Pedigree and Provenance Model and Notation (PPMN)

Version 1.0 Beta 2 – Change Bar

This OMG document replaces the submission document (bmi/22-09-03). It is an OMG Adopted Beta Specification and is currently in the finalization phase. Comments on the content of this document are welcome and should be directed to issues@omg.org by March 24, 2023.

You may view the pending issues for this specification from the OMG revision issues web page https://issues.omg.org/issues/lists.

The FTF Recommendation and Report for this specification will be published in October 2023. If you are reading this after that date, please download the available specification from the OMG Specifications Catalog.

Commented [JB1]: This convenience document provides the changes (deletions and additions) to the PPMN specification based on the resolutions for issues raised for the PPMN FTF.

A comment is attached to each change in the document. The comment identifies the type of change (e.g., a figure update) and the raised Issue and its resolution sub-task. Thus, the issues will be identified as so (e.g.): PPMN-4/PPMN-51.

By searching through the document for a particular issue (e.g., PPMN-2), you can find all the changes to the specification based on the resolution for that issue.

Issues addressed:
PPMN-3/PPMN-88
PPMN-27/PPMN-106
PPMN-12/PPMN-89
PPMN-71/PPMN-92
PPMN-67/PPMN-93
PPMN-73/PPMN-110
PPMN-69/PPMN-111
PPMN-44/PPMN-103
PPMN-50/PPMN-114
PPMN-35-PPMN-104
PPMN-30/PPMN-116
PPMN-49/PPMN-129
PPMN-48/PPMN-133
PPMN-2/PPMN-134
PPMN-70/PPMN-135
PPMN-42/PPMN-137
PPMN-43/PPMN-138
PPMN-37/PPMN-143
PPMN-145/PPMN-146
PPMN-41/PPMN-147
PPMN-21/PPMN-152
PPMN-22/PPMN-153
PPMN-11/PPMN-157
PPMN-94/PPMN-95

NOTE: Original section 12, "SCE Metamodel" removed. Change accepted to reduce size of this document.

Pedigree and Provenance Model and Notation v1.0   2

124

125

126 **OMG's Issue Reporting Procedure**

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

130

Table of Contents

> **Commented [JB2]:** Updates to the Table of Contents, Table of Figures and Table of Tables accepted to reduce file size.

335

Annexes

338

339

Table of Figures

420

421

Table of Tables

556

557

Preface

OMG

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

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

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

OMG Specifications

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

https://www.omg.org/spec

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

OMG Headquarters
109 Highland Avenue
Needham, MA 02494
USA
Tel: +1-781-444-0404
Fax: +1-781-444-0320
Email: *pubs@omg.org*

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

Issues

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.

1   Scope

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

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

<div style="float:right; border:1px solid #c00; padding:4px;">**Commented [JB3]:** Updated to address PPMN-94/PPMN-95.</div>

2   Conformance

2.1   General

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

2.2   PPMN Modeling Conformance

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

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

2.3   Visual Conformance

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

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

- **PPMN** elements MAY have labels (e.g., its name and/or other attributes) placed inside the shape, or above or below the shape, in any direction or location, depending on the preference of the modeler or modeling tool vendor.

- The fills that are used for the graphical elements MAY be white or clear. The notation MAY be extended to use other fill colors to suit the purpose of the modeler or tool (e.g., to highlight the value of an object attribute).

- Graphical elements, shapes, and decorators MAY be of any size that suits the purposes of the modeler or modeling tool with the condition that the additional graphical elements SHALL NOT conflict with any current BPM+ Standard defined graphical element.

- The lines that are used to draw the graphical elements MAY be black.

642   o   The notation MAY be extended to use other line colors to suit the purpose of the modeler or tool (e.g.,
643   to highlight the value of an object attribute).
644   o   The notation MAY be extended to use other line styles to suit the purpose of the modeler or tool (e.g.,
645   to highlight the value of an object attribute) with the condition that the line style SHALL NOT conflict
646   with any current BPM+ Standard defined line style.

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

648 • New decorators or indicators MAY be added to the specified graphical elements. These decorators or
649   indicators could be used to highlight a specific attribute of a **PPMN** element or to represent a new subtype
650   of the corresponding concept with the condition that the additional graphical elements SHALL NOT
651   conflict with any current BPM+ Standard defined decorator or indicator.

652 • A new shape representing a new kind of **PPMN** element MAY be added to a model with the condition that
653   the shape SHALL NOT conflict with the shape specified for any other BPM+ Standard element or
654   decorator.

655 • Graphical elements MAY be colored, and the coloring MAY have specified semantics that extend the
656   information conveyed by the element as specified in this standard.

657 • The line style of a graphical element MAY be changed, but that change SHALL NOT conflict with any
658   other line style REQUIRED by this specification or the other BPM+ Standards.

659 • An extension SHALL NOT change the specified shape of a defined graphical element or decorator. (e.g.,
660   changing a square into a triangle, or changing rounded corners into squared corners, etc.).

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

662

3   References

3.1   Normative References

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

668 • Key words for use in RFCs to Indicate Requirement Levels, S. Bradner, IETF RFC 2119, March 1997
669   http://www.ietf.org/rfc/rfc2119.txt

670 • [BPMN] OMG Business Process and Model Notation (BPMN™): https://www.omg.org/bpmn/

671 • [CMMN] OMG Case Management Model and Model Notation
672   (CMMN™): https://www.omg.org/spec/CMMN/

673 • [DD] Diagram Definition (DD™)

674 • [DMN] OMG Decision Model and Model Notation (DMN™): https://www.omg.org/spec/DMN/

675 • [MOF] Meta Object Facility (MOF™): https://www.omg.org/spec/MOF/

676 • [SCE] Specification Core Elements (SCE): https://www.omg.org/spec/SDMN/

677 • [UML] Unified Modeling Language ™ (UML®): https://www.omg.org/spec/UML

678 • [XMI] XML Metadata Interchange (XMI®) https://www.omg.org/spec/XMI

679

3.2   Non-normative References

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

684　　　　• [MDMI] OMG Model Driven Message Interoperability (MDMI), Version 1.0:
685　　　　　https://www.omg.org/spec/MDMI/

686　　　　• [SysML] OMG Systems Modeling Language (SysML®): https://www.omg.org/spec/SysML/

4　Terms and Definitions

688　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 "*An entity is a physical, digital, conceptual, or other kind of thing with some fixed aspects; entities may be real or imaginary.*"[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.

5   Symbols

6   Additional Information

6.1   Conventions

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

6.2   Typographical and Linguistic Conventions and Style

This document incorporates the following conventions:

- The keywords "MUST," "MUST NOT," "REQUIRED," "SHALL," "SHALL NOT," "SHOULD," "SHOULD NOT," "RECOMMENDED," "MAY," and "OPTIONAL" in this document are to be interpreted as described in RFC-2119.

- A **term** is a word or phrase that has a special meaning. When a term is defined, the term name is highlighted in **bold** typeface.

- A reference to another definition, section, or specification is highlighted with underlined typeface and provides a link to the relevant location in this specification.

- A reference to a graphical element is highlighted with a bold, capitalized word (e.g., **ProcessRef**).

- A reference to a non-graphical element or **PPMN**, **Parties**, or **SCE** concept is highlighted by being italicized (e.g., *Entity*).

- A reference to an attribute or model association will be presented with the `Courier New` font (e.g., `Expression`).

- Non-normative examples are set off in boxes and accompanied by a brief explanation.

- XML and pseudo code is highlighted with `Courier New` typeface. Different font colors MAY be used to highlight the different components of the XML code.

- The cardinality of any content part is specified using the following operators:

712	o	[1] — exactly once
713	o	[0..1] — 0 or 1
714	o	[0..*] — 0 or more
715	o	[1..*] — 1 or more
716	•	Attributes separated by | and grouped within { and } — alternative values
717	o	<value> — default value
718	o	<type> — the type of the attribute

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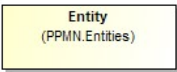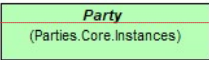
720 The metamodel presented in these sections utilizes the patterns and mechanisms that are used for the current **BPM+**
721 specifications. **BPM+** specifications rarely display the entire metamodel of a technical specification in a single
722 diagram. The entire metamodel would be very large, complicated,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

726 The metamodel diagrams are Unified Modeling Language (UML) structure diagrams. In addition to the metamodel,
727 OMG specifications provide XML schemas which map to the metamodels. In general, it is through XML documents
728 that **BPM+** models are stored and exchanged.
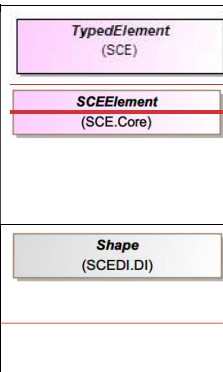
729 Further, some of the metamodel elements are references to elements from other specifications. To clarify the owner
730 of the metamodel element, there is a parenthesized text that identifies the model owner of that element. In addition,
731 colors are used to support the text identification of the owner-language of that element. The colors are used as an aid
732 to distinguish the languages but does not represent a normative aspect of the metamodels nor do they add any
733 semantic information about the metamodels.

734 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 parenthases 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.	**Entity** (PPMN.Entities)
Parties ~~General~~ **Class**	These elements include the namespace in the model of the element in parenthases 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.	*Party* (Parties.Core.Instances)

Commented [JB4]: Updated to address PPMN-69/PPMN-111

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 parenthases below the element name and these elements are color-coded lavender ~~and the border line color is black~~ (see figure to the right).	*TypedElement* (SCE) *SCEElement* (SCE.Core)
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 ~~parenthases~~ parentheses below the element name and these elements are color-coded light-gray. (see figure to the right).	*Shape* (SCEDI.DI)

Commented [JB5]: Updated to address PPMN-69/PPMN-111

Commented [JB6]: Updated to address PPMN-69/PPMN-111

735

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

737 738 739 • Diagram elements MAY have labels (e.g., its name and/or other attributes) placed inside the shape, or above or below the shape, in any direction or location, depending on the preference of the modeler or modeling tool vendor.

740 • The fills that are used for the graphical elements MAY be white or clear.

741 742 o The notation MAY be extended to use other fill colors to suit the purpose of the modeler or tool (e.g., to highlight the value of an object attribute).

743 • Diagram elements and markers MAY be of any size that suits the purposes of the modeler or modeling tool.

744 • The lines that are used to draw the graphical elements MAY be black.

745 746 o The notation MAY be extended to use other line colors to suit the purpose of the modeler or tool (e.g., to highlight the value of an object attribute).

747 748 749 o The notation MAY be extended to use other line styles to suit the purpose of the modeler or tool (e.g., to highlight the value of an object attribute) with the condition that the line style SHALL NOT conflict with any current defined line style of the diagram.

750

6.5   Abbreviations

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

Table 3.   Acronyms

Acronym	Definition
~~BHMN~~	~~BPM+ Harmonization Model and Notation~~
~~BKPMN~~	~~BPM+ Knowledge Package Model and Notation~~
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

Commented [JB7]: Updated to address PPMN-2/PPMN-134

Commented [JB8]: Updated to address PPMN-94/PPMN-95

OMG	Object Management Group
PPMN	Provenance and Pedigree Model and Notation
RFC	~~RequestRemote Function Call~~Request for Comment
SCE	Specification Common Elements
~~SDMNDI~~	~~Shared Data Model and Notation Diagram Interchange~~
SDMN	Shared Data Model and Notation
SysML	Systems Modeling Language
URI	Uniform Resource Identifier
XMI	XML Metadata Interchange
XML	Extensible Markup Language

<div style="float:right">Commented [JB9]: Text updated to address PPMN-2/PPMN-134</div>

<div style="float:right">Commented [JB10]: Text updated to address PPMN-2/PPMN-134</div>

753

6.6 Structure of this Document

755 PPMN's primary conceptual elements comprise *Entities*, *Occurrences*, and *Parties*, all of which are derived from
756 **SCE**. Section 7 "Overview" briefly explains concepts and depicts all relevant packages and their dependencies. It is
757 the architectural blueprint to use for all remaining sections of the document.

758 *Entities*, *Occurrences*, *Parties* and their associated packages fully describe the provenance and pedigree of entities.
759 Section 8 "Pedigree and Provenance Model and Notation" contains normative clauses defining model elements,
760 properties, associations, and packages of *Entities* and *Occurrences* and their relationships to *Parties*. Section 10
761 contains normative clauses defining model elements, properties, associations, and packages of *Parties*.

762 Section 9 "PPMN Library" and Section 11 "Parties Library" contain libraries of terms used within sections 8 and 10,
763 respectively.

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

766 It should be noted that the elements of PPMN and Parties build upon the elements of **SCE**, a separate specification.
767 These relationships are shown where they occur. For more detail on **SCE**, please refer to the "Specification
768 Common Elements" specification.~~This document provides a brief introduction to **SDMN** and its purpose (see the
769 section entitled "Overview"). The introduction is followed by normative clauses that define the elements of the
770 specification and their properties and associations (see the sections entitled "SDMN Metamodel" (Clause 9);
771 "SDMN Model Elements" (Clause 10); "SDMN Models" (Clause 11); "SDMN Library" (Clause 12); "Mapping to
772 BPM+ Models" (Clause 13); and "SDMN Diagram Interchange" (Clause 16)).~~

<div style="float:right">Commented [JB11]: Text updated for PPMN-3/PPMN-88</div>

773 ~~UPDATE~~

~~6.8~~6.7   Acknowledgements

775 **Supporting Organizations**

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

777 Department of Veterans Affairs
778 cébé IT
779 Knowledge Management LLC
780 Thematix Partners LLC.

781

782 **Special Acknowledgements**

783 The following individuals provided major input to this specification:

784 John Butler
785 Claude Baudoin
786 Thomas Beale
787 Elisa Kendall

788   Robert Lario
789   Pete Rivett
790   Evan Wallace
791   Steve White

792

7   Overview

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

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

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

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

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

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

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

823

824
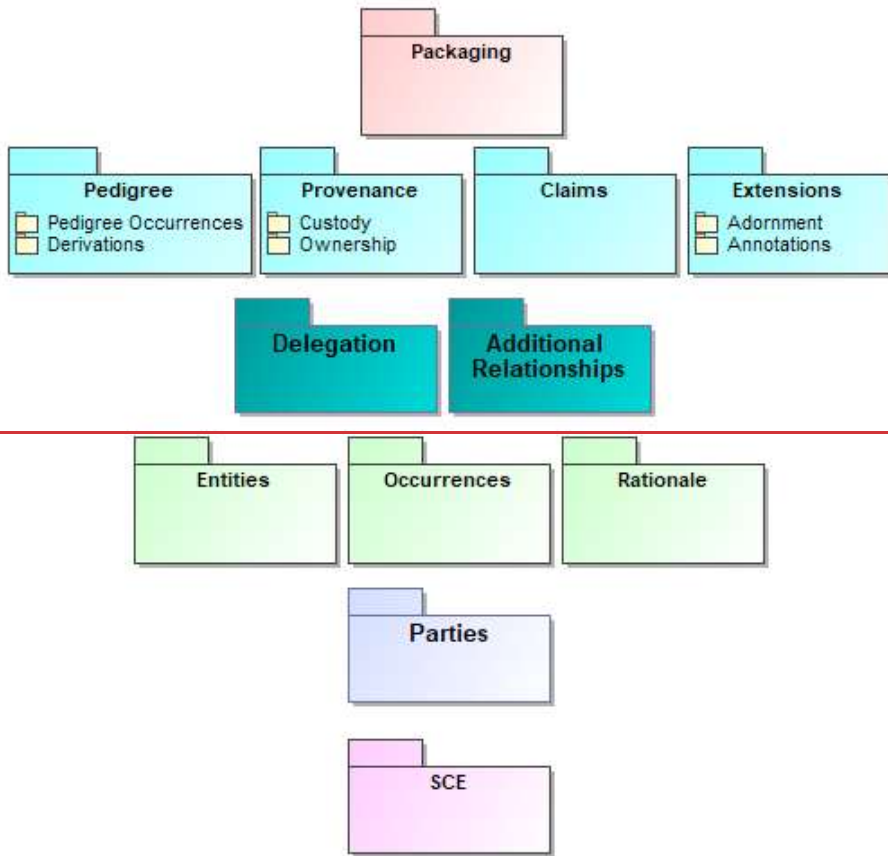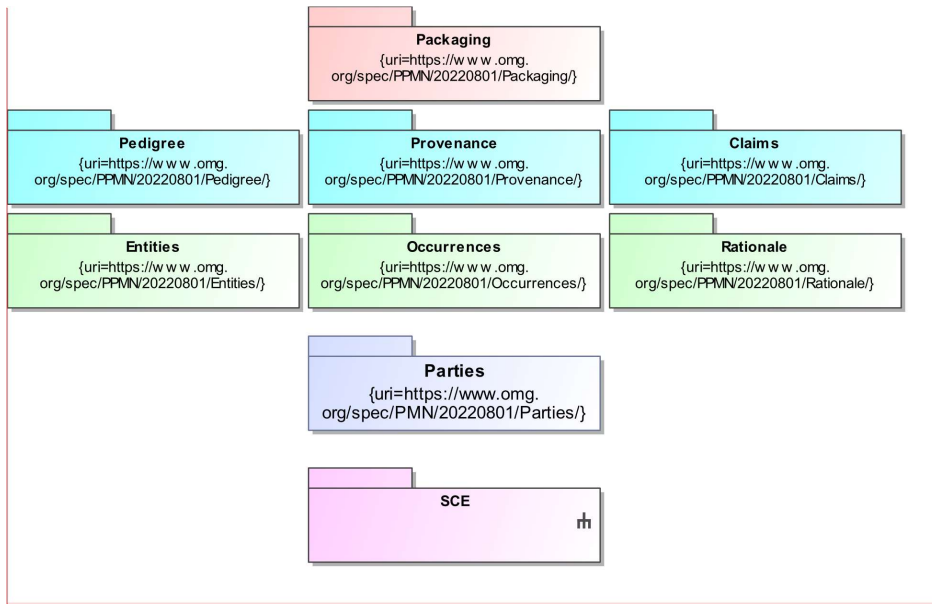
Figure 1:   PPMN Packaging Overview

8   Pedigree and Provenance Model and Notation

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

PPMN includes other packages that provide useful additions to the core model. These include "Packages", "Claims", "Rationale", "Delegation", "Additional relationships", and "Annotations". "Packages" provides a mechanism for effectively grouping elements of a **PPMN** model. "Claims" provide elements that allow users to stipulate that assertions captured in a PPMN model are only claims and may or may not be true. "Rationale" provides elements to substantiate those claims. "Delegation" comprises several elements that specify when one party has acted on behalf of another or been assigned responsibilities of another. "Additional Relationships" includes other less frequently used, but none the less important, relationships in the pedigree and provenance domain. Finally, "Annotations" provides elements that enable the addition of various types of documentation to elements of a **PPMN** model. As shown in the figure below, these domains build from the common elements specified in the Specification Common Elements (SCE) package. PPMN incorporates additional basic elements and primitives that form the foundation of the rest of the model.

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

849 The sections below describe the elements of PPMN in more detail.The sections below describe the elements of
850 PPMN in more detail.The sections below describe the elements of PPMN in more detail.The sections below describe
851 the elements of PPMN in more detail.On top of these basic elements **PPMN** lays further foundation in the form of
852 Parties, Entities and Occurrences. Parties support the specification of the organizations, people and roles they play.
853 Entities support identifying complex "things of interest" as well as references to things that might be external to the
854 system containing the pedigree and provenance metadata. Occurrences provide a general mechanism for identifying
855 *things that happen* in the form of general graphs.

856 Pedigree, Provenance and Extensions are built on top of the packages described above. Both Pedigree and
857 Provenance extend occurrences related to entities show the parties involved in those occurrences. Pedigree
858 occurrences describe the creation and/or evolution of entities while provenance occurrences describe the ownership
859 and/or custody of entities. The Extensions Annotations package provides mechanisms for adding metadata to other
860 elements of the model.

861 Finally, **PPMN** Packaging provides mechanisms for packaging occurrences and occurrence types (expected
862 occurrences) for exchange or other purposes.

