibilityRelationship* element inherits the attributes and/or associations of:

- 1564 • *ElementRelationship* (see the **SCE** specification for more information).

1565 **Properties**

1566 The following table presents the additional attributes and/or associations for *ResponsibilityRelationship*:

Table 45. ResponsibilityRelationship Attributes and/or Associations

Property/Association	Description
end : DateTime [0..1]	The date on which which a <i>Party</i> relinquishes the specified responsibilities with respect to a particular entity.
entity : Entity [1]	The entity for which a <i>Party</i> is responsible from either a custody or ownership perspective.
isUnique : Boolean [] default: false	A boolean that indicates whether or not the responsibility is unique.
kind : ResponsibilityRelationshipKind [0..1]	The kind of <i>ResponsibilityRelationship</i> between <i>PartyTypes</i> and <i>EntityTypes</i> in a given situation. See <i>ResponsibilityRelationshipKind</i> for more details.
responsibilityDescription : Documentation [0..1]	A textual description of the responsibility.
responsibleParty : Party [1..*]	The <i>Party</i> that is responsible from a provenance perspective for a particular entity.
start : DateTime [0..1]	The date on which a <i>Party</i> acquires the responsibilities with respect to a particular entity.
type : ResponsibilityRelationshipType [0..1]	The type of the <i>ResponsibilityRelationship</i> .

1567

1568 **8.4.10 ResponsibilityRelationshipKind**

1569 A class representing the kind of *ResponsibilityRelationship* between *Parties* and *Entities* in some particular situation.

1570 **Generalizations**

1571 The *ResponsibilityRelationshipKind* element inherits the attributes and/or associations of:

- 1572 • *Kind* (see the SCE specification for more information).

1573 **Properties**

1574 The *ResponsibilityRelationshipKind* element does not have any additional attributes and/or associations.

1575 **8.4.11 ResponsibilityRelationshipType**

1576 A kind of *ElementRelationshipType* that specifies an expected *ResponsibilityRelationship* between *PartyTypes* and
1577 *EntityTypes* in some particular situation.

1578 **Generalizations**

1579 The *ResponsibilityRelationshipType* element inherits the attributes and/or associations of:

- 1580 • *ElementRelationshipType* (see the SCE specification for more information).

1581 **Properties**

1582 The following table presents the additional attributes and/or associations for *ResponsibilityRelationshipType*:

Table 46. ResponsibilityRelationshipType Attributes and/or Associations

Property/Association	Description
description : Documentation [0..1]	A textual description of the responsibility.
entityType : EntityType [1..*]	The expected EntityTypes to which the responsibility applies.
kind : ResponsibilityRelationshipKind [1]	A description of the kind of ResponsibilityRelationship between PartyTypes and EntityTypes in a given situation. See ResponsibilityRelationshipKind for more details.
responsiblePartyType : PartyType [1..*]	The PartyType that is expected to have the given ResponsibilityRelationshipType with particular EntityTypes in given situations.

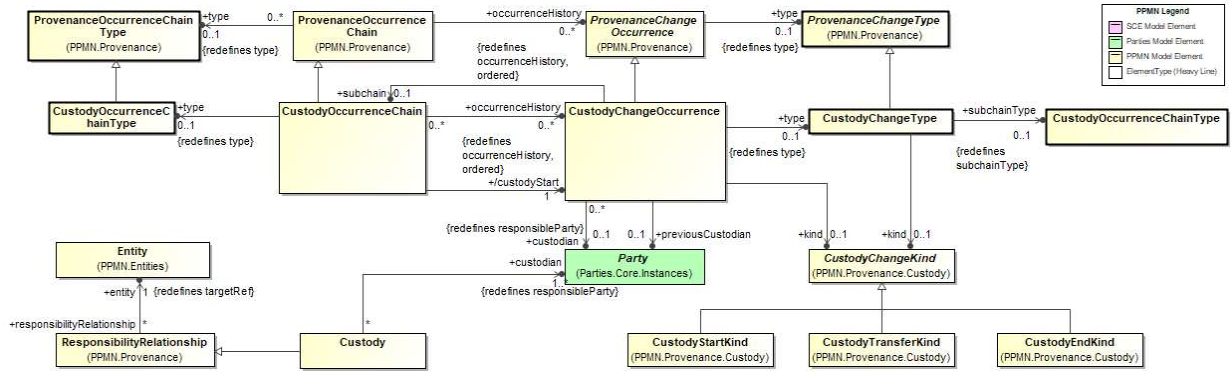
1583

1584 **8.4.12 Custody**

1585 The Custody package provides elements related to the notion of the custody or "physical" control of entities of
1586 interest.

1587 PPMN supports tracking the chain of custody of entities of interest. A *ChainOfCustody* tracks the physical or
1588 electronic holder of an entity of interest. It does this by referencing a series of *CustodyOccurrences* that represent
1589 the custodial history of an entity of interest. A *ChainOfCustody* may have a *ChainOfCustodyType* that defines the
1590 *CustodyOccurrenceTypes* expected for a particular *EntityType*.

1591

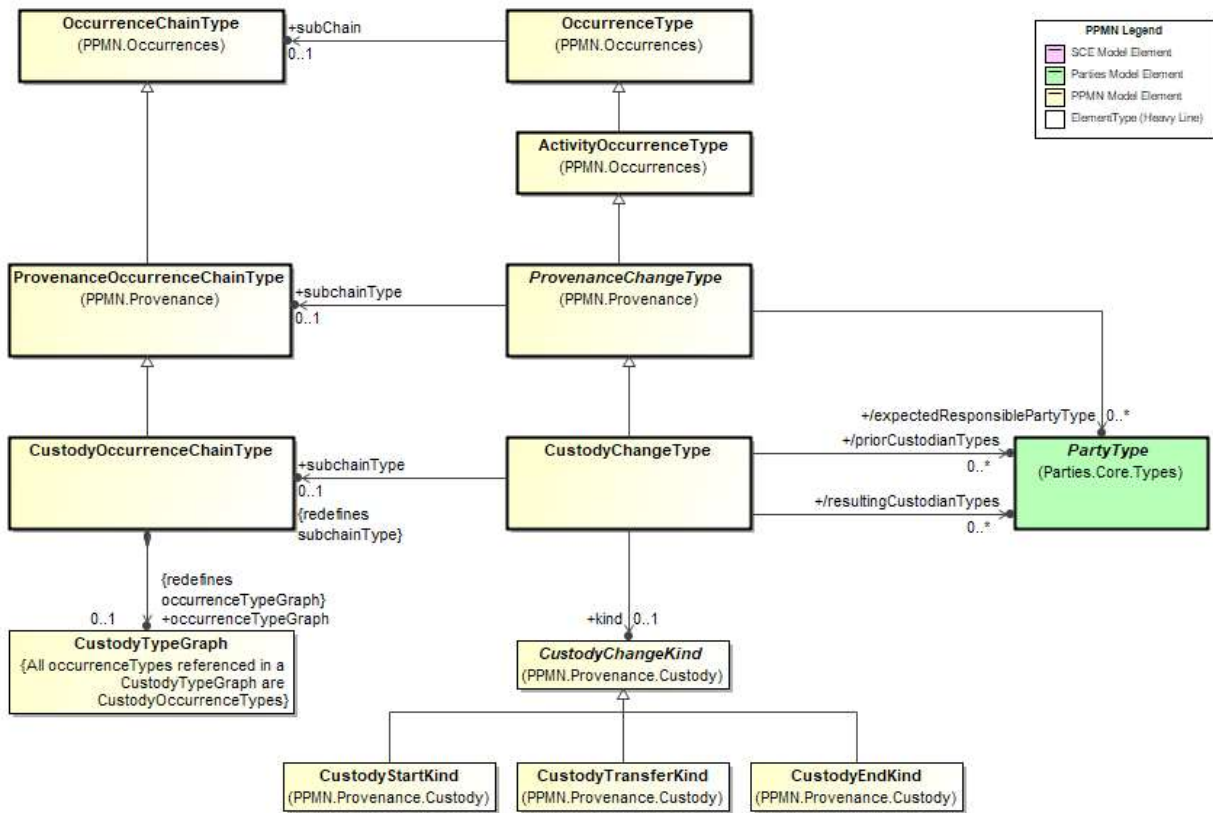


1592

1593 **Figure 32: Custody Occurrence Chains**

1594 Custody-related classes follow the same pattern that PPMN establishes for *Occurrences* generally. This pattern
 1595 supports the "nesting" of a *ChainOfCustody* within a *CustodyOccurrence*. This pattern allows for encapsulation of
 1596 parts of a chain where the details of the occurrences of a part of a larger chain are either not known initially or are
 1597 not deemed important in some context.

1598

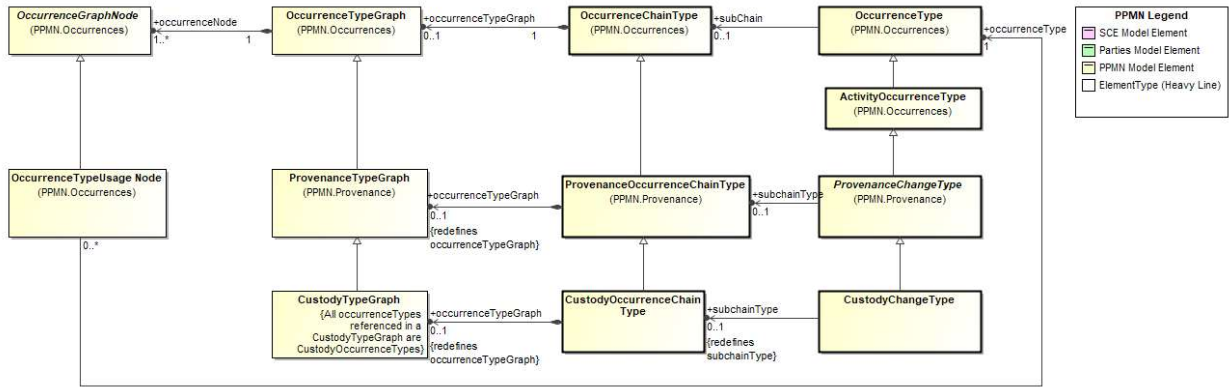


1599

1600 **Figure 33: Custody Occurrence Chain Types**

1601 Custody-related classes follow the same pattern that PPMN establishes for *Occurrences* generally. This pattern
 1602 supports the "nesting" of a *ChainOfCustody* within a *CustodyOccurrence*. This pattern allows for encapsulation of
 1603 parts of a chain where the details of the occurrences of a part of a larger chain are either not known initially or are
 1604 not deemed important in some context.

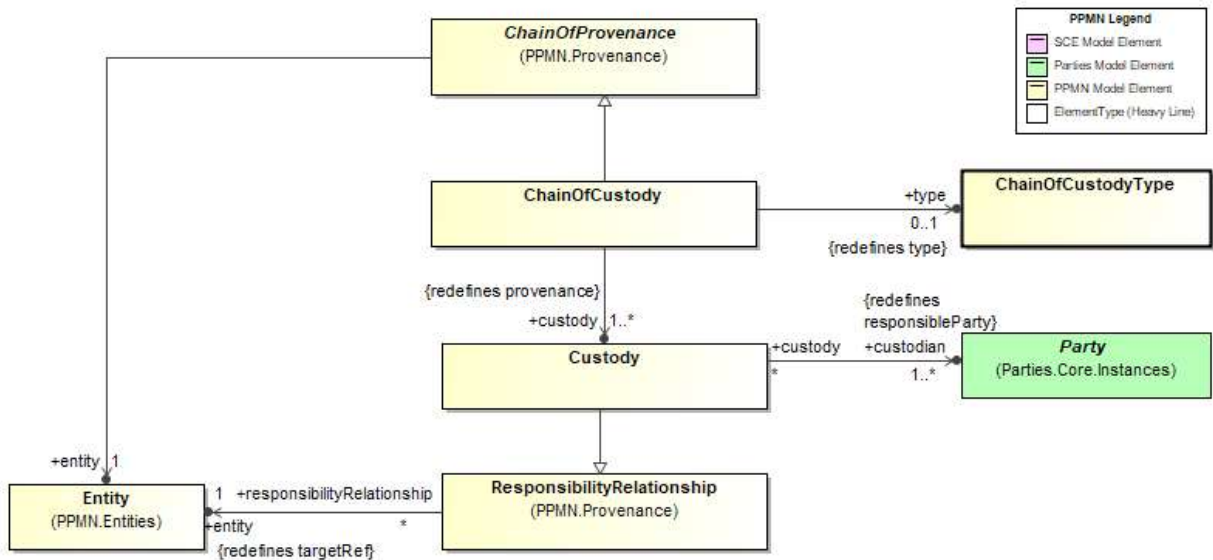
1605



1606
1607 **Figure 34: Custody Occurrence Chain Type Pattern**

1608 TBD.

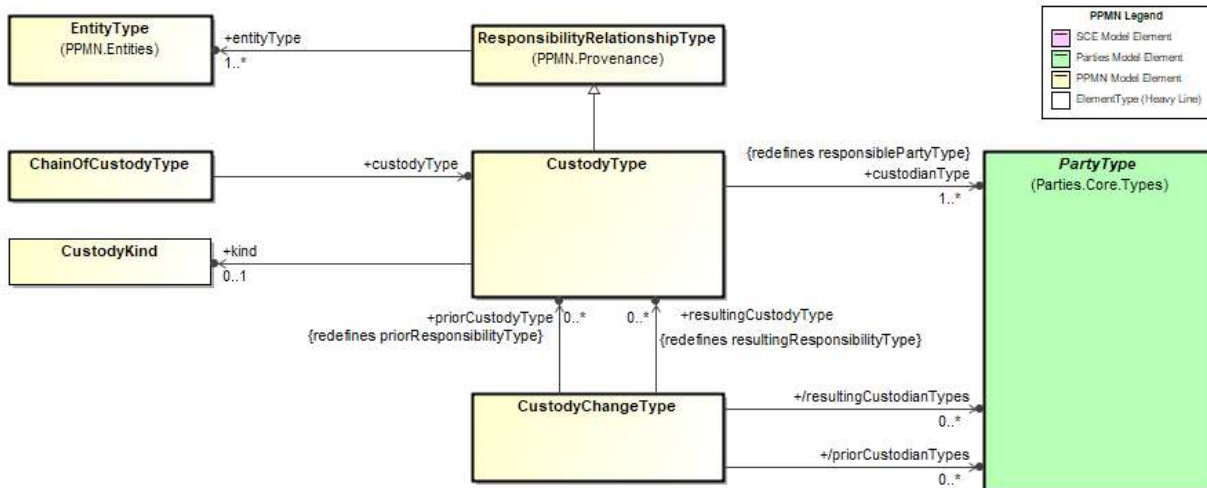
1609



1610
1611 **Figure 35: Chain of Custody**

1612 TBD.

1613



1614
1615 **Figure 36: Chain of Custody Types**

1616 **8.4.12.1 ChainOfCustody**

1617 An ordered set of *Custody* relationships that captures the chain of custody of a particular entity over the course of its
1618 lifecycle.

1619 **Generalizations**

1620 The *ChainOfCustody* element inherits the attributes and/or associations of:

- 1621 • *ChainOfProvenance* (see the section entitled “[ChainOfProvenance](#)” for more information).

1622 **Properties**

1623 The following table presents the additional attributes and/or associations for *ChainOfCustody*:

Table 47. ChainOfCustody Attributes and/or Associations

Property/Association	Description
custody : Custody [1..*]	A set of Custody relationships related to the custody of an entity.
type : ChainOfCustodyType [0..1]	The type of the ChainOfCustody.

1624
1625 **8.4.12.2 ChainOfCustodyType**

1626 A specialization of *ChainOfProvenanceType* that specifies instances of custody chains (*ChainOfCustody*) that
1627 capture an ordered set of *Custody* relationships of type *CustodyType*.

1628 **Generalizations**

1629 The *ChainOfCustodyType* element does not inherit any attributes or associations of from another element.

1630 **Properties**

1631 The following table presents the additional attributes and/or associations for *ChainOfCustodyType*:

Table 48. ChainOfCustodyType Attributes and/or Associations

Property/Association	Description
custodyType : CustodyType []	The <i>CustodyType</i> of the <i>Custody</i> responsibility relationships contained in custody chains of type ChainOfCustodyType.

1632

1633 **8.4.12.3 Custody**

1634 *Custody* is a kind of *ProvenanceRecord* that specifies a *Party* that has physical or electronic control of an entity for a
 1635 particular period of time.

1636 **Generalizations**

1637 The *Custody* element inherits the attributes and/or associations of:

- 1638 • *ResponsibilityRelationship* (see the section entitled “[ResponsibilityRelationship](#)” for more information).

1639 **Properties**

1640 The following table presents the additional attributes and/or associations for *Custody*:

Table 49. Custody Attributes and/or Associations

Property/Association	Description
custodian : Party [1..*]	The <i>Party</i> that acts as the custodian of a particular entity. Redefines <i>responsibleParty</i> .

1641

1642 **8.4.12.4 CustodyChangeKind**

1643 A class indicating the kind of *CustodyChangeOccurrence*.

1644 **Generalizations**

1645 The *CustodyChangeKind* element does not inherit any attributes or associations of from another element.

1646 **Properties**

1647 The *CustodyChangeKind* element does not have any additional attributes and/or associations.

1648 **8.4.12.5 CustodyChangeOccurrence**

1649 An occurrence in the lifecycle of an entity related to the custody of that entity.

1650 **Generalizations**

1651 The *CustodyChangeOccurrence* element inherits the attributes and/or associations of:

- 1652 • *ProvenanceChangeOccurrence* (see the section entitled “[ProvenanceChangeOccurrence](#)” for more
 1653 information).

1654 **Properties**

1655 The following table presents the additional attributes and/or associations for *CustodyChangeOccurrence*:

Table 50. CustodyChangeOccurrence Attributes and/or Associations

Property/Association	Description
custodian : Party [0..1]	The <i>Party</i> that has custody of the entity as a result of the <i>CustodyChangeOccurrence</i> .
kind : CustodyChangeKind [0..1]	The kind of .
previousCustodian : Party [0..1]	The <i>Party</i> that previously had custody of the entity.
subchain : CustodyOccurrenceChain [0..1]	A <i>ChainOfCustody</i> that is encapsulated by the <i>CustodyChangeOccurrence</i> essentially creating a "sub-chain".
type : CustodyChangeType [0..1]	The type of the <i>CustodyChangeOccurrence</i> .

1656

1657 **8.4.12.6 CustodyChangeType**

1658 The type of custody-related occurrences in the lifecycle of an entity that is of interest to some *Party*. Specializations
 1659 of *CustodyOccurrence* will specify the kind of *CustodyOccurrence* that has happened or is expected to happen.

1660 **Generalizations**

1661 The *CustodyChangeType* element inherits the attributes and/or associations of:

- 1662 • *ProvenanceChangeType* (see the section entitled "[ProvenanceChangeType](#)" for more information).

1663 **Properties**

1664 The following table presents the additional attributes and/or associations for *CustodyChangeType*:

Table 51. CustodyChangeType Attributes and/or Associations

Property/Association	Description
kind : CustodyChangeKind [0..1]	The kind of custody change.
priorCustodianTypes : PartyType [0..*]	The type of <i>Party</i> that is expected to relinquish custody of <i>Entities</i> of <i>EntityType</i> as a result of the <i>CustodyOccurrence</i> .
priorCustodyType : CustodyType [0..*]	The <i>CustodyType</i> of the <i>Custody</i> responsibility relationships expected to be in place prior to <i>CustodyChangeOccurrences</i> of type <i>CustodyChangeType</i> .
resultingCustodianTypes : PartyType [0..*]	The type of <i>Party</i> that is expected to have custody of <i>Entities</i> of <i>EntityType</i> as a result of the <i>CustodyOccurrence</i> .
resultingCustodyType : CustodyType [0..*]	The <i>CustodyType</i> expected to be the result of occurrences of type <i>CustodyChangeType</i> .
subchainType : CustodyOccurrenceChainType [0..1]	The expected <i>ChainOfCustodyType</i> that the <i>CustodyOccurrenceType</i> encapsulates.

1665

1666 **8.4.12.7 CustodyEndKind**

1667 A class indicating the CustodyChangeOccurrence was a kind of end.

1668 **Generalizations**

1669 The *CustodyEndKind* element inherits the attributes and/or associations of:

- 1670 • *CustodyChangeKind* (see the section entitled “[CustodyChangeKind](#)” for more information).

1671 In addition, the *CustodyEndKind* element inherits the attributes and/or associations of:

- 1672 • *Kind* (see the SCE specification for information).

1673 **Properties**

1674 The *CustodyEndKind* element does not have any additional attributes and/or associations.

1675 **8.4.12.8 CustodyKind**

1676 A class indicating the kind of *Custody* that a *Party* has with respect to some *Entity*.

1677 **Generalizations**

1678 The *CustodyKind* element does not inherit any attributes or associations of from another element.

1679 **Properties**

1680 The *CustodyKind* element does not have any additional attributes and/or associations.

1681 **8.4.12.9 CustodyOccurrenceChain**

1682 A succession of *CustodyChangeOccurrences* that have happened in the life of an entity that is of interest to some
1683 *Party*.

1684 **Generalizations**

1685 The *CustodyOccurrenceChain* element inherits the attributes and/or associations of:

- 1686 • *ProvenanceOccurrenceChain* (see the section entitled “[ProvenanceOccurrenceChain](#)” for more
1687 information).

1688 **Properties**

1689 The following table presents the additional attributes and/or associations for *CustodyOccurrenceChain*:

Table 52. CustodyOccurrenceChain Attributes and/or Associations

Property/Association	Description
custodyStart : CustodyChangeOccurrence [1]	The occurrence that starts the <i>ChainOfCustody</i> . This is derived by finding the earliest occurrence in the chain.
occurrenceHistory : CustodyChangeOccurrence [0..*]	A set of <i>CustodyOccurrences</i> that comprise the chain.
type : CustodyOccurrenceChainType [0..1]	The type of the <i>ChainOfCustody</i> .

1690

1691 **8.4.12.10 CustodyOccurrenceChainType**

1692 A kind of *ProvenanceChainType* that captures a specification for a series of expected *CustodyOccurrenceTypes* that

1693 are expected for a particular entity type.

1694 **Generalizations**

1695 The *CustodyOccurrenceChainType* element inherits the attributes and/or associations of:

- 1696 • *ProvenanceOccurrenceChainType* (see the section entitled “[ProvenanceOccurrenceChainType](#)” for more
1697 information).

1698 **Properties**

1699 The following table presents the additional attributes and/or associations for *CustodyOccurrenceChainType*:

Table 53. CustodyOccurrenceChainType Attributes and/or Associations

Property/Association	Description
occurrenceTypeGraph : CustodyTypeGraph [0..1]	A graph of <i>CustodyOccurrenceTypes</i> that specifies the sequencing of expected <i>CustodyOccurrences</i> in the lifecycle of an entity of interest to one or more <i>InterestedParties</i> .

1700

1701 **8.4.12.11 CustodyStartKind**

1702 A class indicating the *CustodyChangeOccurrence* was a kind of start.

1703 **Generalizations**

1704 The *CustodyStartKind* element inherits the attributes and/or associations of:

- 1705 • *CustodyChangeKind* (see the section entitled “[CustodyChangeKind](#)” for more information).

1706 In addition, the *CustodyStartKind* element inherits the attributes and/or associations of:

- 1707 • *Kind* (see the SCE specification for more information).

1708 **Properties**

1709 The *CustodyStartKind* element does not have any additional attributes and/or associations.

1710 **8.4.12.12 CustodyTransferKind**

1711 A class indicating the *CustodyChangeOccurrence* was a kind of transfer.

1712 **Generalizations**

1713 The *CustodyTransferKind* element inherits the attributes and/or associations of:

- 1714 • *CustodyChangeKind* (see the section entitled “[CustodyChangeKind](#)” for more information).

1715 **Properties**

1716 The *CustodyTransferKind* element does not have any additional attributes and/or associations.

1717 **8.4.12.13 CustodyType**

1718 A specification of the kind of *Custody* that may exist between *Parties* of type *PartyType* and *Entities* of type
1719 *EntityType*.

1720 **Generalizations**

1721 The *CustodyType* element inherits the attributes and/or associations of:

- *ResponsibilityRelationshipType* (see the section entitled “[ResponsibilityRelationshipType](#)” for more information).

1724 Properties

1725 The following table presents the additional attributes and/or associations for *CustodyType*:

Table 54. CustodyType Attributes and/or Associations

Property/Association	Description
custodianType : PartyType [1..*]	The PartyType expected to have custodial responsibility.
kind : CustodyKind [0..1]	A specification of the kind of custody responsibility.

1726

1727 8.4.12.14 CustodyTypeGraph

1728 A specialized type of *ProvenanceTypeGraph* that captures the *CustodyOccurrenceTypes* that are expected in the
1729 lifecycle of one or more types of entities.

1730 Generalizations

1731 The *CustodyTypeGraph* element inherits the attributes and/or associations of:

- *ProvenanceTypeGraph* (see the section entitled “[ProvenanceTypeGraph](#)” for more information).

1733 Properties

1734 The *CustodyTypeGraph* element does not have any additional attributes and/or associations.

1735 8.4.13 Ownership

1736 An integral aspect of provenance is ownership - the legal or rightful title to an entity. Ownership is important in that
1737 it indicates a legal responsibility for the entity and the right to perform actions on or with the entity in accordance
1738 with applicable laws and regulations. The Ownership package of PPMN provides elements related to the notion of
1739 the ownership of entities of interest by one or more parties.

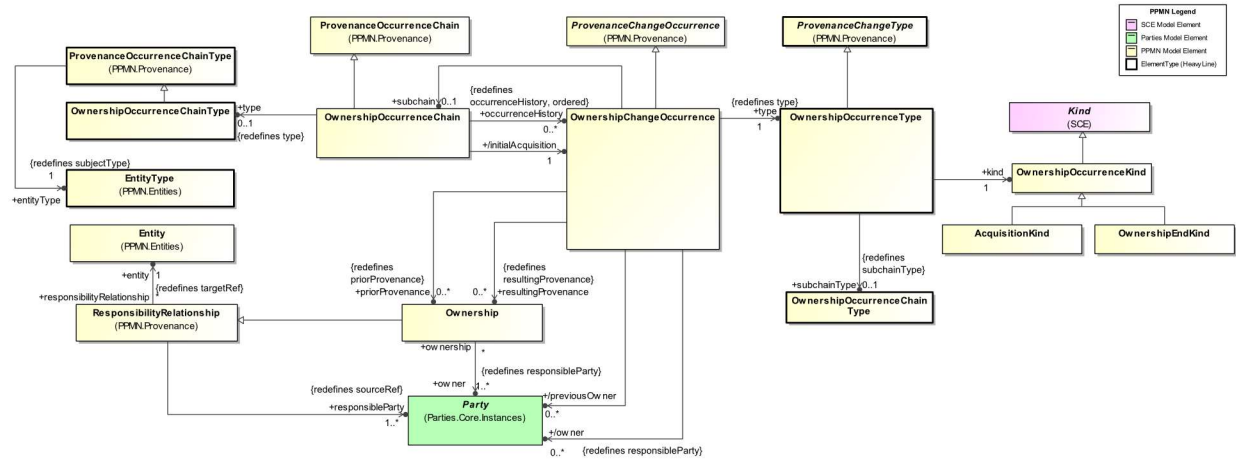
1740 *OwnershipOccurrences* are *Occurrences* that result in some change in ownership such as the acquisition of an entity
1741 by some *Party* or the transfer of ownership of an entity from one *Party* to another. These are useful for two reasons.
1742 First, they link ownership "periods" together and provide greater information about the events or processes that
1743 result in a transition in ownership much like *PedigreeOccurrences* provide insight into how an entity is created or
1744 evolved over time. Second, *Ownership* "records" are generated as a result of *OwnershipOccurrences* and so the
1745 *OwnershipOccurrences* provide insight in how and why ownership has changed..

1746 PPMN supports several kinds of *OwnershipOccurrenceTypes*: *AcquisitionOccurrenceTypes*,
1747 *OwnershipTransferOccurrenceTypes*, and *EndOwnershipOccurrenceTypes*. These specializations support the
1748 typical ownership transitions that may take place in the lifecycle of an entity but are not expected to be only types of
1749 transitions that may occur.

1750 A *ChainOfOwnership* is a kind of *ProvenanceChain* that tracks the ownership-related *Occurrences* of an entity of
1751 interest. A *ChainOfOwnership* may be typed in the same way as *ProvenanceChains* using a
1752 *ChainOfOwnershipType*. *ChainOfOwnershipType* allows stakeholders to define the expected changes in ownership
1753 of entities of a particular type in advance for planning or other purposes.

1754

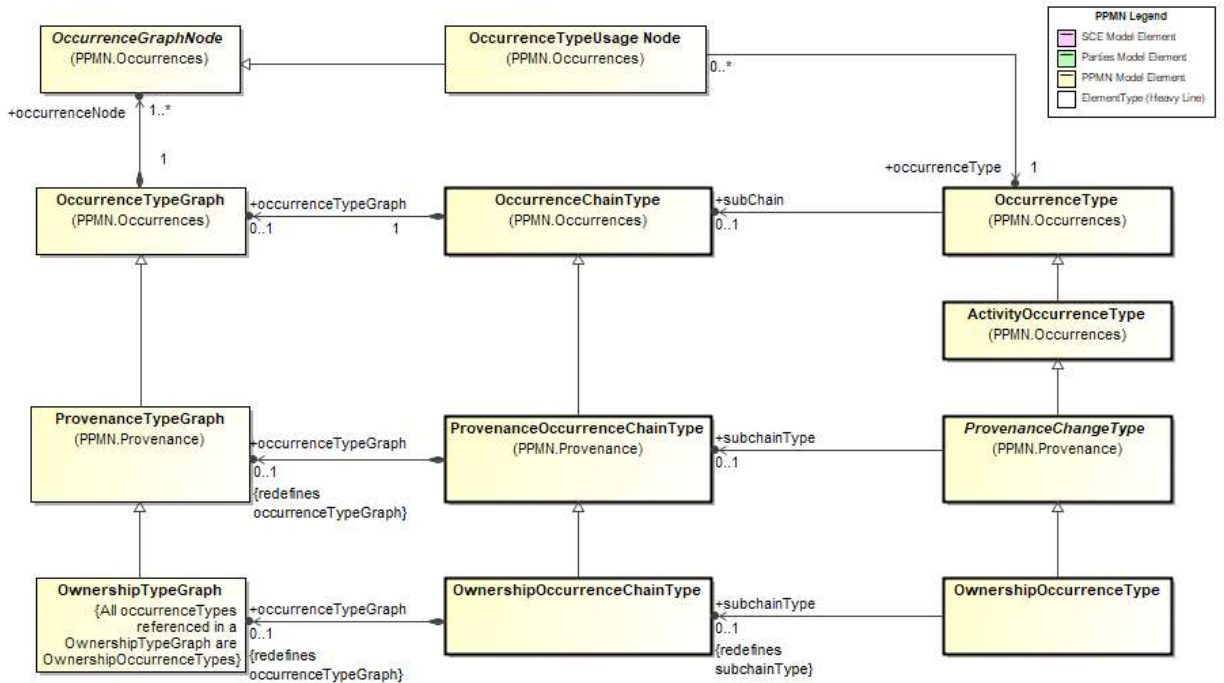
1755



1756
 1757
 1758
 1759
 1760
 1761
 1762
 1763

Figure 37: Ownership Occurrence Chains

ChainOfOwnership, *ChainOfOwnershipType*, *OwnershipOccurrences*, and *OwnershipOccurrenceTypes* follow the same pattern established for other types of occurrences. This pattern supports the "nesting" of a *ChainOfOwnership* within an *OwnershipOccurrence*. This pattern allows for encapsulation of parts of a chain where the details of the *OwnershipOccurrences* of that part of a larger chain are either not known initially or are not deemed important in some context.

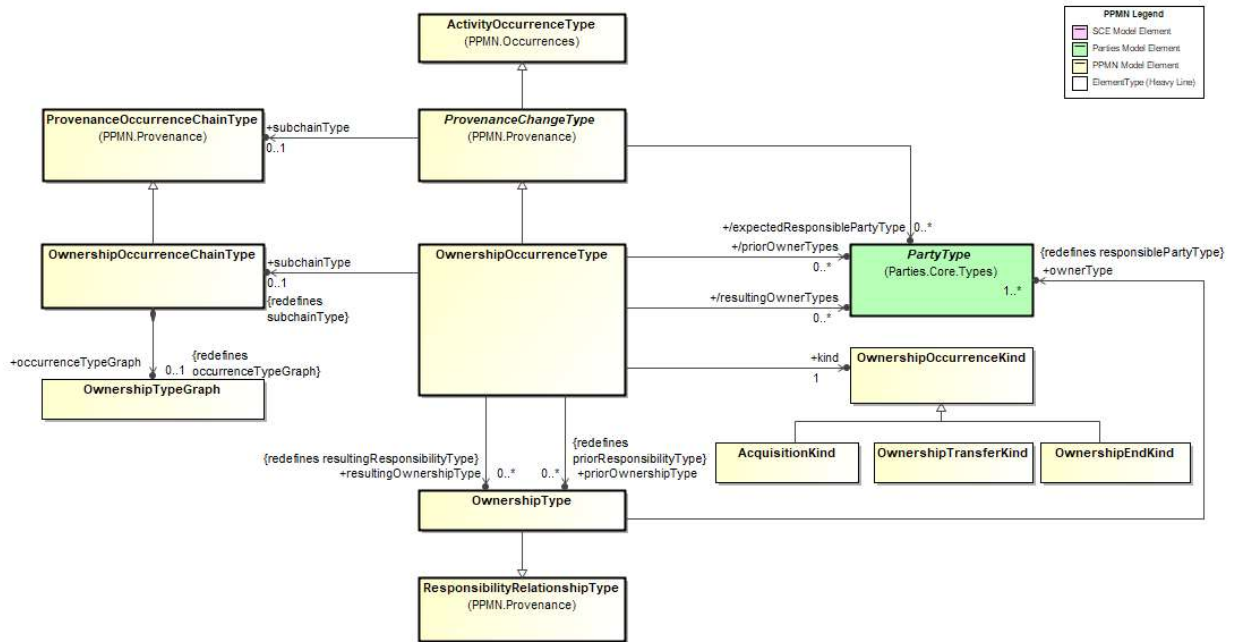


1764
 1765

Figure 38: Ownership Occurrence Chain Type Pattern

ChainOfOwnership, *ChainOfOwnershipType*, *OwnershipOccurrences*, and *OwnershipOccurrenceTypes* follow the same pattern established for other types of occurrences. This pattern supports the "nesting" of a *ChainOfOwnership* within an *OwnershipOccurrence*. This pattern allows for encapsulation of parts of a chain where the details of the *OwnershipOccurrences* of that part of a larger chain are either not known initially or are not deemed important in some context.

1771



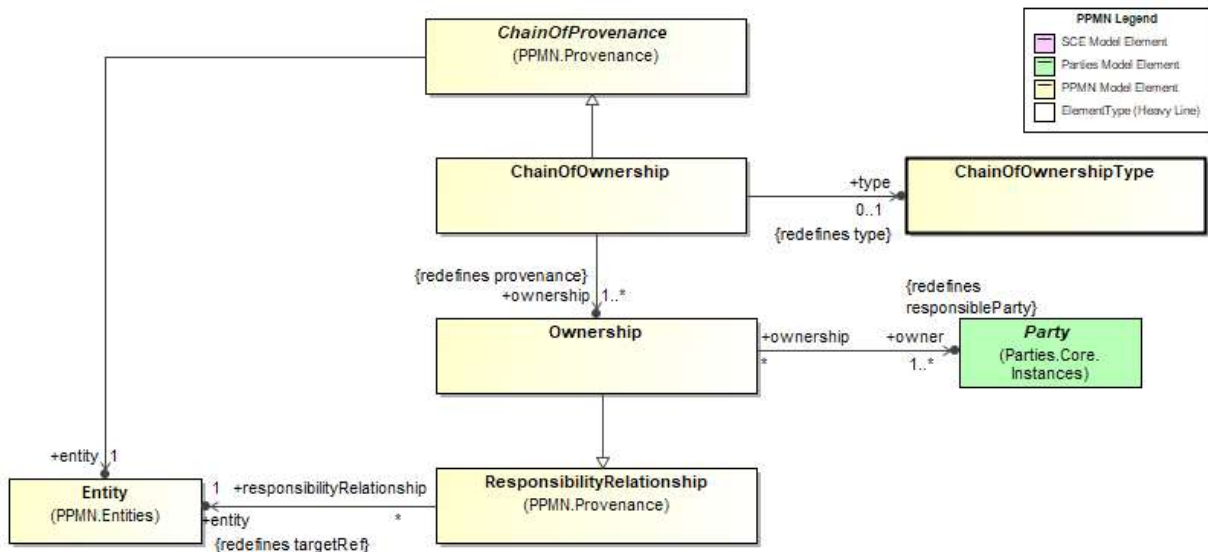
1772

1773

Figure 39: Ownership Occurrence Chain Types

1774

1775



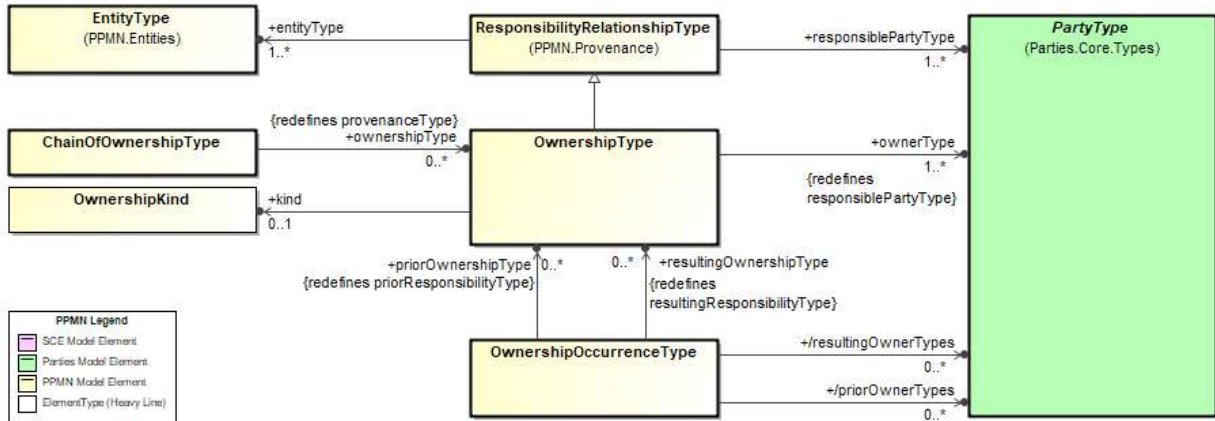
1776

1777

Figure 40: Chain of Ownership

1778

1779



1780
1781 **Figure 41: Chain of Ownership Types**

1782 **8.4.13.1 AcquisitionKind**

1783 A class indicating how a *ChainOfOwnership* was started.

1784 **Generalizations**

1785 The *AcquisitionKind* element inherits the attributes and/or associations of:

- 1786 • *OwnershipOccurrenceKind* (see the section entitled “[OwnershipOccurrenceKind](#)” for more information).

1787 **Properties**

1788 The *AcquisitionKind* element does not have any additional attributes and/or associations.

1789 **8.4.13.2 ChainOfOwnership**

1790 An ordered set of *Ownership* relationships that captures the ownership of a particular entity over the course of its
1791 lifecycle.

1792 **Generalizations**

1793 The *ChainOfOwnership* element inherits the attributes and/or associations of:

- 1794 • *ChainOfProvenance* (see the section entitled “[ChainOfProvenance](#)” for more information).

1795 **Properties**

1796 The following table presents the additional attributes and/or associations for *ChainOfOwnership*:

Table 55. ChainOfOwnership Attributes and/or Associations

Property/Association	Description
ownership : Ownership [1..*]	A set of Ownership relationships related to the ownership of an entity.
type : ChainOfOwnershipType [0..1]	The type of the ChainOfOwnership.

1797

1798 **8.4.13.3 ChainOfOwnershipType**

1799 A specialization of *ChainOfProvenanceType* that specifies instances of ownership chains (*ChainOfOwnership*) that
1800 capture an ordered set of *Ownership* relationships of type *OwnershipType*.

1801 **Generalizations**

1802 The *ChainOfOwnershipType* element inherits the attributes and/or associations of:

- 1803 • *ChainOfProvenanceType* (see the section entitled “[ChainOfProvenanceType](#)” for more information).

1804 **Properties**

1805 The following table presents the additional attributes and/or associations for *ChainOfOwnershipType*:

Table 56. ChainOfOwnershipType Attributes and/or Associations

Property/Association	Description
ownershipType : OwnershipType [0..*]	The <i>OwnershipType</i> of the <i>Ownership</i> responsibility relationships included in <i>ChainOfOwnership</i> s that are of type <i>ChainOfOwnershipType</i> .

1806

1807 **8.4.13.4 Ownership**

1808 A kind of *ProvenanceRecord* relationship that specifies a *Party* is playing the role of *Owner* of an entity for a
1809 particular period of time.

1810 **Generalizations**

1811 The *Ownership* element inherits the attributes and/or associations of:

- 1812 • *ResponsibilityRelationship* (see the section entitled “[ResponsibilityRelationship](#)” for more information).

1813 **Properties**

1814 The following table presents the additional attributes and/or associations for *Ownership*:

Table 57. Ownership Attributes and/or Associations

Property/Association	Description
owner : Party [1..*]	The <i>Party</i> that acts as the owner of a particular entity. Redefines <i>responsibleParty</i> .

1815

1816 **8.4.13.5 OwnershipChangeOccurrence**

1817 An *Occurrence* in the lifecycle of an entity related to the ownership of that entity.

1818 **Generalizations**

1819 The *OwnershipChangeOccurrence* element inherits the attributes and/or associations of:

- 1820 • *ProvenanceChangeOccurrence* (see the section entitled “[ProvenanceChangeOccurrence](#)” for more
1821 information).

1822 **Properties**

1823 The following table presents the additional attributes and/or associations for *OwnershipChangeOccurrence*:

Table 58. OwnershipChangeOccurrence Attributes and/or Associations

Property/Association	Description
owner : Party [0..*]	The <i>Party</i> that has ownership of the entity as a result of the <i>OwnershipOccurrence</i> .
previousOwner : Party [0..*]	The previous owner(s) of the entity.
priorProvenance : Ownership [0..*]	The <i>Ownership</i> relationships prior to the <i>OwnershipChangeOccurrence</i> .
resultingProvenance : Ownership [0..*]	The <i>Ownership</i> relationships that result from the <i>OwnershipChangeOccurrence</i> .
subchain : OwnershipOccurrenceChain [0..1]	A <i>ChainOfOwnership</i> that is encapsulated by the <i>OwnershipOccurrence</i> essentially creating a "sub-chain".
type : OwnershipOccurrenceType [1]	The type of the <i>OwnershipChangeOccurrence</i> .

1824

1825 **8.4.13.6 OwnershipEndKind**

1826 A class indicating how the *ChainOfOwnership* was ended.

1827 **Generalizations**

1828 The *OwnershipEndKind* element inherits the attributes and/or associations of:

- 1829 • *OwnershipOccurrenceKind* (see the section entitled “[OwnershipOccurrenceKind](#)” for more information).

1830 **Properties**

1831 The *OwnershipEndKind* element does not have any additional attributes and/or associations.

1832 **8.4.13.7 OwnershipKind**

1833 A specification of a particular kind of ownership responsibility.

1834 **Generalizations**

1835 The *OwnershipKind* element does not inherit any attributes or associations of from another element.

1836 **Properties**

1837 The *OwnershipKind* element does not have any additional attributes and/or associations.

1838 **8.4.13.8 OwnershipOccurrenceChain**

1839 A succession of *OwnershipOccurrences* that have happened in the life of an entity that is of interest to some *Party*.

1840 **Generalizations**

1841 The *OwnershipOccurrenceChain* element inherits the attributes and/or associations of:

- 1842 • *ProvenanceOccurrenceChain* (see the section entitled “[ProvenanceOccurrenceChain](#)” for more
1843 information).

1844 **Properties**

1845 The following table presents the additional attributes and/or associations for *OwnershipOccurrenceChain*:

Table 59. OwnershipOccurrenceChain Attributes and/or Associations

Property/Association	Description
initialAcquisition : OwnershipChangeOccurrence [1]	The occurrence that starts the <i>ChainOfOwnership</i> . This is derived by finding the earliest occurrence in the chain.
occurrenceHistory : OwnershipChangeOccurrence [0..*]	A set of <i>OwnershipOccurrences</i> that comprise the chain.
type : OwnershipOccurrenceChainType [0..1]	The type of the <i>ChainOfOwnership</i> .

1846

1847 **8.4.13.9 OwnershipOccurrenceChainType**

1848 A kind of *ProvenanceChainType* that captures a specification for a series of expected *OwnershipOccurrenceTypes*
 1849 that are expected for a particular entity type. An *OwnershipOccurrenceType* captures this specification through the
 1850 *occurrenceTypeGraph* property - a graph of *OccurrenceGraphNode*s and *OccurrenceTransitionTypes*.

1851 **Generalizations**

1852 The *OwnershipOccurrenceChainType* element inherits the attributes and/or associations of:

- 1853 • *ProvenanceOccurrenceChainType* (see the section entitled “[ProvenanceOccurrenceChainType](#)” for more
 1854 information).

1855 **Properties**

1856 The following table presents the additional attributes and/or associations for *OwnershipOccurrenceChainType*:

Table 60. OwnershipOccurrenceChainType Attributes and/or Associations

Property/Association	Description
occurrenceTypeGraph : OwnershipTypeGraph [0..1]	A graph of <i>OwnershipOccurrenceTypes</i> that specifies the sequencing of expected <i>OwnershipOccurrences</i> in the lifecycle of an entity of interest to one or more <i>InterestedParties</i> .

1857

1858 **8.4.13.10 OwnershipOccurrenceKind**

1859 A class indicating the kind of *OwnershipOccurrence* that is expected.

1860 **Generalizations**

1861 The *OwnershipOccurrenceKind* element inherits the attributes and/or associations of:

- 1862 • *Kind* (see the SCE specification for more information).

1863 **Properties**

1864 The *OwnershipOccurrenceKind* element does not have any additional attributes and/or associations.

1865 **8.4.13.11 OwnershipOccurrenceType**

1866 The type of *OwnershipOccurrence* in the lifecycle of an entity that is of interest to some *Party*. Specializations of
 1867 *OwnershipOccurrenceType* will specify the kind of *OwnershipOccurrence* that has happened.

1868 **Generalizations**

1869 The *OwnershipOccurrenceType* element inherits the attributes and/or associations of:

- 1870 • *ProvenanceChangeType* (see the section entitled “[ProvenanceChangeType](#)” for more information).

1871 **Properties**

1872 The following table presents the additional attributes and/or associations for *OwnershipOccurrenceType*:

Table 61. OwnershipOccurrenceType Attributes and/or Associations

Property/Association	Description
kind : OwnershipOccurrenceKind [1]	A reference to a definition of the specific kind of <i>OwnershipOccurrenceType</i> .
priorOwnershipType : OwnershipType [0..*]	The <i>OwnershipType</i> expected to exist prior to occurrences of type <i>OwnershipOccurrenceType</i> .
priorOwnerTypes : PartyType [0..*]	The type of <i>Party</i> that is expected to relinquish ownership of <i>Entities</i> of <i>EntityType</i> as a result of the <i>OwnershipOccurrence</i> .
resultingOwnershipType : OwnershipType [0..*]	The <i>OwnershipType</i> expected to be the result of occurrences of type <i>OwnershipOccurrenceType</i> .
resultingOwnerTypes : PartyType [0..*]	The type of <i>Party</i> that is expected to have ownership of <i>Entities</i> of <i>EntityType</i> as a result of <i>Occurrences</i> of the <i>OwnershipOccurrenceType</i> .
subchainType : OwnershipOccurrenceChainType [0..1]	A <i>ChainOfOwnershipType</i> that is encapsulated within the <i>OwnershipOccurrenceType</i> to create a "subchain".

1873

1874 **8.4.13.12 OwnershipTransferKind**

1875 A class indicating how a *ChainOfOwnership* was started.

1876 **Generalizations**

1877 The *OwnershipTransferKind* element does not inherit any attributes or associations of from another element.

1878 **Properties**

1879 The *OwnershipTransferKind* element does not have any additional attributes and/or associations.

1880 **8.4.13.13 OwnershipType**

1881 The type of *Ownership* that may exist between *Parties* of type *PartyType* and *Entities* of type *EntityType*.

1882 **Generalizations**

1883 The *OwnershipType* element inherits the attributes and/or associations of:

- 1884 • *ResponsibilityRelationshipType* (see the section entitled “[ResponsibilityRelationshipType](#)” for more
 1885 information).

1886 **Properties**

1887 The following table presents the additional attributes and/or associations for *OwnershipType*:

Table 62. OwnershipType Attributes and/or Associations

Property/Association	Description
kind : OwnershipKind [0..1]	A specification of the kind of ownership responsibility.
ownerType : PartyType [1..*]	The PartyType expected to have ownership responsibility.

1888

1889 **8.4.13.14 OwnershipTypeGraph**

1890 A specialized type of *ProvenanceTypeGraph* that captures the *OwnershipOccurrenceTypes* that are expected in the
1891 lifecycle of one or more types of entities.

1892 **Generalizations**

1893 The *OwnershipTypeGraph* element inherits the attributes and/or associations of:

- 1894 • *ProvenanceTypeGraph* (see the section entitled "[ProvenanceTypeGraph](#)" for more information).

1895 **Properties**

1896 The *OwnershipTypeGraph* element does not have any additional attributes and/or associations.

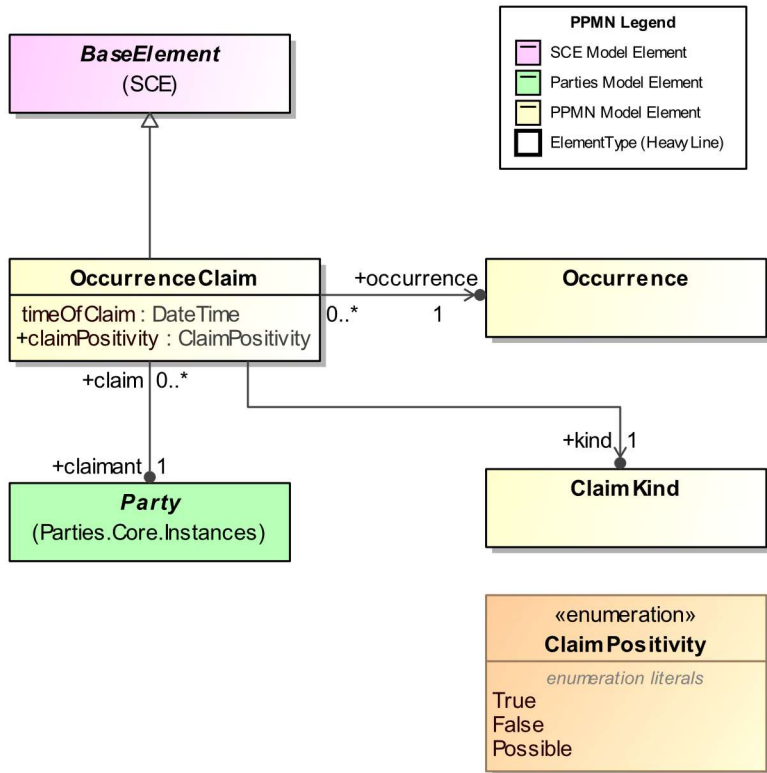
1897 **8.5 Claims**

1898 The Claims package contains elements related to *Claims* made by *Parties* about *Occurrences*.

1899 In many situations, pedigree and/or provenance information about entities is put forth by some party as being true
1900 when in fact, that information may be disputed and even shown to be false. *Claims* provide a mechanism to note the
1901 *Party* (the claimant) that claims an *Occurrence* has happened. The time the claim was made is captured as well as
1902 whether the *Claim* was made in a "positive" or "negative" manner (the `claimPositivity`). `ClaimPositivity`
1903 states whether the *Claim* was made in a "positive" manner, i.e., the *Occurrence* is claimed to have happened, or a
1904 "negative" maner, i.e., the *Occurrence* is claimed *not to have happened*. A `claimPositivity` of "Possible"
1905 means that the *Occurrence* *may* have happened.

1906

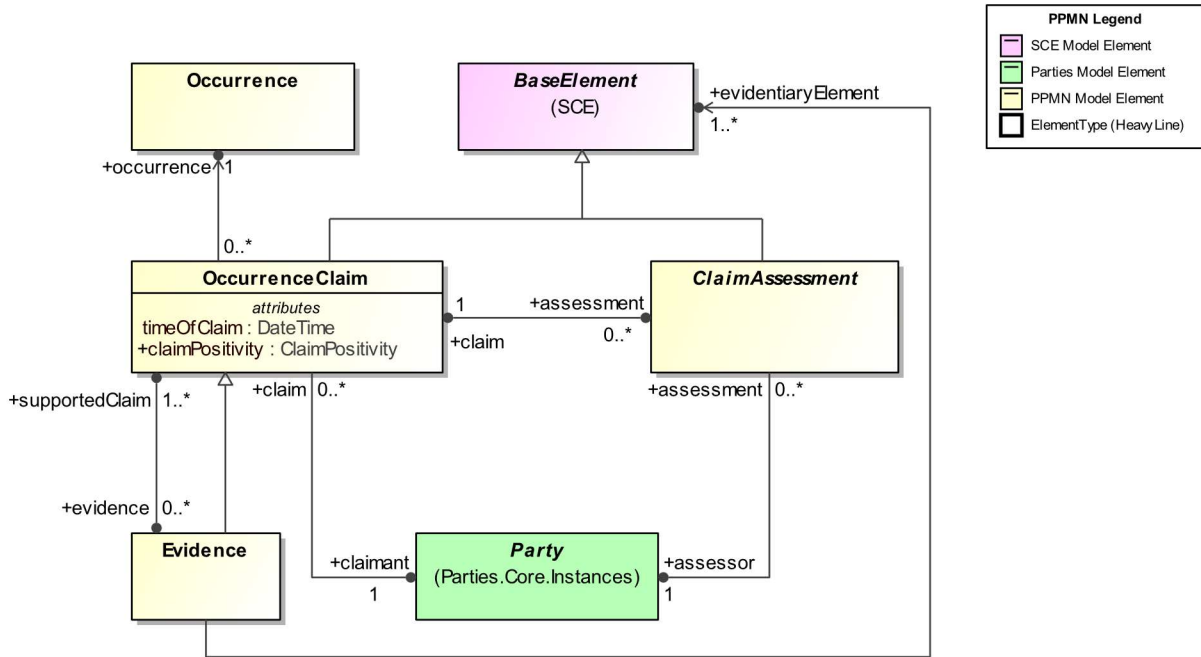
1907



1908
 1909
 1910
 1911
 1912
 1913

Figure 42: Claims

A Claim may be assessed in some way as stated by a ClaimAssessment by some Party (the assessor). The actual method or mechanism of the assessment is outside the scope of this specification.



1914
1915 **Figure 43: Claim Assessments**

1916 **8.5.1 ClaimPositivity**

1917 A enumeration that indicates whether the statement asserted by a Claim is asserted as being true, false, or possible.

Table 63. ClaimPositivity Literals

Literal	Description
False	Indicates that the Claim asserts the Occurrence did not happen.
Possible	Indicates that the Claim asserts the Occurrence may have happened.
True	Indicates that the Claim asserts the Occurrence happened.

1918
1919 **8.5.2 ClaimAssessment**

1920 An assessment of a Claim by an assessor.

1921 **Generalizations**

1922 The *ClaimAssessment* element inherits the attributes and/or associations of:

- 1923 • *SCE SCEElement* (see the *SCE* specification for more information).

1924 **Properties**

1925 The following table presents the additional attributes and/or associations for *ClaimAssessment*:

Table 64. ClaimAssessment Attributes and/or Associations

Property/Association	Description
assessor : Party [1]	The Party that made the assessment.
claim : OccurrenceClaim [1]	The Claim about which the assessment was made.

1926

1927 **8.5.3 ClaimKind**

1928 A class that indicates the kind of *Claim* that has been made.

1929 **Generalizations**

1930 The *ClaimKind* element inherits the attributes and/or associations of:

- 1931 • *Kind* (see the **SCE** specification for more information).

1932 **Properties**

1933 The *ClaimKind* element does not have any additional attributes and/or associations.

1934 **Generalizations**

1935 The *Evidence* element inherits the attributes and/or associations of:

- 1936 • *OccurrenceClaim* (see the section entitled "[OccurrenceClaim](#)" for more information).

1937 **Properties**

1938 The following table presents the additional attributes and/or associations for *Evidence*:

Table 65. Evidence Attributes and/or Associations

Property/Association	Description
evidentiaryElement : SCEElement [1..*]	The elements that comprise the <i>Evidence</i> for the supported <i>Claims</i> .
supportedClaim : OccurrenceClaim [1..*]	The <i>Claims</i> that the <i>Evidence</i> is intended to support.

1939

1940 **8.5.4 OccurrenceClaim**

1941 A statement made by a Party about whether an Occurrence happened or not.

1942 **Generalizations**

1943 The *OccurrenceClaim* element inherits the attributes and/or associations of:

- 1944 • **SCE** *SCEElement* (see the **SCE** specification for more information).

1945 **Properties**

1946 The following table presents the additional attributes and/or associations for *OccurrenceClaim*:

Table 66. OccurrenceClaim Attributes and/or Associations

Property/Association	Description
assessment : ClaimAssessment [0..*]	An assessment of the Claim.
claimant : Party [1]	The Party that made the Claim.
claimPositivity : ClaimPositivity []	A property that states whether the claim is said to be true, false or possible.
evidence : Evidence [0..*]	The <i>Evidence</i> intended to support the <i>Claim</i> .
kind : ClaimKind [1]	The kind of assertion of the Claim.
occurrence : Occurrence [1]	The Occurrence about which the Claim was made.
timeOfClaim : DateTime []	The time the Claim was made.

1947

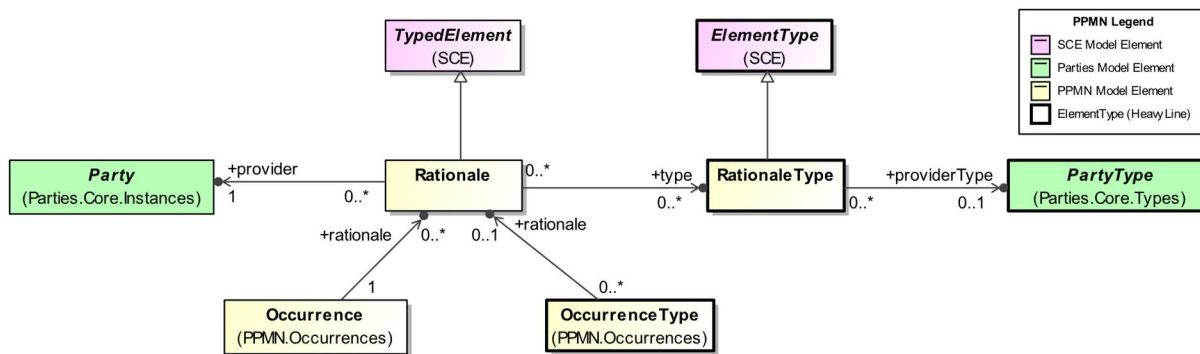
1948 8.6 Rationale

1949 The Rationale package contains elements that provide the ability to capture the rationale for *Occurrences*.

1950 PPMN supports the ability to capture a *Rationale*, the reasoning or justification, for *Occurrences* and
 1951 *OccurrenceTypes*. *RationaleType* enables capture of the type of a particular Rationale or of the kind of *Rationale*
 1952 that is expected in a particular context.

1953

1954



1955

1956 **Figure 44: Rationale**

1957 8.6.1 Rationale

1958 A class representing the basis for an *Occurrence* or *OccurrenceType*.

1959 Generalizations

1960 The *Rationale* element inherits the attributes and/or associations of:

- SCE *TypedElement* (see the section SCE specification for more information).

1962 Properties

1963 The following table presents the additional attributes and/or associations for *Rationale*:

Table 67. Rationale Attributes and/or Associations

Property/Association	Description
provider : Party [1]	The <i>Party</i> that provided the <i>Rationale</i> .
type : RationaleType [0..*]	The class(es) that provide(s) a specification of the <i>Rationale</i> .

1964

1965 **8.6.2 RationaleType**

1966 A class representing the type or classification of a *Rationale*.

1967 **Generalizations**

1968 The *RationaleType* element inherits the attributes and/or associations of:

- 1969 • *SCE ElementType* (see the *SCE* specification for more information).

1970 **Properties**

1971 The following table presents the additional attributes and/or associations for *RationaleType*:

Table 68. RationaleType Attributes and/or Associations

Property/Association	Description
providerType : PartyType [0..1]	The <i>PartyType</i> that is expected to provide the kind of <i>Rationale</i> specified by the <i>RationaleType</i> .

1972

1973 **8.7 Annotations**

1974 The Annotation package contains elements related to the notion of annotation of elements with notes about that
1975 element.

1976 *Annotations* are applied to *BaseElements* for any purpose that suits the business needs of an organization.