863

864

Figure 2 content:

Packaging
Packaging
«import» «import»

Pedigree
Pedigree

Provenance
Custody Ownership
«import» «import»
Provenance
«import»

Claims
Claims
«import»
«import»

Additional Elements
«import»
Delegation Additional Relationships
«import»

Entities
Entities
«import»

Occurrences
Occurrences
«import» «import»

Rationale
Rationale
«import»

Extensions
«import»
Annotations

Parties
Parties

Common Elements
SCE

865
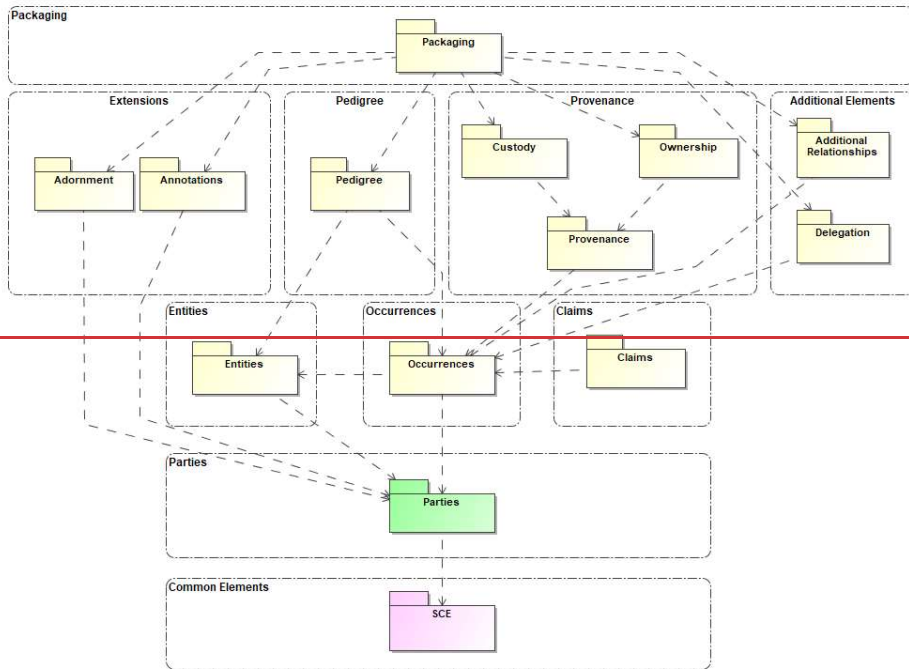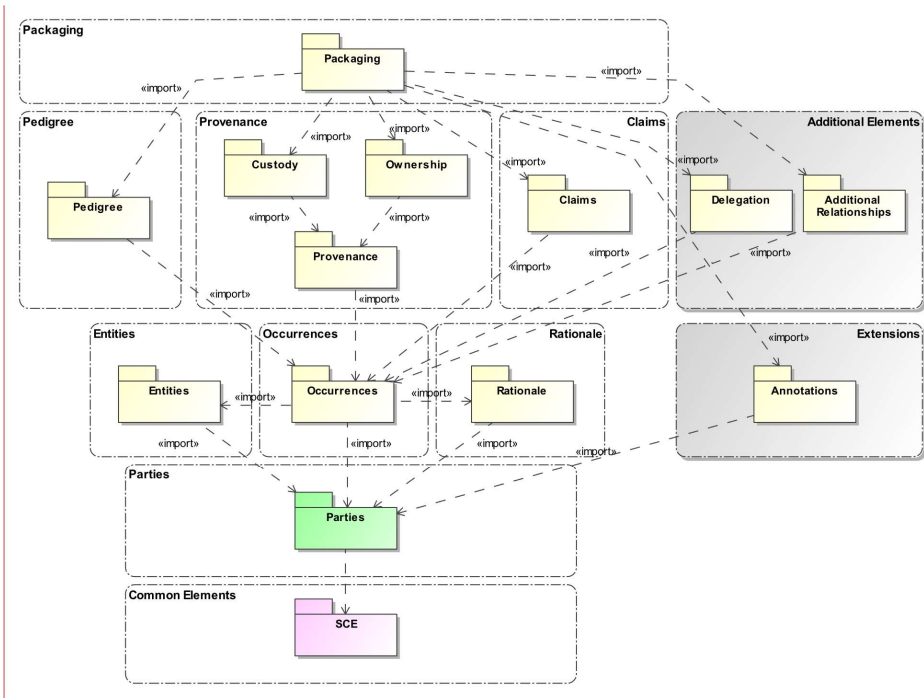
Figure 2:   Pedigree and Provenance Packaging

8.1   Entities

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

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

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

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

885

886
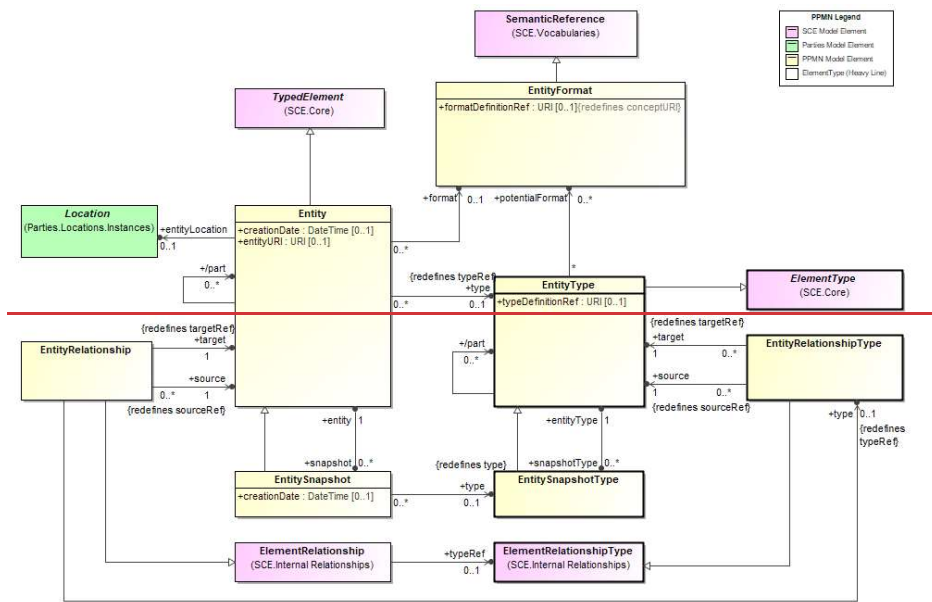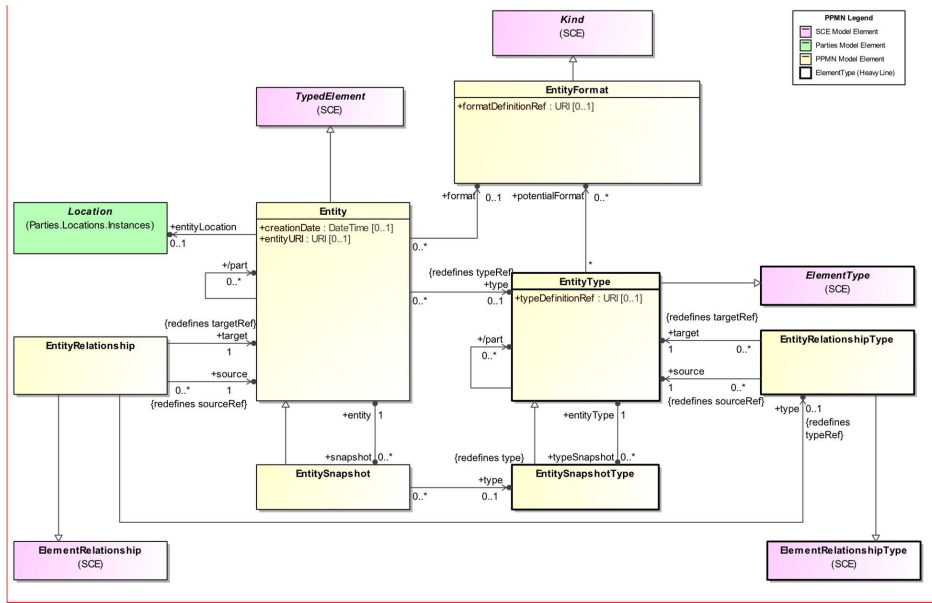
Figure 3: **Entities and EntityTypes**

887

Commented [JB21]: Diagram updated for PPMN-19/PPMN-83.

888 *Entities* and *EntityTypes* are related by *EntityRelationships* and *EntityTypeRelationships*, respectively. (See the
889 figure below). These relationships are used to show how *Entities* and *EntityTypes* are inter-related.
890 *EntityRelationship* is a specialization of **SCE** *ElementRelationship* whose `type` is *ElementRelationshipType* and
891 whose `kind` is a **SCE** *RelationshipKind*. *EntityRelationshipType* is a specialization of **SCE**
892 *ElementRelationshipType* and whose `kind` is also **SCE** *RelationshipKind*.

893
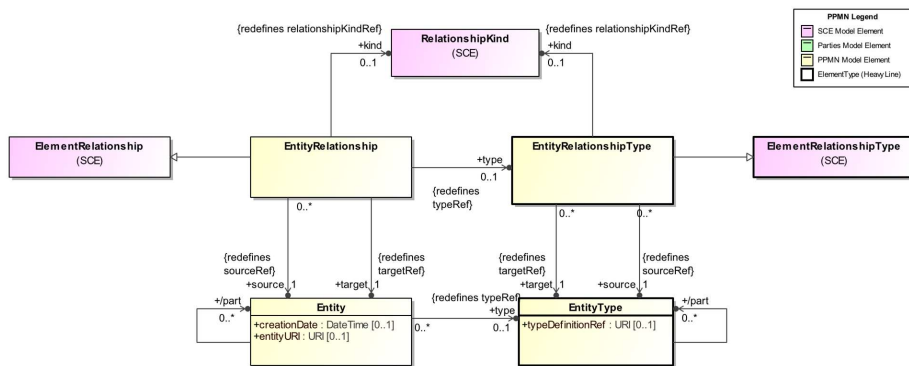
894

895 **Figure 4:   Entity Relationships**

896

897 8.1.1   Entity

898 An individual concept, ~~or~~ informational or physical artifact, or other kind of thing that is concretized in digital or
899 other ~~form.~~media or in a physical representationform. The W3C PROV-DM defines an entity as "*An entity is a*
900 *physical, digital, conceptual, or other kind of thing with some fixed aspects; entities may be real or imaginary.*"[2]
901 Entities may have a type and format, captured through the *EntityType* and *EntityFormat*, respectively. These two
902 classes are used together to support specifying generally what kind of thing an *Entity* is and the form it may take.
903 For example, the *EntityType* might be a "building permit" and the *EntityFormat* might be ".gif". Additionally,
904 Entities may have a location as captured by the `entityLocation` property.

905 Generalizations

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

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

908 Properties

909 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 *Entity* was created.
entityLocation : Location [0..1]	The location of the *Entity*.

---

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

entityURI : URI [0..1]	A URI to the Entity.
format : EntityFormat [0..1]	The format of the *Entity*.
part : Entity [0..*]	A derived property that indicates thethethe~~TtheTthe~~The *Entity* or *Entities* that ~~is/are contained~~ comprise ~~by~~ the *Entity*. This is determined by *EntityRelationships* whose source is the *Entity* and whose `kind` is "Composition". (See the **SCE** specification for more information.)
snapshot : EntitySnapshot [0..*]	The snapshots of the *Entity* that represent the *Entity* at some particular point in time, particular *Location*, or both.
type : EntityType [0..1]	The type of the *Entity*.

Commented [JB24]: Text updated for PPMN-71/PPMN-92.

910

8.1.2 EntityFormat

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

Commented [JB25]: Text updated for PPMN-19/PPMN-83

Generalizations

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

916 • *~~SemanticReference~~ Kind* (see the ~~section entitled "SemanticReference"~~SCE Specification for more
917 information).

Commented [JB26]: Text updated for PPMN-19/PPMN-83

Properties

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

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.

920

8.1.3 EntityRelationship

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

Generalizations

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

926 • *ElementRelationship* (see the ~~section entitled "ElementRelationship"~~SCE Specification for more
927 information).

Commented [JB27]: Text updated for PPMN-19/PPMN-83

Properties

929 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 *Occurrence* that resulted in the relationship.
source : Entity [1]	The source *Entity* of the relationship.
target : Entity [1]	The target *Entity* of the relationship.
type : EntityRelationshipType [0..1]	A specification of the type of EntityRelationship.
kind : RelationshipKind [0..1]	The kind of EntityRelationship.

> **Commented [JB28]:** Text updated for PPMN-71/PPMN-92.

930

8.1.4   EntityRelationshipType

931

932 A kind of
933 *ElementRelationshipTypeElementRelationshipTypeElementRelationshipTypeElementRelationshipTypeElementRelati*
934 *onshipTypeElementRelationship* that represents an expected relationship between two *EntityTypes*. The kind of
935 *EntityRelationshipTypeEntityRelationshipTypeEntityRelationshipTypeEntityRelationshipTypeRelationshipEntityRel*
936 *ationshipTypeRelationshipEntityTypeRelationship* is specified by the ~~type~~ kind property inherited from
937 *ElementRelationshipType.~~ElementRelationship.~~*

> **Commented [JB29]:** Updated to address PPMN-27/PPMN-106

Generalizations

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

940 • *ElementRelationshipType* (see the ~~section entitled "ElementRelationshipType"~~SCE Specification for more
941   information).

> **Commented [JB30]:** Text updated for PPMN-19/PPMN-83

Properties

943 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 *EntityType* of the relationship.
target : EntityType [1]	The target *EntityType* of the relationship.
kind : RelationshipKind [0..1]	The kind of EntityRelationshipType.

> **Commented [JB31]:** Text updated for PPMN-71/PPMN-92.

944

8.1.5   EntitySnapshot

945

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

> **Commented [JB32]:** Spelling

Generalizations

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

951 • *Entity* (see the section entitled "Entity" for more information).

Properties

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

Table 8. **EntitySnapshot Attributes and/or Associations**

Property/Association	Description
~~creationDate : DateTime [0..1]~~	~~The date the *EntitySnapshot* was created.~~
entity : Entity [1]	The *Entity* that the *EntitySnapshot* represents at some particular point in time and potentially some *Location*.
type : EntitySnapshotType [0..1]	The type of the *Entity*.

> **Commented [JB33]:** Text updated to address PPMN-22/PPMN-153

954

955 8.1.6 EntitySnapshotType

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

959 **Generalizations**

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

961 • *EntityType* (see the section entitled "EntityType" for more information).

962 **Properties**

963 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 *EntityType* that the *EntityTypeSnapshot* represents at some particular point in time, particular *Location*, or both.

964

965 8.1.7 EntityType

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

970 **Generalizations**

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

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

973 **Properties**

974 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..*]	The *EntityType* or *EntityTypes* that ~~is/are contained~~ comprise ~~by~~ the *EntityType*. This is determined by *EntityRelationshipTypes* whose source is the *EntityType* and whose `kind` is "Composition". (See the **SCE** specification for more information.)
potentialFormat : EntityFormat [0..*]	Formats in which *Entities* of type *EntityType* may exist.
snapshotType : EntitySnapshotType [0..*]	The snapshots of the *EntityType* that represent the *EntityType* at some particular point in time, particular *Location*, or both.
typeDefinitionRef : URI [0..1]	An external definition of the EntityType.

Commented [JB34]: Text updated for PPMN-67/PPMN-93.

975

8.2   Occurrences

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

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

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

989

990

991
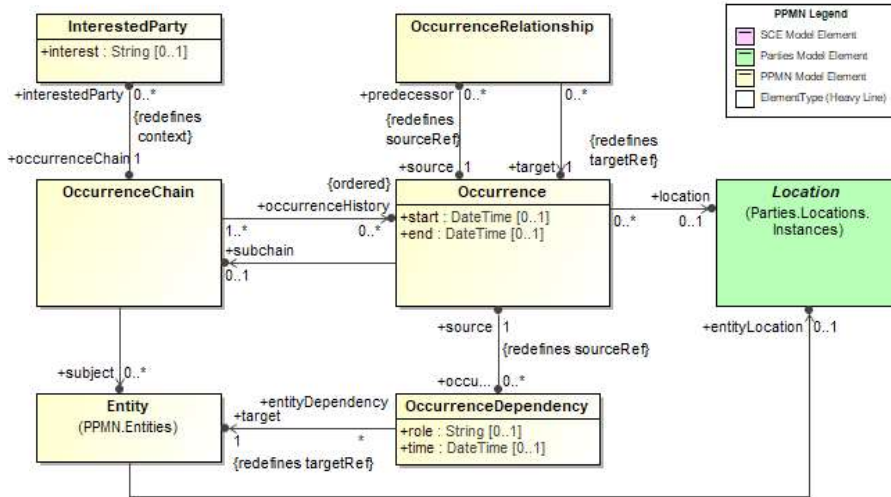
Figure 4:Figure 5:   **Occurrences - Simplified**

993 In addition, *Occurrences* may also result in some number of *ElementRelationships* between elements that were
994 involved in the *Occurrence*. These include *DerivedFrom* relationships (see section 8.3.2, below) as well as
995 *ResponsibilityRelationships* (see section 8.4.9, below).
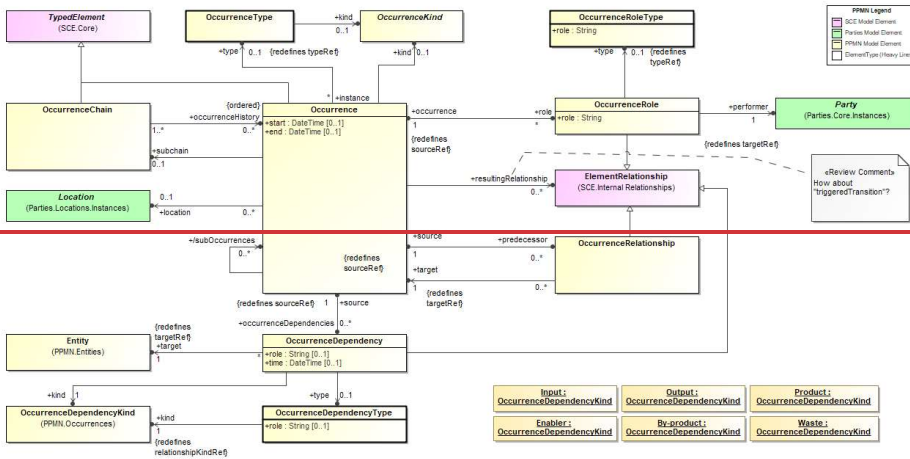
996

997
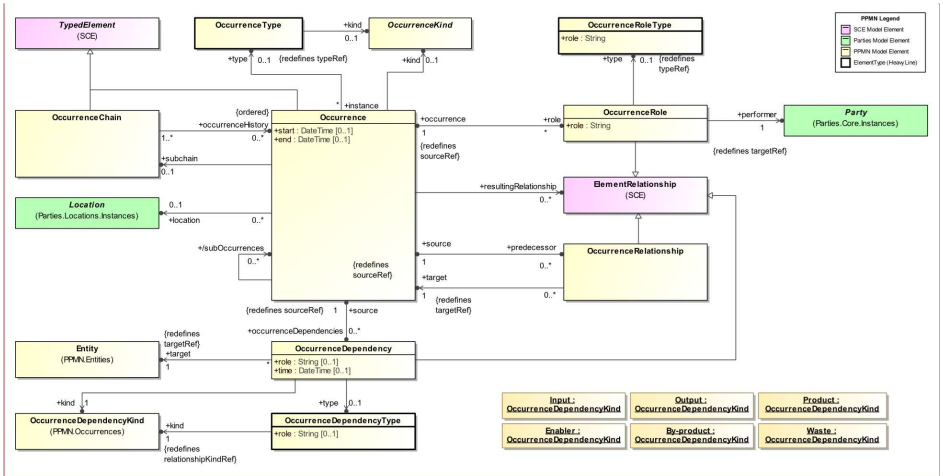
998 ~~Figure 5:~~Figure 6:   **Occurrences**

999 *OccurrenceChains* are *TypedElements* that track some series of *Occurrences* related to one or more *Entities* that acts
1000 as the context of the *Occurrences*.

1001
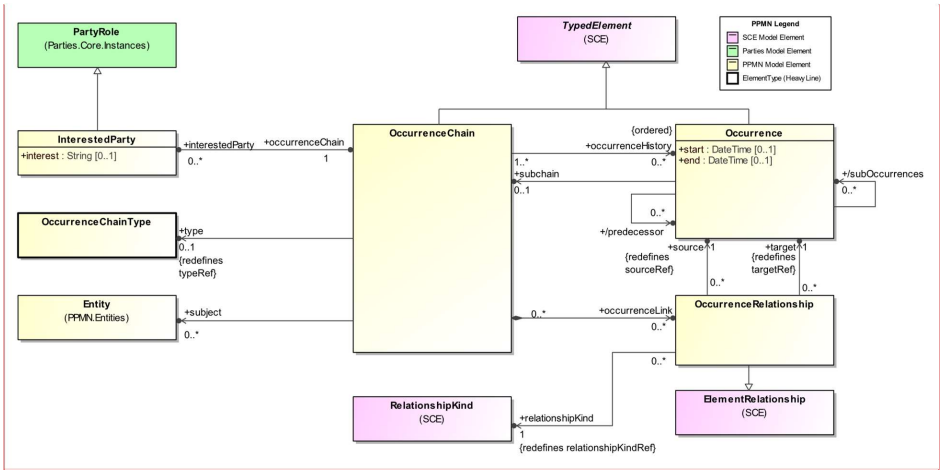
1002

1003

1004
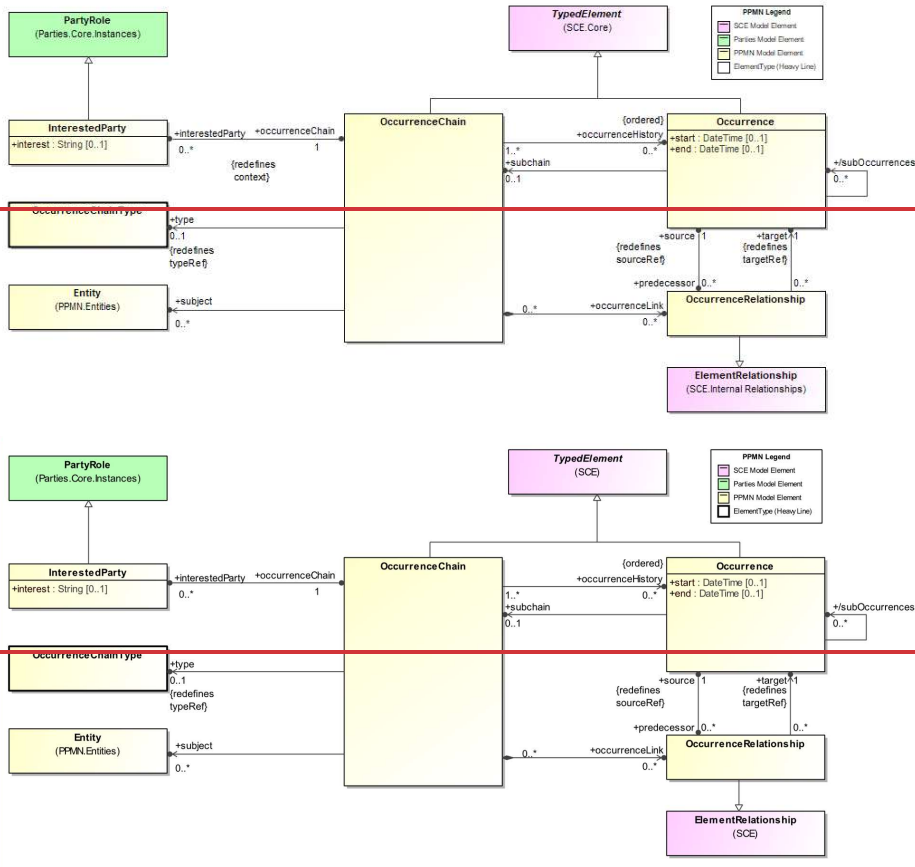
1005 ~~Figure 6:~~Figure 7:   **Occurrence Chains**

1006 *OccurrenceTypes* support the definition of expected Occurrences in an *OccurrenceChain*. Essentially,
1007 *OccurrenceTypes* represent *Occurrence* instances that are expected to happen to entities of a particular type from the
1008 perspective of the *InterestedParties. OccurrenceTypes* can be organized into graphs, *OccurrenceTypeGraphs*, that
1009 show an expected sequence or "chain" of those types of *Occurrences*. Further, *OccurrenceTypes* can optionally
1010 have sub-chain types so that *OccurrenceTypeGraphs* can be nested within one another. *OccurrenceTypeRole*
1011 captures roles expected to be played by *Parties* in those *Occurrences*.