1977 *Annotations* can exist independently of those elements providing a “catalog” of *Annotations*. *AnnotationTemplate*

1978 provides a means of creating base annotations that can be "instantiated" as either *SimpleAnnotations* or

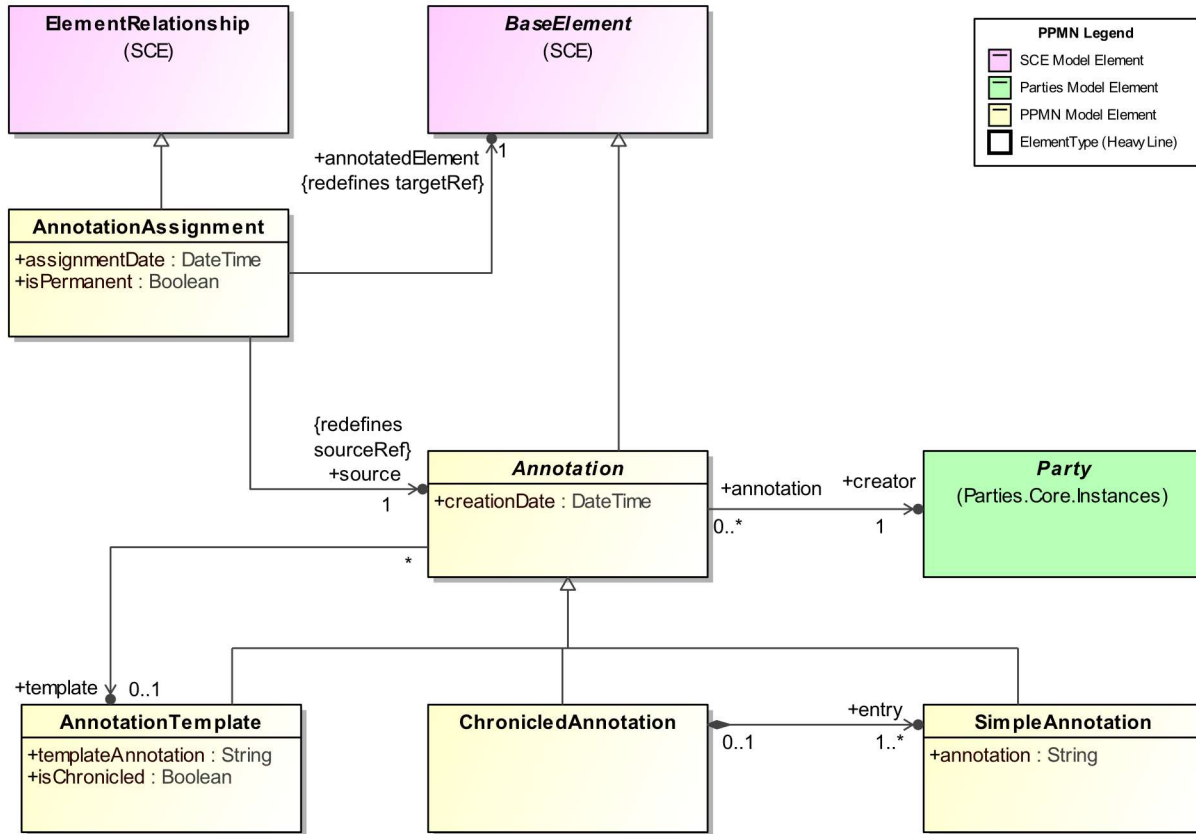
1979 *ChronicledAnnotations*. *Annotations* may have an association to the *AnnotationTemplate* from which they were

1980 created. The *Party* creating an *Annotation* is captured as the creator. That *Party* or another *Party* may assign an

1981 annotation to a *BaseElement* through an *AnnotationAssignment* relationship.

1982

1983



1984

1985 **Figure 45: Annotations**

1986 **8.7.1.1 Annotation**

1987 A note or series of notes related to some BaseElement in a PPM information set.

1988 **Generalizations**

1989 The *Annotation* element inherits the attributes and/or associations of:

- 1990 • SCE BaseElement (see the SCE specification for more information).

1991 **Properties**

1992 The following table presents the additional attributes and/or associations for *Annotation*:

Table 69. Annotation Attributes and/or Associations

Property/Association	Description
creationDate : DateTime []	The Date/Time that the Annotation was created.
creator : Party [1]	The <i>Party</i> that created the annotation.
template : AnnotationTemplate [0..1]	The template from which an <i>Annotation</i> was created.

1993

1994 **8.7.1.2 AnnotationAssignment**

1995 An association that links an *Annotation* to a *BaseElement* in a PPMN information set.

1996 **Generalizations**

1997 The *AnnotationAssignment* element inherits the attributes and/or associations of:

- 1998 • *ElementRelationship* (see the SCE specification for more information).

1999 **Properties**

2000 The following table presents the additional attributes and/or associations for *AnnotationAssignment*:

Table 70. AnnotationAssignment Attributes and/or Associations

Property/Association	Description
annotatedElement : BaseElement [1]	The element to which the <i>Annotation</i> has been assigned.
assignmentDate : DateTime []	The Date/Time the <i>Annotation</i> was applied.
isPermanent : Boolean []	A boolean specifying whether or not the <i>Annotation</i> is intended to be permanent.
source : Annotation [1]	The <i>Annotation</i> that has been assigned to some element.

2001

2002 **8.7.1.3 AnnotationTemplate**

2003 A kind of *Annotation* that is intended to be used as a template for other *Annotations*.

2004 **Generalizations**

2005 The *AnnotationTemplate* element inherits the attributes and/or associations of:

- 2006 • *Annotation* (see the section entitled "[Annotation](#)" for more information).

2007 **Properties**

2008 The following table presents the additional attributes and/or associations for *AnnotationTemplate*:

Table 71. AnnotationTemplate Attributes and/or Associations

Property/Association	Description
isChronicled : Boolean []	A boolean that specifies whether the <i>Annotations</i> created with this template are <i>ChronicledAnnotations</i> or not.
templateAnnotation : String []	A default string that is meant for recurring use.

2009

2010 **8.7.1.4 ChronicledAnnotation**

2011 A kind of *Annotation* that has a series of time-based entries. Individual entries are captured as *SimpleAnnotations*
 2012 with the *isPermenant* flag set to True. The *creationDate* of the *SimpleAnnotations* that represent the entries
 2013 of a *ChronicledAnnotation* captures the date the *ChronicledAnnotation* was updated.

2014 **Generalizations**

2015 The *ChronicedAnnotation* element inherits the attributes and/or associations of:

- 2016 • *Annotation* (see the section entitled “[Annotation](#)” for more information).

2017 **Properties**

2018 The following table presents the additional attributes and/or associations for *ChronicedAnnotation*:

Table 72. ChronicedAnnotation Attributes and/or Associations

Property/Association	Description
entry : SimpleAnnotation [1..*]	A <i>SimpleAnnotation</i> that represents one entry in a <i>ChronicedAnnotation</i> .

2019

2020 **8.7.1.5 SimpleAnnotation**

2021 A kind of *Annotation* that is a simple note related to one or more *BaseElements* in a PPM information set.

2022 **Generalizations**

2023 The *SimpleAnnotation* element inherits the attributes and/or associations of:

- 2024 • *Annotation* (see the section entitled “[Annotation](#)” for more information).

2025 **Properties**

2026 The following table presents the additional attributes and/or associations for *SimpleAnnotation*:

Table 73. SimpleAnnotation Attributes and/or Associations

Property/Association	Description
annotation : String []	A string containing the text of the <i>Annotation</i> .

2027

2028 **8.8 Delegation**

2029 The Delegation package provides elements related to the notion of delegation of responsibility for an entity from one
2030 party to another.

2031 Delegation captures the notion that a Party may assign a set of responsibilities to another party. The responsibilities
2032 being assigned are essentially captured as a Role. The class ActedOnBehalfOf is a relationship that states that one
2033 Party was acting for or representing another Party and that action may be justified by a Delegation. The property
2034 *inRole* allows a model to specify that the *Party* acted on behalf of another *Party* while performing a particular role
2035 in an *Occurrence*.

2036

2047

2048 8.8.2 DelegationAssignment

2049 A kind of *ActivityOccurrence* wherein one *Party* delegates a set of responsibilities to another *Party*.

2050 Generalizations

2051 The *DelegationAssignment* element inherits the attributes and/or associations of:

- 2052 • *ActivityOccurrence* (see the section entitled "[ActivityOccurrence](#)" for more information).

2053 Properties

2054 The following table presents the additional attributes and/or associations for *DelegationAssignment*:

Table 75. DelegationAssignment Attributes and/or Associations

Property/Association	Description
delegation : Delegation [1]	The <i>Delegation</i> that was the result of the <i>DelegationAssignment</i> .
delegator : Party [1]	The <i>Party</i> responsible for the <i>DelegationAssignment</i> .

2055

2056 8.9 Additional Relationships

2057 In addition to Delegation and Derivation, PPMN includes a number of other types of relationships that are important
2058 to pedigree and/or provenance. These additional relationships are described herein.

2059 **PPMN** includes several other types of relationships that may be important to particular stakeholders in addition to
2060 derivations and delegations. These cover the concepts of attribution, specialization, alternates and general
2061 "informing of".

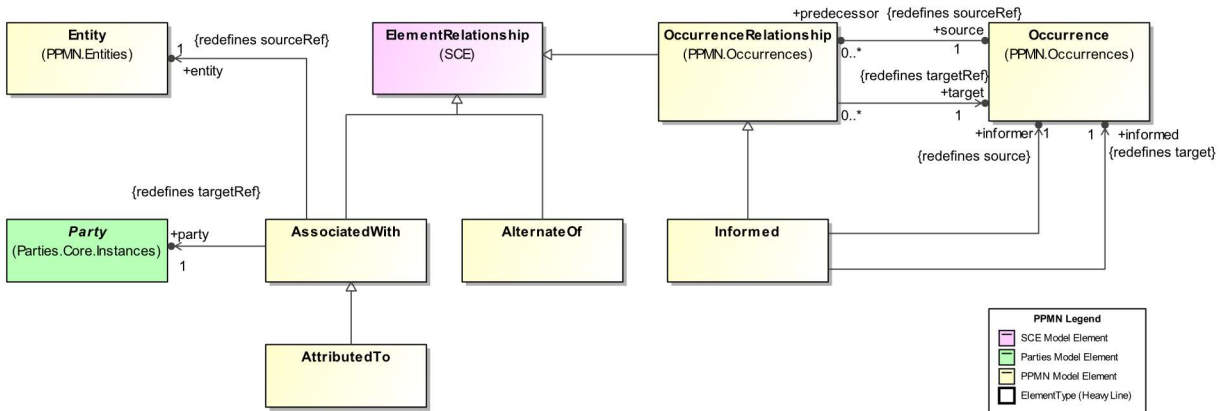
2062 Attribution is captured through the *AttributedTo* relationship. This element states that an entity of interest was
2063 generated through some unknown activity or action of the *Party*.

2064 The *AlternateOf* relationship states that two entities or elements represent the same thing or aspects of the same
2065 thing. The *AlternateOf* relationship will generally between two entities of some type. However, this is not always
2066 the case. It may be useful in certain situations to note alternate *Parties* or *Occurrences*. Note that the `source` and
2067 `target` of the *AlternateOf* must both be of the same general "type". In other words they must both be, for
2068 example, *Entities*, or both be *Parties*, or both be *Occurrences*.

2069 The *Informed* relationship is used to show that one *Occurrence* provided information or insight to or in some way
2070 affected another *Occurrence*. For example a testing process may inform a redesign of an assembly line for a
2071 manufacturer.

2072

2073



2074
2075 **Figure 47: Additional PPMN Relationships**

2076 **8.9.1 AlternateOf**

2077 The *AlternateOf* relationship is a kind of *ElementRelationship* that states that two elements represent the same thing
2078 or aspects of the same thing. The *AlternateOf* relationship will generally between two entities of some type.
2079 However, this is not always the case. It may be useful in certain situations to note alternate *Parties* or *Occurrences*.
2080 Note that the *source* and *target* of the *AlternateOf* must both be of the same general "type". In other words
2081 they must both be, for example, *Entities*, or both be *Parties*, or both be *Occurrences*.

2082 **Generalizations**

2083 The *AlternateOf* element inherits the attributes and/or associations of:

- 2084 • *ElementRelationship* (see the *SCE* specification for more information).

2085 **Properties**

2086 The *AlternateOf* element does not have any additional attributes and/or associations.

2087 **8.9.2 AssociatedWith**

2088 The *AssociatedWith* relationship is a kind of *ElementRelationship* that captures the fact that a *Party* is associated in
2089 some way with an *Entity*.

2090 **Generalizations**

2091 The *AssociatedWith* element inherits the attributes and/or associations of:

- 2092 • *ElementRelationship* (see the *SCE* specification for more information).

2093 **Properties**

2094 The following table presents the additional attributes and/or associations for *AssociatedWith*:

Table 76. AttributedTo Attributes and/or Associations

Property/Association	Description
entity : Entity [1]	An entity that is associated with some <i>Party</i> .
party : Party [1]	The <i>Party</i> to which some entity is associated.

2095

2096 **8.9.3 AttributedTo**

2097 The *AttributedTo* relationship is a kind of *AssociatedWith* relationship that captures the fact that some activity or
2098 action of a *Party* created, transformed, or destroyed an Entity.

2099 **Generalizations**

2100 The *AttributedTo* element inherits the attributes and/or associations of:

- 2101 • *AssociatedWith* (see the section entitled "[AssociatedWith](#)" for more information).

2102 **8.9.4 Informed**

2103 The *Informed* relationship is a kind of *ElementRelationship* that is used to show that one *Occurrence* provided
2104 information or insight to or in some way affected another *Occurrence*.

2105 **Generalizations**

2106 The *Informed* element inherits the attributes and/or associations of:

- 2107 • *OccurrenceRelationship* (see the section entitled "[OccurrenceRelationship](#)" for more information).

2108 **Properties**

2109 The following table presents the additional attributes and/or associations for *Informed*:

Table 77. Informed Attributes and/or Associations

Property/Association	Description
informed : Occurrence [1]	The <i>Occurrence</i> that was informed by the source <i>Occurrence</i> .
informer : Occurrence [1]	The <i>Occurrence</i> that informed another <i>Occurrence</i> .

2110

2111 **8.10 Packaging**

2112 **PPMN Packaging** consists of elements that allow users to group or "package up" sets of occurrences associated with
2113 the pedigree and provenance of entities of interest as well as elements that define expected occurrences. The
2114 packaging follows the pattern laid out in the Specification Common elements (SCE) specification and used in the
2115 Parties specification as well.

2116 The Pedigree and Provenance Metamodel and Notation supports the capture of events that happen in the lifecycle of
2117 entities of interest including creation, evolution, destruction, as well as changes in ownership and custody. In
2118 addition to capturing events that happened in the past, the specification also enables specifying events that are
2119 expected to happen in the future. As stated previously, these elements are loosely referred to as the "instances" and
2120 "types", respectively. The main packaging structures of PPMN support packaging of these elements using
2121 *PPMNInstances* and the *PPMNDefinitions* elements.

2122 *PPMNInstances* are specializations of *PartyInstances* and are designed to group "instances" related to events that
2123 have taken place in the lifecycle of entities of interest. These elements include actual events or *Occurrences*, the
2124 *Entities*, and the *Parties* involved.

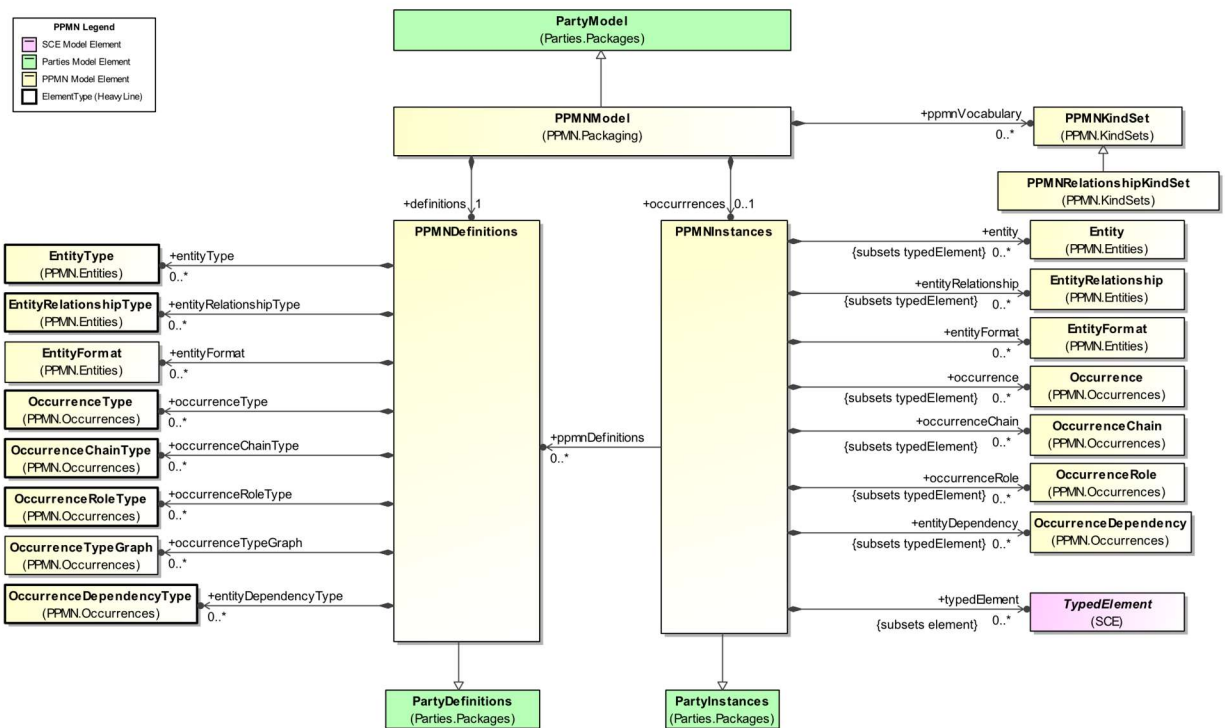
2125 *PPMNDefinitions* are specializations of *PartyDefinitions* and are designed to group the **PPMN** "types", i.e. the
2126 elements related to "expected" *Occurrences*. These elements include *OccurrenceTypes*, *EntityType*s, and *PartyTypes*
2127 among others.

2128 *PPMNInstances* and *PPMNDefinitions* together are included in *PPMNModels* along with relevant *PPMNKinds*.
2129 *PPMNModels* represent the semantics of the model versus the presentation elements contained in the *PPMNDI*
2130 package. *PPMNModels* are specializations of *PartyModels* and so may include *PartyInstances* and *PartyDefinitions*
2131 as well.

2132 All of these elements are brought together as a complete bundle in the *PPMNModelPackage*. *PPMNModelPackages*
 2133 contain both the model elements via the `model` property as well as the presentation elements via the
 2134 `presentation` property. *PPMNModelPackages* are a specialization of *PartyModelPackage* and so may contain
 2135 all of the *Party*-related elements contained therein.

2136 All **PPMN** packages and models are specializations of *SCE Package* and as such can contain other *SCE Packages*
 2137 or their specializations. They can also include imports of external elements through the *Import* element in *SCE*.

2138
 2139
 2140
 2141



2142
 2143 **Figure 48: PPMN Packaging**

2144 **8.10.1 PPMNDefinitions**

2145 A kind of *PartyDefinitions* that is the container for "Type-related" PPMN elements. Type-related elements include
 2146 elements such as *OccurrenceChainTypes* and its specializations, *OccurrenceTypes* and its specializations, and
 2147 profiles. Type-related elements are contained in *TypePackages*.

2148 **Generalizations**

2149 The *PPMNDefinitions* element inherits the attributes and/or associations of:

- 2150 • *PartyDefinitions* (see the section entitled "[PartyDefinitions](#)" for more information).

2151 **Properties**

2152 The following table presents the additional attributes and/or associations for *PPMNDefinitions*:

Table 78. PPMNDefinitions Attributes and/or Associations

Property/Association	Description
entityDependencyType : OccurrenceDependencyType [0..*]	A list of <i>EntityDependencyTypes</i> within the <i>PPMNModel</i> .
entityFormat : EntityFormat [0..*]	A list of the <i>EntityFormats</i> referenced within the <i>PPMNDefinitions</i> package.
entityRelationshipType : EntityRelationshipType [0..*]	A list of <i>EntityRelationshipTypes</i> within the <i>PPMNModel</i> .
entityType : EntityType [0..*]	A list of <i>EntitieTypes</i> within the <i>PPMNModel</i> .
occurrenceChainType : OccurrenceChainType [0..*]	A list of <i>OccurrenceChainTypes</i> within the <i>PPMNModel</i> .
occurrenceRoleType : OccurrenceRoleType [0..*]	A list of <i>OccurrenceRoleTypes</i> within the <i>PPMNModel</i> .
occurrenceType : OccurrenceType [0..*]	A list of <i>OccurrenceTypes</i> within the <i>PPMNModel</i> .
occurrenceTypeGraph : OccurrenceTypeGraph [0..*]	A list of <i>OccurrenceTypeGraphs</i> within the <i>PPMNModel</i> .

2153

2154 **8.10.2 PPMNInstances**

2155 **PPMN** information sets are exchanged in bulk through the *OccurrenceSet* element. The *OccurrenceSet* element
 2156 provides the outermost container for other **PPMN** elements contained in one or more *PPMNPackages*. The
 2157 occurrence chains, occurrences and other "instance-related" elements are contained within one or more
 2158 *OccurrenceSets* while "type-related" elements such as *OccurrenceChainTypes*, and *OccurrenceTypes* if present are
 2159 contained within definitions packages.

2160 **Generalizations**

2161 The *PPMNInstances* element inherits the attributes and/or associations of:

- 2162 • *PartyInstances* (see the section entitled "[PartyInstances](#)" for more information).

2163 **Properties**

2164 The following table presents the additional attributes and/or associations for *PPMNInstances*:

Table 79. PPMNInstances Attributes and/or Associations

Property/Association	Description
entity : Entity [0..*]	A list of <i>Entities</i> of interest within the <i>PPMNModel</i> .
entityDependency : OccurrenceDependency [0..*]	A list of <i>EntityDependencies</i> within the <i>PPMNModel</i> .

entityFormat : EntityFormat [0..*]	A list of the <i>EntityFormats</i> referenced within the <i>PPMNInstances</i> package.
entityRelationship : EntityRelationship [0..*]	A list of <i>EntityRelationships</i> within the <i>PPMNModel</i> .
occurrence : Occurrence [0..*]	A list of <i>Occurrences</i> within the <i>PPMNModel</i> .
occurrenceChain : OccurrenceChain [0..*]	A list of <i>OccurrenceChains</i> within the <i>PPMNModel</i> .
occurrenceRole : OccurrenceRole [0..*]	A list of <i>OccurrenceRoles</i> within the <i>PPMNModel</i> .
ppmnDefinitions : PPMNDefinitions [0..*]	The property refers to zero or more <i>PPMNDefinitions</i> packages that contains the <i>ElementTypes</i> that provide a basis for the instances contained in the <i>PartyInstances</i> package.
typedElement : TypedElement [0..*]	A list of <i>TypedElements</i> within the <i>PPMNModel</i> .

2165

2166 8.10.3 PPMNModel

2167 A *PPMNModel* is the main container for semantic elements of a **PPMN** model including types, instances, and
 2168 KindSets. As a specialization of *PartyModel* it also contains Party-related types, and instances. These elements are
 2169 separate from the visual elements included in the *PPMNModelPackage*.

2170 Generalizations

2171 The *PPMNModel* element inherits the attributes and/or associations of:

- 2172 • *PartyModel* (see the section entitled "[PartyModel](#)" for more information).

2173 Properties

2174 The following table presents the additional attributes and/or associations for *PPMNModel*:

Table 80. PPMNModel Attributes and/or Associations

Property/Association	Description
definitions : PPMNDefinitions [1]	The packages that contain the elements that represent the definitions of a PPMN model. These elements generally include the type elements.
occurrences : PPMNInstances [0..1]	The packages that contain the elements that represent the definitions of a PPMN model. These elements generally include the type elements.
ppmnVocabulary : PPMNVocabulary [0..*]	The <i>ppmnVocabulary</i> is a list of terms (as <i>Kinds</i>) that provide an extensible mechanism to define the elements of enumerations in a <i>PPMNModel</i> .

2175

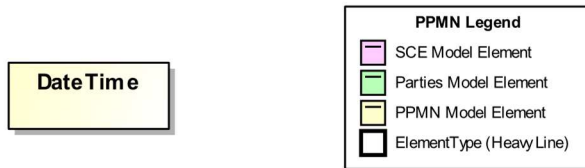
2176

2177 8.11 Primitives

2178 The Primitives package contains primitive data elements used by other packages in PPMN.

2179 PPMN uses the four primitives shown in the figure in addition to other UML primitives.

2180



2181

2182 **Figure 49: PPMN Primitives**

2183 8.11.1 DateTime

2184 A primitive that captures a point in time including a date and the time of day to greatest precision practical.

2185 Generalizations

2186 The *DateTime* element does not inherit any attributes or associations of from another element.

2187 Properties

2188 The *DateTime* element does not have any additional attributes and/or associations.

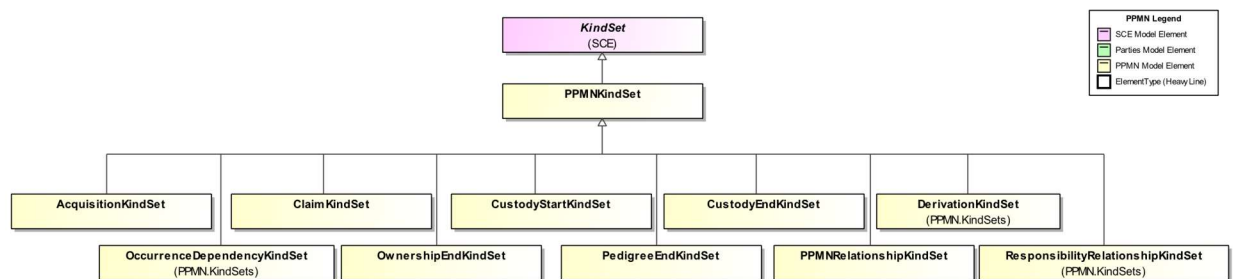
2189 8.12 KindSets

2190 *PPMNKindSet* are sets of terms used within a **PPMN** model that are defined by an external ontology. The terms link to formal definitions for the terms used within the model. The *Kind* element, or a specialization thereof, is used to name the term and provide a link to the definitions. *PPMNVocabularies* are contained within a *PPMNModel* package.

2194 The following figure presents the elements related to the **PPMN KindSets** section:

2195

2196



2197

2198 **Figure 50: PPMN KindSets**

2199 8.12.1 PPMNKindSet

2200 *PPMNKindSet* are lists of terms used as possible values for properties within PPMN. The terms are specializations of the *Kind* element that can be used to relate to the term to an external definition or meaning. The terms themselves

2201

2202 do not represent the definitions or meanings but provide links to an external source. The KindSet mechanism is used
2203 to support extensibility of the specification.

2204 **Generalizations**

2205 The *PPMNKindSet* element inherits the attributes and/or associations of:

- 2206 • *KindSet* (see the SCE specification for more information).

2207 **Properties**

2208 The *PPMNKindSet* element does not have any additional attributes and/or associations.

2209 **8.12.2 AcquisitionKindSet**

2210 A kind of *PPMNKindSet* that includes terms that specify how a *ChainOfOwnership* was started.

2211 **Generalizations**

2212 The *AcquisitionKindSet* element inherits the attributes and/or associations of:

- 2213 • *PPMNKindSet* (see the section entitled “PPMNKindSet” for more information).

2214 **Properties**

2215 The following table presents the additional attributes and/or associations for *AcquisitionKindSet*:

Table 81. AcquisitionKindSet Attributes and/or Associations

Property/Association	Description
term : AcquisitionKind [1..*]	A list of the terms representing valid <i>AcquisitionKinds</i> .

2216

2217 **8.12.3 ClaimKindSet**

2218 A kind of *PPMNKindSet* that includes terms that indicate the kind of *Claim* that has been made.

2219 **Generalizations**

2220 The *ClaimKindSet* element inherits the attributes and/or associations of:

- 2221 • *PPMNKindSet* (see the section entitled “[PPMNKindSet](#)” for more information).

2222 **Properties**

2223 The following table presents the additional attributes and/or associations for *ClaimKindSet*:

Table 82. ClaimKindSet Attributes and/or Associations

Property/Association	Description
term : ClaimKind [1..*]	A list of the terms representing valid <i>ClaimKinds</i> within a PPMN Model.

2224

2225 **8.12.4 CustodyEndKindSet**

2226 A kind of *PPMNKindSet* that includes terms that specify how a *ChainOfCustody* was ended.

2227 **Generalizations**

2228 The *CustodyEndKindSet* element inherits the attributes and/or associations of:

- 2229 • *PPMNKindSet* (see the section entitled “[PPMNKindSet](#)” for more information).

2230 **Properties**

2231 The following table presents the additional attributes and/or associations for *CustodyEndKindSet*:

Table 83. CustodyEndKindSet Attributes and/or Associations

Property/Association	Description
term : CustodyEndKind [1..*]	A list of the terms representing valid <i>CustodyEndKinds</i> within a PPMN Model.

2232

2233 **8.12.5 CustodyStartKindSet**

2234 A kind of *PPMNKindSet* that includes terms that specify how a *ChainOfCustody* was started.

2235 **Generalizations**

2236 The *CustodyStartKindSet* element inherits the attributes and/or associations of:

- 2237 • *PPMNKindSet* (see the section entitled “[PPMNKindSet](#)” for more information).

2238 **Properties**

2239 The following table presents the additional attributes and/or associations for *CustodyStartKindSet*:

Table 84. CustodyStartKindSet Attributes and/or Associations

Property/Association	Description
term : CustodyStartKind [1..*]	A list of the terms representing valid <i>CustodyStartKinds</i> within a PPMN Model.

2240

2241 **8.12.6 DerivationKindSet**

2242 A kind of *PPMNKindSet* that includes terms that specify the type of derivation relationship that exists between two
2243 *Entities*.

2244 **Generalizations**

2245 The *DerivationKindSet* element inherits the attributes and/or associations of:

- 2246 • *PPMNKindSet* (see the section entitled “[PPMNKindSet](#)” for more information).

2247 **Properties**

2248 The following table presents the additional attributes and/or associations for *DerivationKindSet*:

Table 85. DerivationKindSet Attributes and/or Associations

Property/Association	Description
term : DerivationKind [1..*]	A list of the terms representing valid <i>DerivationTypes</i> within a PPMN Model.

2249

2250 8.12.7 OccurrenceDependencyKindSet

2251 A kind of *PPMNKindSet* that includes terms that specify how the type of dependency an *Occurrence* has on an
2252 *Entity*.

2253 Generalizations

2254 The *OccurrenceDependencyKindSet* element inherits the attributes and/or associations of:

- 2255 • *PPMNKindSet* (see the section entitled “[PPMNKindSet](#)” for more information).

2256 Properties

2257 The following table presents the additional attributes and/or associations for *OccurrenceDependencyKindSet*:

Table 86. OccurrenceDependencyKindSet Attributes and/or Associations

Property/Association	Description
term : OccurrenceDependencyKind [1..*]	A list of the terms representing valid OccurrenceDependencies within a PPMN Model.

2258

2259 8.12.8 OwnershipEndKindSet

2260 A kind of *PPMNKindSet* that includes terms that specify how the *ChainOfOwnership* was ended.

2261 Generalizations

2262 The *OwnershipEndKindSet* element inherits the attributes and/or associations of:

- 2263 • *PPMNKindSet* (see the section entitled “[PPMNKindSet](#)” for more information).

2264 Properties

2265 The following table presents the additional attributes and/or associations for *OwnershipEndKindSet*:

Table 87. OwnershipEndKindSet Attributes and/or Associations

Property/Association	Description
term : OwnershipEndKind [1..*]	A list of the terms representing valid <i>OwnershipEndKinds</i> within a PPMN Model.

2266

2267 **8.12.9 PedigreeEndKindSet**

2268 A kind of *PPMNKindSet* that includes terms that specify the kind of relationship between two PPMN elements.

2269 **Generalizations**

2270 The *PedigreeEndKindSet* element inherits the attributes and/or associations of:

- 2271 • *PPMNKindSet* (see the section entitled "[PPMNKindSet](#)" for more information).

2272 **Properties**

2273 The *PedigreeEndKindSet* element does not have any additional attributes and/or associations.

2274 **8.12.10 PPMNRelationshipKindSet**

2275 A kind of *PPMNKindSet* that includes terms that specify the kind of relationship between two PPMN elements.

2276 **Generalizations**

2277 The *PPMNRelationshipKindSet* element inherits the attributes and/or associations of:

- 2278 • *PPMNKindSet* (see the section entitled "[PPMNKindSet](#)" for more information).

2279 **Properties**

2280 The following table presents the additional attributes and/or associations for *PPMNRelationshipKindSet*:

Table 88. PPMNRelationshipKindSet Attributes and/or Associations

Property/Association	Description
term : RelationshipKind [0..*]	A list of the terms representing valid <i>RelationshipKinds</i> within a PPMN model.

2281

2282 **8.12.11 ResponsibilityRelationshipKindSet**

2283 A kind of *PPMNKindSet* that includes terms that specify the kind of *ResponsibilityRelationship* exists between one
2284 or more *Parties* and an *Entity*.

2285 **Generalizations**

2286 The *ResponsibilityRelationshipKindSet* element inherits the attributes and/or associations of:

- 2287 • *PPMNKindSet* (see the section entitled "[PPMNKindSet](#)" for more information).

2288 **Properties**

2289 The following table presents the additional attributes and/or associations for *ResponsibilityRelationshipKindSet*:

Table 89. ResponsibilityRelationshipKindSet Attributes and/or Associations

Property/Association	Description
term : ResponsibilityRelationshipKind [0..*]	A list of the terms representing valid <i>RelationshipKinds</i> within a PPMN model.

2290

2291 **9 PPMN Library**

2292 A Library is included in **PPMN** to provide standard instances that are intended to be implemented by tools
2293 supporting **PPMN**. Currently, **PPMN** defines the instances for *AcquisitionKinds*, *ClaimKinds*, *CustodyStartKinds*,
2294 *CustodyEndKinds*, *OwnershipEndKinds*, *PedigreeEndKinds*, and *RelationshipKinds* (See following sections).

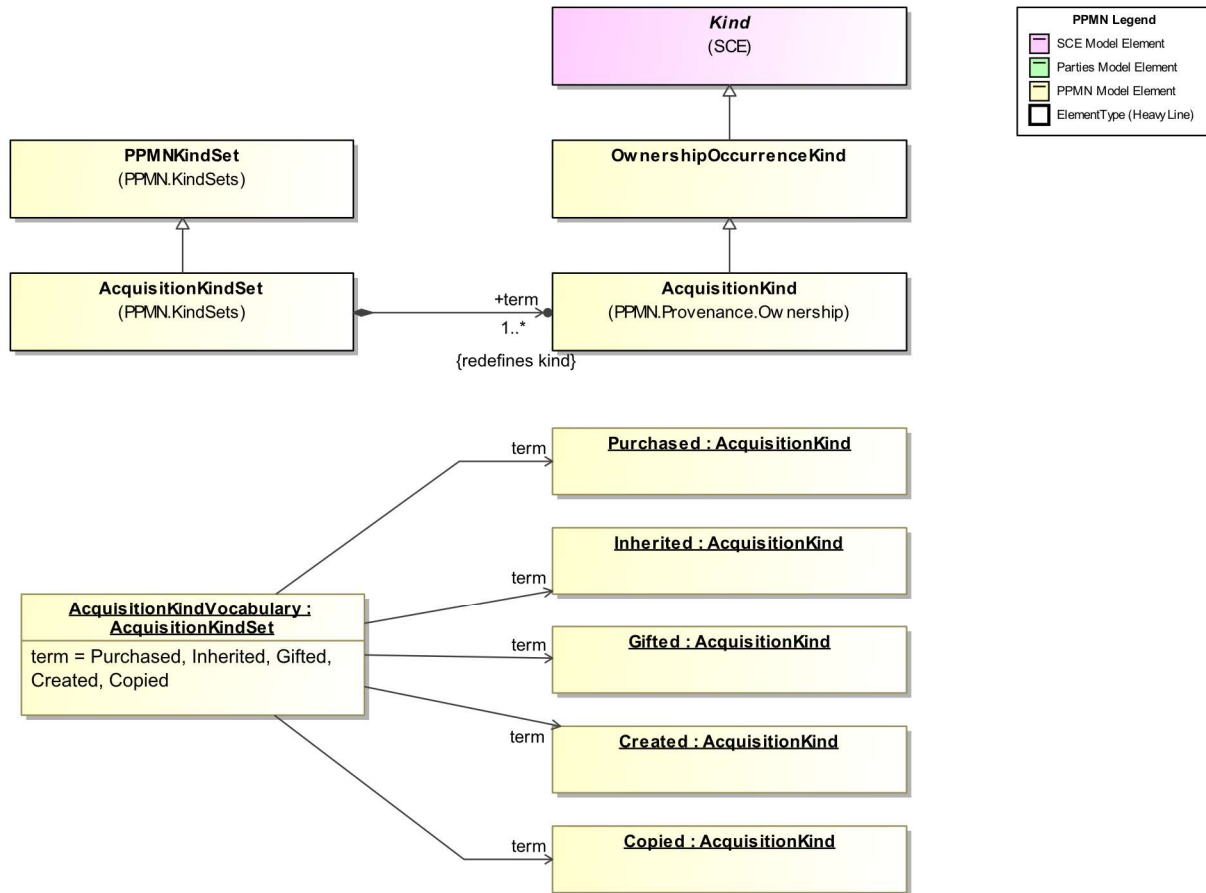
2295 **9.1 AcquisitionKinds**

2296 The *AcquisitionKinds* library contains instances that represent the standard ways in which ownership of an entity
2297 may begin. These elements are instances of *AcquisitionKind*. The set can be extended with additional instances of
2298 *AcquisitionKind* or a specialization thereof.

2299 The following figure presents the instances of the *AcquisitionKind* element that are terms for the
2300 *AcquisitionKindSet*:

2301

2302



2303
2304
2305
2306

Figure 51: AcquisitionKinds

The following table provides a definition of the terms included in the *AcquisitionKinds* set.

Table 90. AcquisitionKinds KindSet

#	Name	Documentation
1	AcquisitionKindSet	A kind of PPMNKindSet that includes terms that specify how a <i>ChainOfOwnership</i> was started.
2	Copied	An instance that indicates that a Party gained ownership of an entity by copying another entity.
3	Created	An instance that indicates that a Party gained ownership of an entity by creating it.
4	Gifted	An instance that indicates that a Party gained ownership of an entity by receiving it as a gift.
5	Inherited	An instance that indicates that a Party gained ownership of an entity as part of an inheritance.
6	Purchased	An instance that indicates that a Party gained ownership of an entity by purchasing the entity.

2307 **9.1.1 AcquisitionKindSet**

2308 A kind of *PPMNKindSet* that includes terms that specify how a *ChainOfOwnership* was started.

2309 **9.1.2 Copied**

2310 An instance that indicates that a Party gained ownership of an entity by copying another entity.

2311 **9.1.3 Created**

2312 An instance that indicates that a Party gained ownership of an entity by creating it.

2313 **9.1.4 Gifted**

2314 An instance that indicates that a Party gained ownership of an entity by receiving it as a gift.

2315 **9.1.5 Inherited**

2316 An instance that indicates that a Party gained ownership of an entity as part of an inheritance.

2317 **9.1.6 Purchased**

2318 An instance that indicates that a Party gained ownership of an entity by purchasing the entity.

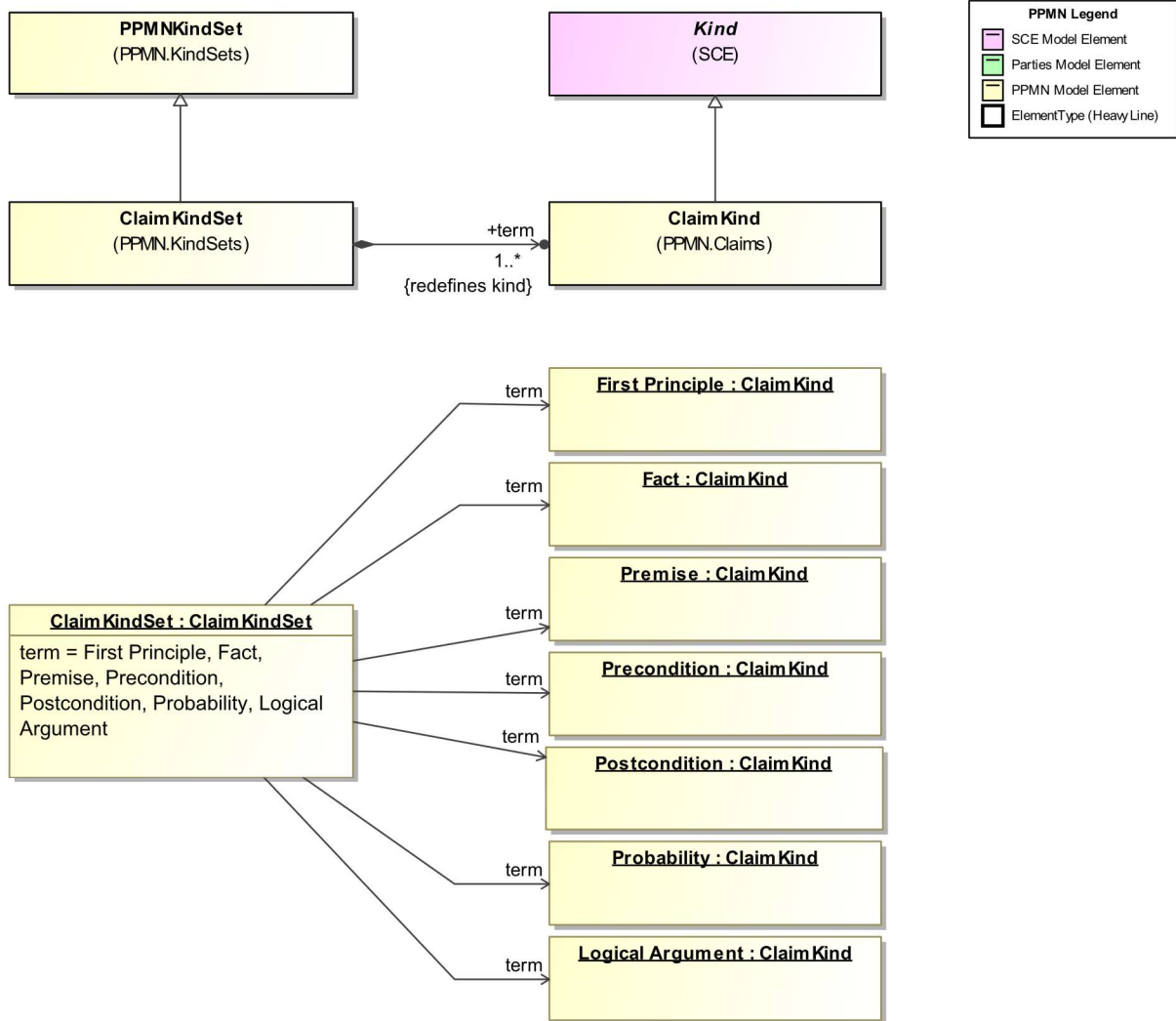
2319 **9.2 ClaimKinds**

2320 The *ClaimKinds* library contains instances that represent the standard types of claims that can be made in regards to
2321 a set of PPMN elements. These elements are instances of *ClaimKind*. The KindSet can be extended with additional
2322 instances of *ClaimKinds* or a specialization thereof.

2323 The following figure presents the instances of the *ClaimKind* element that are terms for the ClaimKindSet:

2324

2325



2326
2327
2328
2329

Figure 52: ClaimKinds

The following table provides a definition of the terms included in the *ClaimKinds* set.

Table 91. ClaimKinds KindSet

#	Name	Documentation
1	ClaimKindSet	A set of terms that specify the kinds of claims may be made.
2	Fact	A basic assertion.
3	First Principle	A foundational assertion that is held as true.
4	Logical Argument	An assertion that is based on other assertions.
5	Postcondition	An assertion that is assumed to be true at the end of a process.
6	Precondition	An assertion that is assumed to be true at the start of a process.
7	Premise	An assertion that is used in a logical argument.

#	Name	Documentation
8	Probability	An assertion that indicates some degree of truth.

2330 **9.2.1 ClaimKindSet**

2331 A set of terms that specify the kinds of claims may be made.

2332 **9.2.2 Fact**

2333 A basic assertion.

2334 **9.2.3 First Principle**

2335 A foundational assertion that is held as true.

2336 **9.2.4 Logical Argument**

2337 An assertion that is based on other assertions.

2338 **9.2.5 Postcondition**

2339 An assertion that is assumed to be true at the end of a process.

2340 **9.2.6 Precondition**

2341 An assertion that is assumed to be true at the start of a process.

2342 **9.2.7 Premise**

2343 An assertion that is used in a logical argument.

2344 **9.2.8 Probability**

2345 An assertion that indicates some degree of truth.

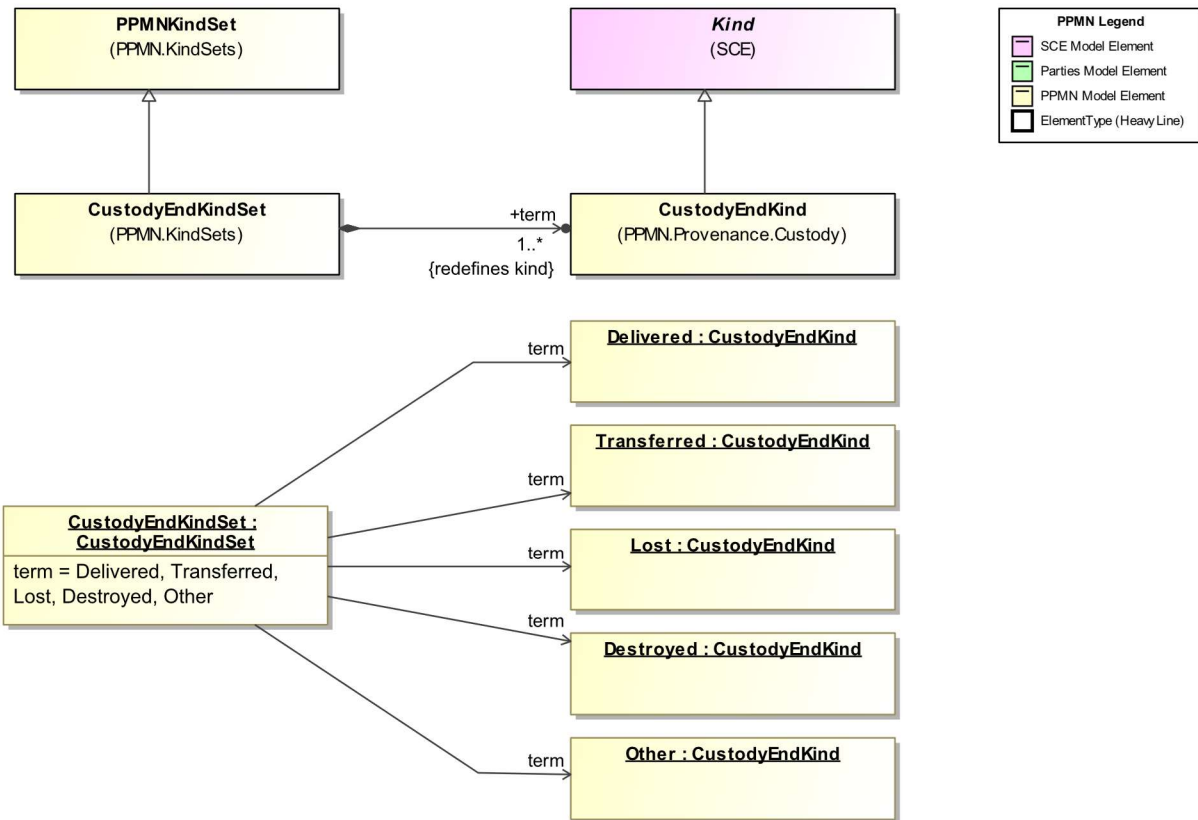
2346 **9.3 CustodyEndKinds**

2347 The *CustodyEndKinds* library contains instances that represent the standard ways in which custody of an entity may
 2348 end. These elements are instances of *CustodyEndKind*. The set can be extended with additional instances of
 2349 *CustodyEndKind* or a specialization thereof.

2350 The following figure presents the instances of the *CustodyEndKind* element that are terms for the
 2351 *CustodyEndKindSet*:

2352

2353



2354
2355
2356
2357

Figure 53: CustodyEndKinds

The following table provides a definition of the terms included in the *CustodyEndKinds* set.

Table 92. CustodyEndKinds KindSet

#	Name	Documentation
1	CustodyEndKindSet	A set of terms that specify the kind of <i>CustodyOccurrence</i> that results in the end of a <i>ChainOfCustody</i> .
2	Delivered	An instance that specifies that an entity was delivered to some other <i>Party</i> .
3	Destroyed	An instance that specifies that an entity was destroyed.
4	Lost	An instance that specifies that an entity was lost.
5	Other	An instance that specifies that custody of an entity was relinquished in some other way.
6	Transferred	An instance that specifies that an entity was transferred to some other <i>Party</i> .

2358
2359
2360
2361

9.3.1 CustodyEndKindSet

A set of terms that specify the kind of *CustodyOccurrence* that results in the end of a *ChainOfCustody*.

9.3.2 Delivered

An instance that specifies that an entity was delivered to some other *Party*.

2362 **9.3.3 Destroyed**
2363 An instance that specifies that an entity was destroyed.

2364 **9.3.4 Lost**
2365 An instance that specifies that an entity was lost.

2366 **9.3.5 Other**
2367 An instance that specifies that custody of an entity was relinquished in some other way.

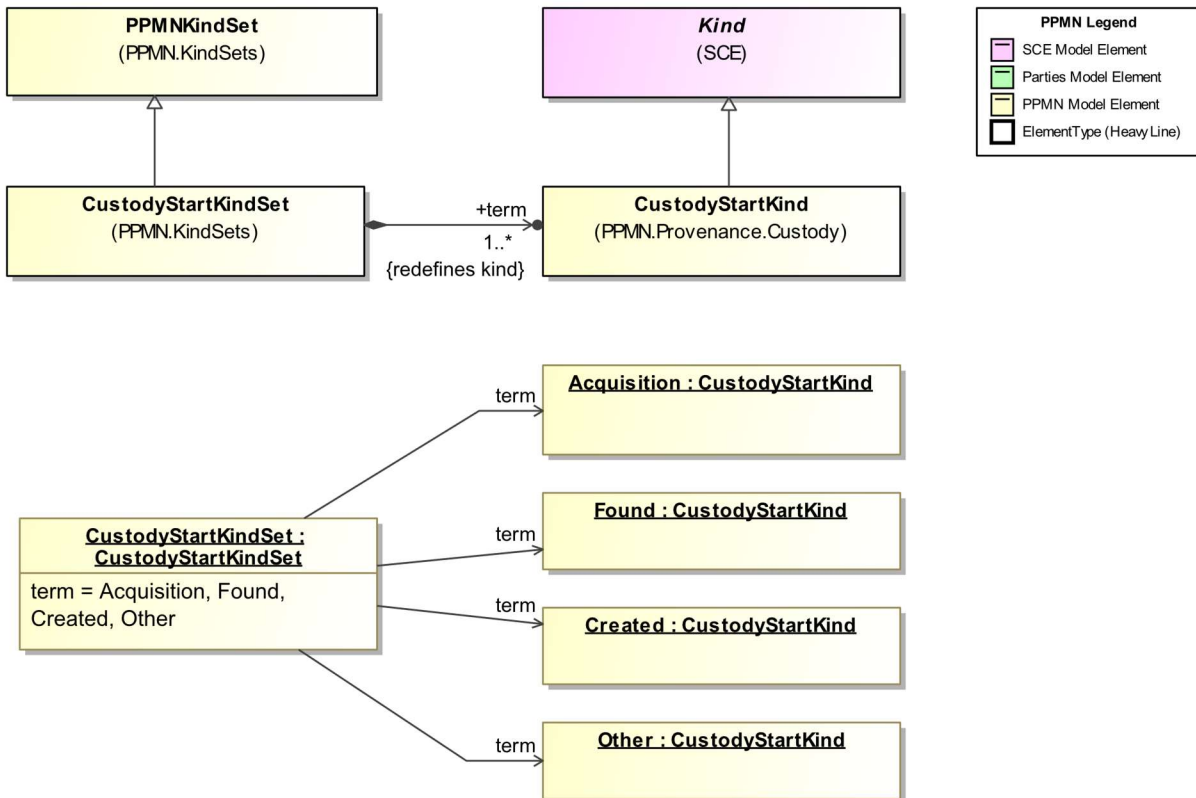
2368 **9.3.6 Transferred**
2369 An instance that specifies that an entity was transferred to some other *Party*.

2370 **9.4 CustodyStartKinds**

2371 The *CustodyStartKinds* library contains instances that represent the standard ways in which custody of an entity may
2372 begin. These elements are instances of *CustodyStartKind*. The set can be extended with additional instances of
2373 *CustodyStartKind* or a specialization thereof.

2374 The following figure presents the instances of the *CustodyStartKind* element that are terms for the
2375 *CustodyStartKindSet*:

2376
2377



2378
2379 **Figure 54: CustodyStartKinds**

2380 The following table provides a definition of the terms included in the *CustodyStartKinds* set.

2381

Table 93. CustodyStartKinds KindSet

#	Name	Documentation
1	CustodyStartKindSet	A set of terms that specify the kind of <i>CustodyOccurrence</i> that results in the start of a <i>ChainOfCustody</i> .
2	Acquisition	An instance that indicates that a <i>Party</i> gains custody of an entity through some type of acquisition.
3	Created	An instance that indicates that a <i>Party</i> gains custody of an entity by creation of the entity.
4	Found	An instance that indicates that a <i>Party</i> gains custody of an entity when the entity is found.
5	Other	An instance that indicates that a <i>Party</i> gains custody of an entity by some other event.

2382 **9.4.1 CustodyStartKindSet**

2383 A set of terms that specify the kind of *CustodyOccurrence* that results in the start of a *ChainOfCustody*.

2384 **9.4.2 Acquisition**

2385 An instance that indicates that a *Party* gains custody of an entity through some type of acquisition.

2386 **9.4.3 Created**

2387 An instance that indicates that a *Party* gains custody of an entity by creation of the entity.

2388 **9.4.4 Found**

2389 An instance that indicates that a *Party* gains custody of an entity when the entity is found.

2390 **9.4.5 Other**

2391 An instance that indicates that a *Party* gains custody of an entity by some other event.

2392 **9.5 DerivationKinds**

2393 The following table provides a definition of the terms included in the *DerivationKinds* set.

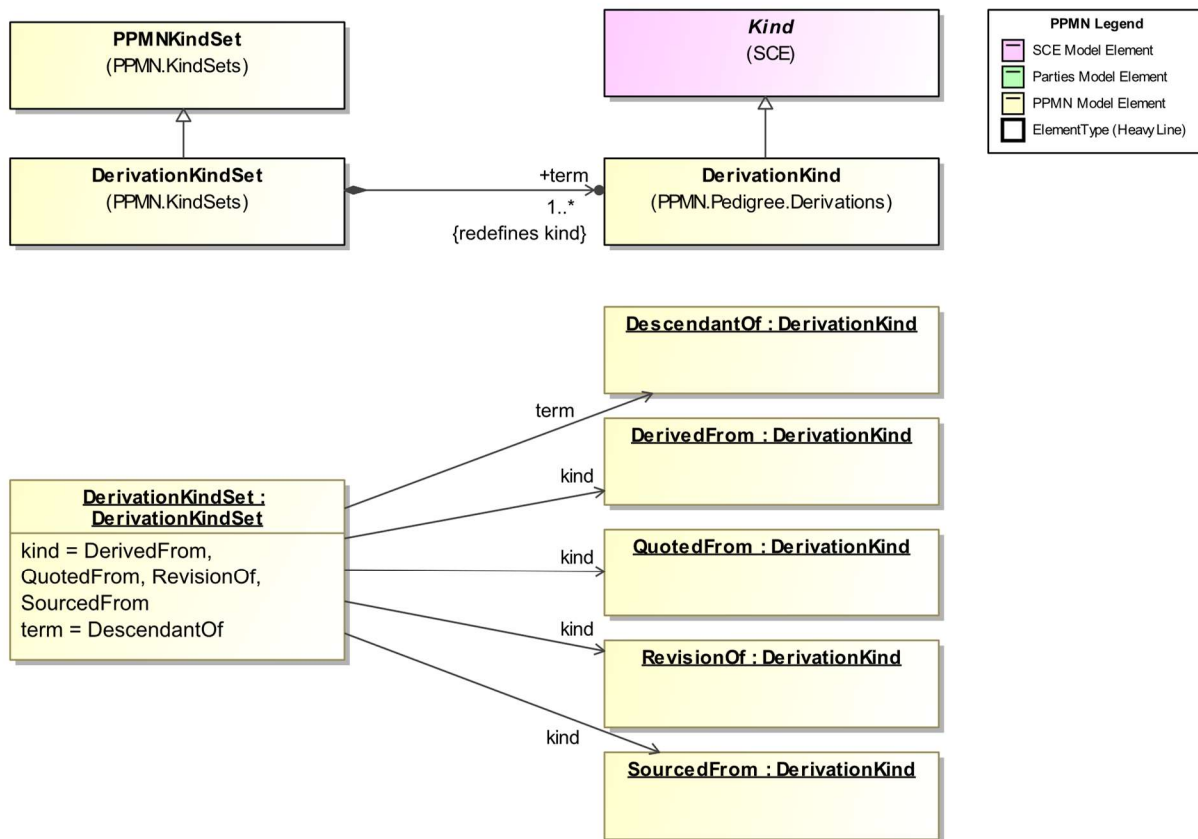
2394

Table 94. DerivationKinds KindSet

#	Name	Documentation
1	DerivationKindSet	A set of terms that specify the kind of derivation that exists between two <i>Entities</i> .
2	DerivedFrom	<i>DerivedFrom</i> indicates that source <i>EntityType</i> s are derived in some way from target <i>EntityType</i> s.
3	DescendantOf	<i>DescendantOf</i> indicates that source <i>EntityType</i> is a descendant of the target <i>EntityType</i> .
4	QuotedFrom	<i>QuotedFrom</i> indicates that source <i>EntityType</i> s are quoted from target <i>EntityType</i> s.

#	Name	Documentation
5	RevisionOf	RevisionOf indicates that source <i>EntityTypes</i> are revisions of target <i>EntityTypes</i> .
6	SourcedFrom	SourcedFrom indicates that source <i>EntityTypes</i> are sourced from from target <i>EntityTypes</i> which in turn are produced by some party potentially with some special experience or knowledge.

2395 The following figure presents the instances of the *RelationshipKind* element that are terms for the
 2396 PPMNRelationshipKindSet:
 2397
 2398



2399
 2400 **Figure 55: DerivationKinds**

2401 **9.5.1 DerivationKindSet**

2402 A set of terms that specify the kind of derivation that exists between two *Entities*.

2403 **9.5.2 DerivedFrom**

2404 DerivedFrom indicates that source *EntityTypes* are derived in some way from target *EntityTypes*.

2405 **9.5.3 DescendantOf**

2406 DescendantOf indicates that source *EntityType* is a descendant of the target *EntityType*.

2407 **9.5.4 QuotedFrom**

2408 QuotedFrom indicates that source *EntityTypes* are quoted from target *EntityTypes*.

2409 **9.5.5 RevisionOf**

2410 RevisionOf indicates that source *EntityTypes* are revisions of target *EntityTypes*.

2411 **9.5.6 SourcedFrom**

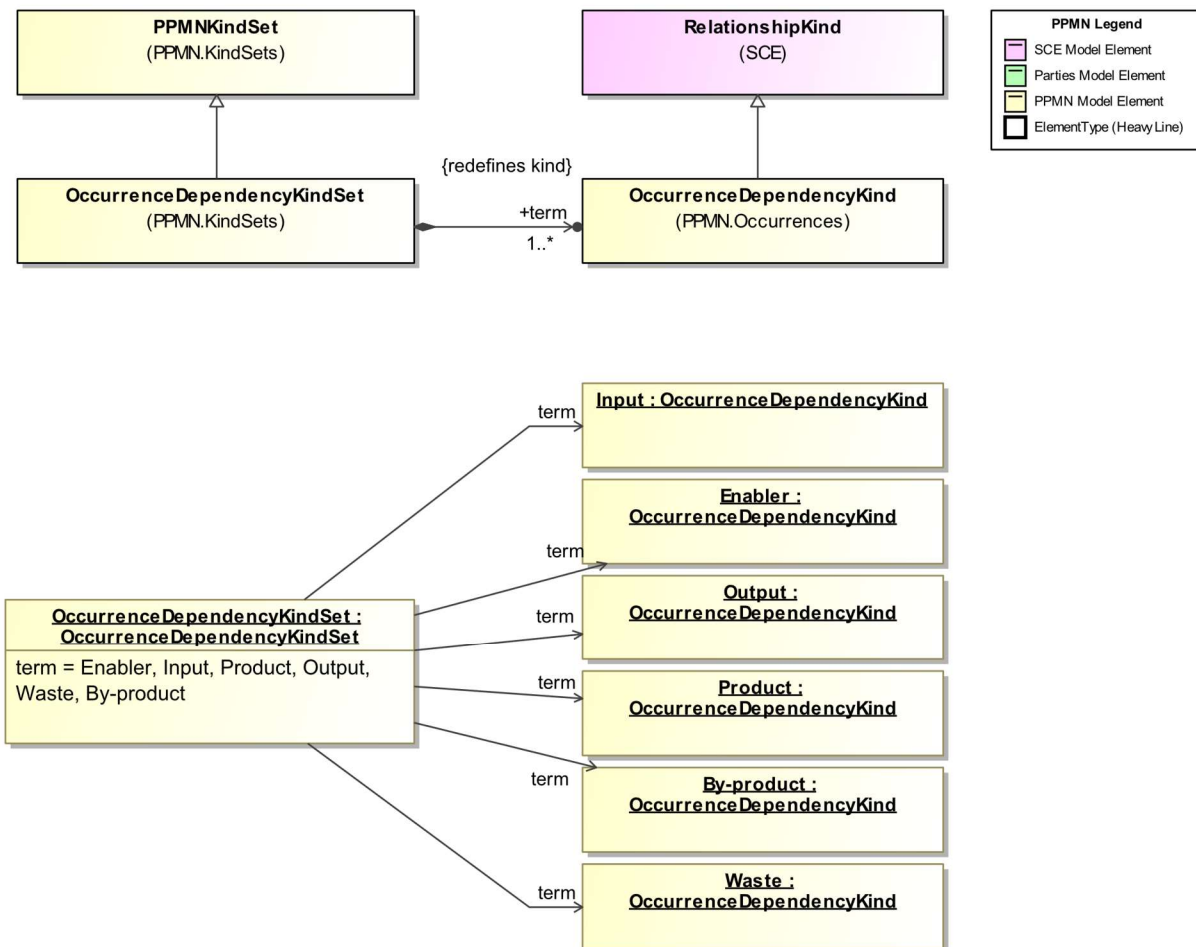
2412 SourcedFrom indicates that source *EntityTypes* are sourced from from target *EntityTypes* which in turn are
2413 produced by some party potentially with some special experience or knowledge.