1012
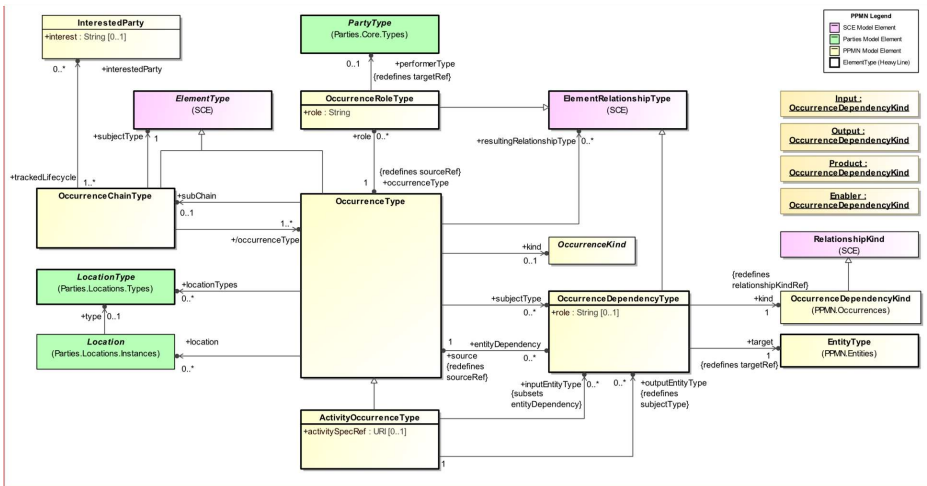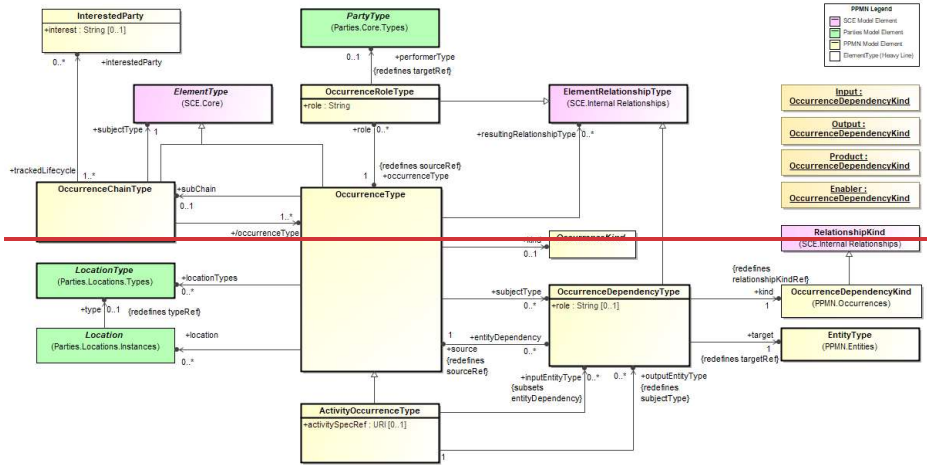
1013

1014

Figure 7:Figure 8: **Occurrence Types**

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

1021
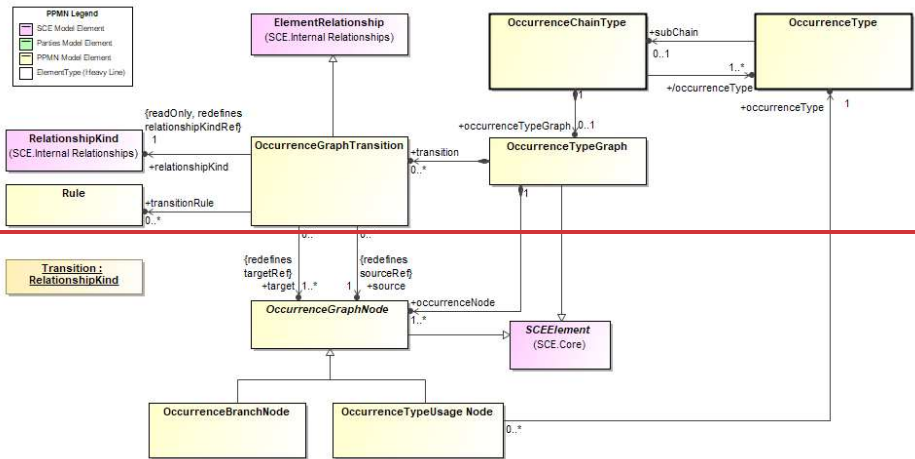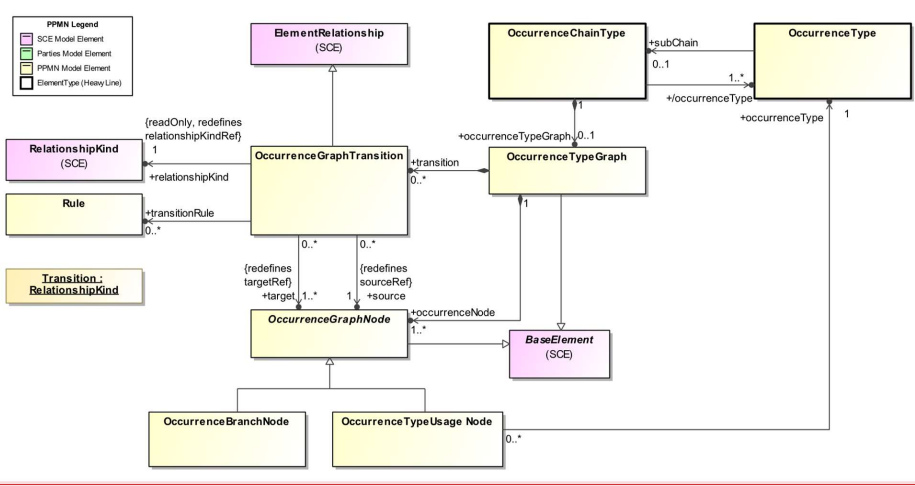
1022

1023 ~~Figure 8:~~Figure 9:　　**Occurrence Type Graphs**

1024 **PPMN** establishes a pattern of elements that supports the "nesting" of *OccurrenceChains* within an *Occurrence*.
1025 This pattern allows for encapsulation of parts of a chain where the details of the *Occurrences* of that part of a larger
1026 chain are either not known initially or are not deemed important in some context. The figure below illustrates ~~the~~
1027 this pattern at the "type" level.

1028
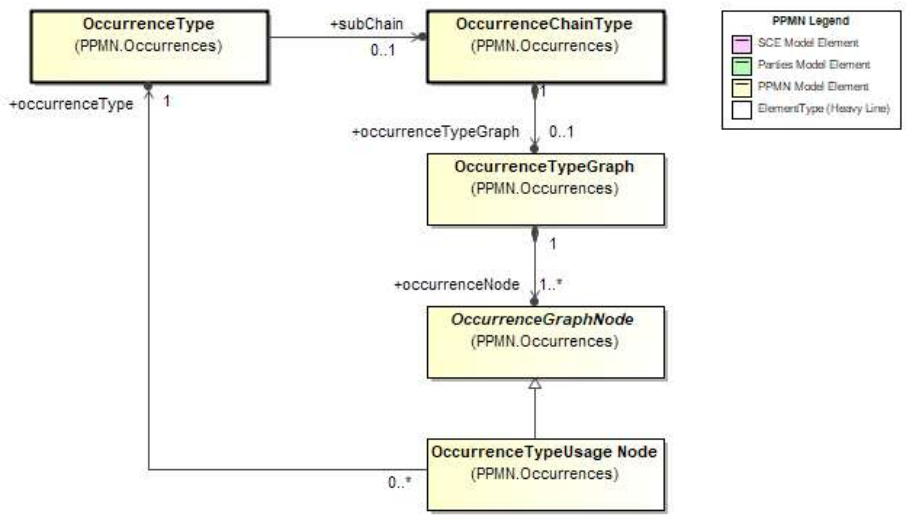
1029

1030 ~~Figure 9:~~Figure 10:   **Occurrences Type Pattern**

1031

1032

1033
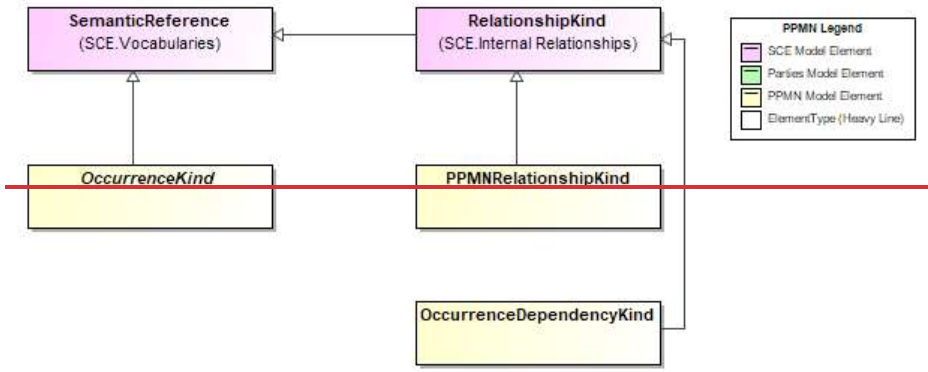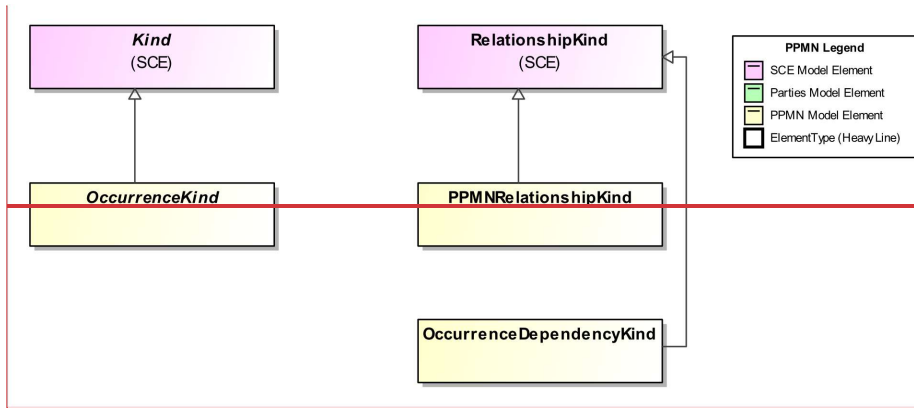
1034   ~~**Figure 10:   Occurrence Kinds**~~

1035   An *ActivityOccurrence* is a kind of *Occurrence* that represents some activity that produces or modifies one or more
1036   entities. The *ActivityOccurrenceType*~~The *ActivityType*~~ specifies the type of activity of the *ActivityOccurrence*
1037   providing a URI for a specification of the *ActivityOccurrenceType*. ~~*ActivityType.*~~

1038   *ActivityOccurrence* has a name (inherited), a URI reference to a specification of the instance if one exists, and a
1039   description. *ActivityOccurrence* includes zero or more references to *Parties* that play a part in the activity through
1040   the inherited *OccurrenceRole* property and references to the entities used in the activity through the `inputEntity`
1041   property which holds a collection of *OccurrenceDependencies*. Output entities of the activity are captured through
1042   the ~~outputEntities~~ outputEntity property.

1043

Figure 11:   Activity Occurrences

8.2.1   ActivityOccurrence

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

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

ActivityOccurrences have a name (inherited), a URI reference to an instance if one exists, and a description. *ActivityOccurrences* include 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 ~~outputEntities~~ `outputEntity` property of *PedigreeOccurrence*.
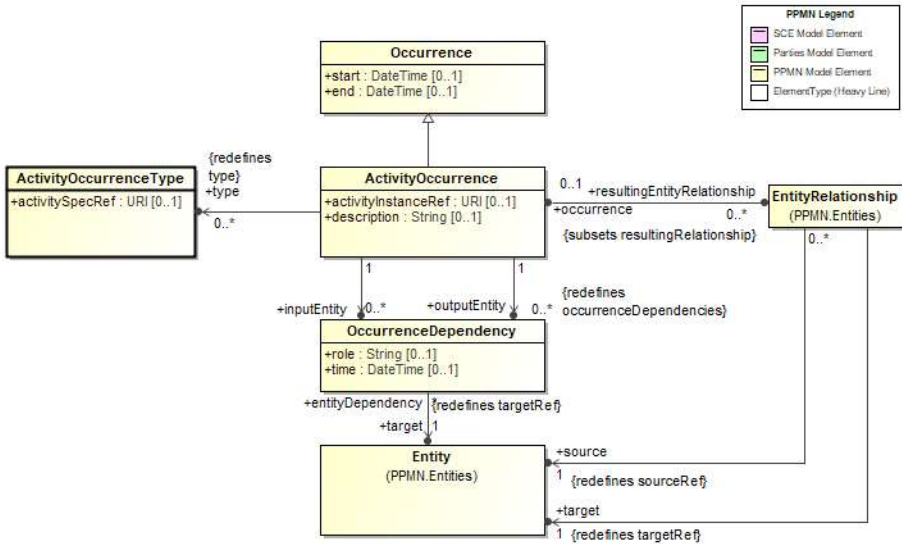
Figure 12: Activity Occurrence

Generalizations

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

- *Occurrence* (see the section entitled "Occurrence" for more information).

Properties

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 *ActivityOccurrence* 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 *ActivityOccurrence*.
outputEntity : OccurrenceDependency [0..*]	A set of dependencies on entities that were outputs or results of the *ActivityOccurrence*.
resultingEntityRelationship : EntityRelationship [0..*]	EntityRelationships created as a result of the Occurrence.
type : ActivityOccurrenceType [0..*]	The type of the *ActivityOccurrence*.

1066 ### 8.2.2 ActivityOccurrenceType

1067 A potentially complex *OccurrenceType* that identifies an expected activity that may have input and output entities of
1068 interest.

1069 **Generalizations**

1070 The *ActivityOccurrenceType* element inherits the attributes and/or associations of:

1071 • *OccurrenceType* (see the section entitled "OccurrenceType" for more information).

1072 **Properties**

1073 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 *OccurrenceType*.
outputEntityType : OccurrenceDependencyType [0..*]	A set of dependencies that point to the types of entities that are expected to be produced by instances of the *OccurrenceType*.

1074

1075 ### 8.2.3 InterestedParty

1076 A kind of *PartyRole* that captures the fact that a *Party* has some interest in a particular occurrence chain as specified
1077 by its occurenceChain property or so some set of *OccurrenceChains* as defined by an *OccurrenceChainType*.

1078 **Generalizations**

1079 The *InterestedParty* element inherits the attributes and/or associations of:

1080 • *PartyRole* (see the section entitled "PartyRole" for more information).

1081 **Properties**

1082 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 *Party* has in the *Occurrences*.
occurrenceChain : OccurrenceChain [1]	The *OccurrenceChains* of interest to some *Party*.

1083

1084 ### 8.2.4 Occurrence

1085 A *Occurrence* or "happening" of importance in a domain in some context.

1086 **Generalizations**

1087 The *Occurrence* element inherits the attributes and/or associations of:

1088 • **SCE** *TypedElement* (see the section **SCE** specification for more information).

1089 **Properties**

1090 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 *DateTime* of the end of the *Occurrence*.
kind : OccurrenceKind [0..1]	A reference to a definition of the specific kind of Occurrence.
location : Location [0..1]	The location at which an *Occurrence* took place.
occurrenceDependencies : OccurrenceDependency [0..*]	A dependency on the subject(s) of the *Occurrence*.
predecessor : Occurrence~~Relationship~~ [0..*]	A derived property indicating a dependency on ~~a~~ one or more preceding *Occurrences*. ~~Occurrence.~~
rationale : Rationale [0..*]	The *Rationale* given for the *Occurrence*.
resultingRelationship : ElementRelationship [0..*]	The relationships generated by the *Occurrence*.
role : OccurrenceRole [*]	A role played by some *Party* in an *Occurrence*.
start : DateTime [0..1]	The *DateTime* of the start of the *Occurrence*.
subchain : OccurrenceChain [0..1]	An *OccurrenceChain* that is encapsulated by the *Occurrence* - essentially a "sub-chain".
subOccurrences : Occurrence [0..*]	A set of *Occurrences* that happen as part of the parent *Occurrence*. These *Occurrences* are normally part of a "sub-chain".
type : OccurrenceType [0..1]	The type of an *Occurrence*.

Commented [JB47]: Updated to address PPMN-73/PPMN-110.

1091

8.2.5   OccurrenceBranchNode

1093 A kind of *OccurrenceGraphNode* that allows for branching or other kinds of connections between other
1094 *OccurrenceGraphNodes*.

1095 **Generalizations**

1096 The *OccurrenceBranchNode* element inherits the attributes and/or associations of:

1097 • *OccurrenceGraphNode* (see the section entitled "OccurrenceGraphNode" for more information).

1098 **Properties**

1099 The *OccurrenceBranchNode* element does not have any additional attributes and/or associations.

8.2.6   OccurrenceChain

1101 A succession of *Occurrences* (events or activities) that have happened in the life of some *RootElement*
1102 ~~*NamedElement*~~*RootElement* that are of interest to some *Party*.

Commented [JB48]: Text updated for PPMN-19/PPMN-83

1103 **Generalizations**

1104 The *OccurrenceChain* element inherits the attributes and/or associations of:

1105   •   **SCE** *TypedElement* (see the section **SCE** specification for more information).

1106 **Properties**

1107 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 *Parties* that have some interest in *Occurrences* related to the `subject` elements.
occurrenceHistory : Occurrence [0..*]	A set of *Occurrences* that comprise the chain.
occurrenceLink : OccurrenceRelationship [0..*]	The *OccurrenceRelationship(s)* that show(s) the relationship(s) between *Occurrences* in the chain.
subject : Entity [0..*]	The element(s) that is(are) the result of the *Occurrences* in the chain.
type : OccurrenceChainType [0..1]	The type of the *OccurrenceChain*.

1108

1109 ## 8.2.7   OccurrenceChainType

1110 An *OccurrenceChainType* is a kind of *ElementType* that captures a specification for a series potential *Occurrences*
1111 that are expected in a particular context. An *OccurrenceChainType* captures this specification through the
1112 occurrenceTypeGraph property - a graph of *OccurrenceGraphNodes* and *OccurrenceTransitionTypes*.

1113 **Generalizations**

1114 The *OccurrenceChainType* element inherits the attributes and/or associations of:

1115   •   **SCE** *ElementType* (see the section **SCE** specification for more information).

1116 **Properties**

1117 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 *OccurrenceChainType*.
occurrenceType : OccurrenceType [1..*]	The `occurrenceType` derived property is based on the series of relationships between from *OccurrenceChainType* through other classes to *OccurrenceType*: `OccurrenceChainType.occurrenceTypeGraph.occurrenceNode.occurrenceType`.
occurrenceTypeGraph : OccurrenceTypeGraph [0..1]	A graph of *OccurrenceTypes* that specifies the sequencing of expected *Occurrences* in the lifecycle of an entity of interest to one or more *InterestedParties*.

subjectType : ElementType [1]	The subject of the *OccurrenceChainType*.

1118

8.2.8   OccurrenceDependency

1119

1120 A type of relationship that records the dependence on an entity of interest for some particular purpose. That purpose
1121 is captured as the `role`.

1122 *OccurrenceDependencies* indicate how *Entities* are used within an *Occurrence*.

1123
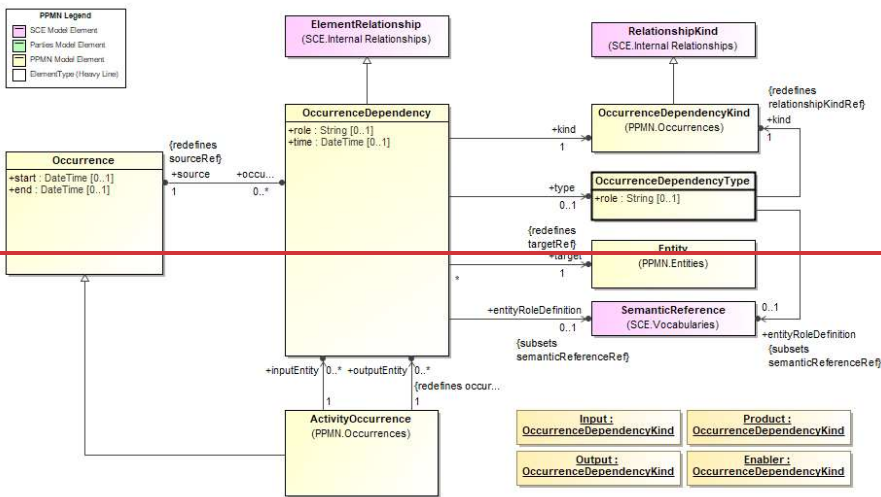
1124
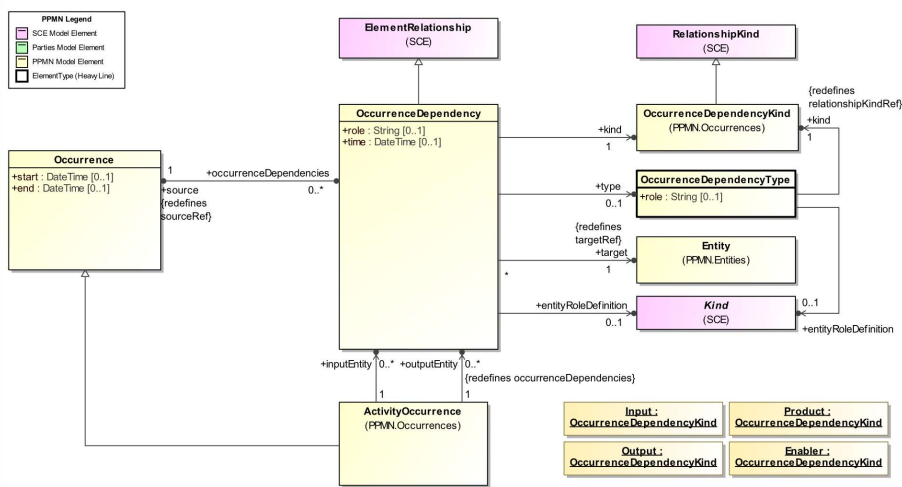
Figure 13:   OccurrencesDependencies

Generalizations

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

- *ElementRelationship* (see the ~~section entitled "ElementRelationship"~~ SCE Specification for more information).

> **Commented [JB50]:** Text updated for PPMN-19/PPMN-83

Properties

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

Table 17.   OccurrenceDependency Attributes and/or Associations

Property/Association	Description
entityRoleDefinition : ~~SemanticReference~~Kind [0..1]	A ~~SemanticReference~~ *Kind* ~~to~~ that provides a definition of the way the *Entity* was used in the *Occurrence*.
kind : OccurrenceDependencyKind [1] default: Output	A description of the type of dependency an *OccurrenceType* has on an *EntityType*. See *RelationshipKind*, below, for more details.
relationshipKind : RelationshipKind [1] default: Dependency	A description of the type of the relationship. See *RelationshipKind*, 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 *Occurrence*.
source : Occurrence [1]	The *Occurrence* that has some dependency on the target ~~*NamedElement*~~*Occurrence*.
target : Entity [1]	The ~~*NamedElement*~~ *Entity* on which some *Occurrence* depends.
time : DateTime [0..1]	The time that the *Occurrence* had the dependency on the *Entity*.

> **Commented [JB51]:** Text updated for PPMN-19/PPMN-83
>
> **Commented [JB52]:** Text updated for PPMN-19/PPMN-83
>
> **Commented [JB53]:** Text updated for PPMN-19/PPMN-83

type : OccurrenceDependencyType [0..1]	The type of the *EntityDependency*.

1133

8.2.9 OccurrenceDependencyKind

1135 A class indicating the kind of dependency an *Occurrence* has on an *Entity*.

Generalizations

1137 The *OccurrenceDependencyKind* element inherits the attributes and/or associations of:

1138 • *RelationshipKind* (see the ~~section entitled "RelationshipKind"~~SCE Specification for more information).

Commented [JB54]: Text updated for PPMN-19/PPMN-83

Properties

1140 The *OccurrenceDependencyKind* element does not have any additional attributes and/or associations.

8.2.10 OccurrenceDependencyType

1142 A kind of *ElementRelationship* that captures a dependency of a type of *Occurrence* on a particular type of entity and
1143 the role the entity plays in that type of *Occurrence*.

1144 *OccurrenceRoleTypes* indicate how *Parties are expected to* participate in an *Occurrence*.

1145

1146

Figure 14:   Occurrence Dependency Types

Generalizations

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

- *ElementRelationshipType* (see the ~~section entitled "ElementRelationshipType"~~ **SCE** Specification for more information).

Properties

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

Table 18.   OccurrenceDependencyType Attributes and/or Associations

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

8.2.11   OccurrenceGraphNode

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

Generalizations

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

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

Commented [JB55]: Diagram updated for PPMN-19/PPMN-83.

Commented [JB56]: Text updated for PPMN-19/PPMN-83

Commented [JB57]: Text updated for PPMN-19/PPMN-83

Commented [JB58]: Text updated for PPMN-19/PPMN-83

Properties

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

8.2.12   OccurrenceGraphTransition

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

Generalizations

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

- *ElementRelationship* (see the ~~section entitled "ElementRelationship"~~ **SCE** Specification for more information).

Properties

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 *RelationshipKind* in the **SCE** specification~~, below,~~ for more details. This property is read only and set to Transition.
source : OccurrenceGraphNode [1]	The *OccurrenceGraphNode* from which the transition leaves.
target : OccurrenceGraphNode [1..*]	The *OccurrenceGraphNode* to which the transition leads.
transitionRule : Rule [0..*]	The *Rules* that constrain the *OccurrenceTransitionType*.

8.2.13   OccurrenceKind

A class indicating the specific kind of *Occurrence*.

Generalizations

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

- ~~*SemanticReference*~~ *Kind* (see the ~~section entitled "SemanticReference"~~**SCE** specification for more information).

Properties

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

8.2.14   OccurrenceRelationship

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

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

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

Commented [JB59]: Text updated for PPMN-19/PPMN-83

Commented [JB60]: Text updated for PPMN-19/PPMN-83

Commented [JB61]: Text updated for PPMN-19/PPMN-83

Commented [JB62]: Updated to address PPMN-73/PPMN-110

Generalizations

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

- *ElementRelationship* (see the ~~section entitled "ElementRelationship"~~**SCE** specification for more information).

Properties

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 *Occurrence*.
target : Occurrence [1]	The *Occurrence* on which the source *Occurrence* depends.
relationshipKind : RelationshipKind [1]	A description of the kind of the relationship between the two *Occurrences*. See *RelationshipKind* in the **SCE** specification for more details. This property is read only and set to Transition.

8.2.15   OccurrenceRole

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

OccurrenceRoles indicate how a *Party* participated in an *Occurrence*.
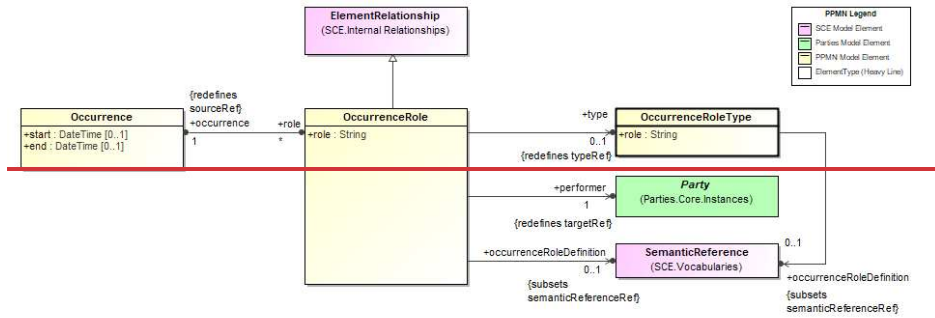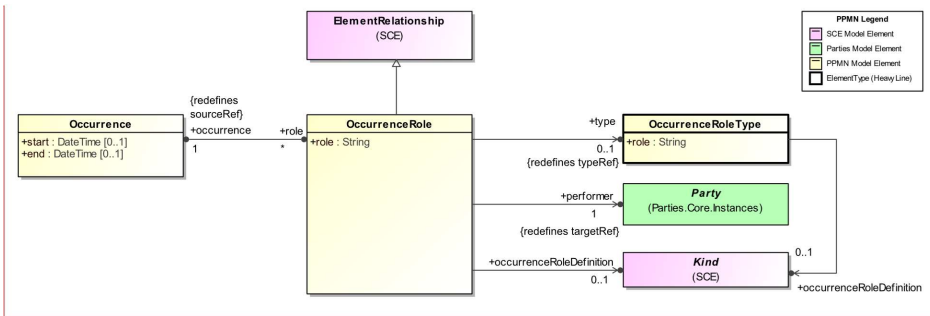
Figure 15:   OccurrencesRoles

Generalizations

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

- *ElementRelationship* (see the ~~section entitled "ElementRelationship"~~**SCE specification** for more information).

Properties

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 *Occurrence* in which the *Party* plays the role.
occurrenceRoleDefinition : ~~SemanticReference~~ Kind [0..1]	A ~~SemanticReference to a~~Kind that provides a definition of the role the *Party* played in the *Occurrence*.
performer : Party [1]	The *Party* that plays the role in an *Occurrence* specified by the *OccurrenceRole*.
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` *Party* in the *Occurrence*.

8.2.16   OccurrenceRoleType

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

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

Commented [JB65]: Diagram updated for PPMN-19/PPMN-83.

Commented [JB66]: Text updated for PPMN-19/PPMN-83

Commented [JB67]: Text updated for PPMN-19/PPMN-83

Commented [JB68]: grammar
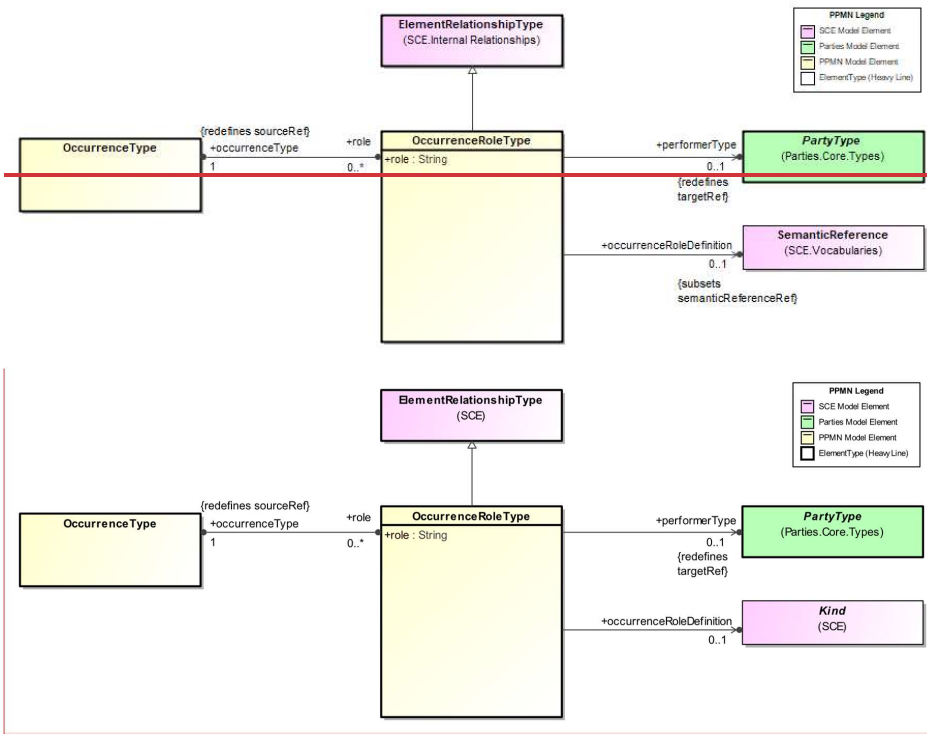
Figure 16: Occurrence Role Types

Generalizations

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

- *ElementRelationshipType* (see the ~~section entitled "ElementRelationshipType"~~SCE specification for more information).

Properties

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

Table 22.   **OccurrenceRoleType Attributes and/or Associations**

Property/Association	Description
occurrenceRoleDefinition : ~~SemanticReference~~Kind [0..1]	A ~~*SemanticReference*~~*Kind* that provides ~~to~~ a definition of the role the *PartyType* is expected to play in the *OccurrenceType*.
occurrenceType : OccurrenceType [1]	The type of *Occurrence* in which the `expectedPerformer` to perform in the role.
performerType : PartyType [0..1]	The *Party* that is expected to perform in a particular role in an *Occurrence*.

role : String []	A textual description of the role in the *Occurrence*.

1228

8.2.17 OccurrenceType

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

Generalizations

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

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

Properties

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

1237

8.2.18 OccurrenceTypeGraph

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

Generalizations

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

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

Properties

1244 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 *OccurrenceGraphNodes* included in the *OccurrenceTypeGraph*.
transition : OccurrenceGraphTransition [0..*]	The *OccurrenceTypeTransitions* included in the *OccurrenceTypeGraph*.

1245

8.2.19   OccurrenceTypeUsage Node

1247 A kind of *OccurrenceGraphNode* that identifies the usage of an *OccurrenceType* in an *OccurrenceTypeGraph*.

Generalizations

1249 The *OccurrenceTypeUsage Node* element inherits the attributes and/or associations of:

1250 • *OccurrenceGraphNode* (see the section entitled "OccurrenceGraphNode" for more information).

Properties

1252 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 *OccurrenceType* that the node represents.

1253

8.2.20   PPMNRelationshipKind

1255 A class indicating the kind of relationship between two **PPMN** elements.

Generalizations

1257 The *PPMNRelationshipKind* element inherits the attributes and/or associations of:

1258 • *RelationshipKind* (see the ~~section entitled "RelationshipKind"~~SCE specification for more information).

> **Commented [JB74]:** Text updated for PPMN-19/PPMN-83

Properties

1260 The *PPMNRelationshipKind* element does not have any additional attributes and/or associations.

8.2.21   Rule

1262 A condition that can be evaluated in some context as being either True or False.

Generalizations

1264 The *Rule* element does not inherit any attributes or associations of from another element.

Properties

1266 The *Rule* element does not have any additional attributes and/or associations.

8.3   Pedigree

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

8.3.1   Pedigree Occurrences

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

PedigreeChains record the actual events or processes that happen as part of the history of an entity of interest. *PedigreeChains* also record a reference to the entity to which the *Occurrences* relate through the `entity` property. Conceptually, *PedigreeChains* are "instances" of *PedigreeChainTypes* and as such may be governed by the relations established in the *PedigreeChainType*. These occurrences represent actual events or activities in the history of one or more *Entities* that are of interest to some *Party*.
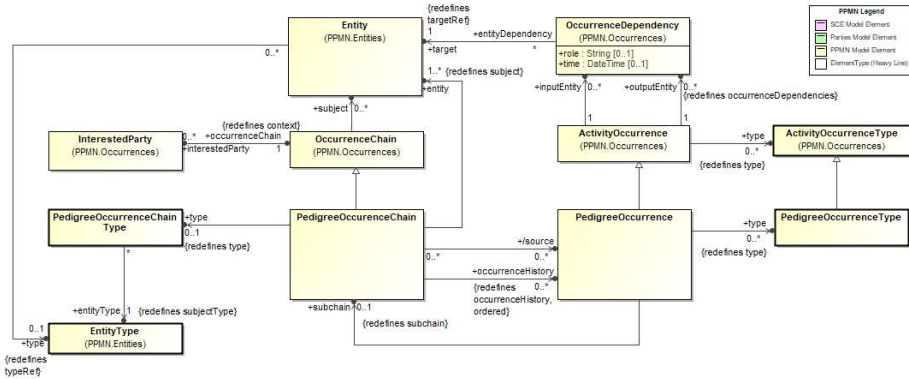
Figure 17:   Pedigree Occurrence Chains - Overview

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

> **Commented [JB75]:** Sentence fragment

Commented [JB76]: Diagram updated for PPMN-19/PPMN-83.
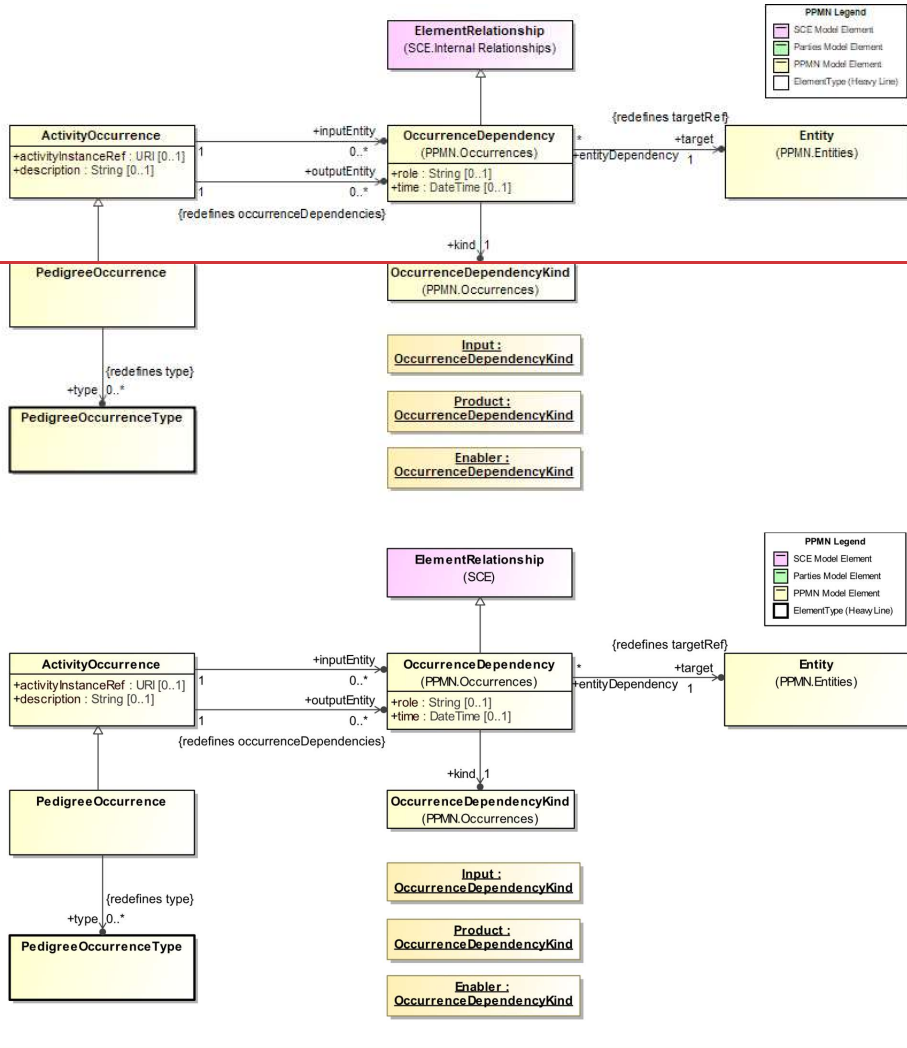
1286

1287

1288

Figure 18:   Pedigree Occurrences

1289 *PedigreeOccurrenceChains* record the actual *PedigreeOccurrences* that happen as part of the
1290 `occurrenceHistory` property, an ordered list. *PedigreeOccurrenceChains* include a reference to the *Entity* or
1291 *Entities* to which the *Occurrences* relate through the `entity` property. *PedigreeOccurrenceChains* are essentially
1292 instances of *PedigreeChainTypes* and as such are governed by the relations established in the *PedigreeChainType*.

1293 *PedigreeOccurrences* are instances of *PedigreeOccurrenceTypes*. These occurrences represent actual events or
1294 activities in the history of an *Entity* that is of interest to some *Party*.

1295

1296

1297

1298 **Figure 19:   Pedigree Occurrence Chains**

1299   *PedigreeChainType* supports the definition of types of occurrences expected in *PedigreeChains* related to an
1300   *EntityType* in which some *Party* is interested. *PedigreeChainTypes* are modeled as simple graphs so that rich
1301   definitions of entity lifecycles can be created (though they are not required). The model also supports simple
1302   definitions of valid *PedigreeOccurrenceTypes* or no lifecycle definitions at all.

1303   *InterestedParty* is a kind of *PartyRole* that indicates that a *Party* has some interest in the with respect to an entity.
1304   *PedigreeChainTypes* are specific to one or more *InterestedParties*. As an example, an automobile manufacturer
1305   may be interested a set of occurrences related to the building of a car such as StartAssembly, InstallEngine,
1306   PaintCar, TestCar, and ShipCar. A dealership on the other hand would likely be interested in tracking other events
1307   such as BuildCar, ShipCar, ReceiveCar, and SellCar.

> **Commented [JB77]:** grammar

1308

1309
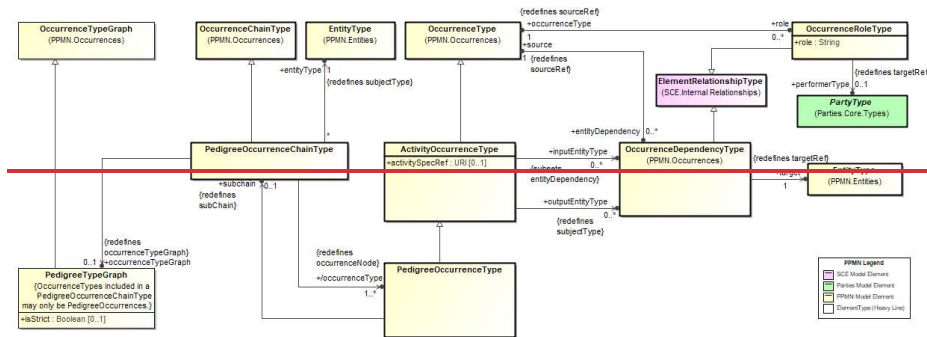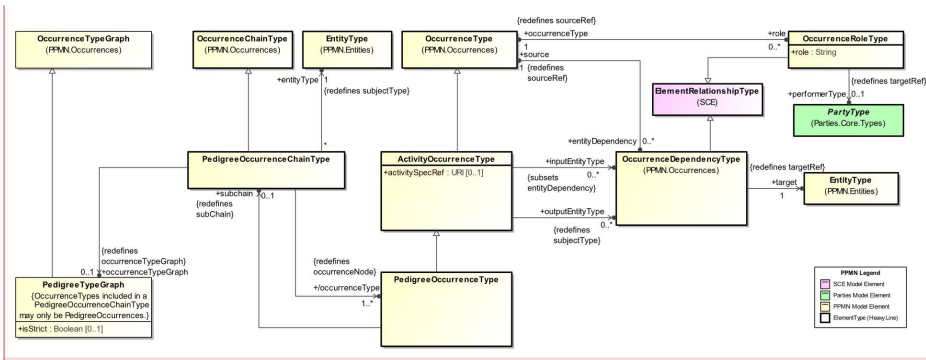
1310

Figure 20: Pedigree Occurrence Chain Type

1311

1312 *PedigreeChainType* supports the definition of types of occurrences expected in *PedigreeChains* related to an entity
1313 type in which some *Party* is interested. *PedigreeChainTypes* are modeled as simple graphs so that rich definitions
1314 of entity lifecycles can be created (though they are not required). The model also supports simple definitions of
1315 valid *PedigreeOccurrenceTypes* or no lifecycle definitions at all.
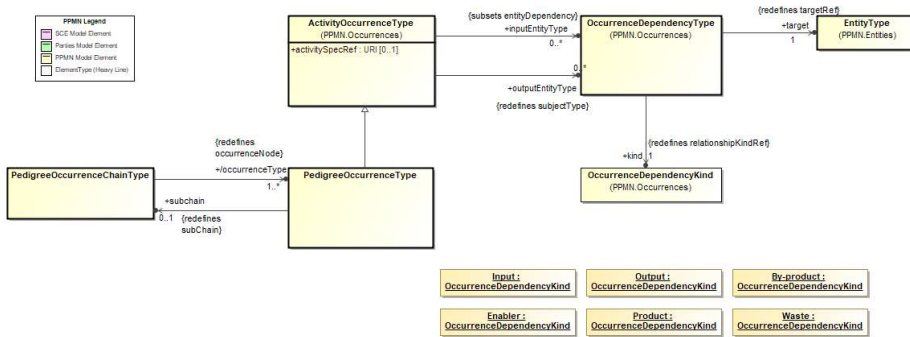
1316

Figure 21: Pedigree Occurrence Types

1317

1318 The lineage of an *Entity*, herein referred to as its "pedigree" or "pedigree chain", is a lattice comprising *Entities* as
1319 nodes and derivations (*DerivedFrom* relationships) as edges. Pedigree chains are created by *Occurrences* that result
1320 in some number of *Entities* being used to create one or more new *Entities* or evolve one or more existing *Entities*.
1321 These *Occurrences* result one or more derivations between "input" *Entities* and the "output" *Entities*.

1322 Given that a particular *Occurrence* may encapsulate a sub-chain of *Occurrences*, derivations may involve a series of
1323 one or more *Occurrences* that create or evolve an entity of interest into another. In these cases, the *Occurrences* that
1324 comprise the sub-chain would also potentially result in derivations that would combine to result in the derivations of
1325 the containing *Occurrence*. As stated above, derivations are noted in the form of a *DerivedFrom* relationship
1326 between one *Entity* that is the ~~derivee~~ derivationSource and another that is the derivedEntity. The
1327 derivation may be related to an *ActivityOccurrence* that caused the transformation. This may specifically be a
1328 *PedigreeOccurrence* but may also be a more general *ActivityOccurrence*. Often, the activities that result in
1329 derivations are not easily tracked or quantified and so just noting the *Entity* or *Entities* from which the entity of
1330 interest is derived is all that is necessary or in some cases even possible.