2414 **9.6 OccurrenceDependencyKinds**

2415 The *OccurrenceDependencyKinds* library contains instances that represent the standard ways in which an
2416 *Occurrence* may depend on an *Entity*. These elements are instances of *OccurrenceDependencyKind*. The set can be
2417 extended with additional instances of *OccurrenceDependencyKind* or a specialization thereof.

2418 The following figure presents the instances of the *OccurrenceDependencyKind* element that are terms for the
2419 *OccurrenceDependencyKindSet*:

2420
2421



2422

2423 **Figure 56: OccurrenceDependencyKinds**

2424 The following table provides a definition of the terms included in the *OccurrenceDependencyKinds* set.

2425

Table 95. OccurrenceDependencyKinds KindSet

#	Name	Documentation
1	OccurrenceDependencyKindSet	A set of terms that specify the kind of <i>OccurrenceDependency</i> that exists between two <i>Occurrences</i> .
2	By-product	By-product indicates that the source <i>Occurrence</i> produces or creates the target <i>Entity</i> as a by-product during the course of the <i>Occurrence</i> .
3	Enabler	Enabler indicates that the source <i>Occurrence</i> uses the target <i>Entity</i> in some way that enables the <i>Occurrence</i> . However, the <i>Entity</i> is not used or become a part of any of the products or by-products of the <i>Occurrence</i> .
4	Input	Input indicates that the target <i>Entity</i> is an input to the source <i>Occurrence</i> is an input during the course of the <i>Occurrence</i> .
5	Output	Output indicates that the target <i>Entity</i> is an output of some kind of the <i>Occurrence</i> ..
6	Product	Product indicates that the source <i>Occurrence</i> produces or creates the target <i>Entity</i> during the course of the <i>Occurrence</i> . This is a more specific type of Output.
7	Waste	Waste indicates that the source <i>Occurrence</i> produces or creates the target <i>Entity</i> as waste during the course of the <i>Occurrence</i> . This is a more specific type of Output.

2426 **9.6.1 OccurrenceDependencyKindSet**

2427 A set of terms that specify the kind of *OccurrenceDependency* that exists between two *Occurrences*.

2428 **9.6.2 By-product**

2429 By-product indicates that the source *Occurrence* produces or creates the target *Entity* as a by-product
2430 during the course of the *Occurrence*.

2431 **9.6.3 Enabler**

2432 Enabler indicates that the source *Occurrence* uses the target *Entity* in some way that enables the
2433 *Occurrence*. However, the *Entity* is not used or become a part of any of the products or by-products of the
2434 *Occurrence*.

2435 **9.6.4 Input**

2436 Input indicates that the target *Entity* is an input to the source *Occurrence* during the course of the
2437 *Occurrence*.

2438 **9.6.5 Output**

2439 Output indicates that the target *Entity* is an output of some kind of the *Occurrence*..

2440 **9.6.6 Product**

2441 Product indicates that the source *Occurrence* produces or creates the target *Entity* during the course of the
 2442 *Occurrence*.

2443 **9.6.7 Waste**

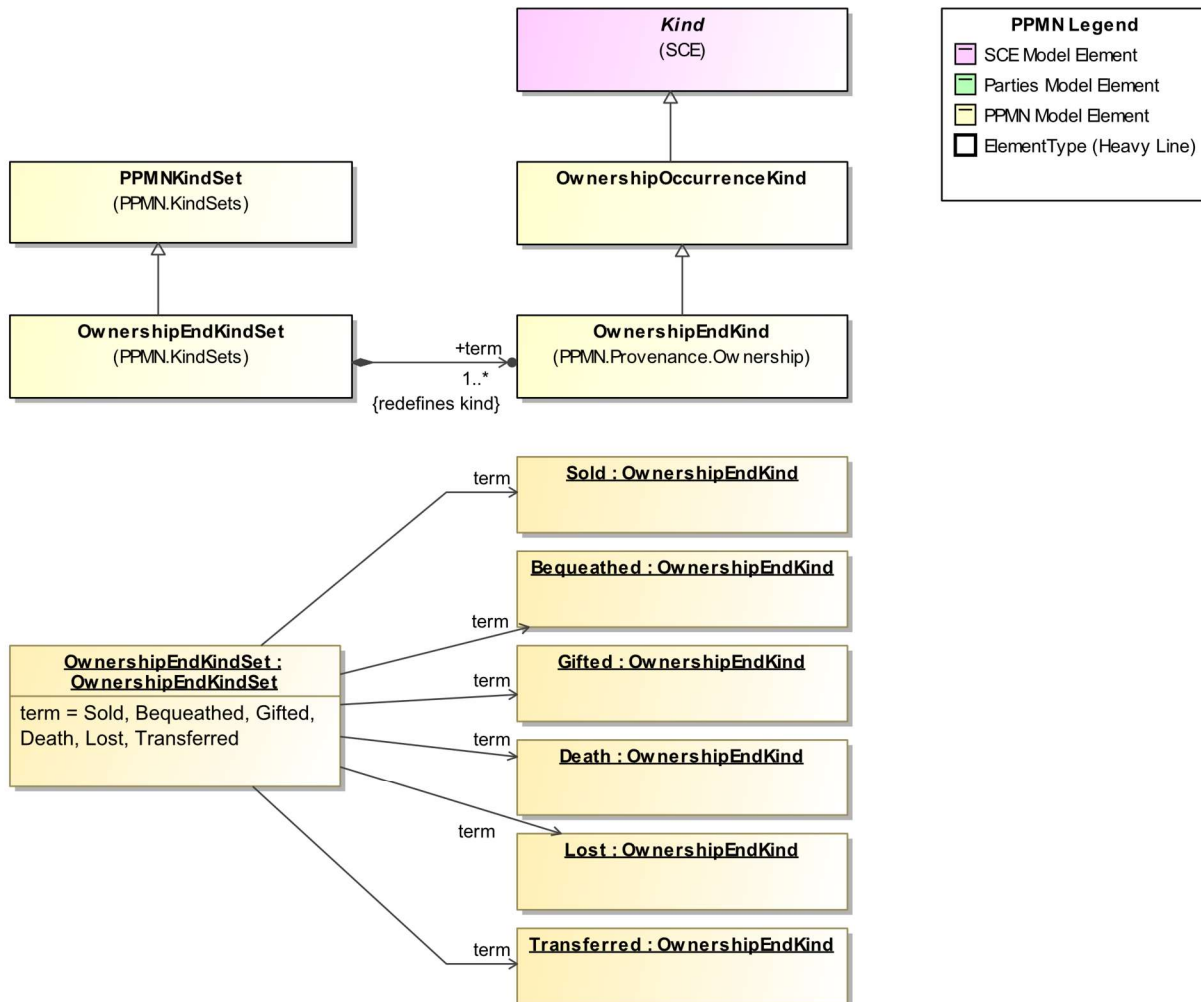
2444 Waste indicates that the source *Occurrence* produces or creates the target *Entity* as waste during the course of
 2445 the *Occurrence*.

2446 **9.7 OwnershipEndKinds**

2447 The *OwnershipEndKinds* library contains instances that represent the standard ways in which ownership of an entity
 2448 may end. These elements are instances of *OwnershipEndKind*. The set can be extended with additional instances of
 2449 *OwnershipEndKind* or a specialization thereof.

2450 The following figure presents the instances of the *OwnershipEndKind* element that are terms for the
 2451 *OwnershipEndKindSet*:

2452
 2453



2454
 2455 **Figure 57: OwnershipEndKinds**

2456 The following table provides a definition of the terms included in the *OwnershipEndKinds* set.

2457

Table 96. OwnershipEndKinds KindSet

#	Name	Documentation
1	OwnershipEndKindSet	A set of terms that specify how the <i>ChainOfOwnership</i> was ended.
2	Bequeathed	An instance that specifies that an entity was bequeathed to some other party.
3	Death	An instance that specifies that an entity died.
4	Gifted	An instance that specifies that an entity was gifted to some other <i>Party</i> .
5	Lost	An instance that specifies that an entity was lost.
6	Sold	An instance that specifies that an entity was sold to some other <i>Party</i> .
7	Transferred	An instance that specifies that ownership of an entity was transferred to some other <i>Party</i> .

2458 **9.7.1 OwnershipEndKindSet**

2459 A set of terms that specify how the *ChainOfOwnership* was ended.

2460 **9.7.2 Bequeathed**

2461 An instance that specifies that an entity was bequeathed to some other party.

2462 **9.7.3 Death**

2463 An instance that specifies that an entity died.

2464 **9.7.4 Gifted**

2465 An instance that specifies that an entity was gifted to some other *Party*.

2466 **9.7.5 Lost**

2467 An instance that specifies that an entity was lost.

2468 **9.7.6 Sold**

2469 An instance that specifies that an entity was sold to some other *Party*.

2470 **9.7.7 Transferred**

2471 An instance that specifies that ownership of an entity was transferred to some other *Party*.

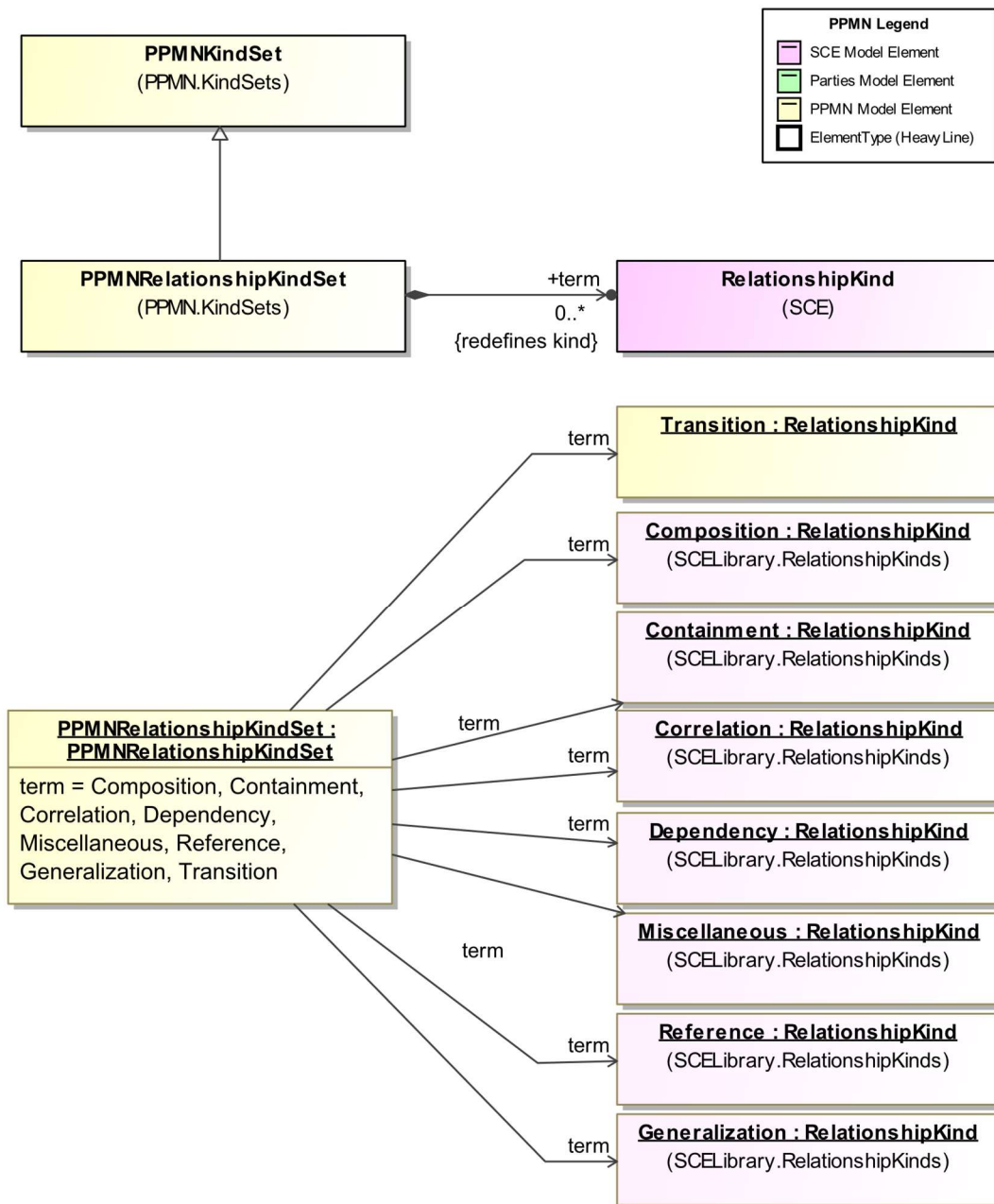
2472 **9.8 PPMNRelationshipKinds**

2473 The *PPMNRelationshipKinds* library contains instances that represent the standard types of relationships between
2474 **PPMN** elements. This library extends the **SCE RelationshipKinds** library of terms to add *Transition*. These
2475 elements are instances of *SCE RelationshipKind*. The set can be extended with additional instances of
2476 *RelationshipKind* or a specialization thereof.

2477 The following figure presents the instances of the *RelationshipKind* element that are terms for the
2478 *PPMNRelationshipKindSet*:

2479

2480



2481

2482

Figure 58: PPMNRelationshipKinds

2483

The following table provides a definition of the terms included in the *PPMNRelationshipKinds* Vocabulary.

2484

Table 97. PPMNRelationshipKinds KindSet

#	Name	Documentation
1	PPMNRelationshipKinds	A kind of PPMNKindSet that includes terms that specify the kind of relationship between two PPMN elements.
2	Composition	Composition indicates that the source element is composed of, in part, the target element. Other elements could be included in this composition.
3	Containment	Containment indicates that the source element is a container for the target element.
4	Correlation	Correlation indicates that the source element is correlated with the target element. This is often used when a mapping is required between the structures of two data elements.
5	Dependency	Dependency indicates that target element is dependent in some way on the source element.
6	Generalization	Generalization indicates that the source element is a generalization of the target element (which is based on and extends the source).
7	Miscellaneous	Miscellaneous indicates that source element has some relationship with the target element that is of a kind that is not expressed through the other <i>RelationshipKind</i> instances.
8	Reference	Reference indicates that source element references the target element.
9	Transition	Transition indicates that "flow" or sequencing moves from the source element to the target element.

2485 **9.8.1 PPMNRelationshipKinds**

2486 A kind of PPMNKindSet that includes terms that specify the kind of relationship between two PPMN elements.

2487 **9.8.2 Transition**

2488 Transition indicates that "flow" or sequencing moves from the source element to the target element.

2489 **9.8.3 Additional Terms from SCE**

2490 **9.8.3.1 Reference**

2491 Reference indicates that source element references the target element.

2492 **9.8.3.2 Miscellaneous**

2493 Miscellaneous indicates that source element has some relationship with the target element that is of a kind
2494 that is not expressed through the other *RelationshipKind* instances.

2495 **9.8.3.3 Composition**

2496 Composition indicates that the source element is composed of, in part, the target element. Other elements
2497 could be included in this composition.

2498 **9.8.3.4 Dependency**

2499 Dependency indicates that target element is dependent in some way on the source element.

2500 **9.8.3.5 Containment**

2501 Containment indicates that the source element is a container for the target element.

2502 **9.8.3.6 Correlation**

2503 Correlation indicates that the source element is correlated with the target element. This is often used
2504 when a mapping is required between the structures of two data elements.

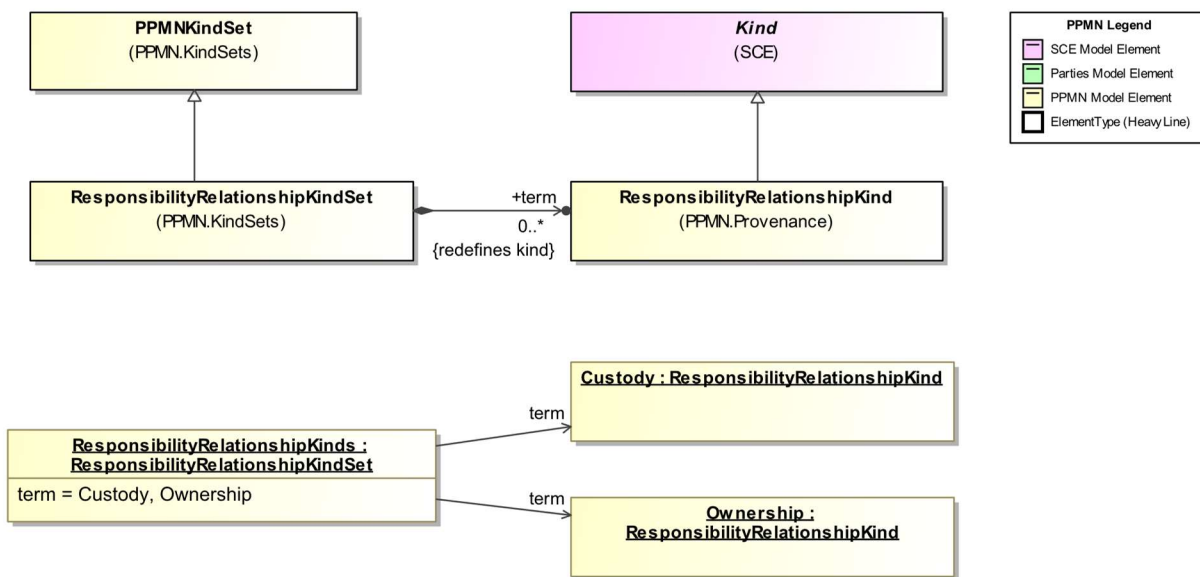
2505 **9.8.3.7 Generalization**

2506 Generalization indicates that the source element is a generalization of the target element (which is based
2507 on and extends the source).

2508 **9.9 ResponsibilityRelationshipKinds**

2509 The following figure presents the instances of the *ResponsibilityRelationshipKind* element that are terms for the
2510 *ResponsibilityRelationshipKindSet*:

2511
2512



2513
2514 **Figure 59: ResponsibilityRelationshipKinds**

2515 The following table provides a definition of the terms included in the *PPMNRelationshipKinds* set.

2516

Table 98. ResponsibilityRelationshipKinds KindSet

#	Name	Documentation
1	ResponsibilityRelationshipKinds	A kind of PPMNKindSet that includes terms that specify the kind of responsibility a <i>Party</i> has with respect to an <i>Entity</i> .
2	Custody	Custody indicates that the source element has custody (immediate charge of or control over) the target element.
3	Ownership	Ownership indicates that the source element owns the target element.

2517 **9.9.1 ResponsibilityRelationshipKinds**
2518 A kind of PPMNKindSet that includes terms that specify the kind of responsibility a *Party* has with respect to an
2519 *Entity*.

2520 **9.9.2 Custody**
2521 *Custody* indicates that the *source* element has custody (immediate charge of or control over) the *target*
2522 element.

2523 **9.9.3 Ownership**
2524 *Ownership* indicates that the *source* element owns the *target* element.

2525 **10 Parties Model**

2526 This section defines the semantic elements of the **Parties** Metamodel. The main topics are organized into Core
2527 Elements, Locations, Packages, Vocabularies, and Primitives.

2528

2529 **10.1 Core**

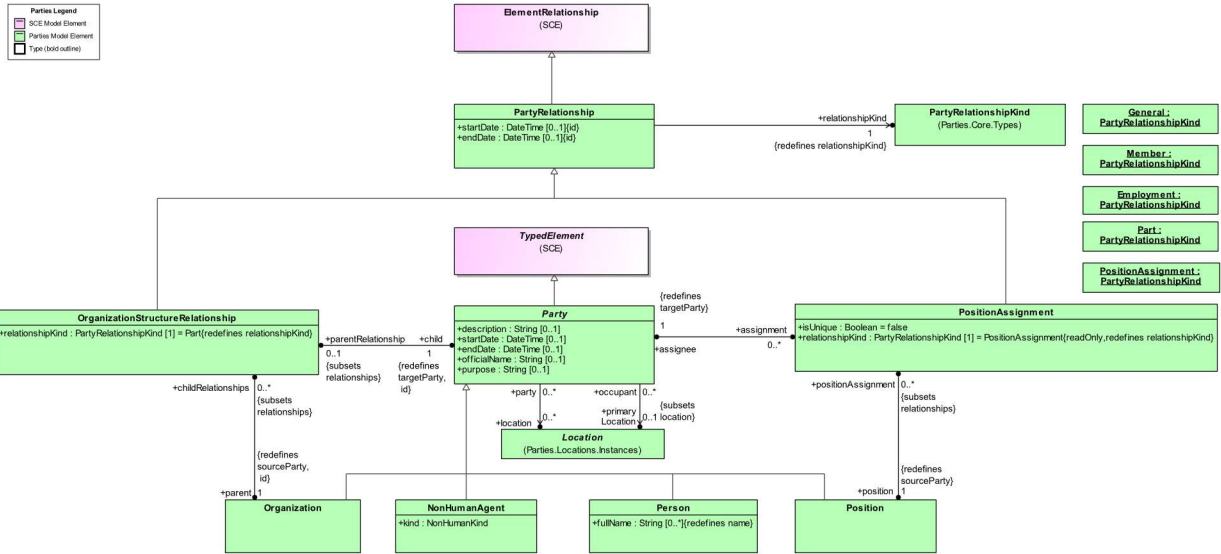
2530 The Core elements of the **Parties** metamodel contains elements related to people, organizations, roles, automated
2531 systems and the relationships between them. The elements are separated into Instances and Types. The Instances
2532 section defines elements that enable modeling specific Parties (i.e., people, organizations, positions and roles and
2533 their interrelationships). The Types section defines elements that enable modeling the kinds of Parties that are of
2534 interest in some context.

2535 **10.1.1 Instances**

2536 The Core.Instances section of the **Parties** metamodel contains elements related to people, organizations, roles,
2537 automated systems and the relationships between them. These elements enable modeling specific Parties. Elements
2538 in the Core.Instances section are generally specializations of **SCE TypedElements** and as such may have an
2539 *ElementType* specified. The corresponding types are described below in the *Core.Types* section.

2540 A *Party* is an abstract concept intended to generalize the notions of *Organization*, *Person*, *Position* or *Non-Human*
2541 *Agent* - essentially things that can be proactive and play a part in a business context. This generalization
2542 acknowledges the fact that many of the same business interactions can be defined regardless of the particular type of
2543 party involved. For instance, in the sale of a parcel of land, the seller might be a *Person* or an *Organization* or even
2544 a *Position* in an *Organization* wherein that *Position* is responsible for handling real estate transactions. Likewise for
2545 the buyer. The *Party* pattern captures this notion in a succinct manner that has broad applicability.

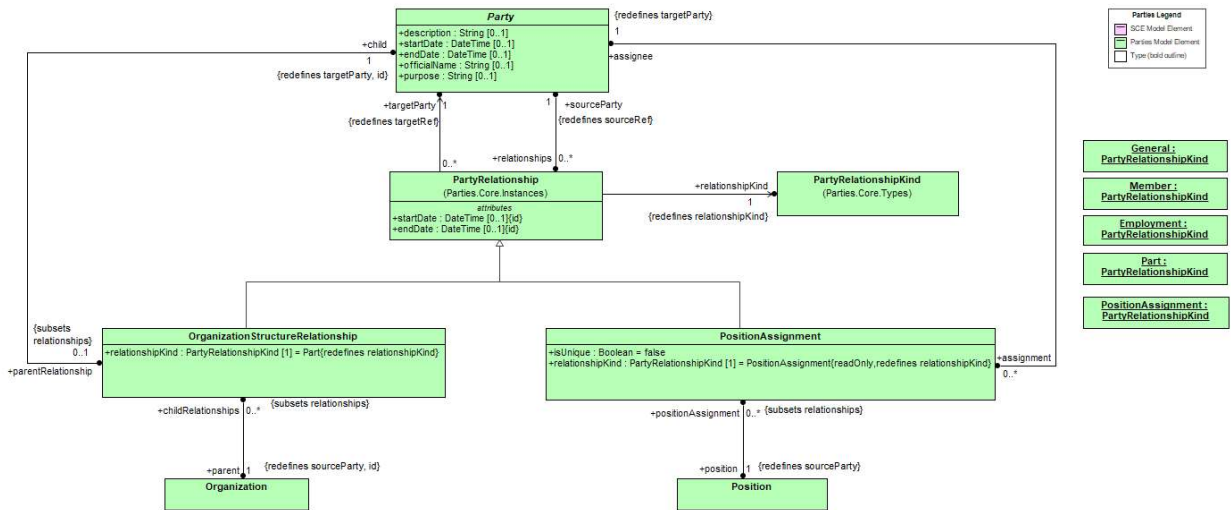
2546



2547
 2548
 2549
 2550
 2551
 2552
 2553
 2554

Figure 60: Parties

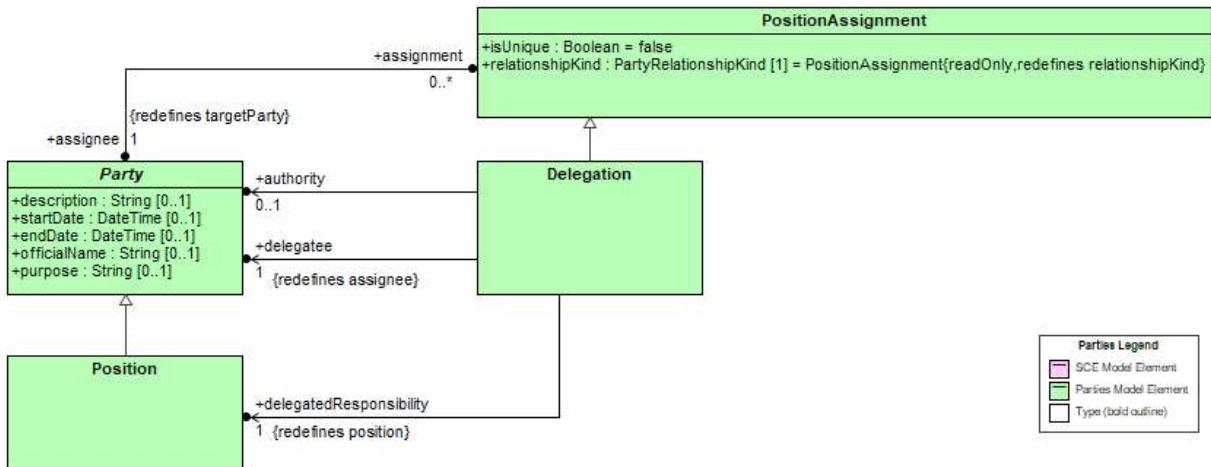
PartyRelationships capture relationships between *Parties*. The precise kind of relationship is specified by the *relationshipKind* property. There are two specializations of *PartyRelationship*: *OrganizationalStructureRelationship* and *PositionAssignment*. *OrganizationalStructureRelationship* supports the specification of the structure of an *Organization* while *PositionAssignment* supports the assignment of *Parties* to *Positions*.



2555
 2556
 2557
 2558
 2559

Figure 61: Party Relationships

Delegation captures the notion that a *Party* may assign a set of responsibilities to another party. The responsibilities being assigned are essentially captured as a *Position*.

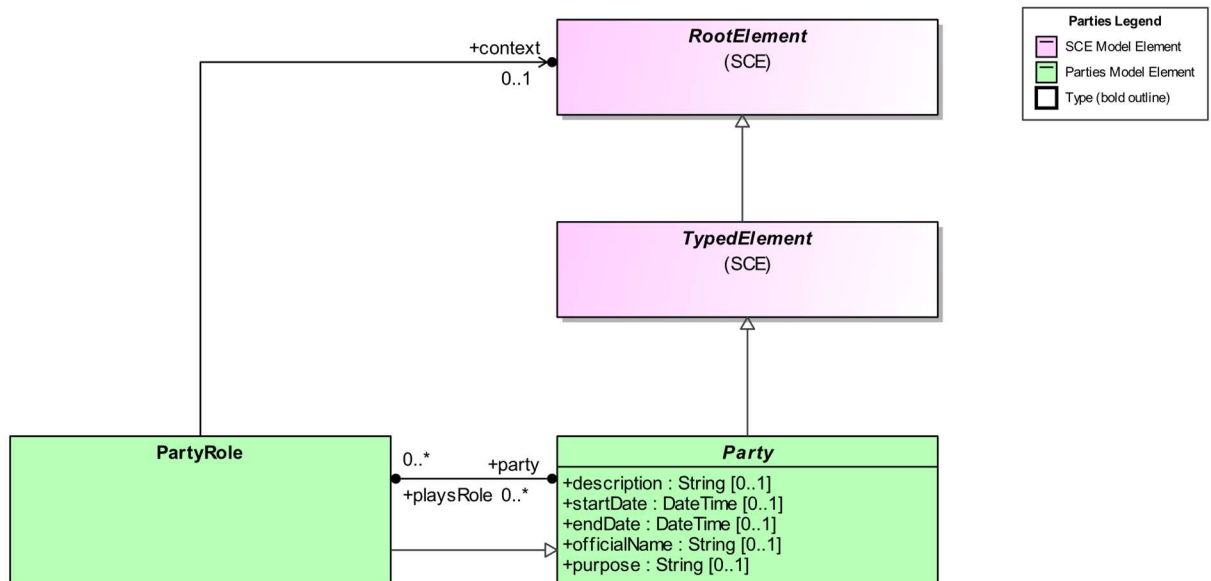


2560

2561 **Figure 62: Delegation**

2562 *PartyRoles* represent a role that a *Party* may play in some context. For instance, in the sale of a parcel of land, the
 2563 Seller might be a *Person* or an *Organization* or even a *Position* in an *Organization* wherein that *Position*
 2564 responsible for handling real estate transactions. Likewise for the buyer. The *PartyRole* captures this notion in a
 2565 succinct manner that has broad applicability.

2566



2567

2568 **Figure 63: Party Role**

2569 This diagram shows the mapping of *Party* and its specializations to *PartyType* and its specializations.

2570

Table 100. NonHumanAgent Attributes and/or Associations

Property/Association	Description
kind : NonHumanKind []	An instance that indicates the kind of NonHumanAgent the element represents.
type : IndividualType [0..1]	The class that provides a specification of the <i>Automation</i> .

2589

2590 10.1.1.3 Organization

2591 *Organization* is used to represent a group of *Parties*. The group may be a company, a department within a company,
2592 a club, a consortium, or some other group.

2593 Generalizations

2594 The *Organization* element inherits the attributes and/or associations of:

- 2595 • *Party* (see the section entitled “[Party](#)” for more information).

2596 Properties

2597 The following table presents the additional attributes and/or associations for *Organization*:

Table 101. Organization Attributes and/or Associations

Property/Association	Description
childRelationships : OrganizationStructureRelationship [0..*]	A set of relationships to the members of the <i>Organization</i> .
type : OrganizationType [0..1]	The class that provides a specification of the <i>Organization</i> .

2598

2599 10.1.1.4 OrganizationStructureRelationship

2600 A specialization of *PartyRelationship* used to indicate internal structural relationships of a *Party*.

2601 Generalizations

2602 The *OrganizationStructureRelationship* element inherits the attributes and/or associations of:

- 2603 • *PartyRelationship* (see the section entitled “[PartyRelationship](#)” for more information).