1331
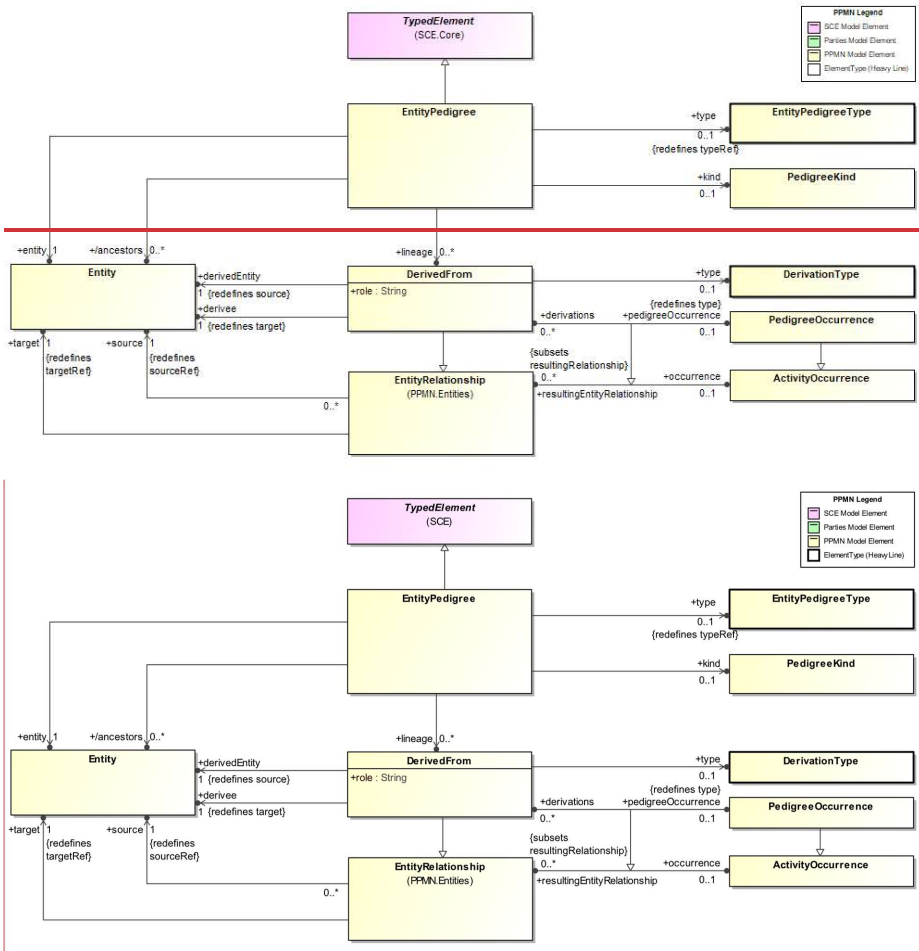
Figure 22:   Pedigree "Chains"

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

1342
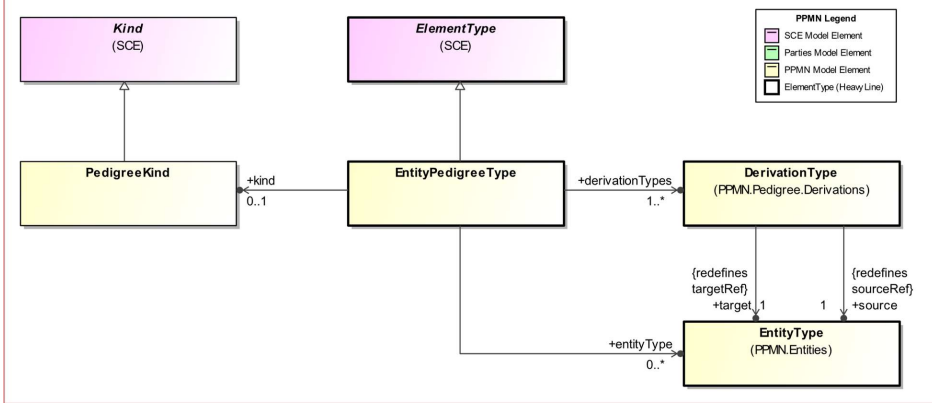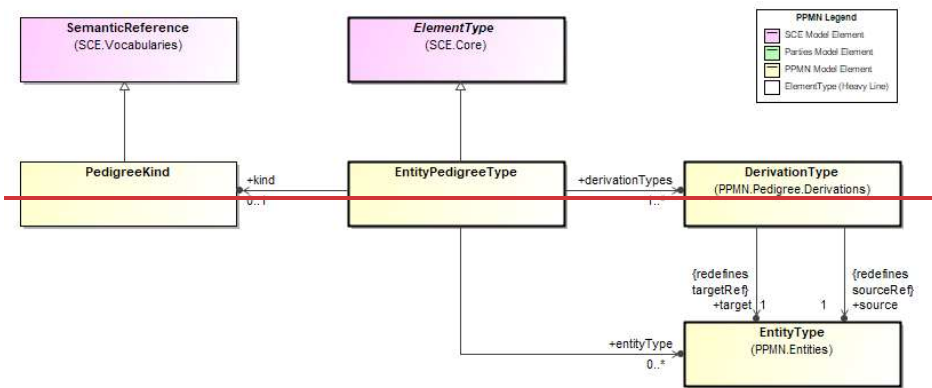
1343

Figure 23:   Pedigree Chains Types

8.3.1.1   EntityPedigree

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

Generalizations

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

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

Properties

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

1352

8.3.1.2   EntityPedigreeType

1354   The type of pedigree or lineage between *Entities* of type entityType.

Generalizations

1356   The *EntityPedigreeType* element inherits the attributes and/or associations of:

1357   •   **SCE** *ElementType* (see the ~~section~~ **SCE** specification for more information).

> **Commented [JB84]:** Text updated for PPMN-19/PPMN-83

Properties

1359   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 *EntityType(s)* to which the EntityPedigreeType applies.
kind : PedigreeKind [0..1]	The kind of entity pedigree or lineage the EntityPedigreeType represents.

1360

8.3.1.3   PedigreeKind

1362   A class that indicates the kind of pedigree or lineage between *Entities*.

Generalizations

1364   The *PedigreeKind* element inherits the attributes and/or associations of:

1365 • *SemanticReferenceKind* (see the ~~section entitled "SemanticReference"~~SCE specification for more
1366 information).

Properties
1367

1368 The *PedigreeKind* element does not have any additional attributes and/or associations.

8.3.1.4 PedigreeOccurrenceChain
1369

1370 A succession of *PedigreeOccurrences* that have happened in the life of an entity that is of interest to some *Party*.

Generalizations
1371

1372 The *PedigreeOccurenceChain* element inherits the attributes and/or associations of:

1373 • *OccurrenceChain* (see the section entitled "OccurrenceChain" for more information).

Properties
1374

1375 The following table presents the additional attributes and/or associations for *PedigreeOccurenceChain*:

Table 28. PedigreeOccurenceChain Attributes and/or Associations

Property/Association	Description
entity : Entity [1]	The *Entity* or *Entities* for which the *PedigreeChain* represents the history of *PedigreeOccurrences*.
occurrenceHistory : PedigreeOccurrence [0..*]	A sequence of *PedigreeOccurrences* that represent the history of *PedigreeOccurrences* that took place with respect to a particular entity.
source : PedigreeOccurrence [0..*]	The *PedigreeOccurrences* that were the original sources for ancestor entities of the subject entity.
type : PedigreeOccurrenceChainType [0..1]	The type of the *PedigreeChain*.

1376

8.3.1.5 PedigreeOccurrence
1377

1378 An *ActivityOccurrence* in the lifecycle of an entity related to the source or evolution of that entity that is of interest
1379 to some *Party*.

Generalizations
1380

1381 The *PedigreeOccurrence* element inherits the attributes and/or associations of:

1382 • *ActivityOccurrence* (see the section entitled "ActivityOccurrence" for more information).

Properties
1383

1384 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 PedigreeOccurrence.

subchain : PedigreeOccurenceChain [0..1]	A sequence of *PedigreeOccurrences* that take the ~~inputEntities~~ inputEntity of the *PedigreeOccurrence* and transform them into the ~~outputEntities~~ outputEntity of the *PedigreeOccurrence* and are encapsulated by the *PedigreeOccurrence*.
type : PedigreeOccurrenceType [0..*]	The type of the *PedigreeOccurrence*.

<div style="float:right; border:1px solid red; padding:2px;">**Commented [JB85]:** Updated to address PPMN-30/PPMN-116.</div>

1385

8.3.1.6 PedigreeOccurrenceChainType

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

Generalizations

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

- *OccurrenceChainType* (see the section entitled "OccurrenceChainType" for more information).

Properties

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 `occurrenceType` derived property is based on the series of relationships between from *PedigreeChainType* through other classes to *PedigreeOccurrenceType*: `OccurrenceChainType.occurrenceTypeGraph.occurrenceNode.occurrenceType`.
occurrenceTypeGraph : PedigreeTypeGraph [0..1]	A graph of *PedigreeOccurrenceTypes* that are expected in the lifecycle of a particular type of entity.

1394

8.3.1.7 PedigreeOccurrenceType

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

Generalizations

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

- *ActivityOccurrenceType* (see the section entitled "ActivityOccurrenceType" for more information).

Properties

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 *PedigreeChainType* that is encapsulated within the *PedigreeOccurrenceType* to create a "subchain".

1402

8.3.1.8   PedigreeTypeGraph

1403

1404 A *PedigreeChainType* is a specification for the types of *Occurrences* that happen with respect to an entity that are of
1405 interest to a particular *Party*. If the property isStrict=True, then only the *Occurrences* of type
1406 *PedigreeOccurrenceType* will be included in related *PedigreeChains*. If the property is False then *Occurrences* of
1407 other types may be included in related *PedigreeChains*.

Generalizations

1408

1409 The *PedigreeTypeGraph* element inherits the attributes and/or associations of:

1410 • *OccurrenceTypeGraph* (see the section entitled "OccurrenceTypeGraph" for more information).

Properties

1411

1412 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 *PedigreeTypeGraph* is strict or not. If the value is True, then only the *Occurrences* of type *PedigreeOccurrenceType* will be included in related *PedigreeChains*. If the value is False then *Occurrences* of other types may be included in related *PedigreeChains*.

1413

8.3.2   Derivations

1414

1415 The Derivations package contains elements that capture the derivation relationships between Entities. These
1416 elements, in conjuction with Entities, capture the lineage or pedigree of Entities.

1417 Derivations capture the lineal relationships between Entities or Entity Snapshots. Derivations are noted in the form
1418 of a *DerivedFrom* relationship, or one of its specializations, between one *Entity* that is the
1419 derivationSourceee derivee and another that is the derivedEntity. A derivation may be the result of
1420 a general *ActivityOccurrence* or specifically a *PedigreeOccurrence*. Please note that the activities that result in
1421 derivations are not always easily tracked or quantified and so just noting the entity from which the entity of interest
1422 is derived is all that is possible.

1423 **PPMN** specifies four types of derivation: revision, quotation, sourcing, and descendant. Revision is captured in the
1424 form of the *RevisionOf* relationshipby setting the kind attribute of the *DerivationType* to *RevisionOf*.. *RevisionOf*
1425 is an instance ofa specialization of *DerivedFrom DerivedKind* and is used in situations where one entity is a revision
1426 of another as in a report or publication. Quotation is captured by setting the kind attribute of the *DerivationType* to
1427 *QuotedFrom* the *QuotedFrom* specialization of *DerivedFrom* and specifies that part of all of one entity is a repeat of
1428 part or all of another entity, presumably some textual report or publication. The quotation may or may not be by the
1429 original author of the quoted entity. *SourcedFrom* is a specialization of *DerivedFrom*captured by setting the kind
1430 attribute of the *DerivationType* to *SourcedFrom* that and identifies specifies that the entity of interest came from
1431 another entity which was in turn produced by some party potentially with some special experience or knowledge.

Commented [JB86]: Update with issue numbers 50/114

Commented [JB87]: Updated to address typographical error found by AB review.

1432 Finally, *DescendantOf* ~~specializes *DerivedFrom*~~is captured by setting the `kind` attribute of the *DerivationType* to
1433 *DescendantOf* and indicates that the entity of interest is a descendant of the ancestor *Entity*.

1434

1435
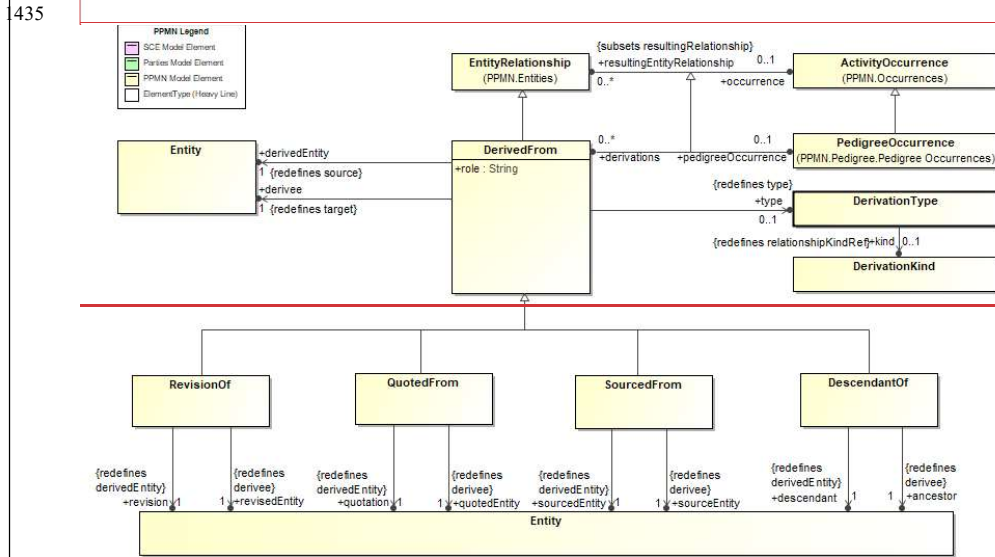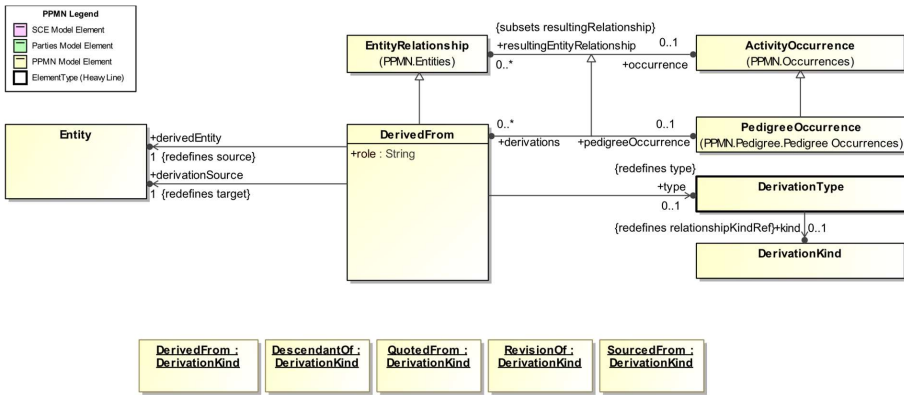
1436
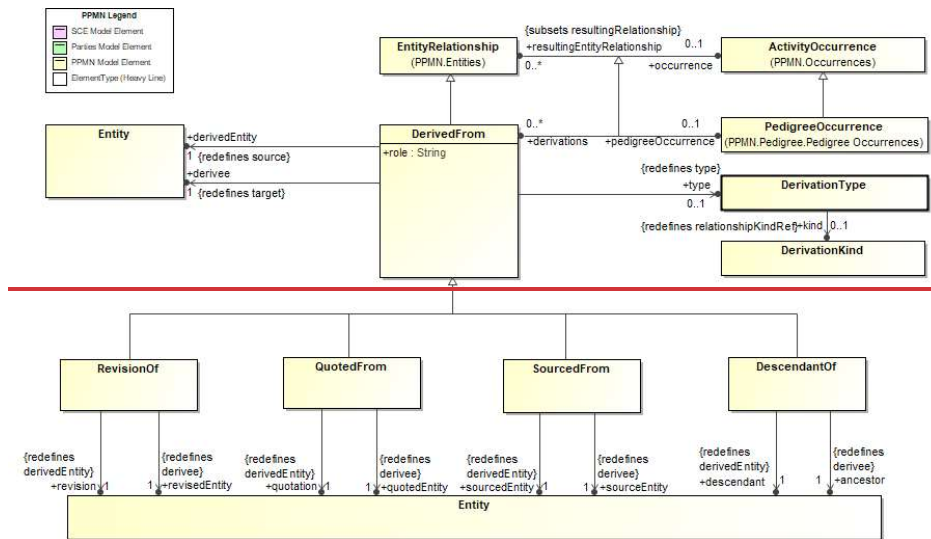
Figure 24:   Derivations

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

ElementRelationshipType
(SCE.Internal Relationships)

SemanticReference
(SCE.Vocabularies)

DerivationType
(PPMN.Pedigree.Derivations)

+kind
0..1

DerivationKind
(PPMN.Pedigree.Derivations)

{redefines
relationshipKindRef}

{redefines
targetRef}
+target 1

{redefines
sourceRef}
1 +source

DerivedFrom : DerivationKind

EntityType
(PPMN.Entities)
attributes
+typeDefinitionRef : URI [0..1]

QuotedFrom : DerivationKind

RevisionOf : DerivationKind

SourcedFrom : DerivationKind

PPMN Legend
SCE Model Element
Parties Model Element
PPMN Model Element
ElementType (Heavy Line)

442

1444   **Figure 25:   Derivation Types**

1445   ### 8.3.2.1   DerivationKind

1446   A class indicating the kind of derivation that exists between two *Entities*.

1447   **Generalizations**

1448   The *DerivationKind* element inherits the attributes and/or associations of:

1449   • *SemanticReferenceKind* (see the ~~section entitled "SemanticReference"~~**SCE** specification for more
1450   information).

> **Commented [JB91]:** Text updated for PPMN-19/PPMN-83

1451   **Properties**

1452   The *DerivationKind* element does not have any additional attributes and/or associations.

1453   ### 8.3.2.2   DerivationType

1454   A kind of *ElementRelationship* that captures the type of derivation between one particular *EntityType* and another.

1455   **Generalizations**

1456   The *DerivationType* element inherits the attributes and/or associations of:

1457   • *ElementRelationshipType* (see the ~~section entitled "ElementRelationshipType"~~**SCE** specification for more
1458   information).

> **Commented [JB92]:** Text updated for PPMN-19/PPMN-83

1459   **Properties**

1460   The following table presents the additional attributes and/or associations for *DerivationType*:

> **Commented [JB90]:** Diagram updated for PPMN-19/PPMN-83.

Table 33.   DerivationType Attributes and/or Associations

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

1461

8.3.2.3   DerivedFrom

1463 Derivations are noted in the form of a *DerivedFrom* relationship between one *Entity* that is the ~~derivee~~
1464 derivationSource and another that is the derivedEntity. The derivation may be related to an
1465 *ActivityOccurrence* that specifies the particular *Occurrence* that caused the transformation. Often, the activities that
1466 result in derivations are not easily tracked or quantified and so just noting the entity from which the entity of interest
1467 is derived is all that is necessary.

Generalizations

1469 The *DerivedFrom* element inherits the attributes and/or associations of:

1470 • *EntityRelationship* (see the section entitled "EntityRelationship" for more information).

Properties

1472 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 *Entity* that was derived.
~~derivee~~ derivationSource : Entity [1]	The *Entity* from which the derivedEntity was derived.
pedigreeOccurrence : PedigreeOccurrence [0..1]	The *PedigreeOccurrence* that resulted in the derivation.
role : String []	A string that captures the role in the derivationOccurrence that produced the element.
type : DerivationType [0..1]	The type of derivation.

1473

8.3.2.4   DescendantOf

1475 *DescendantOf* is a specialization of *DerivedFrom* that identifies that the entity of interest is a descendant of another
1476 *Entity*.

Generalizations

1478 The *DescendantOf* element inherits the attributes and/or associations of:

Commented [JB93]: Updated to address PPMN-50/PPMN-114

Commented [JB94]: Updated to address PPMN-50/PPMN-114

1479   • *DerivedFrom* (see the section entitled "DerivedFrom" for more information).

Properties

1481   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 *Entity*.
descendant : Entity [1]	The descendant *Entity*.

1482

8.3.2.5   QuotedFrom

1484   Quotation is captured by the *QuotedFrom* specialization of *DerivedFrom* and specifies that part of all of one entity
1485   is a repeat of part or all of another entity, presumably some textual report or publication. The quotation may or may
1486   not be by the original author of the quoted entity.

Generalizations

1488   The *QuotedFrom* element inherits the attributes and/or associations of:

1489   • *DerivedFrom* (see the section entitled "DerivedFrom" for more information).

Properties

1491   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.

1492

8.3.2.6   RevisionOf

1494   Revision is captured in the form of the *RevisionOf* relationship. *RevisionOf* is a specialization of *DerivedFrom* and
1495   is used in situations where one entity is a revision of another as in a report or publication.

Generalizations

1497   The *RevisionOf* element inherits the attributes and/or associations of:

1498   • *DerivedFrom* (see the section entitled "DerivedFrom" for more information).

Properties

1500   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.

1501

8.3.2.7   SourcedFrom

1502

1503 *SourcedFrom* is a specialization of *DerivedFrom* that identifies that the entity of interest came from another entity
1504 which was in turn produced by some party potentially with some special experience or knowledge.

Generalizations

1505

1506 The *SourcedFrom* element inherits the attributes and/or associations of:

1507 • *DerivedFrom* (see the section entitled "DerivedFrom" for more information).

Properties

1508

1509 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 sourcedEntity was sourced.

1510

8.4   Provenance

1511

1512 The Provenance package contains elements related to the notion of the ownership and custody of entities of interest.
1513 This includes the *Occurrences* that result in changes in the ownership or custody of those entities of interest.

1514 *ProvenanceOccurrences* are specializations of *Occurrence* related to changes in ownershp or custody of an entity.
1515 *ProvenanceOccurrences are* instances of *ProvenanceOccurrenceType* or one of its specializations. Similar to
1516 *OccurrenceType*, *ProvenanceOccurrenceType* is a specification of "expected" *ProvenanceOccurrences.* They
1517 capture the *Parties* expected to be involved in the instances. Expected types of entities to which the occurrences
1518 refer are noted through the entityType property.

1519 A *ProvenanceChain* records the provenance-related events that happen as part of the lifecycle of an entity. These
1520 events are recorded as part of the occurrenceHistory property, an ordered list of *ProvenanceOccurrences*. A
1521 *ProvenanceChain* also records a reference to the entity to which the *Occurrences* relate through the entity
1522 property. *ProvenanceChains* are essentially instances of Provenance*ChainTypes* and as such are governed by the
1523 relations established in the *ProvenanceChainType*. If the *ProvenanceChainType* isStrict property is set to
1524 "True" then the types of occurrences maintained in the *ProvenanceChain* are constrained to those included in the
1525 *ProvenanceChainType*.

1526

1527

1528
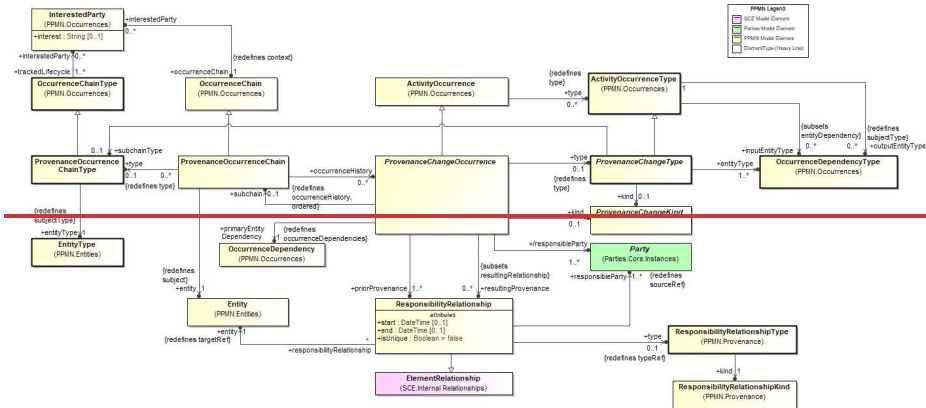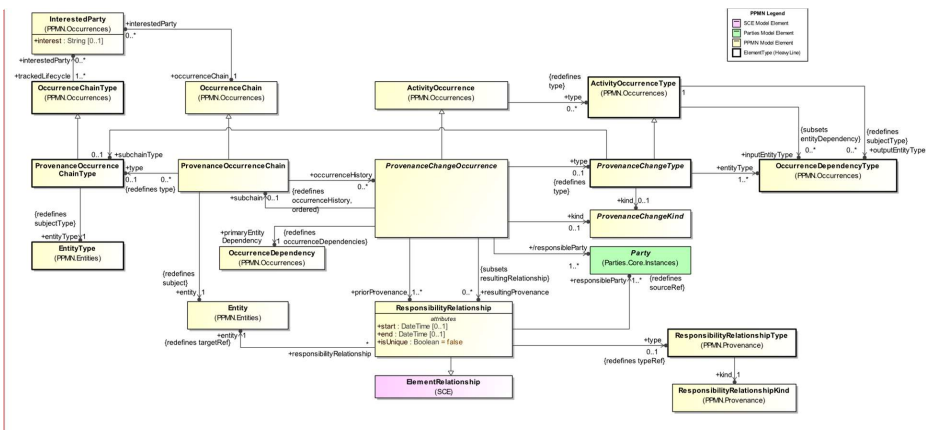
Figure 26:   Provenance Occurrence Chains

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

1536

1537
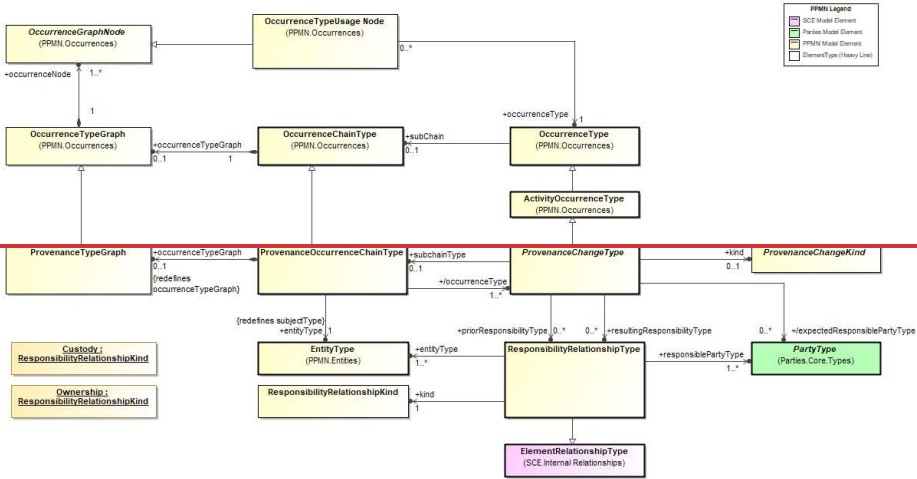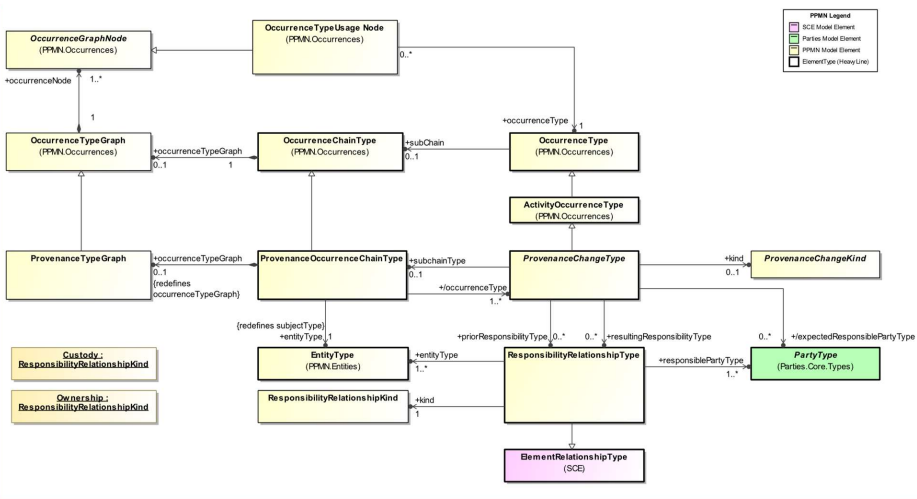
Figure 27:   Provenance Occurrence Chain Types

1538

1539   In addition to tracking changes in ownership or custody for an entity of interest over time, stakeholders also require
1540   the ability to make direct statements about who owns or has custody of an entity at a particular point in time. The
1541   *Ownership* and *Custody* classes provide this capability. Both *Ownership* and *Custody* specializations of
1542   *ResponsibilityRelationship* and, as such, capture the *Party* that owns or has custody of, respectively, a particular
1543   *Entity* for a particular period of time. These provenance "records" can either be maintained in real time or generated
1544   based on *Occurrences* that have been tracked for an entity.

1545
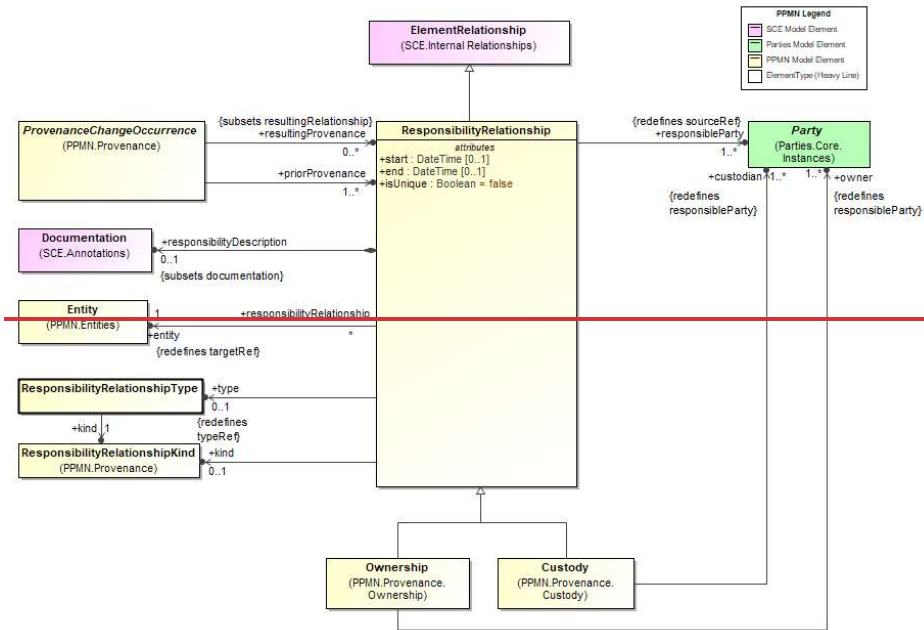
1546

1547

1548 **Figure 28:   Provenance "Records"**

1549 TBD.

1550
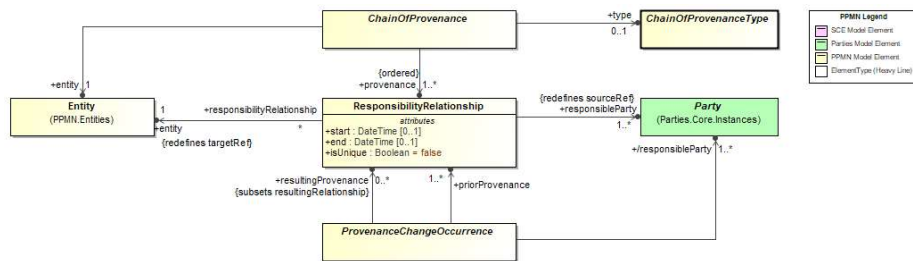
1551

1552 **Figure 29:   Chain of Provenance**

1553 TBD.

1554

ElementRelationshipType
(SCE.Internal Relationships)

PPMN Legend
SCE Model Element
Parties Model Element
PPMN Model Element
ElementType (Heavy Line)

Documentation
(SCE.Annotations)

ResponsibilityRelationshipType

+kind

ResponsibilityRelationshipKind
(PPMN.Provenance)

Custody :
ResponsibilityRelationshipKind

+description
0..1
{subsets
documentation}

1

Ownership :
ResponsibilityRelationshipKind

EntityType
(PPMN.Entities)

+entityType
1..*

+responsiblePartyType
1..*

PartyType
(Parties.Core.Types)

1555

ElementRelationshipType
(SCE)

PPMN Legend
SCE Model Element
Parties Model Element
PPMN Model Element
ElementType (Heavy Line)

Documentation
(SCE)

ResponsibilityRelationshipType

+kind

ResponsibilityRelationshipKind
(PPMN.Provenance)

Custody :
ResponsibilityRelationshipKind

+description
0..1
{subsets
documentation}

1

Ownership :
ResponsibilityRelationshipKind

EntityType
(PPMN.Entities)

+entityType
1..*

+responsiblePartyType
1..*

PartyType
(Parties.Core.Types)

1556

1557 **Figure 30:   Provenance Record Types**

1558 TBD.

1559

PPMN Legend
SCE Model Element
Parties Model Element
PPMN Model Element
ElementType (Heavy Line)

ChainOfProvenanceType
(PPMN.Provenance)

+entityType 1..*

+provenanceType 1..*

EntityType
(PPMN.Entities)

+entityType
1..*

ResponsibilityRelationshipType
(PPMN.Provenance)

+responsiblePartyType
1..*

PartyType
(Parties.Core.Types)

0..*

+/expectedResponsiblePartyType

ResponsibilityRelationshipKind
(PPMN.Provenance)

+kind
1

+priorResponsibilityType   0..*

0..*   +resultingResponsibilityType

ProvenanceChangeType
(PPMN.Provenance)

1560

1561 **Figure 31:   Chain of Provenance Types**

1562 ## 8.4.1   ChainOfProvenance

1563 An ordered set of *ResponsibilityRelationships* that captures the provenance of a particular entity over the course of
1564 its lifecycle.

1565 **Generalizations**

1566 The *ChainOfProvenance* element does not inherit any attributes or associations of from another element.

1567 **Properties**

1568 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 *ChainOfProvenance* refers.
provenance : ResponsibilityRelationship [1..*]	A set of ResponsibilityRelationships related to the provenance of an entity.
type : ChainOfProvenanceType [0..1]	The type of the *ChainOfProvenance*.

1569

1570 **8.4.2   ChainOfProvenanceType**

1571 An *ElementType* that specifies a set of expected provenance chains (*ChainOfProvenance*) that capture an ordered set
1572 of *ResponsibilityRelationships* of type *ResponsibilityRelationshipType*.

1573 **Generalizations**

1574 The *ChainOfProvenanceType* element does not inherit any attributes or associations of from another element.

1575 **Properties**

1576 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 *EntityType* for which the *ChainOfProvenanceType* applies.
provenanceType : ResponsibilityRelationshipType [1..*]	The type of the responsibility relationships expected to be included in provenance chains of type *ChainOfProvenanceType*.

1577

1578 **8.4.3   ProvenanceChangeKind**

1579 A class indicating the kind of provenance change that is expected.

1580 **Generalizations**

1581 The *ProvenanceChangeKind* element does not inherit any attributes or associations of from another element.

1582 **Properties**

1583 The *ProvenanceChangeKind* element does not have any additional attributes and/or associations.

1584 **8.4.4   ProvenanceChangeOccurrence**

1585 An *Occurrence* in the lifecycle of an entity related to the custody and/or ownership of that entity.

Generalizations

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

- *ActivityOccurrence* (see the section entitled "ActivityOccurrence" for more information).

Properties

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 *OccurrenceDependency* whose `target` is the *Entity* to which the *ProvenanceOccurrence applies.*
priorProvenance : ResponsibilityRelationship [1..*]	The *ResponsibilityRelationships* prior to the *ProvenanceChangeOccurrence.*
responsibleParty : Party [1..*]	The *Party* that has responsibility for the entity as a result of the *ProvenanceOccurrence.*
resultingProvenance : ResponsibilityRelationship [0..*]	The *ResponsibilityRelationships* that result from the *ProvenanceChangeOccurrence.*
subchain : ProvenanceOccurrenceChain [0..1]	A *ProvenanceChain* that is encapsulated by the *ProvenanceOccurrence,* essentially creating a "sub-chain".
type : ProvenanceChangeType [0..1]	The type of the *ProvenanceOccurrence.*

8.4.5   ProvenanceChangeType

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

Generalizations

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

- *ActivityOccurrenceType* (see the section entitled "ActivityOccurrenceType" for more information).

Properties

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 *ProvenanceChangeType.*

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

1599

8.4.6   ProvenanceOccurrenceChain

1601   A succession of *ProvenanceOccurrences* that have happened in the life of an entity that is of interest to some *Party*.

Generalizations

1603   The *ProvenanceOccurrenceChain* element inherits the attributes and/or associations of:

1604   • *OccurrenceChain* (see the section entitled "OccurrenceChain" for more information).

Properties

1606   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 *ProvenanceChain*.
occurrenceHistory : ProvenanceChangeOccurrence [0..*]	A set of *ProvenanceOccurrences* that comprise the chain.
type : ProvenanceOccurrenceChainType [0..1]	The type of the *ProvenanceChain*.

1607

8.4.7   ProvenanceOccurrenceChainType

1609   A kind of *OccurrenceChainType* that captures a specification for a series of potential *ProvenanceOccurrences* that
1610   are expected in a particular context. A *ProvenanceChainType* captures this specification through the
1611   `occurrenceTypeGraph` property - a graph of *OccurrenceGraphNodes* and *OccurrenceTransitionTypes*.

Generalizations

1613   The *ProvenanceOccurrenceChainType* element inherits the attributes and/or associations of:

1614   • *OccurrenceChainType* (see the section entitled "OccurrenceChainType" for more information).

Properties

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 *ProvenanceChainType*.
occurrenceType : ProvenanceChangeType [1..*]	A derived property that holds the set of *ProvenanceOccurrenceTypes* that represent the types of *ProvenanceOccurrences* expected to occur as part of *ProvenanceChains* that the *ProvenanceChainType* specifies.
occurrenceTypeGraph : ProvenanceTypeGraph [0..1]	A graph of *ProvenanceOccurrenceTypes* that specifies the sequencing of expected *ProvenanceOccurrences* in the lifecycle of an entity of interest to zero or more *InterestedParties*.

8.4.8   ProvenanceTypeGraph

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

Generalizations

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

- *OccurrenceTypeGraph* (see the section entitled "OccurrenceTypeGraph" for more information).

Properties

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

8.4.9   ResponsibilityRelationship

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

Generalizations

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

- *ElementRelationship* (see the ~~section entitled "ElementRelationship"~~ **SCE specification** for more information).

> **Commented [JB99]:** Text updated for PPMN-19/PPMN-83

Properties

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 *Party* relinquishes the specified responsibilities with respect to a particular entity.
entity : Entity [1]	The entity for which a *Party* 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 ResponsibilityRelationship between PartyTypes and EntityTypes in a given situation. See ResponsibilityRelationshipKind for more details.
responsibilityDescription : Documentation [0..1]	A textual description of the responsibility.
responsibleParty : Party [1..*]	The *Party* that is responsible from a provenance perspective for a particular entity.
start : DateTime [0..1]	The date on which a *Party* acquires the responsibilities with respect to a particular entity.
type : ResponsibilityRelationshipType [0..1]	The type of the *ResponsibilityRelationship*.

1635

8.4.10   ResponsibilityRelationshipKind

1637   A class representing the kind of *ResponsibilityRelationship* between *Parties* and *Entities* in some particular situation.

Generalizations

1639   The *ResponsibilityRelationshipKind* element inherits the attributes and/or associations of:

1640   • *SemanticReferenceKind* (see the ~~section entitled "SemanticReference~~**SCE** specification" for more
1641   information).

Properties

1643   The *ResponsibilityRelationshipKind* element does not have any additional attributes and/or associations.

8.4.11   ResponsibilityRelationshipType

1645   A kind of *ElementRelationshipType* that specifies an expected *ResponsibilityRelationship* between *PartyTypes* and
1646   *EntityTypes* in some particular situation.

Generalizations

1648   The *ResponsibilityRelationshipType* element inherits the attributes and/or associations of:

1649   • *ElementRelationshipType* (see the ~~section entitled "ElementRelationshipType"~~**SCE** specification for more
1650   information).

Properties

1652   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.

1653

8.4.12   Custody

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

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

1661
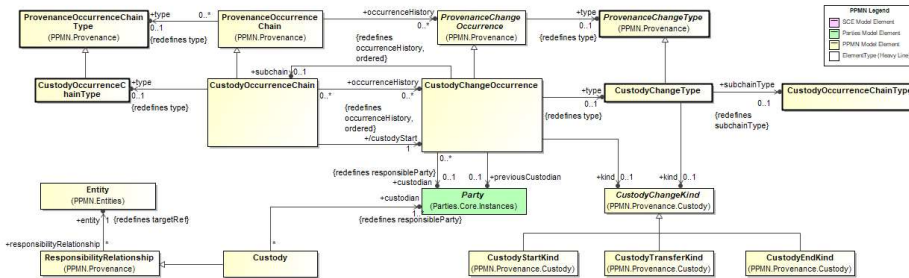
Figure 32:   Custody Occurrence Chains

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

1668
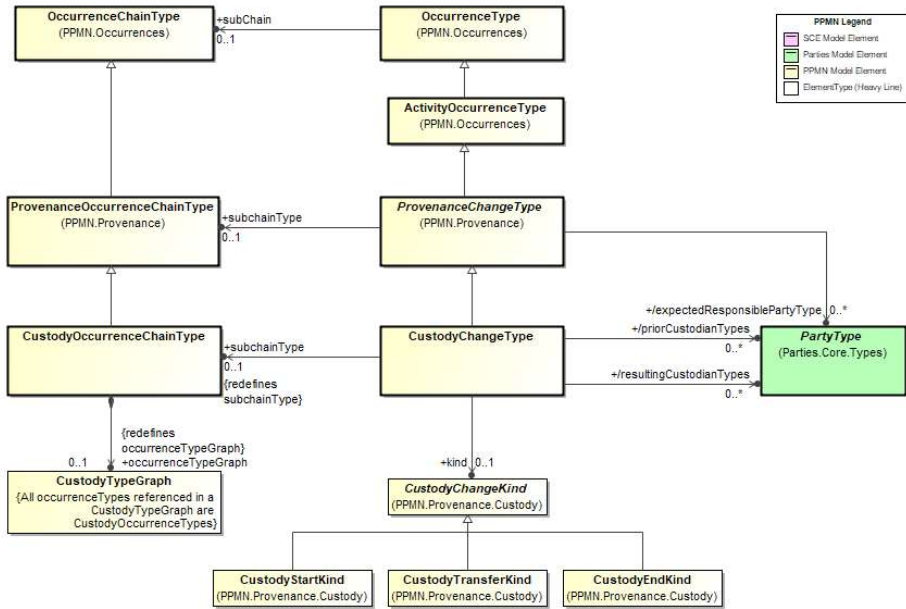
1669

Figure 33: Custody Occurrence Chain Types

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

1675
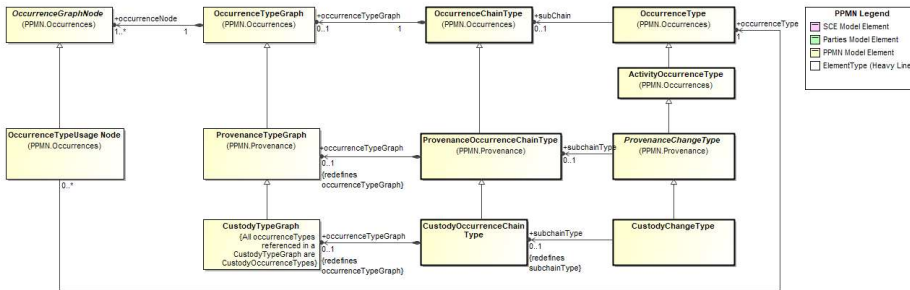
1676

Figure 34: Custody Occurrence Chain Type Pattern

TBD.

1679
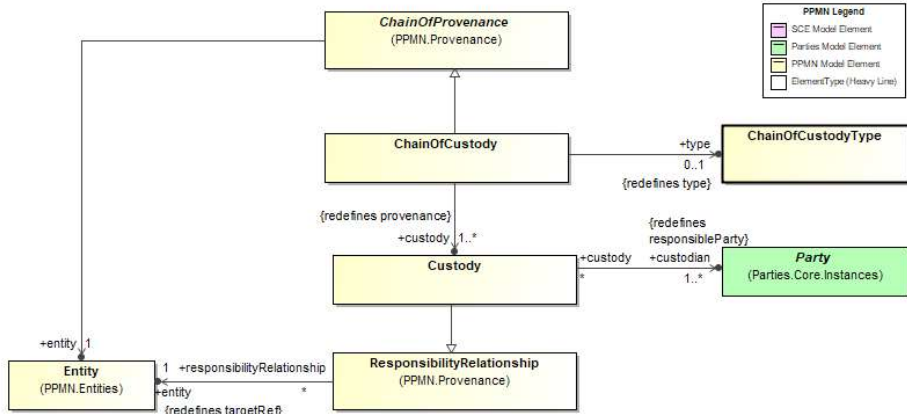
1680

Figure 35: Chain of Custody

1682 TBD.

1683
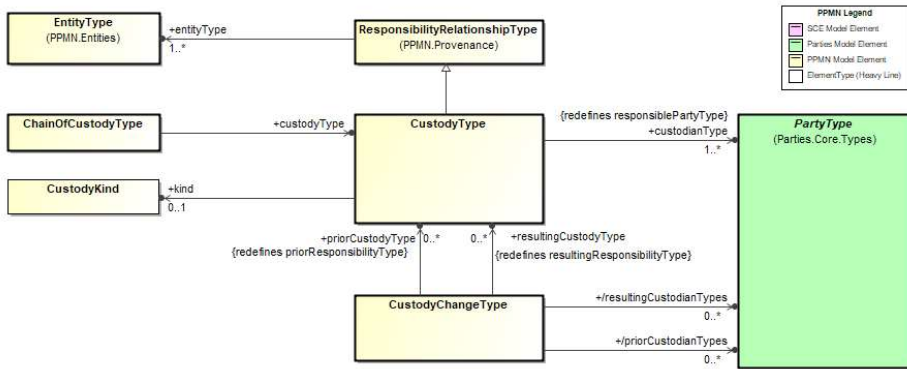
1684

Figure 36: Chain of Custody Types

8.4.12.1 ChainOfCustody

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

Generalizations

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

- *ChainOfProvenance* (see the section entitled "ChainOfProvenance" for more information).