2604 Properties

2605 The following table presents the additional attributes and/or associations for *OrganizationStructureRelationship*:

Table 102. OrganizationStructureRelationship Attributes and/or Associations

Property/Association	Description
child : Party [1]	The <i>Party</i> that is a member of the organization.
parent : Organization [1]	The <i>Organization</i> in which the <i>Party</i> is a member.

relationshipKind : PartyRelationshipKind [1] default: Part	The kind of structural relationship an Organization has with another Party.
---	---

2606

2607 **10.1.1.5 Party**

2608 *Party* is an abstract concept representing a *Person*, *Role*, *Organization*, or other entity involved in some activity,
2609 interaction or endeavor.

2610 **Generalizations**

2611 The *Party* element inherits the attributes and/or associations of:

- **SCE TypedElement** (see the SCE specification for more information).

2613 **Properties**

2614 The following table presents the additional attributes and/or associations for *Party*:

Table 103. Party Attributes and/or Associations

Property/Association	Description
assignment : PositionAssignment [0..*]	A relationship indicating a <i>Position</i> to which the <i>Party</i> has been assigned.
description : String [0..1]	A textual description of the <i>Party</i> .
endDate : DateTime [0..1]	The effective end date of the <i>Party</i> .
location : Location [0..*]	The location of the <i>Party</i> .
officialName : String [0..1]	The official name of the <i>Party</i> .
parentRelationship : OrganizationStructureRelationship [0..1]	A set of relationships to the <i>Organizations</i> in which the <i>Party</i> has membership.
playsRole : PartyRole [0..*]	The roles played by a <i>Party</i> .
primaryLocation : Location [0..1]	The primary location of the <i>Party</i> .
purpose : String [0..1]	The purpose of the <i>Party</i> with respect to the pedigree and/or provenance context.
relationships : PartyRelationship [0..*]	<i>PartyRelationships</i> in which the Party is involved.

startDate : DateTime [0..1]	The effective start date of the <i>Party</i> .
type : PartyType [0..1]	The class that provides a specification of the <i>Party</i> .

2615

2616 10.1.1.6 PartyRelationship

2617 A kind of *ElementRelationship* that indicates a relationship between two *Parties*.

2618 Generalizations

2619 The *PartyRelationship* element inherits the attributes and/or associations of:

- 2620 • *ElementRelationship* (see the **SCE** specification for more information).

2621 Properties

2622 The following table presents the additional attributes and/or associations for *PartyRelationship*:

Table 104. PartyRelationship Attributes and/or Associations

Property/Association	Description
endDate : DateTime [0..1]	The effective end date of the relationship.
relationshipKind : PartyRelationshipKind [1]	The kind of relationship between two <i>Parties</i> .
sourceParty : Party [1]	The source <i>Party</i> of the relationship.
startDate : DateTime [0..1]	The effective start date of the relationship.
targetParty : Party [1]	The target <i>Party</i> of the relationship.
type : PartyRelationshipType [0..1]	The class that provide a specification of the <i>PartyRelationship</i> .

2623

2624 10.1.1.7 PartyRole

2625 A role played by a *Party* in some context. For instance, a Buyer or a Supplier.

2626 Generalizations

2627 The *PartyRole* element inherits the attributes and/or associations of:

- 2628 • *Party* (see the section entitled “[Party](#)” for more information).

2629 Properties

2630 The following table presents the additional attributes and/or associations for *PartyRole*:

Table 105. PartyRole Attributes and/or Associations

Property/Association	Description
context : BaseElement [0..1]	The context in which the <i>Party</i> plays the role.
party : Party [0..*]	The <i>Party</i> that plays the role.
type : PartyRoleType [0..1]	The class that provides a specification of the <i>PartyRole</i> .

2631

2632 **10.1.1.8 Person**

2633 An individual homo sapiens.

2634 **Generalizations**

2635 The *Person* element inherits the attributes and/or associations of:

- 2636 • *Party* (see the section entitled “[Party](#)” for more information).

2637 **Properties**

2638 The following table presents the additional attributes and/or associations for *Person*:

Table 106. Person Attributes and/or Associations

Property/Association	Description
fullName : String [0..*]	The full name of the <i>Person</i> .
type : IndividualType [0..1]	The class that provides a specification of the <i>Person</i> .

2639

2640 **10.1.1.9 Position**

2641 A *Position* is a formally defined role in an *Organization* filled by some *Person*. *Positions* are often associated with a
2642 set of responsibilities in some context.

2643 Examples of *Positions* include Chief Executive Officer or Technical Staff Member.

2644

2645 **Generalizations**

2646 The *Position* element inherits the attributes and/or associations of:

- 2647 • *Party* (see the section entitled “[Party](#)” for more information).

2648 **Properties**

2649 The following table presents the additional attributes and/or associations for *Position*:

Table 107. Position Attributes and/or Associations

Property/Association	Description
positionAssignment : PositionAssignment [0..*]	A <i>PositionAssignment</i> that indicates the <i>Party</i> that fills the <i>Position</i> .
type : PositionType [0..1]	The class that provides a specification of the <i>Position</i> .

2650

2651 **10.1.1.10 PositionAssignment**

2652 *PositionAssignment* indicates a *Party* is assigned to a particular *Position* for a particular period of time.

2653 **Generalizations**

2654 The *PositionAssignment* element inherits the attributes and/or associations of:

- 2655 • *PartyRelationship* (see the section entitled "[PartyRelationship](#)" for more information).

2656 **Properties**

2657 The following table presents the additional attributes and/or associations for *PositionAssignment*:

Table 108. PositionAssignment Attributes and/or Associations

Property/Association	Description
assignee : Party [1]	The <i>Party</i> that fills or filled the <i>Position</i> .
isUnique : Boolean [] default: false	A boolean stating whether only one <i>Party</i> filled a particular <i>Role</i> during that particular date range.
position : Position [1]	The <i>Position</i> filled by the noted <i>Party</i> .
relationshipKind : PartyRelationshipKind [1] default: PositionAssignment	The kind of relationship between an <i>Organization</i> and a <i>Position</i> within that <i>Organization</i> .
type : PositionAssignmentType [0..1]	The class that provides a specification of the <i>PositionAssignment</i> .

2658

2659 **10.1.2 Types**

2660 The Core.Types section of the **Parties** metamodel contains elements related to the kinds of people, organizations,
2661 roles, automated systems and the relationships between them that are of interest in some context. These elements
2662 enable modeling kinds of Parties rather than particular Parties. Elements in the Core.Types section are generally
2663 specializations of **SCE ElementTypes** and as such provide a specification Parties to be created using elements in the
2664 Core.Instances section described above.

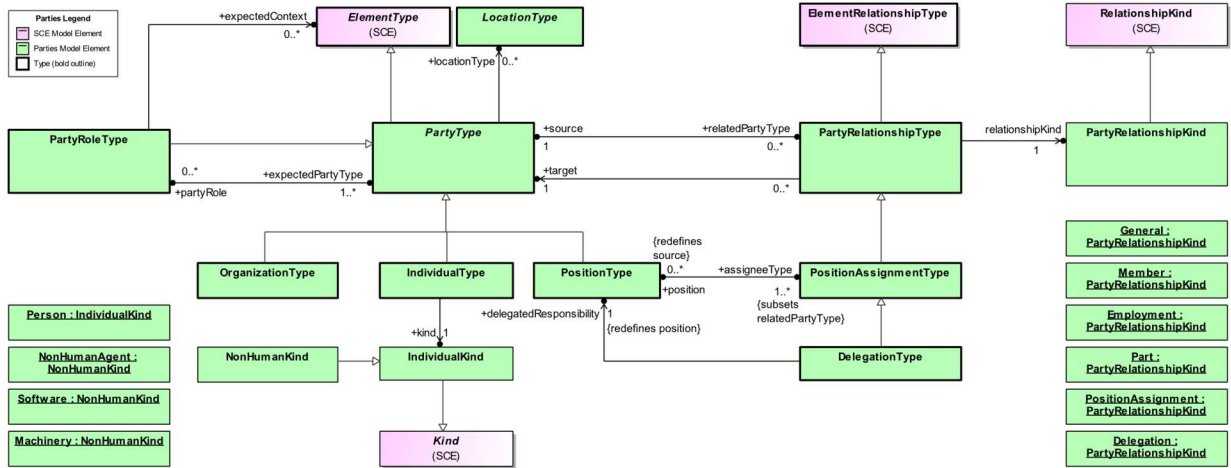
2665 *PartyTypes* define the types or classifications of *Parties*. *PartyTypes* provide the ability to specify or "configure"
2666 organizational structures for different kinds of parties such as companies, non-profits, community organizations and
2667 many others. *PartyType* configurations can be used to provide a constraint mechanism on the *Parties* created in
2668 some context though the Party metamodel does not require their use.

2669 While *PartyType* itself is abstract, the Party metamodel includes the concrete specializations *OrganizationType*,
2670 *IndividualType*, and *PositionType*. These types correspond to the concrete specializations of *Party* where

2671 IndividualType is used as the type for *Person*, *Automation*, and *SoftwareAgent* with the *kind* property set
 2672 appropriately,

2673 *PartyRelationshipTypes* capture the possible relationships between *PartyTypes*. *PartyRelationshipTypes* have a
 2674 *PartyRelationshipKind* that specifies the kind of relationship: Part, Member, Assignment, or General. (See
 2675 *PartyRelationshipKind*.) *PositionAssignmentType* captures the particular relationship type wherein one or more
 2676 *PartyTypes* are expected to fill (or be assigned to) a particular *PositionType*.

2677

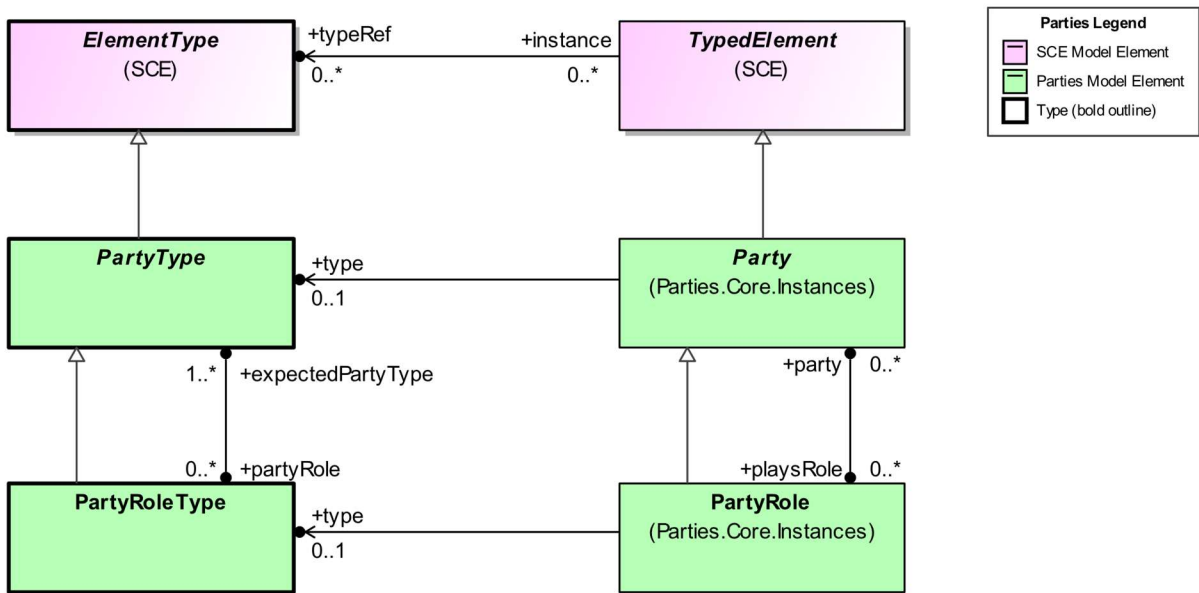


2678

2679 **Figure 65: Party Types**

2680 *PartyRoles* define the types or classifications of the roles that may be played by one or more kinds of *Parties* (i.e.,
 2681 *PartyTypes*) in some context. The *expectedPartyType* property specifies which *PartyTypes* are expected to
 2682 play *PartyRoles* of that *PartyRoleType*.

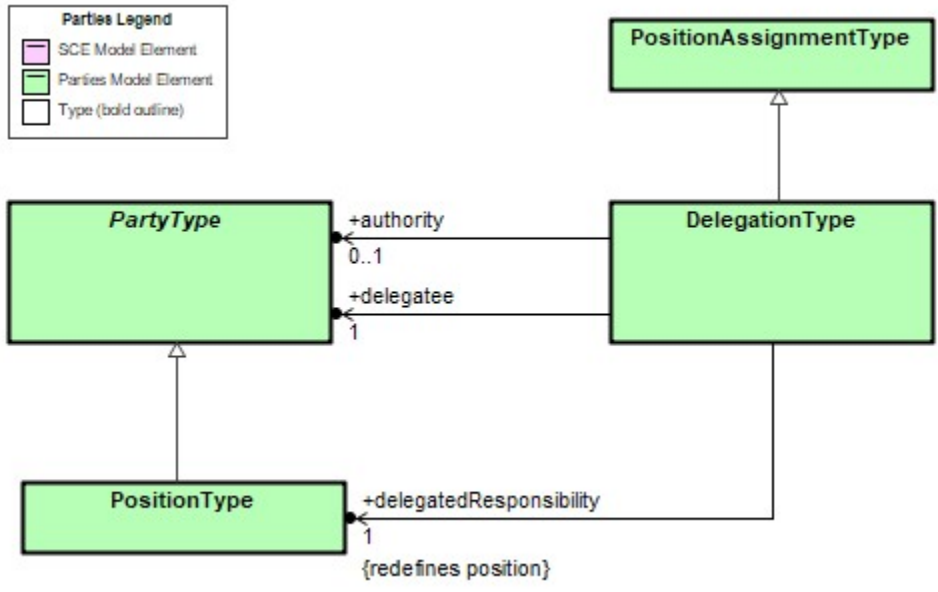
2683



2684

2685 **Figure 66: Party Role Type**

2686 Delegation captures the notion that a Party may assign a set of responsibilities to another party. *DelegationType*
 2687 supports the ability to state that the responsibilities associated with a *PositionType* may be delegated to particular
 2688 *PartyTypes* on the authority of some *PartyType*.
 2689



2690
 2691 **Figure 67: Delegation Types**

2692 **10.1.2.1 DelegationType**

2693 *DelegationType* indicates a particular *PartyType* is delegated responsibility for particular *PositionType* by an
 2694 authority.

2695 **Generalizations**

2696 The *DelegationType* element inherits the attributes and/or associations of:

- 2697 • *PositionAssignmentType* (see the section entitled “[PositionAssignmentType](#)” for more information).

2698 **Properties**

2699 The following table presents the additional attributes and/or associations for *DelegationType*:

Table 109. DelegationType Attributes and/or Associations

Property/Association	Description
authority : PartyType [0..1]	The <i>PartyType</i> expected to be the authority by which the delegation approved.
delegatedResponsibility : PositionType [1]	The set of responsibilities as defined by a <i>PositionType</i> that may be delegated.
delegatee : PartyType [1]	The <i>PartyType</i> to whom the responsibilities are expected to be delegated.

2700

2701 **10.1.2.2 IndividualKind**

2702 *IndividualKind* is a specialization of *Kind* that serves as the foundation for `terms` in a *PartyKindSet* that is used to
2703 specify the kinds of *IndividualTypes* in a **Parties** model. Instead of being defined a fixed enumerated list, the kinds
2704 are defined through instances of *IndividualKind* present in the *IndividualKinds* library. The instances defined in that
2705 library SHALL be included in any **Parties** implementation. However, the implementation can allow additional kinds
2706 of individuals through the addition of new instances of *IndividualKind* in the *IndividualKinds* library.

2707

2708 **Generalizations**

2709 The *IndividualKind* element inherits the attributes and/or associations of:

- 2710 • *Kind* (see the SCE specification for more information).

2711 **Properties**

2712 The *IndividualKind* element does not have any additional attributes and/or associations.

2713 **10.1.2.3 IndividualType**

2714 A kind of *PartyType* representing the type or classification of a *Party* of interest that is an individual such as a
2715 *Person*, *Automation*, or *SoftwareAgent*.

2716 **Generalizations**

2717 The *IndividualType* element inherits the attributes and/or associations of:

- 2718 • *PartyType* (see the section entitled “[PartyType](#)” for more information).

2719 **Properties**

2720 The following table presents the additional attributes and/or associations for *IndividualType*:

Table 110. IndividualType Attributes and/or Associations

Property/Association	Description
kind : IndividualKind [1]	An instance that indicates the kind of individual the <i>IndividualType</i> represents.

2721

2722 **10.1.2.4 NonHumanKind**

2723 *NonHumanKind* is a kind of *IndividualKind* that serves as the foundation for `terms` in a *PartyVocabulary* that is
2724 used to specify the kinds of *NonHumanAgents* in a **Parties** model. Instead of being defined as a fixed enumerated
2725 list, the kinds are defined through instances of *NonHumanKind* present in the *IndividualKinds* library. The instances
2726 defined in that library SHALL be included in any **Parties** implementation. However, the implementation can allow
2727 additional kinds of individuals through the addition of new instances of *NonHumanKind* in the *IndividualKinds*
2728 library.

2729 **Generalizations**

2730 The *NonHumanKind* element inherits the attributes and/or associations of:

- 2731 • *IndividualKind* (see the section entitled “[IndividualKind](#)” for more information).

2732 **Properties**

2733 The *NonHumanKind* element does not have any additional attributes and/or associations.

2734 **10.1.2.5 OrganizationType**

2735 A kind of *PartyType* that represents the type or classification of an *Organization*.

2736 **Generalizations**

2737 The *OrganizationType* element inherits the attributes and/or associations of:

- 2738 • *PartyType* (see the section entitled “[PartyType](#)” for more information).

2739 **Properties**

2740 The *OrganizationType* element does not have any additional attributes and/or associations.

2741 **10.1.2.6 PartyRelationshipKind**

2742 *PartyRelationshipKind* is a specialization of *RelationshipKind* that serves as the foundation for terms for a
2743 *PartiesKindSet* that is used to specify the kind of relationship that exists between two *PartyTypes* related by a
2744 *PartyRelationshipType*. Instead of being defined a fixed enumerated list, the kinds are defined through instances of
2745 *PartyRelationshipKind* present in the *PartyRelationshipKinds* library. The instances defined in the **Parties Library**
2746 SHALL be included in any **Parties** implementation. However, the implementation can allow additional kinds of
2747 relationship types through the addition of new instances of *PartyRelationshipKind* in the *PartyRelationshipKinds*
2748 library.

2749 **Generalizations**

2750 The *PartyRelationshipKind* element inherits the attributes and/or associations of:

- 2751 • *RelationshipKind* (see the section entitled “[RelationshipKind](#)” for more information).

2752 **Properties**

2753 The *PartyRelationshipKind* element does not have any additional attributes and/or associations.

2754 **10.1.2.7 PartyRelationshipType**

2755 A kind of *ElementRelationship* that indicates a relationship between two *PartyTypes*.

2756 **Generalizations**

2757 The *PartyRelationshipType* element inherits the attributes and/or associations of:

- 2758 • *ElementRelationshipType* (see the **SCE** specification for more information).

2759 **Properties**

2760 The following table presents the additional attributes and/or associations for *PartyRelationshipType*:

Table 111. PartyRelationshipType Attributes and/or Associations

Property/Association	Description
relationshipKind : PartyRelationshipKind [1]	A specification of the kind of relationship of expected to exist between two Parties or PartyTypes.
source : PartyType [1]	The source <i>PartyType</i> of the relationship.
target : PartyType [1]	The target <i>PartyType</i> of the relationship.

2761

2762 **10.1.2.8 PartyRoleType**

2763 A type or classification of a role that may be played by a particular *PartyType* in some context. For instance, a Buyer
2764 or a Supplier.

2765 **Generalizations**

2766 The *PartyRoleType* element inherits the attributes and/or associations of:

- 2767 • *PartyType* (see the section entitled “[PartyType](#)” for more information).

2768 **Properties**

2769 The following table presents the additional attributes and/or associations for *PartyRoleType*:

Table 112. PartyRoleType Attributes and/or Associations

Property/Association	Description
expectedContext : ElementType [0..*]	The context in which instances of the <i>PartyRoleType</i> are expected to occur.
expectedPartyType : PartyType [1..*]	The type of <i>Party</i> that is expected to play the role specified by the <i>PartyRoleType</i> .

2770

2771 **10.1.2.9 PartyType**

2772 An abstract class representing the type or classification of a *Party* of interest.

2773 **Generalizations**

2774 The *PartyType* element inherits the attributes and/or associations of:

- 2775 • *SCE ElementType* (see the *SCE* specification for more information).

2776 **Properties**

2777 The following table presents the additional attributes and/or associations for *PartyType*:

Table 113. PartyType Attributes and/or Associations

Property/Association	Description
locationType : LocationType [0..*]	The type of <i>Location</i> at which the instances of the <i>PartyType</i> are expected to be located.
partyRole : PartyRoleType [0..*]	The type(s) of roles that <i>Parties</i> of type <i>PartyType</i> are expected to play.
relatedPartyType : PartyRelationshipType [0..*]	The related <i>PartyType</i> of a relationship.

2778

2779 **10.1.2.10 PositionAssignmentType**

2780 *PositionAssignmentType* indicates a particular *PartyType* is expected to fill particular *PositionType*.

2781 **Generalizations**

2782 The *PositionAssignmentType* element inherits the attributes and/or associations of:

2783 • *PartyRelationshipType* (see the section entitled “[PartyRelationshipType](#)” for more information).

2784 Properties

2785 The following table presents the additional attributes and/or associations for *PositionAssignmentType*:

Table 114. PositionAssignmentType Attributes and/or Associations

Property/Association	Description
kind : [1]	The kind relationship between the <i>PartyTypes</i> that is set to <i>Assignment</i>
position : PositionType [0..*]	The <i>PositionType</i> that will be filled by the <i>PartyType</i> referenced by the target of the <i>PositionTypeAssignment</i> .

2786

2787 10.1.2.11 PositionType

2788 A kind of *PartyType* that represents the type or classification of a *Position*.

2789 Generalizations

2790 The *PositionType* element inherits the attributes and/or associations of:

2791 • *PartyType* (see the section entitled “[PartyType](#)” for more information).

2792 Properties

2793 The following table presents the additional attributes and/or associations for *PositionType*:

Table 115. PositionType Attributes and/or Associations

Property/Association	Description
assigneeType : PositionAssignmentType [1..*]	A <i>PositionAssignmentType</i> that indicates the <i>PartyType</i> that may fill the <i>PositionType</i> .

2794

2795 10.2 Locations

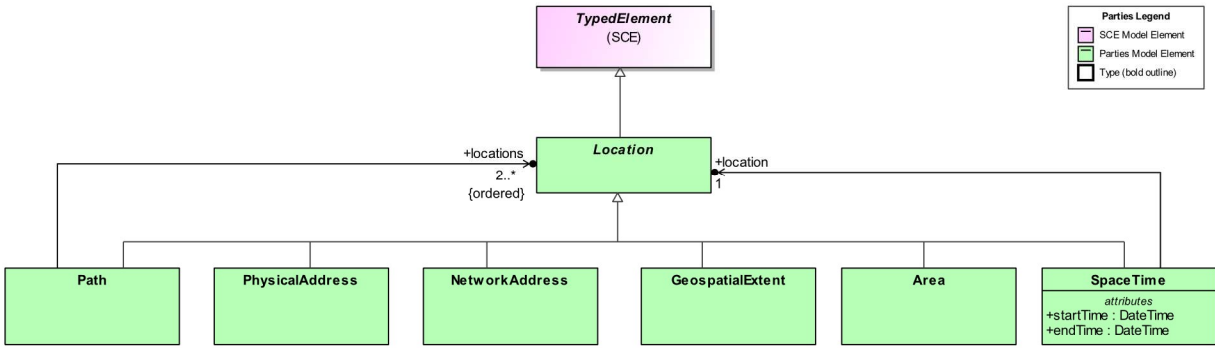
2796 The Locations package contains elements related to physical or virtual locations.

2797 10.2.1 Instances

2798 The Locations.Instances section of the **Parties** metamodel contains elements related to locations and the relationships
2799 between them. These elements enable modeling specific locations at which Parties may reside. Elements in the
2800 Locations.Instances section are generally specializations of **SCE TypedElements** and as such may have an
2801 **ElementType** specified. The corresponding types are described below in the Locations.Types section.

2802 Organizations may deem the location at which an occurrence took place to be of significance. In those situations a
2803 Location, either physical or virtual, may be captured in conjunction with an Occurrence.

2804



2805
2806 **Figure 68: Locations**

2807 **10.2.1.1 Area**

2808 A kind of location that encompasses some region in the world.

2809 **Generalizations**

2810 The *Area* element inherits the attributes and/or associations of:

- 2811 • *Location* (see the section entitled “[Location](#)” for more information).

2812 **Properties**

2813 The following table presents the additional attributes and/or associations for *Area*:

Table 116. Area Attributes and/or Associations

Property/Association	Description
type : AreaType [0..1]	The class that provides a specification of the <i>Area</i> .

2814

2815 **10.2.1.2 GeospatialExtent**

2816 A location that is a volume in the world such as a container or a room.

2817 **Generalizations**

2818 The *GeospatialExtent* element inherits the attributes and/or associations of:

- 2819 • *Location* (see the section entitled “[Location](#)” for more information).

2820 **Properties**

2821 The following table presents the additional attributes and/or associations for *GeospatialExtent*:

Table 117. GeospatialExtent Attributes and/or Associations

Property/Association	Description
type : VolumeType [0..1]	The class that provides a specification of the <i>GeospatialExtent</i> .

2822

2823 **10.2.1.3 Location**

2824 A particular place or position.

2825 **Generalizations**

2826 The *Location* element inherits the attributes and/or associations of:

- 2827 • *SCE TypedElement* (see the **SCE** specification for more information).

2828 **Properties**

2829 The following table presents the additional attributes and/or associations for *Location*:

Table 118. Location Attributes and/or Associations

Property/Association	Description
description : String [0..1]	A description of the <i>Location</i> .
type : LocationType [0..1]	The class that provides a specification of the <i>Location</i> .

2830

2831 **10.2.1.4 NetworkAddress**

2832 The address of an element or node on a network.

2833 **Generalizations**

2834 The *NetworkAddress* element inherits the attributes and/or associations of:

- 2835 • *Location* (see the section entitled "[Location](#)" for more information).

2836 **Properties**

2837 The following table presents the additional attributes and/or associations for *NetworkAddress*:

Table 119. NetworkAddress Attributes and/or Associations

Property/Association	Description
type : NetworkAddressType [0..1]	The class that provides a specification of the <i>NetworkAddress</i> .

2838

2839 **10.2.1.5 Path**

2840 An ordered collection of *Locations*.

2841 **Generalizations**

2842 The *Path* element inherits the attributes and/or associations of:

- 2843 • *Location* (see the section entitled "[Location](#)" for more information).

2844 **Properties**

2845 The following table presents the additional attributes and/or associations for *Path*:

Table 120. Path Attributes and/or Associations

Property/Association	Description
locations : Location [2..*]	The locations that specify the <i>Path</i> .
type : PathType [0..1]	The class that provides a specification of the <i>Path</i> .

2846

2847 **10.2.1.6 PhysicalAddress**

2848 A physical location in the real world that has an identifiable address.

2849 **Generalizations**

2850 The *PhysicalAddress* element inherits the attributes and/or associations of:

- 2851 • *Location* (see the section entitled “[Location](#)” for more information).

2852 **Properties**

2853 The following table presents the additional attributes and/or associations for *PhysicalAddress*:

Table 121. PhysicalAddress Attributes and/or Associations

Property/Association	Description
type : PointType [0..1]	The class that provides a specification of the <i>PhysicalAddress</i> .

2854

2855 **10.2.1.7 SpaceTime**

2856 A *Location* at a particular point in time.

2857 **Generalizations**

2858 The *SpaceTime* element inherits the attributes and/or associations of:

- 2859 • *Location* (see the section entitled “[Location](#)” for more information).

2860 **Properties**

2861 The following table presents the additional attributes and/or associations for *SpaceTime*:

Table 122. SpaceTime Attributes and/or Associations

Property/Association	Description
endTime : DateTime []	The ending time of the <i>SpaceTime</i> .
location : Location [1]	The location of the <i>SpaceTime</i> .
startTime : DateTime []	The starting time of the <i>SpaceTime</i> .
type : SpaceTimeType [0..1]	The class that provides a specification of the <i>SpaceTime</i> .

2862

2863 **10.2.2 Types**

2864 The *Locations.Types* section of the **Parties** metamodel contains elements related to the kinds of locations and the
2865 relationships between them that are of interest in some context. These elements enable modeling kinds of *Locations*
2866 rather than particular *Locations*. Elements in the *Locations.Types* section are generally specializations of **SCE**
2867 *ElementTypes* and as such provide a specification of *Locations* to be created using elements in the
2868 *Locations.Instances* section described above.

2869 **10.2.2.1 AreaType**

2870 A kind of *LocationType* that states that a *Location* is a region or surface in the world.

2871 **Generalizations**

2872 The *AreaType* element inherits the attributes and/or associations of:

- 2873 • *LocationType* (see the section entitled "[LocationType](#)" for more information).

2874 **Properties**

2875 The *AreaType* element does not have any additional attributes and/or associations.

2876 **10.2.2.2 LocationType**

2877 A class representing the type or classification of a *Location*..

2878 **Generalizations**

2879 The *LocationType* element inherits the attributes and/or associations of:

- 2880 • **SCE** *ElementType* (see the **SCE** specification for more information).

2881 **Properties**

2882 The *LocationType* element does not have any additional attributes and/or associations.

2883 **10.2.2.3 NetworkAddressType**

2884 A class that specifies that *Locations* of this type are *NetworkAddresses*.

2885 **Generalizations**

2886 The *NetworkAddressType* element inherits the attributes and/or associations of:

- 2887 • *LocationType* (see the section entitled "[LocationType](#)" for more information).

2888 **Properties**

2889 The *NetworkAddressType* element does not have any additional attributes and/or associations.

2890 **10.2.2.4 PathType**

2891 A kind of *LocationType* that states that a *Location* is a path.

2892 **Generalizations**

2893 The *PathType* element inherits the attributes and/or associations of:

- 2894 • *LocationType* (see the section entitled "[LocationType](#)" for more information).

2895 **Properties**

2896 The *PathType* element does not have any additional attributes and/or associations.

2897 **10.2.2.5 PointType**

2898 A kind of *LocationType* that states that a *Location* is a specific point in the world.

2899 **Generalizations**

2900 The *PointType* element inherits the attributes and/or associations of:

- 2901 • *LocationType* (see the section entitled “[LocationType](#)” for more information).

2902 **Properties**

2903 The *PointType* element does not have any additional attributes and/or associations.

2904 **10.2.2.6 SpaceTimeType**

2905 A kind of *LocationType* that states that a *Location* is a *Location* at a particular time.

2906 **Generalizations**

2907 The *SpaceTimeType* element inherits the attributes and/or associations of:

- 2908 • *LocationType* (see the section entitled “[LocationType](#)” for more information).

2909 **Properties**

2910 The *SpaceTimeType* element does not have any additional attributes and/or associations.

2911 **10.2.2.7 VolumeType**

2912 A kind of *LocationType* that states that a *Location* is a volume in the world such as a container or room.

2913 **Generalizations**

2914 The *VolumeType* element inherits the attributes and/or associations of:

- 2915 • *LocationType* (see the section entitled “[LocationType](#)” for more information).

2916 **Properties**

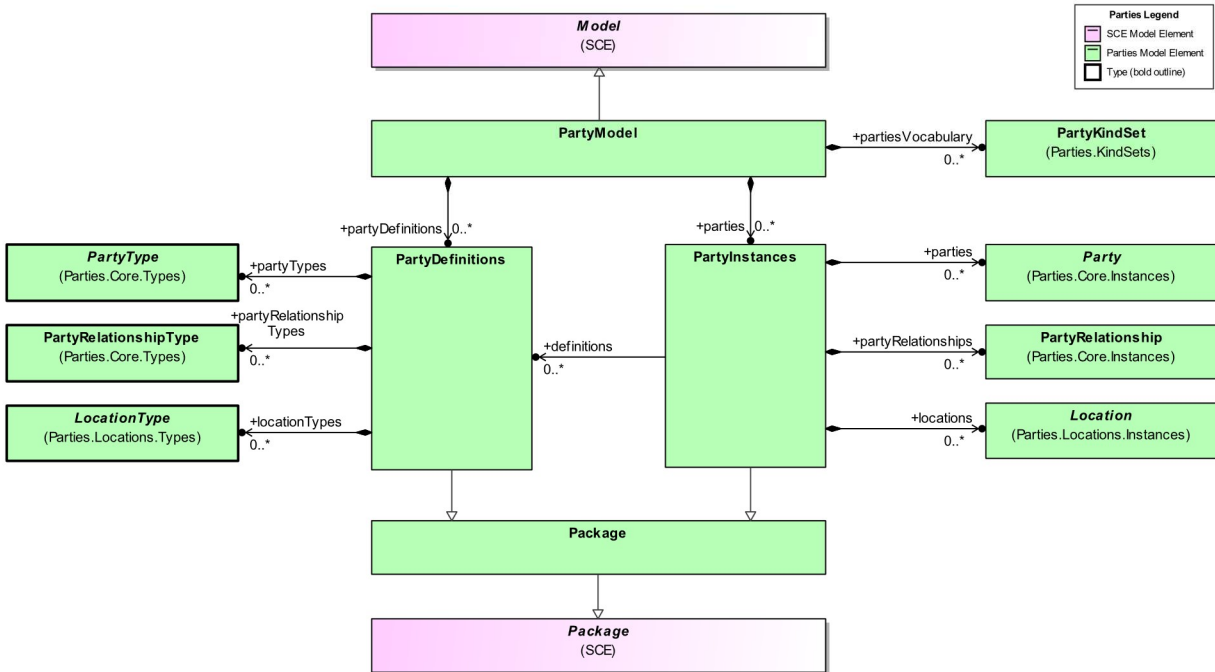
2917 The *VolumeType* element does not have any additional attributes and/or associations.

2918 **10.3 Packages**

2919 The Packages package provides elements to support the packaging of Parties-related elements.

2920 The following figure presents the attributes and associations for the **Parties** packaging elements, including details
2921 about the elements they contain:

2922



2923
2924 **Figure 69: Party Packages**

2925 **10.3.1 Package**

2926 *Package* is a kind of *SCE Package* that that is concrete. **Parties Packages** are a general packaging mechanism that
2927 can be used to hold any group of model elements.

2928 **Generalizations**

2929 The *Package* element inherits the attributes and/or associations of:

- 2930 • *SCE Package* (see the **SCE specification** for more information).

2931 **Properties**

2932 *Package* has no additional properties.

2933 **10.3.2 PartyDefinitions**

2934 *PartyDefinitions* is a kind of *Package* that contains the definitions of *PartyTypes* that are used to specify types of
2935 *Party* structures.

2936 **Generalizations**

2937 The *PartyDefinitions* element inherits the attributes and/or associations of:

- 2938 • *Package* (see the section entitled “[Package](#)” for more information).

2939 **Properties**

2940 The following table presents the additional attributes and/or associations for *PartyDefinitions*:

Table 123. PartyDefinitions Attributes and/or Associations

Property/Association	Description
locationTypes : LocationType [0..*]	The <code>locationTypes</code> property references the <i>LocationTypes</i> contained within the <i>PartyDefinitions</i> package.
partyRelationshipTypes : PartyRelationshipType [0..*]	The <code>partyRelationshipTypes</code> property references the <i>PartyRelationshipTypes</i> contained within the <i>PartyDefinitions</i> package.
partyTypes : PartyType [0..*]	The <code>partyTypes</code> property references the <i>PartyTypes</i> contained within the <i>PartyDefinitions</i> package.

2941

2942 10.3.3 PartyInstances

2943 *PartyInstances* is kind of *Package* package that contains *Parties*, *PartyRelationships*, and their *Locations*.

2944 Generalizations

2945 The *PartyInstances* element inherits the attributes and/or associations of:

- 2946 • *Package* (see the section entitled “[Package](#)” for more information).

2947 Properties

2948 The following table presents the additional attributes and/or associations for *PartyInstances*:

Table 124. PartyInstances Attributes and/or Associations

Property/Association	Description
definitions : PartyDefinitions [0..*]	The property refers to zero or more <i>SCEDefinitions</i> packages that contains the <i>ElementTypes</i> that provide a basis for the instances contained in the <i>PartyInstances</i> package.
locations : Location [0..*]	The <code>locations</code> property references the <i>Location</i> elements contained within the <i>PartyInstances</i> package.
parties : Party [0..*]	The <code>parties</code> property references the <i>Party</i> elements contained within the <i>PartyInstances</i> package.
partyRelationships : PartyRelationship [0..*]	The <code>partRelationships</code> property references the <i>PartyRelationship</i> elements contained within the <i>PartyInstances</i> package.

2949

2950 **10.3.4 PartyModel**

2951 *PartyModel* is kind of *SCE Model* that contains definitions of types of *Parties* as well as specifications of *Party*
 2952 structures themselves.

2953 **Generalizations**

2954 The *PartyModel* element inherits the attributes and/or associations of:

- 2955 • *SCE Model* (see the *SCE* specification for more information).

2956 **Properties**

2957 The following table presents the additional attributes and/or associations for *PartyModel*:

Table 125. PartyModel Attributes and/or Associations

Property/Association	Description
parties : PartyInstances [0..*]	The <code>parties</code> property subsets the <i>SCEModel</i> <code>instances</code> property. It contains a list of all the <i>PartyInstance</i> sub-packages contained within a <i>SCEModel</i> .
partiesVocabulary : PartyVocabulary [0..*]	The <code>partiesVocabulary</code> is a list of terms (as <i>Kinds</i>) that provide an extensible mechanism to define the elements of enumerations in a <i>PartiesModel</i> .
partyDefinitions : PartyDefinitions [0..*]	The <code>partyDefinitions</code> property subsets the <i>SCEModel</i> <code>definitions</code> property. It contains a list of all the <i>PartyDefinitions</i> sub-packages contained within a <i>PartyModel</i> .

2958

2959 **10.4 Primitives**

2960 The *Primitives* package provides primitive data elements used by other **Parties** elements.

2961 The following figure presents the primitive elements used in the **Parties** metamodel:

2962



2963

2964 **Figure 70: Primitives**

2965 **10.4.1 DateTime**

2966 A primitive that captures a point in time including a date and the time of day to greatest precision practical.

2967 **Generalizations**

2968 The *DateTime* element does not inherit any attributes or associations of from another element.

2969 **Properties**

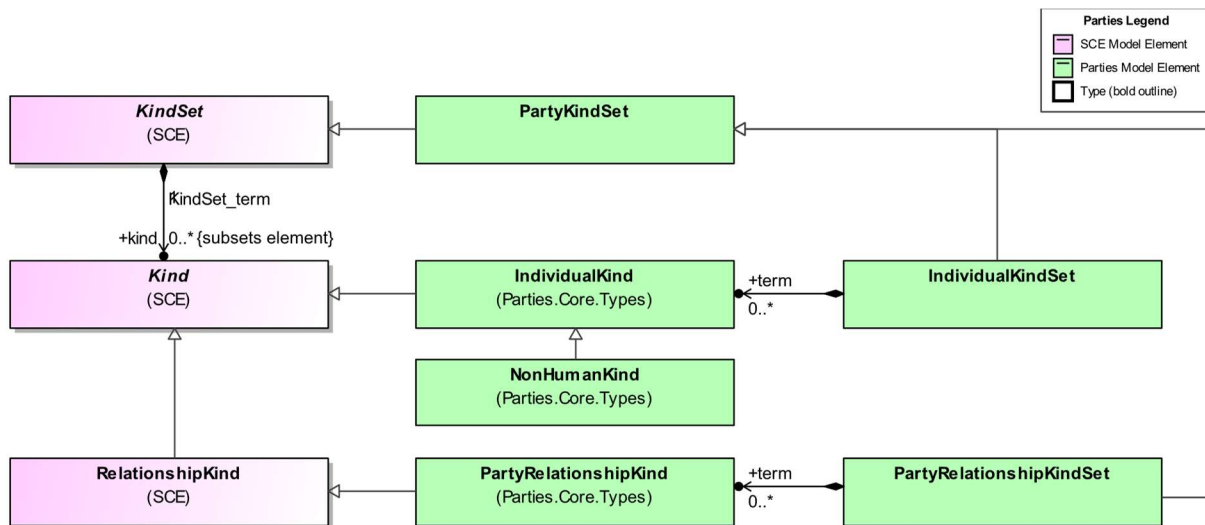
2970 The *DateTime* element does not have any additional attributes and/or associations.

2971 10.5 PartyKindSets

2972 *PartyKindSets* are sets of terms used within a **Parties** model that are defined by an external ontology. The terms link
2973 to formal definitions for the terms used within the model. The *Kind* element is used to name the term provide a link
2974 to the definitions. *PartyVocabularies* are contained within an *PartiesModel* package.

2975 The following figure presents the elements related to the *PartyKindSet* section:

2976



2977

2978 **Figure 71: PartyKindSets**

2979 10.5.1 PartyKindSet

2980 A *PartyKindSet* is a kind of *SCEVocabulary* that includes a list of terms defined as instances of the *Kind* element.
2981 As instances of *Kind*, or a specialization thereof, the instances can be used to relate the terms to external definitions
2982 of the meaning of the term. The terms themselves do not represent the definitions or meanings but provide links to
2983 an external source. The **Parties** model contains two *KindSets*: *PartyRelationshipKinds* and *IndividualKinds*.

2984

2985 Generalizations

2986 The *PartyKindSet* element inherits the attributes and/or associations of:

- 2987 • *KindSet* (see the **SCE** specification for more information).

2988 Properties

2989 The *PartyKindSet* element does not have any additional attributes and/or associations.

2990 10.5.2 IndividualKindSet

2991 A *IndividualKindSet* is a kind of *PartiesKindSet* that includes a list of terms defined as instances of *IndividualKind*,
2992 itself a *Kind*. As instances of a specialization of *Kind*, the instances can be used to relate the terms to external
2993 definitions of the meaning of the term. The terms themselves do not represent the definitions or meanings but
2994 provide links to an external source.

2995 Generalizations

2996 The *IndividualKindSet* element inherits the attributes and/or associations of:

- *PartyKindSet* (see the section entitled “[PartyKindSet](#)” for more information).

2998 Properties

2999 The following table presents the additional attributes and/or associations for *IndividualKindSet*:

Table 126. IndividualKindSet Attributes and/or Associations

Property/Association	Description
term : IndividualKind [0..*]	A list of the terms representing valid IndividualKinds.

3000

3001 10.5.3 PartyRelationshipKindSet

3002 A *PartyRelationshipKindSet* is a kind of *PartiesVocabulary* that includes a list of terms defined as instances of
 3003 *PartyRelationshipKind*, itself a specialization of *Kind*. As instances of a specialization of *Kind*, the instances can be
 3004 used to relate the terms to external definitions of the meaning of the term. The terms themselves do not represent the
 3005 definitions or meanings but provide links to an external source.

3006 Generalizations

3007 The *PartyRelationshipKindSet* element inherits the attributes and/or associations of:

- *PartyKindSet* (see the section entitled “[PartyKindSet](#)” for more information).

3009 Properties

3010 The following table presents the additional attributes and/or associations for *PartyRelationshipKindSet*:

Table 127. PartyRelationshipKindSet Attributes and/or Associations

Property/Association	Description
term : PartyRelationshipKind [0..*]	A list of the terms representing valid PartyRelationshipKinds.

3011

3012 11 Parties Library

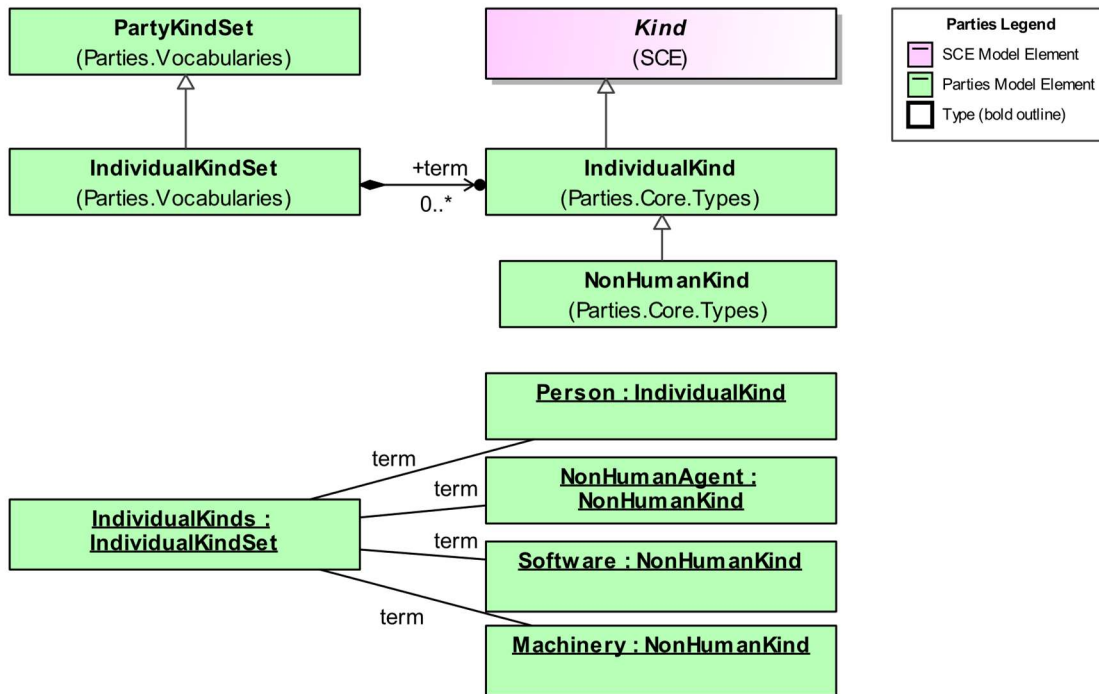
3013 A Library is included in the **Parties** specification to provide standard values that that are intended to be provided by
 3014 tools implementing the **Parties** specification. Currently, **Parties** defines the standard values for two vocabularies:
 3015 *IndividualKinds* and *PartyRelationshipKinds* (See next sections).

3016 11.1 IndividualKinds

3017 The *IndividualKinds* package contains the instances representing the standard *IndividualKinds* set. This set provides
 3018 a standard set of terms for the kinds of Individuals that can be instantiated within a Parties model. These elements
 3019 include an instance of a *PartiesKindSet*, *IndividualKinds*, which represents the set itself as well as instances
 3020 of *IndividualKind* representing the kinds of Individuals that may be instantiated.

3021 The *IndividualKind* element is used to indicate a specific kind *IndividualType* that is to be created. The instances
 3022 defined in this Library SHALL be included in any **Parties** implementation. However, the implementation can allow
 3023 additional instances of the class to represent new *IndividualTypes*.

3024 The following figure presents the instances for the *IndividualKind* element that are terms for the
 3025 IndividualKinds set.
 3026



3027
 3028 **Figure 72: IndividualKinds**

3029 The following table provides a definition of the terms included in the *IndividualKinds* set.

3030 **Table 128. IndividualKinds KindSet**

#	Name	Documentation
1	IndividualKinds	IndividualKinds is an instance of <i>PartiesKindSet</i> that includes terms for the kinds of <i>PartyRelationships</i> that may be created in a Parties model.
2	Machinery	Machinery indicates that the type of <i>NonHumanKind</i> is a machine of some kind.
3	NonHumanAgent	NonHumanAgent indicates that the type of individual is an automated system of some kind.
4	Person	Person indicates that the type of individual is a person.
5	Software	Software indicates that the type of individual is a software module of some kind.

3031 **11.1.1 IndividualKinds**

3032 IndividualKinds is an instance of *PartiesVocabulary* that includes terms for the kinds of *PartyRelationships*
 3033 that may be created in a **Parties** model.

3034 **11.1.2 Machinery**

3035 Machinery indicates that the type of *NonHumanKind* is a machine of some kind.

3036 **11.1.3 NonHumanAgent**

3037 `NonHumanAgent` indicates that the type of individual is an automated system of some kind.

3038 **11.1.4 Person**

3039 `Person` indicates that the type of individual is a person.

3040 **11.1.5 Software**

3041 `Software` indicates that the type of individual is a software module of some kind.

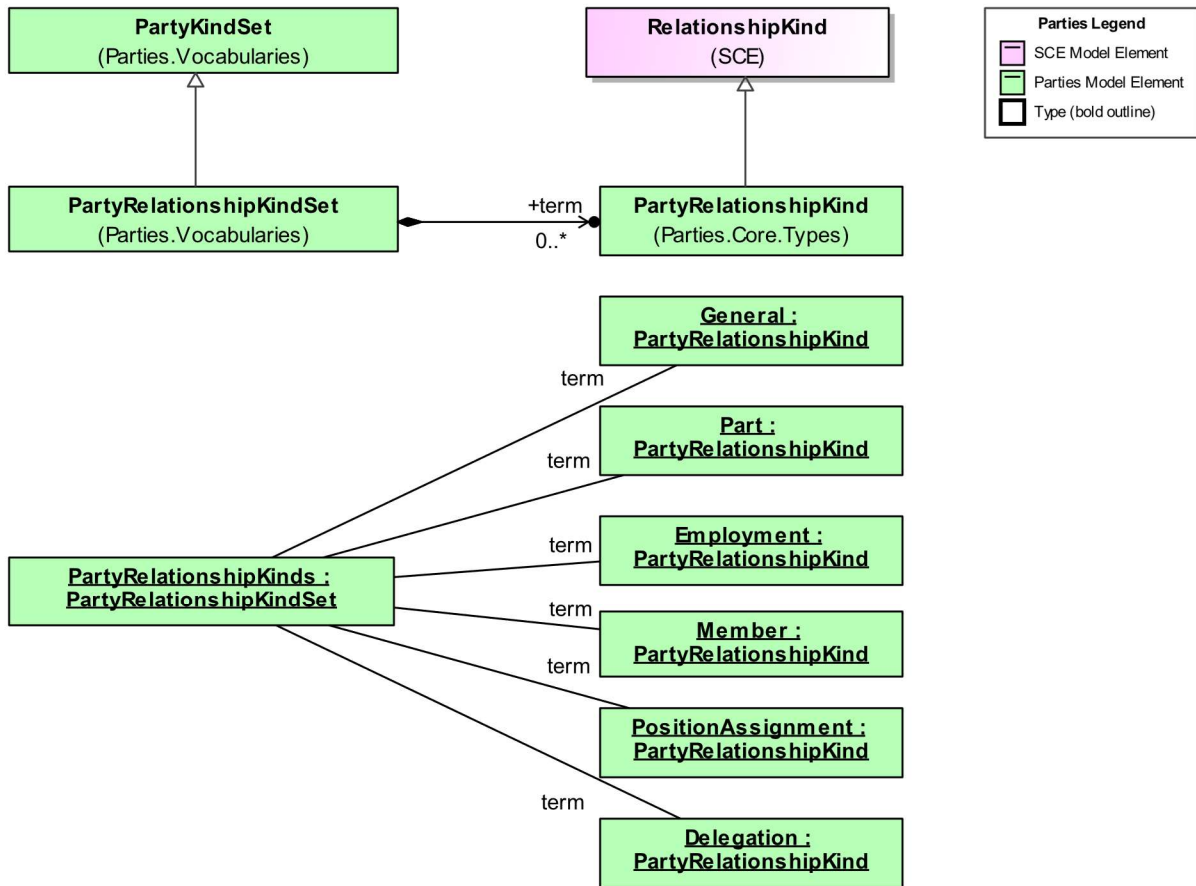
3042 **11.2 PartyRelationshipKinds**

3043 The *PartyRelationshipKinds* package contains one instance of an **SCE KindSet**: `PartyRelationshipKind`
3044 which is provided by the **Parties** Library. The purpose of this set is to provide a set of standard terms for the
3045 different types of relationships between Parties. These terms will be represented by instances of the
3046 *PartyRelationshipKind* element.

3047 The instances defined in this Library SHALL be included in any **Parties** implementation. However, the
3048 implementation can allow additional instances of the class if required for a particular modeling situation. Specifying
3049 the kinds of Party relationships using this instantiation mechanism rather than a fixed enumerated list enables
3050 extension of the kinds of relationships that are possible without having to modify the standard.

3051 The following figure presents the instances for the *PartyRelationshipKind* element that are terms for the instance
3052 (`PartyRelationshipKinds`) of the *PartiesKindSet* element.

3053



3054
 3055
 3056
 3057
 3058

Figure 73: PartyRelationshipKinds

The following table provides a definition of the terms included in the *PartyRelationshipKinds* Vocabulary.

Table 129. PartyRelationshipKinds KindSet

#	Name	Documentation
1	PartyRelationshipKinds	PartyRelationshipKinds is an instance of <i>PartiesKindSet</i> that includes terms for the kinds of <i>PartyRelationships</i> that may be created in a Parties model.
2	Delegation	Delegation indicates that the target element of the <i>PartyRelationship</i> , either a <i>Party</i> or <i>PartyType</i> has been delegated the responsibilities associated with the source element, either a <i>Position</i> or <i>PositionType</i> , respectively.
3	Employment	Employment indicates that the targetParty element of the <i>PartyRelationship</i> is employed by the sourceParty.
4	General	General indicates the existence of some general relationship between the source element of the <i>PartyRelationship</i> is a member of the target element.
5	Member	Member indicates that the target element of the <i>PartyRelationship</i> is a member of the source element.

#	Name	Documentation
6	Part	Part indicates that the target element of the <i>PartyRelationship</i> is a part of the source element.
7	PositionAssignment	Assignment indicates that the source element of the <i>PartyRelationship</i> , either a <i>Party</i> or <i>PartyType</i> is assigned to the target element, either a <i>Position</i> or <i>PositionType</i> , respectively.

3059 **11.2.1 PartyRelationshipKinds**

3060 *PartyRelationshipKinds* is an instance of *PartiesKindSet* that includes terms for the kinds of
3061 *PartyRelationships* that may be created in a **Parties** model.

3062 **11.2.2 Delegation**

3063 *Delegation* indicates that the target element of the *PartyRelationship*, either a *Party* or *PartyType* has been
3064 delegated the responsibilities associated with the source element, either a *Position* or *PositionType*, respectively.

3065 **11.2.3 Employment**

3066 *Employment* indicates that the targetParty element of the *PartyRelationship* is employed by the
3067 sourceParty.

3068 **11.2.4 General**

3069 *General* indicates the existence of some general relationship between the source element of the
3070 *PartyRelationship* is a member of the target element.

3071 **11.2.5 Member**

3072 *Member* indicates that the target element of the *PartyRelationship* is a member of the source element.

3073 **11.2.6 Part**

3074 *Part* indicates that the target element of the *PartyRelationship* is a part of the source element.

3075 **11.2.7 PositionAssignment**

3076 *Assignment* indicates that the source element of the *PartyRelationship*, either a *Party* or *PartyType* is assigned
3077 to the target element, either a *Position* or *PositionType*, respectively.
3078

3079 **12 PPMN and Parties Diagram Interchange (PPMN** 3080 **DI and Parties DI)**

3081 **12.1 Scope**

3082 This chapter describes the **PPMN** and **Parties** Diagram Interchange (**PPMN DI** and **Parties DI**, respectively).
3083 **PPMN DI** extends the **Parties DI**. The **Parties DI** uses the diagram interchange capabilities provided in **SCE** (see
3084 the **SCE 1.0 Beta 1** specification (dte/22-01-04)). The **PPMN DI** is meant to facilitate the interchange of **PPMN** and
3085 **Parties** diagrams between tools rather than being used for internal diagram representation by the tools. The simplest
3086 interchange approach to ensure the unambiguous rendering of **PPMN** and **Parties** diagrams was chosen. As such,
3087 **PPMN DI** does not aim to preserve or interchange any “tool smarts” between the source and target tools (e.g.,
3088 layout smarts, efficient styling, etc.).

3089 **PPMN DI** does not ascertain that **PPMN** or **Parties** diagrams are syntactically or semantically correct.

3090 **12.2 Diagram Definition and Interchange**

3091 **PPMN DI** and **Parties DI**, through their extension of the **SCE DI** meta-model are defined as a MOF-based meta-
3092 models. As such, their instances can be serialized and interchanged using XMI. **PPMN DI** and **Parties DI** are also
3093 defined by the **SCEDI** XML schema. Thus, their instances can also be serialized and interchanged using XML.

3094 The **SCE DI** (see the **SCE 1.0 Beta 1** specification) is harmonized with the OMG Diagram Definition (DD)
3095 standard version 1.1. The referenced DD contains two main parts: the Diagram Commons (DC) and the Diagram
3096 Interchange (DI). The DC defines common types like bounds and points, while the DI provides a framework for
3097 defining domain-specific diagram models. As a domain-specific DI, **SCE DI** defines a few new meta-model classes
3098 that derive from the abstract classes DI.

3099 The focus of **PPMN DI** and **Parties DI** is the interchange of laid out shapes and edges that constitute **PPMN** and
3100 **Parties** diagrams, respectively. Each shape and edge references a particular **PPMN** or **Parties** model element. The
3101 referenced model elements are all part of an actual **PPMN** or **Parties** model. As such, **PPMN DI** and **Parties DI** are
3102 meant to only contain information that is neither present nor derivable, from the original model whenever possible.
3103 Simply put, to render a **PPMN** or **Parties** diagram both the proper **DI** instance(s) (including **PPMN**, **Parties**, and
3104 **SCE DI** instances) as well as the referenced **PPMN** and/or **Parties** model instance(s) are REQUIRED.

3105 From the **PPMN DI** perspective, a **PPMN** diagram is a particular snapshot of a **PPMN** model at a certain point in
3106 time. Multiple **PPMN** diagrams can be exchanged referencing model elements from the same **PPMN** model. Each
3107 diagram may provide an incomplete or partial depiction of the content of the **PPMN** model. The exporting tool is
3108 free to decide how many diagrams are exported and the importing tool is free to decide if and how to present the
3109 contained diagrams to the user. Similarly for **Parties DI**.

3110 **12.3 Notation**

3111 As a specification that contains elements that can notated graphically, **PPMN** specifies the depiction for **PPMN**
3112 diagram elements, including **Parties** elements and **SCE DiagramArtifact** elements.

3113 Serializing a **PPMN** diagram (including those that contain only **Parties** model elements) for interchange requires the
3114 specification of a collection of **SCEShape**(s) and **SCEEdge**(s) in the **SCEDiagram**. The **SCEShape**(s) and
3115 **SCEEdge**(s) attributes must be populated in such a way as to allow the unambiguous rendering of the **PPMN**
3116 diagram by the receiving party. More specifically, the **SCEShape**(s) and **SCEEdge**(s) MUST reference **PPMN** (or
3117 **Parties**) model elements. If no **BaseElement** is referenced or if the reference is invalid, it is expected that this shape
3118 or edge will not be depicted.

3119 When rendering a **PPMN** diagram, the correct depiction of an **SCEShape** or **SCEEdge** depends mainly on the
3120 referenced model element and its particular attributes and/or references. The purpose of this clause is to: provide a
3121 library of the **PPMN** and **Parties** element depictions, and to provide an unambiguous resolution between the
3122 referenced model element [**BaseElement**] and their depiction. Depiction resolution tables are provided below for
3123 both **SCEShape** and **SCEEdge**.

3124 **12.3.1 Labels**

3125 Both **SCEShape** and **SCEEdge** elements may have labels (its name attribute) placed on the shape/edge, or above or
3126 below the shape/edge, in any direction or location, depending on the preference of the modeler or modeling tool
3127 vendor.

3128 Labels are optional for **SCEShape** and **SCEEdge**. When there is a label, the position of the label is specified by the
3129 bounds of the **SCELabel** of the **SCEShape** or **SCEEdge**. Simply put, label visibility is defined by the presence of the
3130 **SCELabel** element.

3131 The bounds of the **SCELabel** are optional and always relative to the containing **SCEDiagram**'s origin point. The
3132 depiction resolution tables provided below exemplify default label positions if no bounds are provided for the
3133 **SCELabel** (for **SCEShape** kinds and **SCEEdge** kinds (see sections above)).

3134 When the **SCELabel** is contained in a **SCEShape**, the text to display is the name of the **BaseElement**.

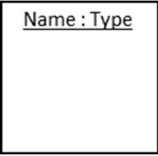




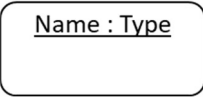
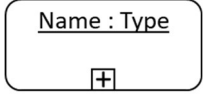
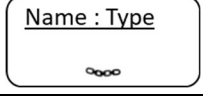
3135 **12.3.2 Shape Resolution**

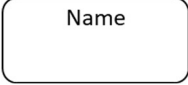


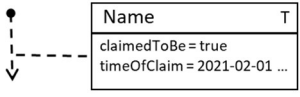
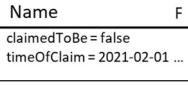
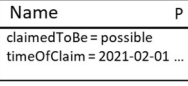
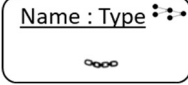
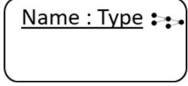
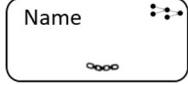
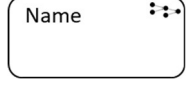
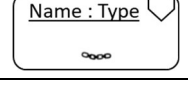
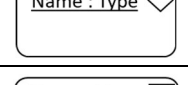
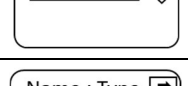
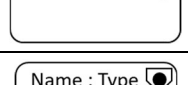
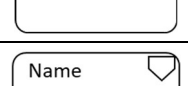
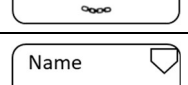
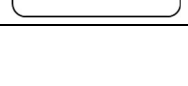
3136 *SCEShape* can be used to represent any of the non-relationship elements from **PPMN** and **Parties** models. These
 3137 include elements such as *Entity*, *EntityType*, *Occurrence*, *OccurrenceType*, *Organization*, and
 3138 *OrganizationType*. When a *SCEShape* is used to depict a diagram element the actual shape is determined by the
 3139 referred PPMN or Parties model element.