Properties

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.

1694

8.4.12.2   ChainOfCustodyType

1696   A specialization of *ChainOfProvenanceType* that specifies instances of custody chains (*ChainOfCustody*) that
1697   capture an ordered set of *Custody* relationships of type *CustodyType.*

Generalizations

1699   The *ChainOfCustodyType* element does not inherit any attributes or associations of from another element.

Properties

1701   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 *CustodyType* of the *Custody* responsibility relationships contained in custody chains of type ChainOfCustodyType.

1702

8.4.12.3   Custody

1704   *Custody* is a kind of *ProvenanceRecord* that specifies a *Party* that has physical or electronic control of an entity for a
1705   particular period of time.

Generalizations

1707   The *Custody* element inherits the attributes and/or associations of:

1708   • *ResponsibilityRelationship* (see the section entitled "ResponsibilityRelationship" for more information).

Properties

1710   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 *Party* that acts as the custodian of a particular entity. Redefines `responsibleParty.`

1711

8.4.12.4   CustodyChangeKind

1713   A class indicating the kind of CustodyChangeOccurrence.

1714 **Generalizations**

1715 The *CustodyChangeKind* element does not inherit any attributes or associations of from another element.

1716 **Properties**

1717 The *CustodyChangeKind* element does not have any additional attributes and/or associations.

1718 ### 8.4.12.5 CustodyChangeOccurrence

1719 An occurrence in the lifecycle of an entity related to the custody of that entity.

1720 **Generalizations**

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

1722 • *ProvenanceChangeOccurrence* (see the section entitled "ProvenanceChangeOccurrence" for more
1723 information).

1724 **Properties**

1725 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 *Party* that has custody of the entity as a result of the *CustodyChangeOccurrence*.
kind : CustodyChangeKind [0..1]	The kind of .
previousCustodian : Party [0..1]	The *Party* that previously had custody of the entity.
subchain : CustodyOccurrenceChain [0..1]	A *ChainOfCustody* that is encapsulated by the *CustodyChangeOccurrence* essentially creating a "sub-chain".
type : CustodyChangeType [0..1]	The type of the *CustodyChangeOccurrence*.

1726

1727 ### 8.4.12.6 CustodyChangeType

1728 The type of custody-related occurrences in the lifecycle of an entity that is of interest to some *Party*. Specializations
1729 of CustodyOccurrence will specify the kind of CustodyOccurrence that has happened or is expected to happen.

1730 **Generalizations**

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

1732 • *ProvenanceChangeType* (see the section entitled "ProvenanceChangeType" for more information).

1733 **Properties**

1734 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 *Party* that is expected to relinquish custody of *Entities* of *EntityType* as a result of the *CustodyOccurrence*.
priorCustodyType : CustodyType [0..*]	The *CustodyType* of the *Custody* responsibility relationships expected to be in place prior to *CustodyChangeOccurrences* of type *CustodyChangeType*.
resultingCustodianTypes : PartyType [0..*]	The type of *Party* that is expected to have custody of *Entities* of *EntityType* as a result of the *CustodyOccurrence*.
resultingCustodyType : CustodyType [0..*]	The *CustodyType* expected to be the result of occurrences of type *CustodyChangeType*.
subchainType : CustodyOccurrenceChainType [0..1]	The expected *ChainOfCustodyType* that the *CustodyOccurrenceType* encapsulates.

1735

8.4.12.7   CustodyEndKind

1737   A class indicating the CustodyChangeOccurrence was a kind of end.

Generalizations

1739   The *CustodyEndKind* element inherits the attributes and/or associations of:

1740   • *CustodyChangeKind* (see the section entitled "CustodyChangeKind" for more information).

1741   In addition, the *CustodyEndKind* element inherits the attributes and/or associations of:

1742   • *SemanticReferenceKind* (see the ~~section entitled "SemanticReference" for~~ **more**SCE specification for
1743   information).

Properties

1745   The *CustodyEndKind* element does not have any additional attributes and/or associations.

8.4.12.8   CustodyKind

1747   A class indicating the kind of *Custody* that a *Party* has with respect to some *Entity*.

Generalizations

1749   The *CustodyKind* element does not inherit any attributes or associations of from another element.

Properties

1751   The *CustodyKind* element does not have any additional attributes and/or associations.

8.4.12.9   CustodyOccurrenceChain

1753   A succession of *CustodyChangeOccurrences* that have happened in the life of an entity that is of interest to some
1754   *Party*.

Generalizations

1756   The *CustodyOccurrenceChain* element inherits the attributes and/or associations of:

1757   • *ProvenanceOccurrenceChain* (see the section entitled "ProvenanceOccurrenceChain" for more
1758   information).

Properties

1760   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 *ChainOfCustody*. This is derived by finding the earliest occurrence in the chain.
occurrenceHistory : CustodyChangeOccurrence [0..*]	A set of *CustodyOccurrences* that comprise the chain.
type : CustodyOccurrenceChainType [0..1]	The type of the *ChainOfCustody*.

8.4.12.10 CustodyOccurrenceChainType

A kind of *ProvenanceChainType* that captures a specification for a series of expected *CustodyOccurrenceTypes* that are expected for a particular entity type.

Generalizations

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

- *ProvenanceOccurrenceChainType* (see the section entitled "ProvenanceOccurrenceChainType" for more information).

Properties

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 *CustodyOccurrenceTypes* that specifies the sequencing of expected *CustodyOccurrences* in the lifecycle of an entity of interest to one or more *InterestedParties*.

8.4.12.11 CustodyStartKind

A class indicating the CustodyChangeOccurrence was a kind of start.

Generalizations

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

- *CustodyChangeKind* (see the section entitled "CustodyChangeKind" for more information).

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

- *SemanticReferenceKind* (see the **SCE** specification section entitled "SemanticReference" for more information).

Properties

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

8.4.12.12 CustodyTransferKind

A class indicating the CustodyChangeOccurrence was a kind of transfer.

Generalizations

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

- *CustodyChangeKind* (see the section entitled "CustodyChangeKind" for more information).

Properties

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

8.4.12.13 CustodyType

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

Generalizations

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

- *ResponsibilityRelationshipType* (see the section entitled "ResponsibilityRelationshipType" for more information).

Properties

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.

8.4.12.14 CustodyTypeGraph

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

Generalizations

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

- *ProvenanceTypeGraph* (see the section entitled "ProvenanceTypeGraph" for more information).

Properties

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

8.4.13   Ownership

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

OwnershipOccurrences are *Occurrences* that result in some change in ownership such as the acquisition of an entity by some *Party* or the transfer of ownership of an entity from one *Party* to another. These are useful for two reasons.

1814   First, they link ownership "periods" together and provide greater information about the events or processes that
1815   result in a transition in ownership much like *PedigreeOccurrences* provide insight into how an entity is created or
1816   evolved over time. Second, *Ownership* "records" are generated as a result of *OwnershipOccurrences* and so the
1817   *OwnershipOccurrences* provide insight in how and why ownership has changed..

1818   **PPMN** supports several kinds of *OwnershipOccurrenceTypes*: *AcquisitionOccurrenceTypes*,
1819   *OwnershipTransferOccurrenceTypes*, and *EndOwnershipOccurrenceTypes*. These specializations support the
1820   typical ownership transitions that may take place in the lifecycle of an entity but are not expected to be only types of
1821   transitions that may occur.

1822   A *ChainOfOwnership* is a kind of *ProvenanceChain* that tracks the ownership-related *Occurrences* of an entity of
1823   interest. A *ChainOfOwnership* may be typed in the same way as *ProvenanceChains* using a
1824   *ChainOfOwnershipType*. *ChainOfOwnershipType* allows stakeholders to define the expected changes in ownership
1825   of entities of a particular type in advance for planning or other purposes.

1826
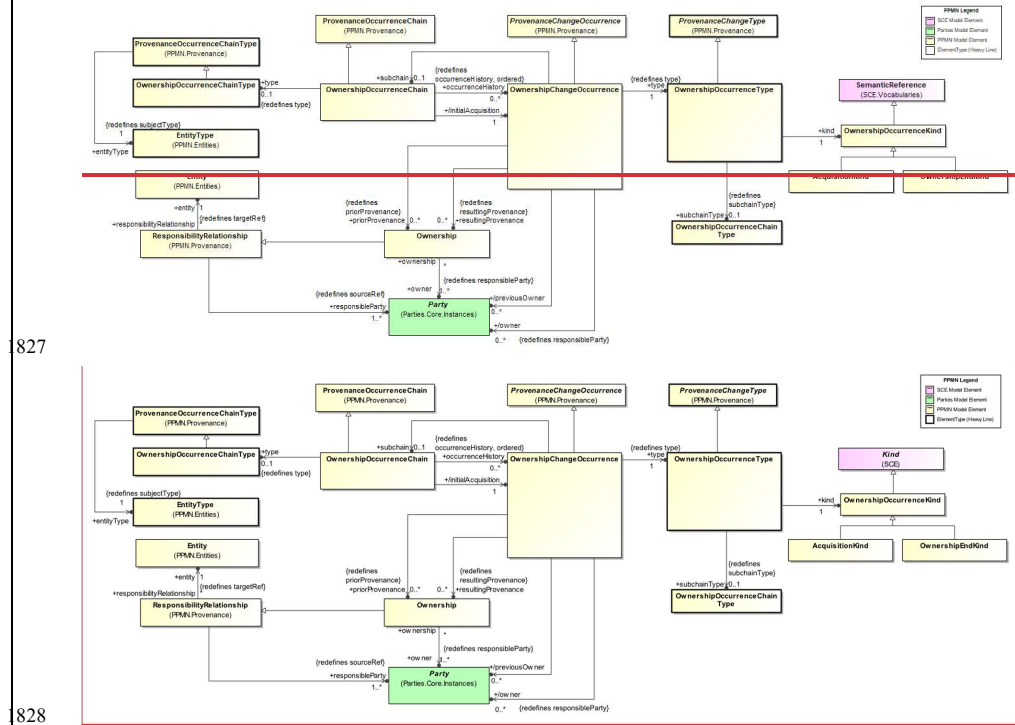
1827

1828

Figure 37:   Ownership Occurrence Chains

1829

1830   *ChainOfOwnership*, *ChainOfOwnershipType*, *OwnershipOccurrences*, and *OwnershipOccurrenceTypes* follow the
1831   same pattern established for other types of occurrences. This pattern supports the "nesting" of a *ChainOfOwnership*
1832   within an *OwnershipOccurrence*. This pattern allows for encapsulation of parts of a chain where the details of the
1833   *OwnershipOccurrences* of that part of a larger chain are either not known initially or are not deemed important in
1834   some context.

1835

> **Commented [JB100]:** Diagram updated for PPMN-19/PPMN-83.
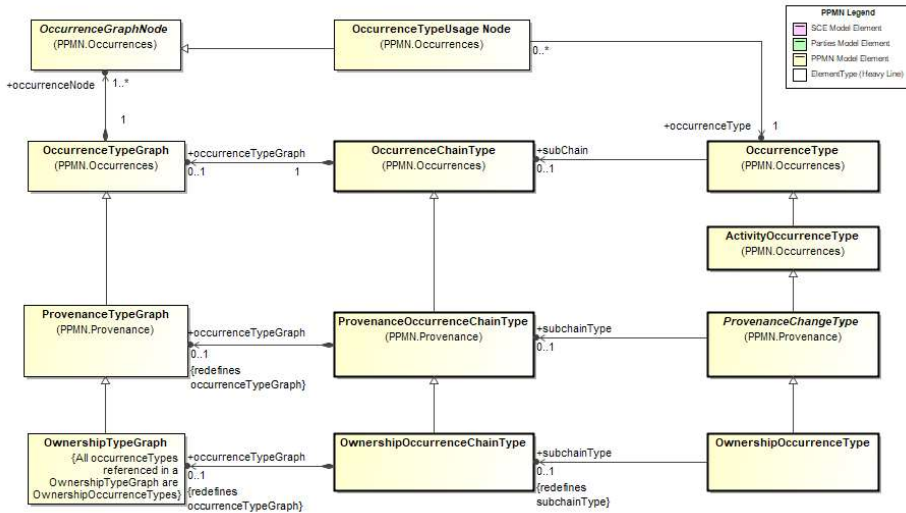
Figure 38:   Ownership Occurrence Chain Type Pattern

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.

Figure 39:   Ownership Occurrence Chain Types

1846

1847
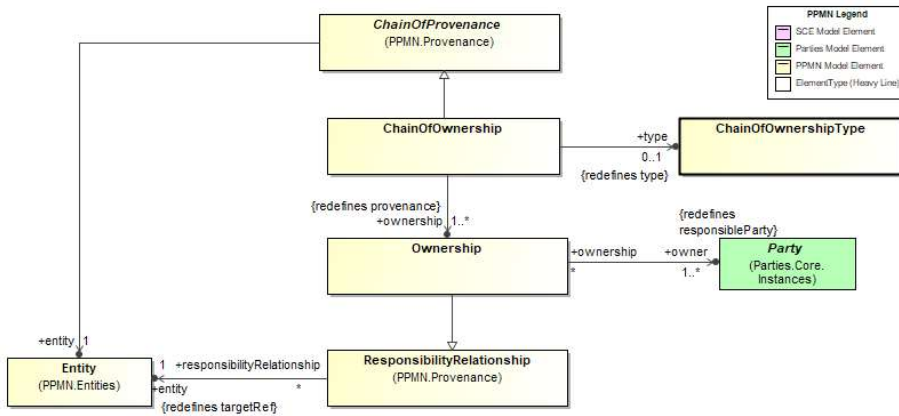
1848
1849 **Figure 40:   Chain of Ownership**

1850

1851
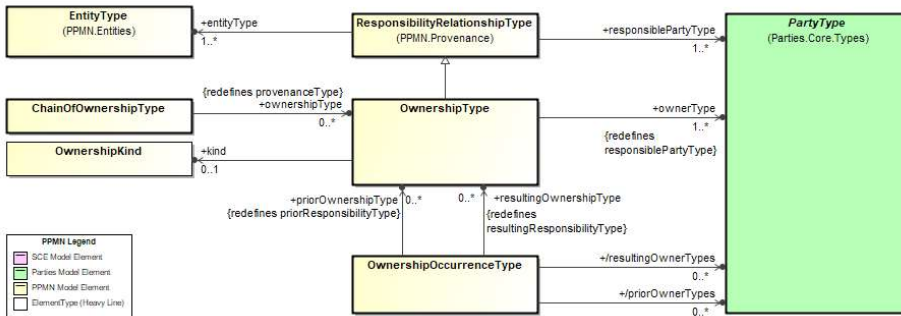
1852
1853 **Figure 41:   Chain of Ownership Types**

1854 ### 8.4.13.1   AcquisitionKind

1855 A class indicating how a *ChainOfOwnership* was started.

1856 **Generalizations**

1857 The *AcquisitionKind* element inherits the attributes and/or associations of:

1858 • *OwnershipOccurrenceKind* (see the section entitled "OwnershipOccurrenceKind" for more information).

1859 **Properties**

1860 The *AcquisitionKind* element does not have any additional attributes and/or associations.

8.4.13.2 ChainOfOwnership

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

Generalizations

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

- *ChainOfProvenance* (see the section entitled "ChainOfProvenance" for more information).

Properties

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.

8.4.13.3 ChainOfOwnershipType

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

Generalizations

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

- *ChainOfProvenanceType* (see the section entitled "ChainOfProvenanceType" for more information).

Properties

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 *OwnershipType* of the *Ownership* responsibility relationships included in *ChainOfOwnershps* that are of type *ChainOfOwnershipType*.

8.4.13.4 Ownership

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

Generalizations

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

- *ResponsibilityRelationship* (see the section entitled "ResponsibilityRelationship" for more information).

Properties

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 *Party* that acts as the owner of a particular entity. Redefines `responsibleParty`.

8.4.13.5   OwnershipChangeOccurrence

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

Generalizations

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

- *ProvenanceChangeOccurrence* (see the section entitled "ProvenanceChangeOccurrence" for more information).

Properties

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

8.4.13.6   OwnershipEndKind

A class indicating how the *ChainOfOwnership* was ended.

Generalizations

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

1901   •   *OwnershipOccurrenceKind* (see the section entitled "OwnershipOccurrenceKind" for more information).

Properties

1903   The *OwnershipEndKind* element does not have any additional attributes and/or associations.

8.4.13.7   OwnershipKind

1905   A specification of a particular kind of ownership responsibility.

Generalizations

1907   The *OwnershipKind* element does not inherit any attributes or associations of from another element.

Properties

1909   The *OwnershipKind* element does not have any additional attributes and/or associations.

8.4.13.8   OwnershipOccurrenceChain

1911   A succession of *OwnershipOccurrences* that have happened in the life of an entity that is of interest to some *Party*.

Generalizations

1913   The *OwnershipOccurrenceChain* element inherits the attributes and/or associations of:

1914   •   *ProvenanceOccurrenceChain* (see the section entitled "ProvenanceOccurrenceChain" for more
1915   information).

Properties

1917   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 *ChainOfOwnership*. This is derived by finding the earliest occurrence in the chain.
occurrenceHistory : OwnershipChangeOccurrence [0..*]	A set of *OwnershipOccurrences* that comprise the chain.
type : OwnershipOccurrenceChainType [0..1]	The type of the *ChainOfOwnership*.

1918

8.4.13.9   OwnershipOccurrenceChainType

1920   A kind of *ProvenanceChainType* that captures a specification for a series of expected *OwnershipOccurrenceTypes*
1921   that are expected for a particular entity type. An *OwnershipOccurrenceType* captures this specification through the
1922   occurrenceTypeGraph property - a graph of *OccurrenceGraphNodes* and *OccurrenceTransitionTypes*.

Generalizations

1924   The *OwnershipOccurrenceChainType* element inherits the attributes and/or associations of:

1925   •   *ProvenanceOccurrenceChainType* (see the section entitled "ProvenanceOccurrenceChainType" for more
1926   information).

Properties

1928 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 *OwnershipOccurrenceTypes* that specifies the sequencing of expected *OwnershipOccurrences* in the lifecycle of an entity of interest to one or more *InterestedParties*.

1929

8.4.13.10 OwnershipOccurrenceKind

1931 A class indicating the kind of *OwnershipOccurrence* that is expected.

Generalizations

1933 The *OwnershipOccurrenceKind* element inherits the attributes and/or associations of:

1934 • *SemanticReferenceKind* (see the **SCE** specification section entitled "SemanticReference" for more
1935   information).

Properties

1937 The *OwnershipOccurrenceKind* element does not have any additional attributes and/or associations.

8.4.13.11 OwnershipOccurrenceType

1939 The type of *OwnershipOccurrence* in the lifecycle of an entity that is of interest to some *Party*. *S*pecializations of
1940 *OwnershipOccurrenceType* will specify the kind of *OwnershipOccurrence* that has happened.

Generalizations

1942 The *OwnershipOccurrenceType* element inherits the attributes and/or associations of:

1943 • *ProvenanceChangeType* (see the section entitled "ProvenanceChangeType" for more information).

Properties

1945 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 *OwnershipOccurrenceType*.
priorOwnershipType : OwnershipType [0..*]	The *OwnershipType* exected to exist prior to occurrences of type *OwnershipOccurrenceType*.
priorOwnerTypes : PartyType [0..*]	The type of *Party* that is expected to relinquish ownership of *Entities* of *EntityType* as a result of the *OwnershipOccurrence*.
resultingOwnershipType : OwnershipType [0..*]	The *OwnershipType* expected to be the result of occurrences of type *OwnershipOccurrenceType*.
resultingOwnerTypes : PartyType [0..*]	The type of *Party* that is expected to have owership of *Entities* of *EntityType* as a result of *Occurrences* of the *OwnershipOccurrenceType*.

subchainType : OwnershipOccurrenceChainType [0..1]	A *ChainOfOwnershipType* that is encapsulated within the *OwnershipOccurrenceType* to create a "subchain".

1946

8.4.13.12 OwnershipTransferKind

1947

1948 A class indicating how a *ChainOfOwnership* was started.

Generalizations

1949

1950 The *OwnershipTransferKind* element does not inherit any attributes or associations of from another element.

Properties

1951

1952 The *OwnershipTransferKind* element does not have any additional attributes and/or associations.

8.4.13.13 OwnershipType

1953

1954 The type of *Ownership* that may exist between *Parties* of type *PartyType* and *Entities* of type *EntityType*.

Generalizations

1955

1956 The *OwnershipType* element inherits the attributes and/or associations of:

1957
1958 • *ResponsibilityRelationshipType* (see the section entitled "ResponsibilityRelationshipType" for more information).

Properties

1959

1960 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.

1961

8.4.13.14 OwnershipTypeGraph

1962

1963 A specialized type of *ProvenanceTypeGraph* that captures the *OwnershipOccurrenceTypes* that are expected in the
1964 lifecycle of one or more types of entities.

Generalizations

1965

1966 The *OwnershipTypeGraph* element inherits the attributes and/or associations of:

1967 • *ProvenanceTypeGraph* (see the section entitled "ProvenanceTypeGraph" for more information).

Properties

1968

1969 The *OwnershipTypeGraph* element does not have any additional attributes and/or associations.

8.5   Claims

1970

1971 The Claims package contains elements related to *Claims* made by *Parties* about *Occurrences*.

1972 In many situations, pedigree and/or provenance information about entities is put forth by some party as being true
1973 when in fact, that information may be disputed and even shown to be false. *Claims* provide a mechanism to note the
1974 *Party* (the claimant) that claims an *Occurrence* has happened. The time the claim was made is captured as well as
1975 whether the *Claim* was made in a "positive" or "negative" manner (the `claimPositivity`). ClaimPositivity
1976 states whether the Claim was made in a "positive" manner, i.e., the Occurrence is claimed to have happened, or a
1977 "negative" maner, i.e., the Occurrence is claimed *not to have happened.* A `claimPositivity` of "Possible"
1978 means that the Occurrence *may* have happened.