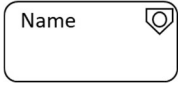
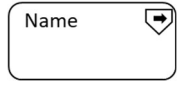
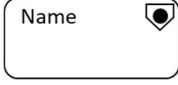
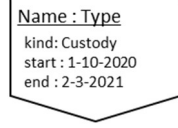
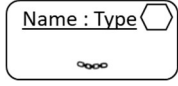
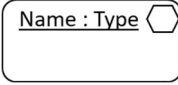
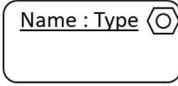
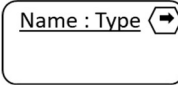
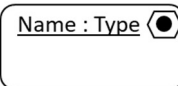

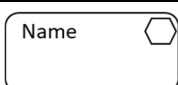
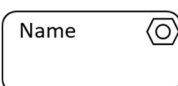
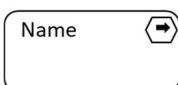
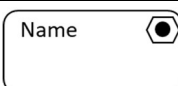

3140 **12.3.2.1 Depiction for PPMN Diagram Elements**

3141 The following table presents the depiction resolutions for **PPMN** elements:

Table 130. Depiction Resolution of PPMN Shapes

PPMN Element	PPMN Element Attributes	Depiction
Entity		
EntityType		
EntitySnapshot		
EntityTypeSnapshot		
EntityFormat		
Occurrence		
Occurrence (with Subchain)		
OccurrenceChain		

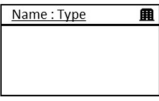
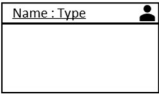
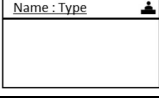
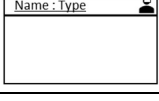
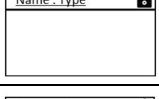
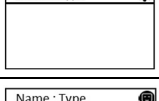
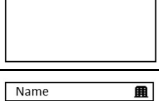
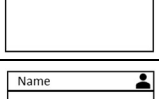
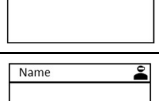
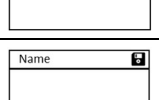
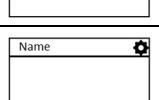
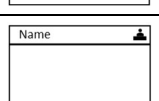
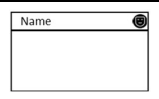


OccurrenceType		
OccurrenceChainType (with Subchain)		
OccurrenceBranchNode		
Claim (as shape)	claimedToBe = true	
Claim (as shape)	claimedToBe = false	
Claim (as shape)	claimedToBe = possible	
PedigreeChain		
PedigreeOccurrence		
PedigreeChainType		
PedigreeOccurrenceType		
CustodyChain		
CustodyOccurrence		
CustodyOccurrence (Custody Start)	kind = instance of CustodyStartKind	
CustodyOccurrence (Custody Transfer)	kind = instance of CustodyTransferKind	
CustodyOccurrence (Custody End)	kind = instance of CustodyEndKind	
CustodyChainType		
CustodyOccurrenceType		

CustodyOccurrenceType (CustodyStart type)	kind = instance of CustodyStartKind	
CustodyOccurrenceType (CustodyTransfer type)	kind = instance of CustodyTransferKind	
CustodyOccurrenceType (CustodyEnd type)	kind = instance of CustodyEndKind	
Custody (with attributes)		
OwnershipOccurrenceChain		
OwnershipChangeOccurrence		
OwnershipChangeOccurrence (Acquisition)	kind = instance of OwnershipStartKind	
OwnershipChangeOccurrence (Ownership Change)	kind = instance of OwnershipTransferKind	
OwnershipChangeOccurrence (End of Ownership Chain)	kind = instance of OwnershipEndKind	
OwnershipChainType		
OwnershipOccurrenceType		
OwnershipOccurrenceType (Ownership Start)	kind = instance of OwnershipStartKind	
OwnershipOccurrenceType (Ownership Transfer)	kind = instance of OwnershipTransferKind	
OwnershipOccurrenceType (ownership End)	kind = instance of OwnershipEndKind	
Ownership (with attributes)		

3142 **12.3.2.2 Depiction for Parties Diagram Elements**

3143 The following table presents the depiction resolutions for **Parties** elements:

Table 131. Depiction Resolution of Parties Shapes

Parties Element	Parties Element Attributes	Depiction
Organization		
Person		
Position		
NonHumanAgent		
Software		
Machinery		
PartyRole		
OrganizationType		
IndividualType (Person)	kind = Person	
IndividualType (NonHumanAgent)	kind = NonHumanAgent	
IndividualType (Software)	kind = Software	
IndividualType (Machinery)	kind = Machinery	
PositionType		
PartyRoleType		
Area		

Path		
PhysicalAddress		
NetworkAddress		
GeospacialExtent		
SpaceTime		

3144

3145 12.3.3 Edge Resolution

3146 *SCE*Edge can be used to represent and of the **PPMN** or **Parties** relationships including relationships such as
 3147 *EntityRelationship*, *OccurrenceDependency*, and *PartyRelationship*.

3148 12.3.3.1 Depiction for PPMN Diagram Elements

3149 The following table presents the depiction resolutions for **PPMN** edges:

Table 132. Depiction Resolution of PPMN Edges

PPMN Element	PPMN Element Attribute	Depiction
EntityRelationship (Generalization)	relationshipKindRef = Generalization	
EntityRelationship (Containment)	relationshipKindRef = Containment	
EntityRelationship (Composition)	relationshipKindRef = Composition	
EntityRelationship (Dependency)	relationshipKindRef = Dependency	
EntityRelationship (Miscellaneous)	relationshipKindRef = Miscellaneous	
EntityRelationship (Reference)	relationshipKindRef = Reference	
DerivedFrom		
RevisionOf		
QuotedFrom		
SourcedFrom		
DerivationType (DerivedFrom)	kind = DerivedFrom	

DerivationType (RevisionOf)	kind = RevisionOf	-----«revisionOf»----->
DerivationType (QuotedFrom)	kind = QuotedFrom	-----«quotedFrom»----->
DerivationType (SourcedFrom)	kind = SourcedFrom	-----«sourcedFrom»----->
OccurrenceRelationship		—————>
OccurrenceDependency	kind = Inputrole name> «input»
OccurrenceDependency	kind = Enablerrole name> «enabler»
OccurrenceDependency	kind = Outputrole name> «output»
OccurrenceDependency	kind = Productrole name> «product»
OccurrenceDependency	kind = By-productrole name> «by-product»
OccurrenceDependency	kind = Wasterole name> «waste»
OccurrenceDependencyType	kind = Inputrole name> «input»
OccurrenceDependencyType	kind = Enablerrole name> «enabler»
OccurrenceDependencyType	kind = Outputrole name> «output»
OccurrenceDependencyType	kind = Productrole name> «product»

OccurrenceDependencyType	kind = By-product	role name -----> «by-product»
OccurrenceDependencyType	kind = Waste	role name -----> «waste»
OccurrenceRole		role name ▶-----
OccurrenceRoleType		role type name ▶-----
OccurrenceGraphTransition	relationshipKind = Transition	—————>
Custody (as relationship)		●—————>
CustodyType (as relationship)		●—————>
Ownership (as relationship)		●—————>
OwnershipType (as relationship)		●—————>

3150 **12.3.3.2 Depiction for Parties Diagram Elements**

3151 The following table presents the depiction resolutions for **Parties** edges:

Table 133. Depiction Resolution of Parties Edges

Parties Element	Parties Element Attribute	Depiction
PartyRelationship (General)	relationshipKind = General	name -----
PartyRelationship (Member)	relationshipKind = Member	name -----> «member»
PartyRelationship (Employment)	relationshipKind = Employment	name -----> «employment»
OrganizationalStructureRelationship	relationshipKind = Part	name ◊-----
PositionAssignment	relationshipKind = PositionAssignment	name ----->
Delegation (without Authority shown)		name -----> «delegation»
Delegation (with Authority shown)	authority = <i>not null</i>	<div style="border: 1px solid black; padding: 2px; width: fit-content; margin: 0 auto;">Name : Type</div> -----> «delegation» name
PartyRelationshipType (General)	relationshipKind = Member	name -----

PartyRelationshipType (Member)	relationshipKind = Member	
PartyRelationshipType (Employment)	relationshipKind = Employment	
PartyRelationshipType (Part)	relationshipKind = Part	
PositionAssignmentType	relationshipKind = PositionAssignment	
DelegationType (without Authority shown)	relationshipKind = Delegation	
DelegationType (with Authority shown)	relationshipKind = Delegation	

3152

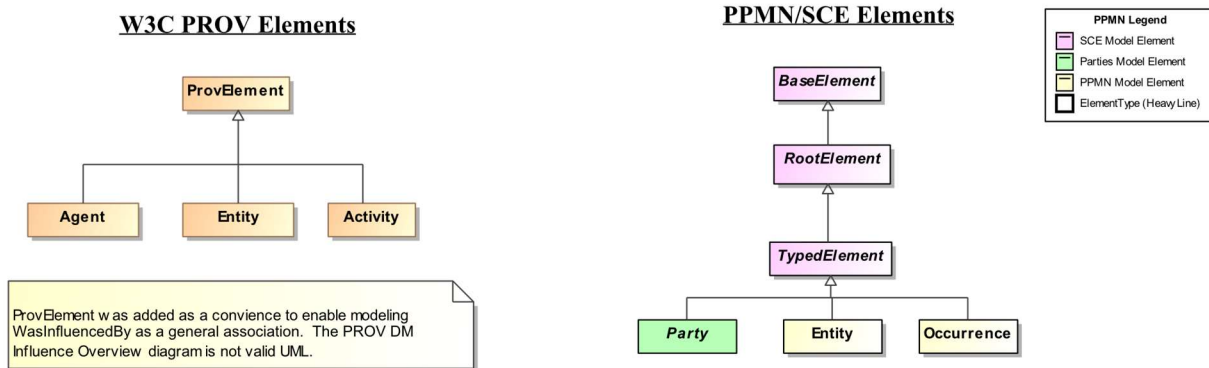
Annex A: PROV Traceability

(informative)

3153
3154
3155
3156
3157
3158
3159
3160
3161
3162

A key requirement of PPMN is to support all the capabilities available in the [W3C PROV](#) specification. This ANNEX describes the traceability of PPMN elements to elements in W3C PROV. Please note that the model of the W3C PROV specification presented herein is an interpretation in UML of that specification by the PPMN authors.

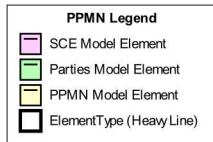
This diagram shows the PPMN and W3C PROV concepts related to the primary three PROV elements - Agent, Entity, and Activity.



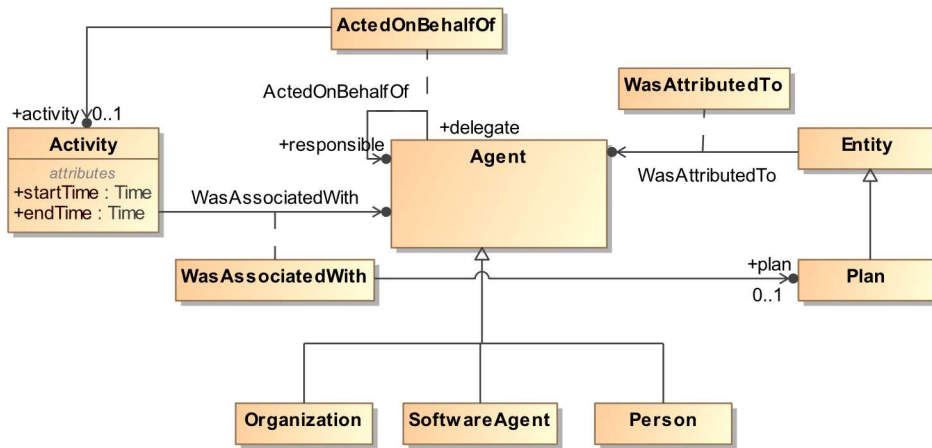
3163
3164
3165
3166

Figure 74: PPMN Trace to PROV - Primary PROV Elements

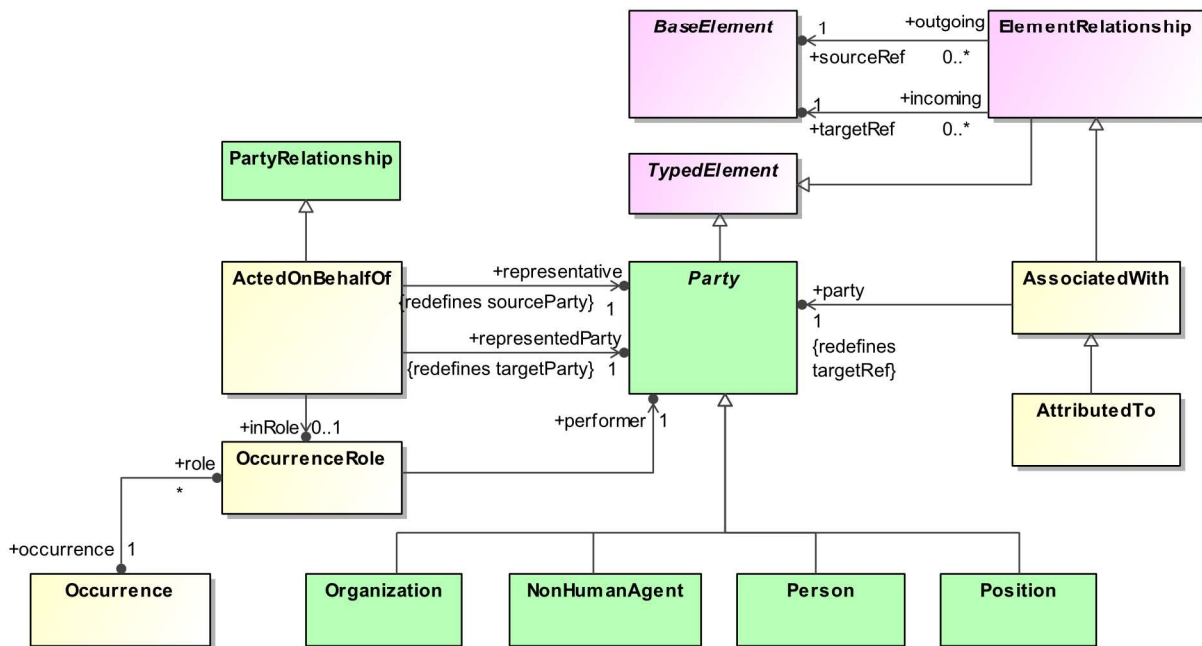
This diagram shows the PPMN and W3C PROV concepts related to Agents, Responsibility, and Influence.



W3C PROV Elements

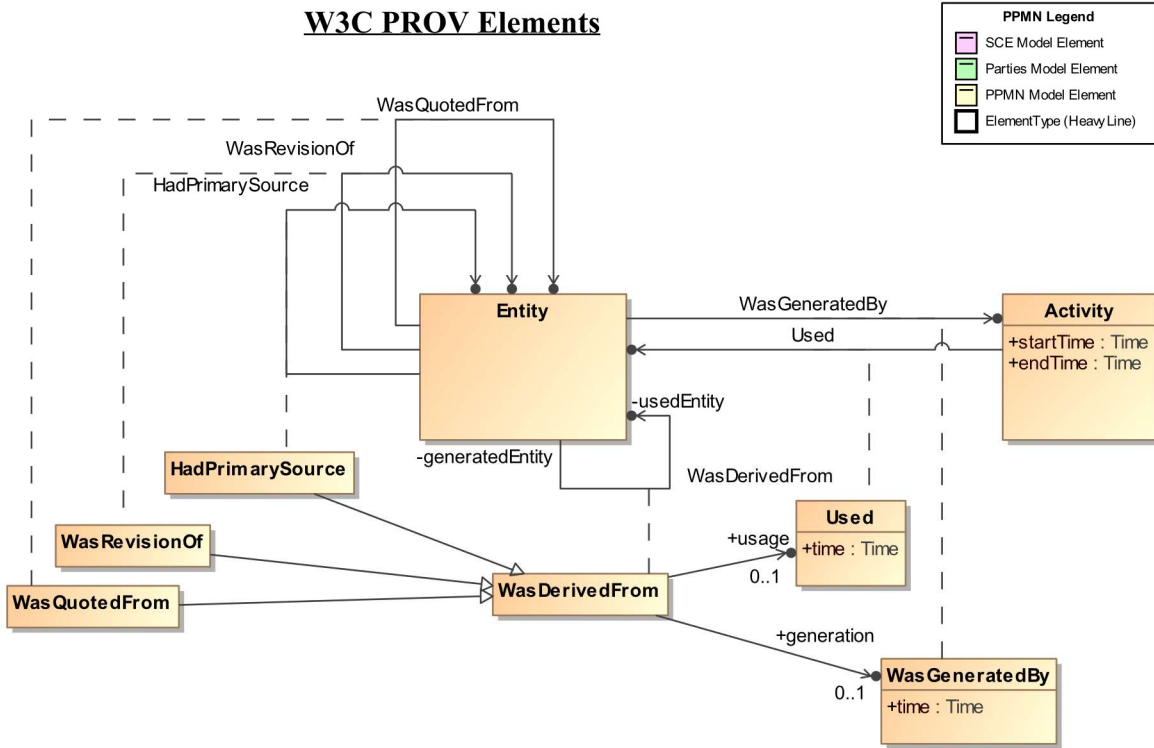


PPMN/SCE Elements

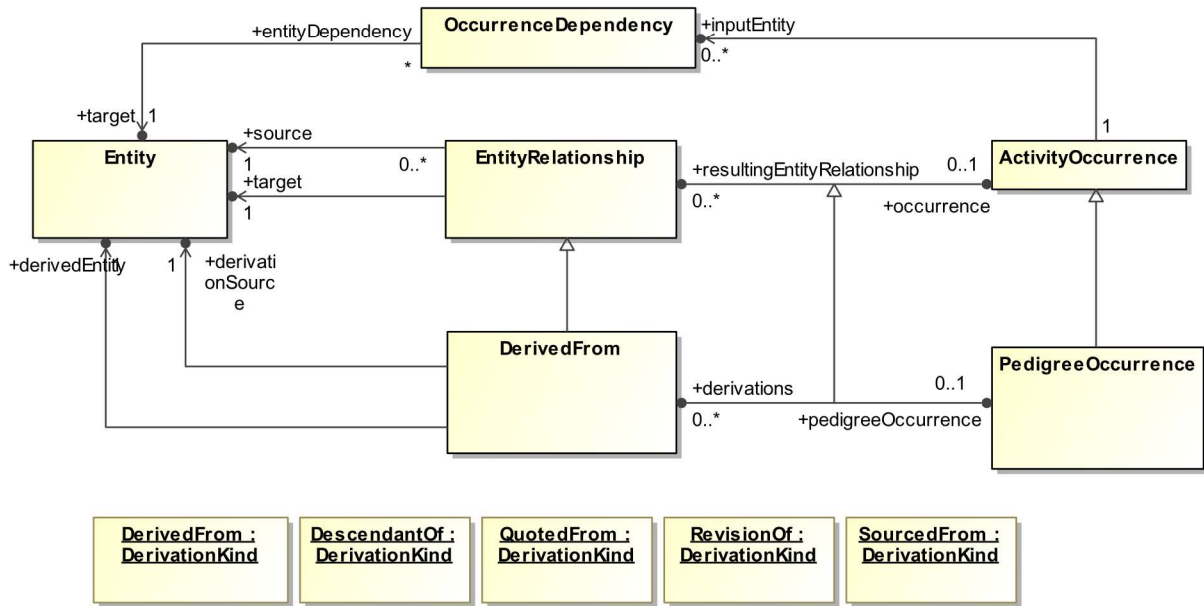


3167
 3168 **Figure 75: PPMN Trace to PROV - Agents, Responsibility, and Influence**
 3169 This diagram shows the PPMN and W3C PROV concepts related to Derivations.
 3170

W3C PROV Elements



PPMN/SCE Elements

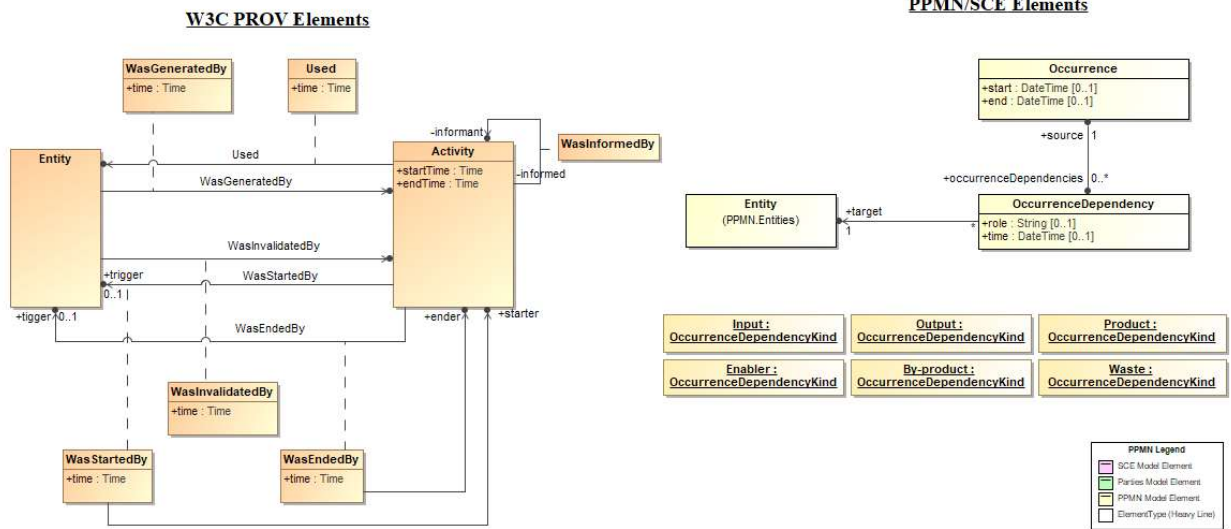


- DerivedFrom : DerivationKind
- DescendantOf : DerivationKind
- QuotedFrom : DerivationKind
- RevisionOf : DerivationKind
- SourcedFrom : DerivationKind

3171
3172 **Figure 76: PPMN Trace to PROV - Derivations**

3173 This diagram shows the PPMN and W3C PROV concepts related to Entities and their relationships to Activities (or
3174 Occurrences in PPMN).

3175

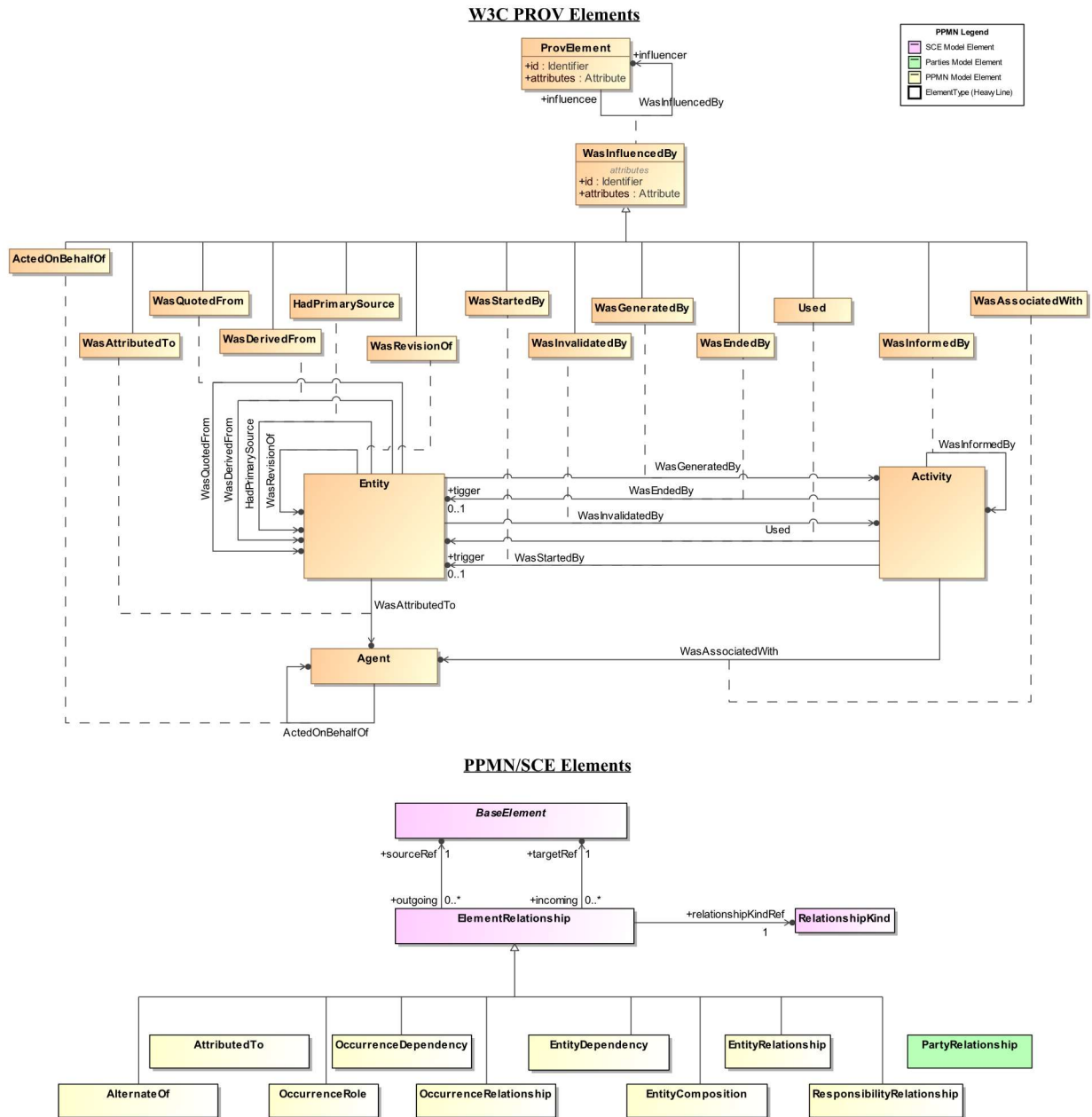


3176

3177 **Figure 77: PPMN Trace to PROV - Entities and Activities**

3178 This diagram shows the PPMN and W3C PROV concepts related to Influence.

3179



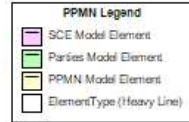
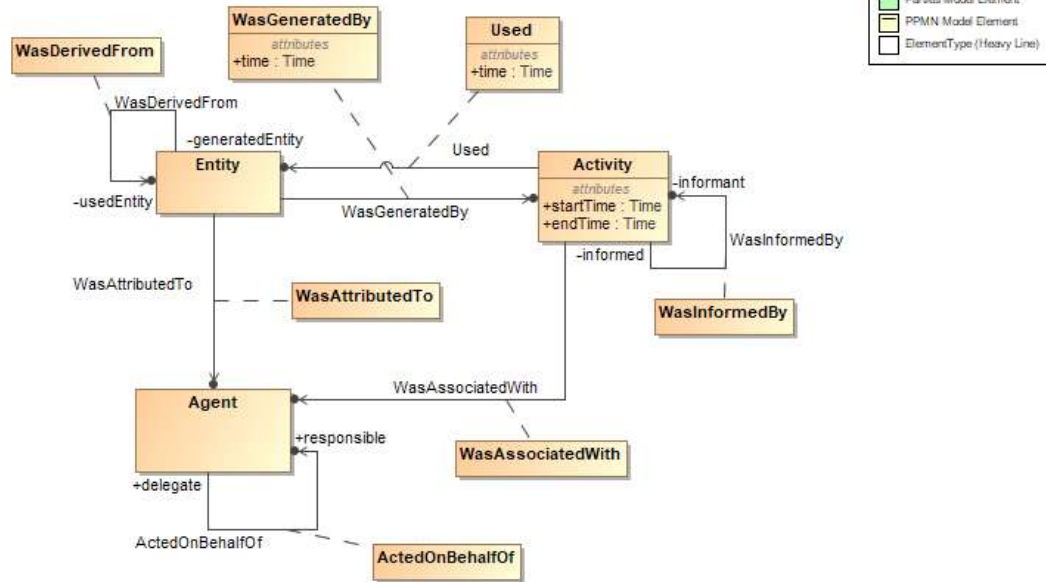
3180

3181 **Figure 78: PPMN Trace to PROV - Influence**

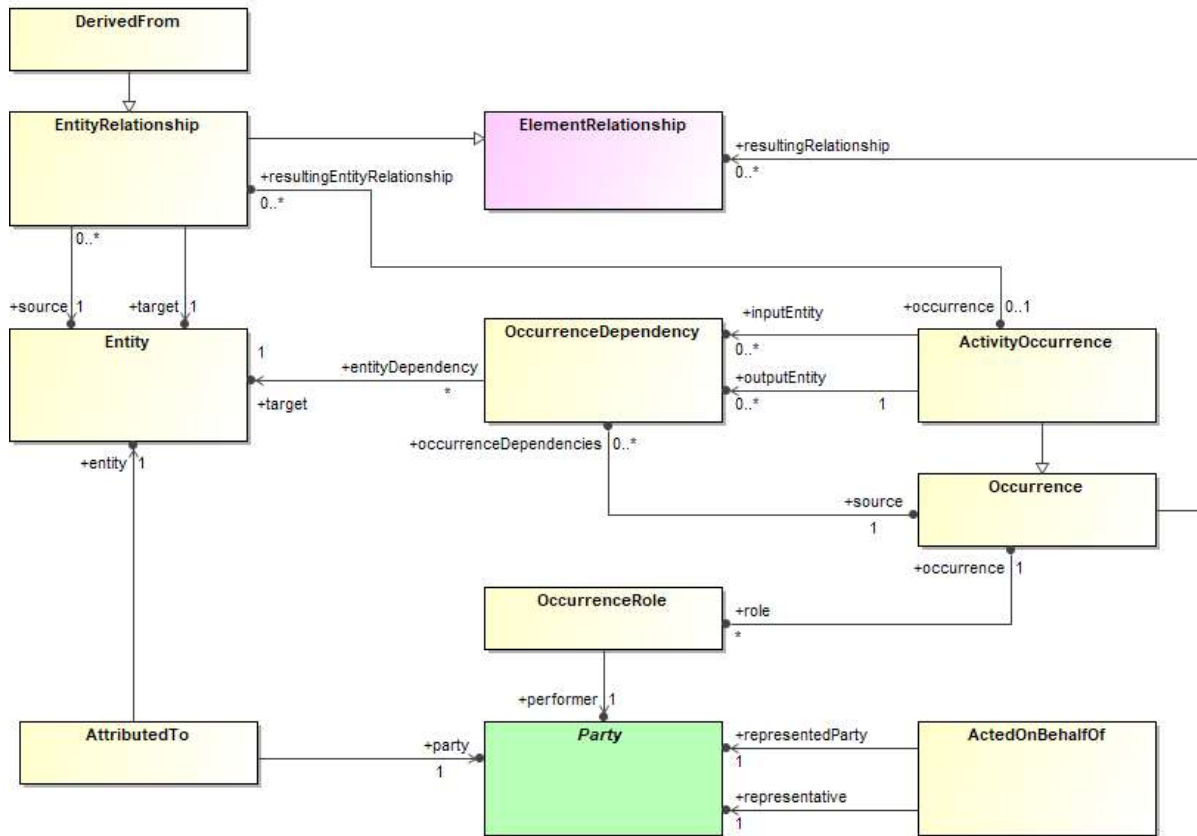
3182 This diagram shows the PPMN and W3C PROV concepts related to the core PROV elements.

3183

W3C PROV Elements



PPMN/SCE Elements



3184

3185 **Figure 79: PPMN Trace to PROV - PROV Core Structures**

3186