1979
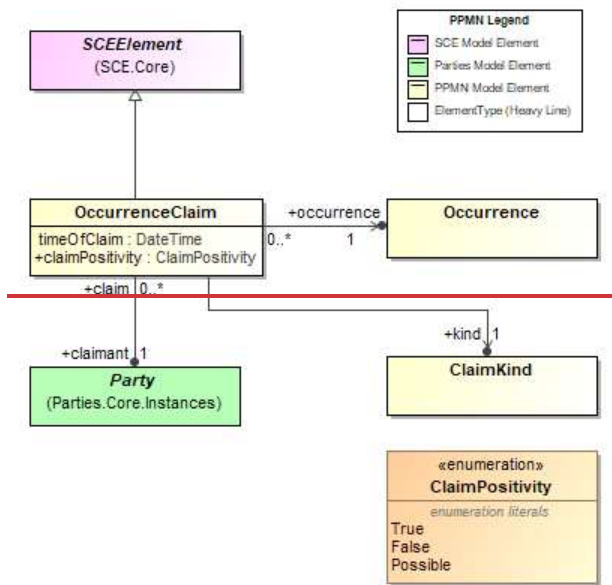
1980
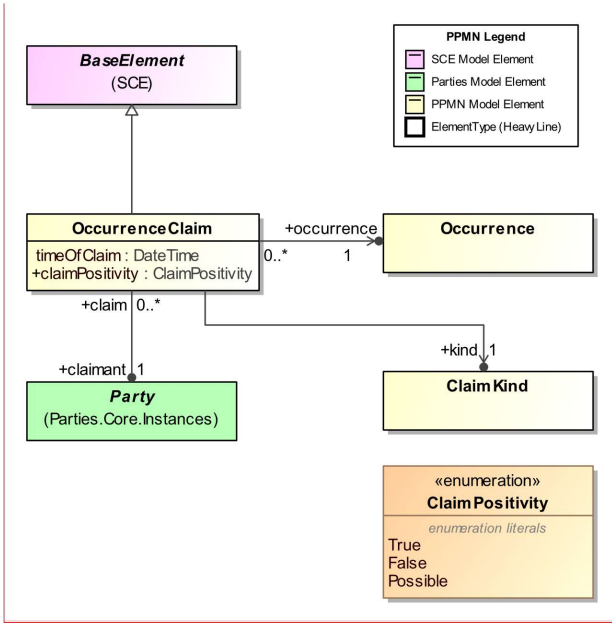
1981

Figure 42:   Claims

1983 A Claim may be assessed in some way as stated by a ClaimAssessment by some Party (the assessor). The actual
1984 method or mechanism of the assessment is outside the scope of this specification.

1985
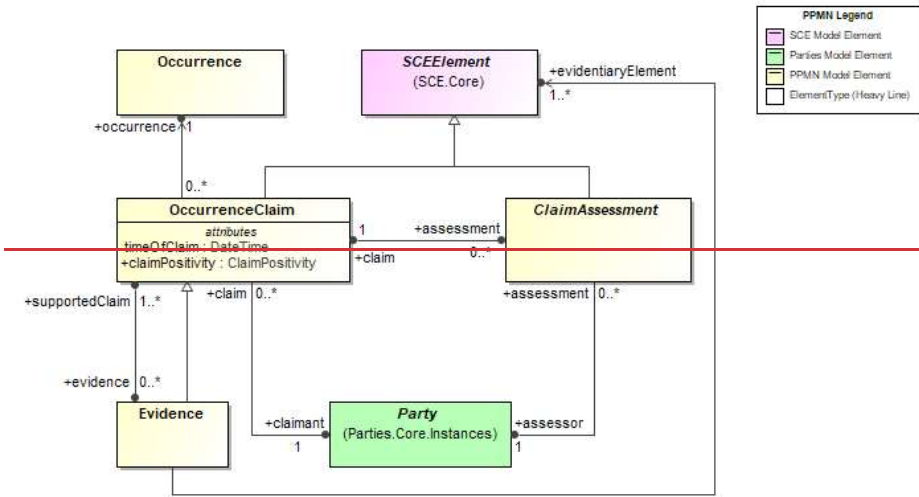
1986
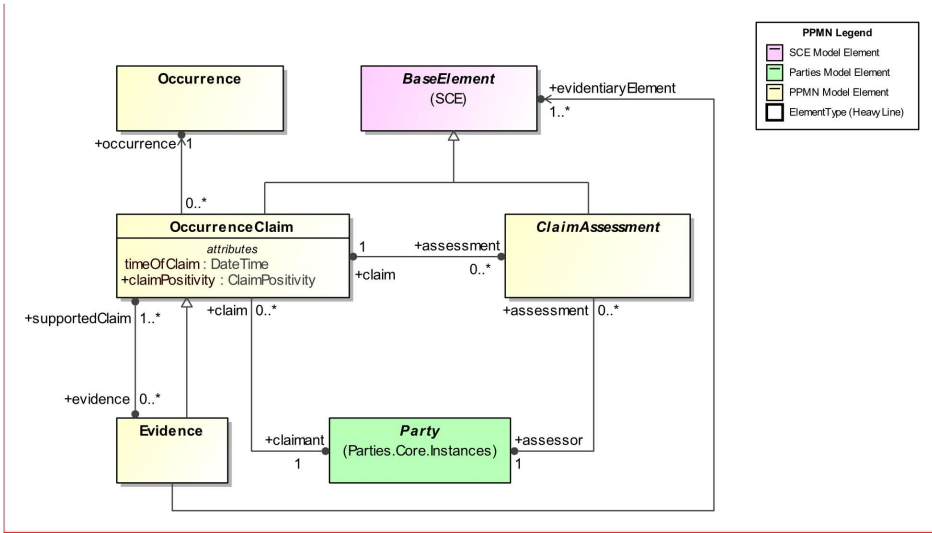
Figure 43: Claim Assessments

8.5.1 ClaimPositivity

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.

8.5.2 ClaimAssessment

An assessment of a Claim by an assessor.

Generalizations

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

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

Properties

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.

1999

8.5.3   ClaimKind

2000

2001   A class that indicates the kind of *Claim* that has been made.

2002   **Generalizations**

2003   The *ClaimKind* element inherits the attributes and/or associations of:

2004   • *SemanticReferenceKind* (see the **SCE** specification ~~section entitled "SemanticReference"~~ for more
2005   information).

2006   **Properties**

2007   The *ClaimKind* element does not have any additional attributes and/or associations.

2008   **Generalizations**

2009   The *Evidence* element inherits the attributes and/or associations of:

2010   • *OccurrenceClaim* (see the section entitled "OccurrenceClaim" for more information).

2011   **Properties**

2012   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 *Evidence* for the supported *Claims*.
supportedClaim : OccurrenceClaim [1..*]	The *Claims* that the *Evidence* is intended to support.

2013

8.5.4   OccurrenceClaim

2014

2015   A statement made by a Party about whether an Occurrence happened or not.

2016   **Generalizations**

2017   The *OccurrenceClaim* element inherits the attributes and/or associations of:

2018   • **SCE** *SCEElement* (see the ~~section~~ **SCE** specification for more information).

> **Commented [JB104]:** Text updated for PPMN-19/PPMN-83

2019   **Properties**

2020   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 *Evidence* intended to support the *Claim*.
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.

8.6   Rationale

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

PPMN supports the ability to capture a *Rationale*, the reasoning or justification, for *Occurrences* and *OccurrenceTypes*. *RationaleType* enables capture of the type of a particular Rationale or of the kind of *Rationale* that is expected in a particular context.
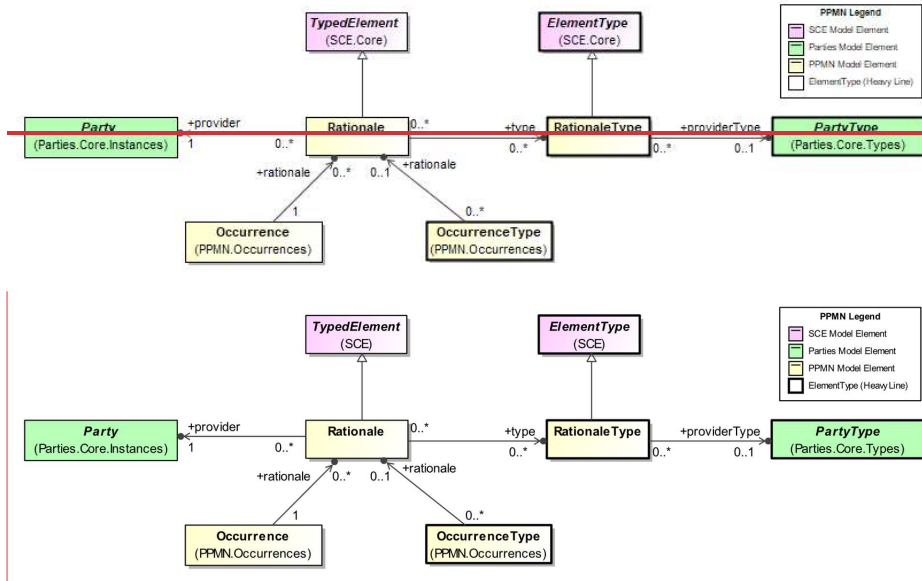
Figure 44:   Rationale

> **Commented [JB105]:** Diagram updated for PPMN-19/PPMN-83.

8.6.1 Rationale

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

Generalizations

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

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

Properties

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 *Party* that provided the *Rationale*.
type : RationaleType [0..*]	The class(es) that provide(s) a specification of the *Rationale*.

8.6.2 RationaleType

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

Generalizations

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

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

Properties

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 *PartyType* that is expected to provide the kind of *Rationale* specified by the *RationaleType*.

8.7   ~~Extensions~~

~~PPMN includes two mechanisms for extension: Adornments and Annotations. Descriptions of these two mechanisms are described herein.~~

8.9.0   ~~Adornment~~

~~The Adornment package contains elements that support the extension of elements with addtional attributes using the adornment pattern.~~

~~PPMN *AdornmentProfiles* extend the **SCE** extension mechanism to allow for the addition of attributes to any *BaseElement* in Pedigree and Provenance information without having to modify that element. The approach is analogous to the Gang of Four adornment design pattern wherein additional features are added to elements without those extensions having to be known when the original element is created. The **SCE** extension mechanism allows for *BaseElements* include extension attributes and values that have been defined by a tool that implements the **SCE**~~

2058 specification. The attributes become part of the BaseElement. **PPMN** *AdornmentProfiles* provide the additional
2059 ability to "adorn" the elements with attributes.

2060 **PPMN** *AdornmentProfile* extends the **SCE** extension mechanism with a number of key features. *AdornmentProfiles*
2061 specialize *ExtensionDefinition* to include a version number, a set of *AdornedElements*, and a set of
2062 *AdornmentAttributeDefinitions*. The *AdornedElements* referenced by the *AdornmentProfile* specify which
2063 *AdornmentAttributeDefinitions* may adorn which *BaseElements*. The *AdornmentProfile's* set of *attributeDefinitions*
2064 are an additional set of definitions that may be applied generally rather than to specific *BaseElements*.

2065 *AdornmentAttributeDefinition* extends **SCE** *Extension.AttributeDefinition* to provide additional detail about the
2066 characteristics of the adornment attributes. *AdornmentValues*, as specified by *AdornmentAttributeDefinition*, may
2067 be chronicled (successive versions are tracked), removable or not, modifiable or not, and required at creation of the
2068 *BaseElement* to which they are applied. *AdornmentAttributeDefinition* also includes a pointer to an *AttributeType*
2069 that specifies whether the *AttributeValue* will be an integer, a string, or a date/time.

2070
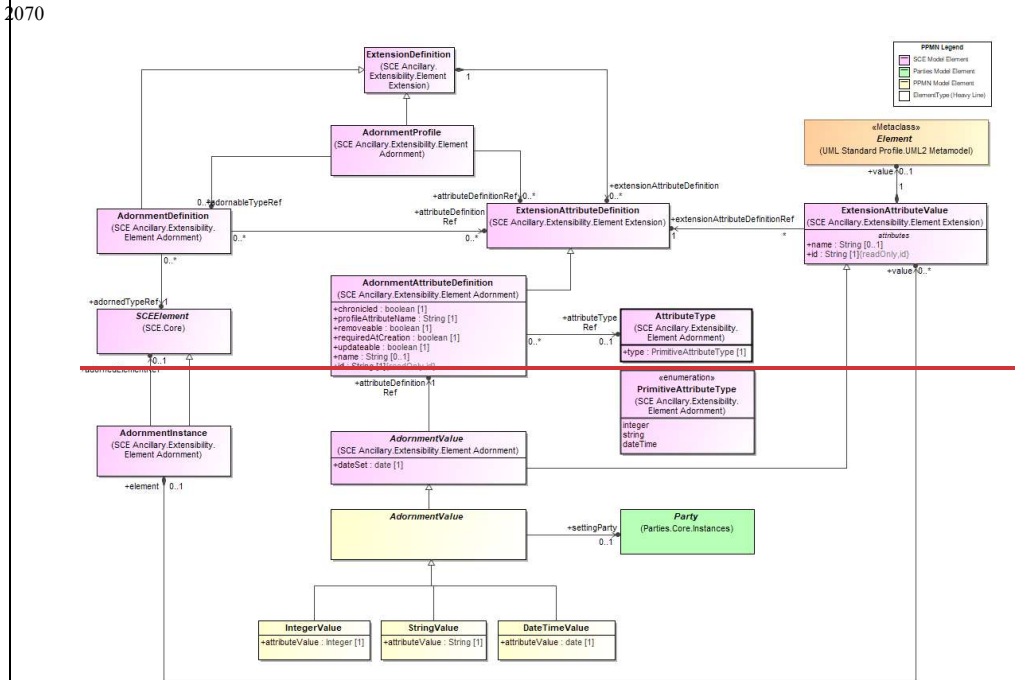
2071
2072 **Figure 8:   Adornment Profiles**

2073 ### 8.17.0.0   AdornmentValue

2074 A value of an attribute associated with an AdornedElement. PPMN AdornmentValue is specialization of SCE
2075 AdornmentValue that extends the SCE AdornmentValue to include the party that set the value.
2076 **Generalizations**

2077 The *AdornmentValue* element inherits the attributes and/or associations of:

2078   • *AdornmentValue* (see the section entitled "AdornmentValue" for more information).

Properties

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

Table 8.　　AdornmentValue Attributes and/or Associations

Property/Association	Description
settingParty : Party [0..1]	The *Party* that set the adornment value.

8.32.0.0　DateTimeValue

An AdornmentValue that is an DateTime type.

Generalizations

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

- *AdornmentValue* (see the section entitled "AdornmentValue" for more information).

Properties

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

Table 8.　　DateTimeValue Attributes and/or Associations

Property/Association	Description
attributeValue : date [1]	The actual value of the *DataTimeValue*.

8.47.0.0　IntegerValue

An AdornmentValue that is an Integer type.

Generalizations

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

- *AdornmentValue* (see the section entitled "AdornmentValue" for more information).

Properties

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

Table 8.　　IntegerValue Attributes and/or Associations

Property/Association	Description
attributeValue : Integer [1]	The actual value of the *IntegerValue*.

8.62.0.0　StringValue

An AdornmentValue that is a String type.

Generalizations

2101 ~~The *StringValue* element inherits the attributes and/or associations of:~~

2102 • ~~*AdornmentValue* (see the section entitled "AdornmentValue" for more information).~~

2103 **~~Properties~~**

2104 ~~The following table presents the additional attributes and/or associations for *StringValue*:~~

~~Table 8.   StringValue Attributes and/or Associations~~

~~Property/Association~~	~~Description~~
~~**attributeValue** : String [1]~~	~~The actual value of the *StringValue*.~~

2105

8.77~~8.7~~ Annotations

2106

2107 The Annotation package contains elements related to the notion of annotation of elements with notes about that
2108 element.

2109 *Annotations* are applied to ~~*NamedElement*~~*BaseElements* for any purpose that suits the business needs of an
2110 organization. *Annotations* can exist independently of those elements providing a "catalog" of *Annotations*.
2111 *AnnotationTemplate* provides a means of creating base annotations that can be "instantiated" as either
2112 *SimpleAnnotations* or *ChronicledAnnotations*. *Annotations* may have an association to the *AnnotationTemplate*
2113 from which they were created. The *Party* creating an *Annotation* is captured as the creator. That *Party* or another
2114 *Party* may assign an annotation to a ~~*NamedElement*~~*BaseElement* through an *AnnotationAssignment* relationship.

2115

2116
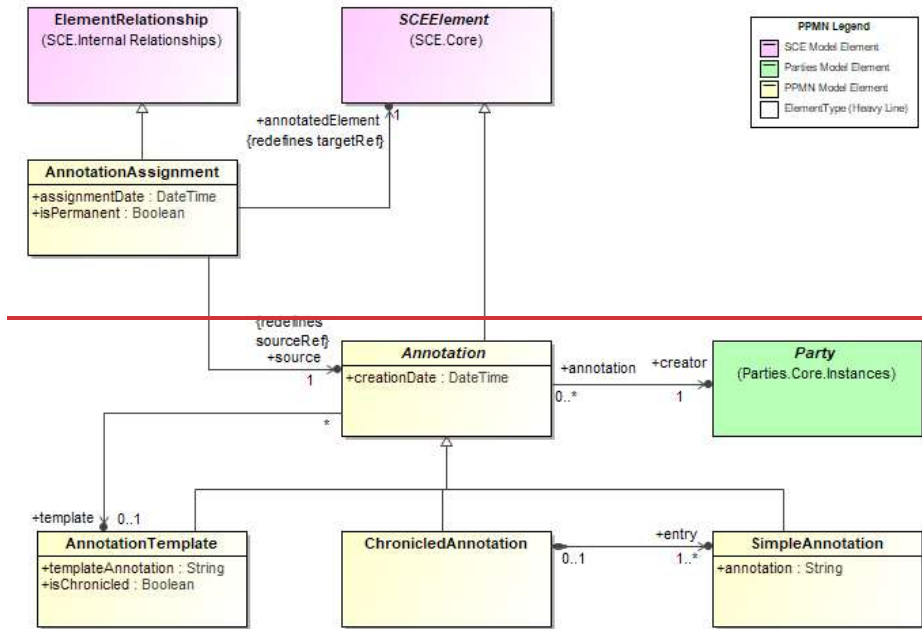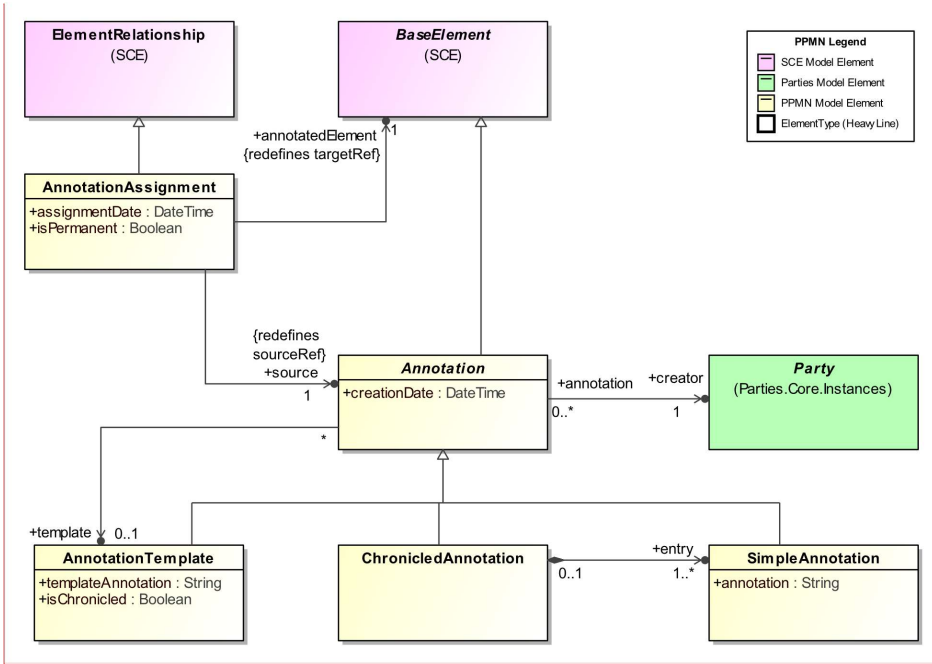
Commented [JB109]: Diagram updated for PPMN-19/PPMN-83.

Figure 45:   Annotations

8.77.1.18.7.1.1   Annotation

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

Generalizations

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

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

Commented [JB110]: Text updated for PPMN-19/PPMN-83

Properties

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 *Party* that created the annotation.
template : AnnotationTemplate [0..1]	The template from which an *Annotation* was created.

2127 **8.77.1.28.7.1.2   AnnotationAssignment**

2128 An association that links an *Annotation* to a *NamedElementBaseElement* in a PPMN information set.

2129 **Generalizations**

2130 The *AnnotationAssignment* element inherits the attributes and/or associations of:

2131 • *ElementRelationship* (see the section entitled "ElementRelationship"SCE specification for more
2132 information).

2133 **Properties**

2134 The following table presents the additional attributes and/or associations for *AnnotationAssignment*:

Table 70.   AnnotationAssignment Attributes and/or Associations

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

2135

2136 **8.77.1.38.7.1.3   AnnotationTemplate**

2137 A kind of *Annotation* that is intended to be used as a template for other *Annotations*.

2138 **Generalizations**

2139 The *AnnotationTemplate* element inherits the attributes and/or associations of:

2140 • *Annotation* (see the section entitled "Annotation" for more information).

2141 **Properties**

2142 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 *Annotations* created with this template are *ChronicledAnnotations* or not.
templateAnnotation : String []	A default string that is meant for recurring use.

2143

2144 **8.77.1.48.7.1.4   ChronicledAnnotation**

2145 A kind of *Annotation* that has a series of time-based entries. Individual entries are captured as *SimpleAnnotations*
2146 with the isPermenant flag set to True. The creationDate of the *SimpleAnnotations* that represent the entries
2147 of a *ChronicledAnnotation* captures the date the *ChronicledAnnotation* was updated.

2148 **Generalizations**

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

2150 • *Annotation* (see the section entitled "Annotation" for more information).

2151 **Properties**

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

Table 72.   ChronicledAnnotation Attributes and/or Associations

Property/Association	Description
entry : SimpleAnnotation [1..*]	A *SimpleAnnotation* that represents one entry in a *ChronicledAnnotation*.

2153

2154 **8.7.7.1.58.7.1.5   SimpleAnnotation**

2155 A kind of *Annotation* that is a simple note related to one or more *BaseElements* ~~NamedElementBaseElements~~ in a
2156 PPM information set.

(comment)

2157 **Generalizations**

2158 The *SimpleAnnotation* element inherits the attributes and/or associations of:

2159 • *Annotation* (see the section entitled "Annotation" for more information).

2160 **Properties**

2161 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 *Annotation*.

2162

2163 **8.78 8.8 Delegation**

2164 The Delegation package provides elements related to the notion of delegation of ~~responsiblities~~responsibility for an
2165 entity from one party to another.

2166 Delegation captures the notion that a Party may assign a set of responsibilities to another party. The responsibilities
2167 being assigned are essentially captured as a Role. The class ActedOnBehalfOf is a relationship that states that one
2168 Party was acting for or representing another Party and that action may be justified by a Delegation. The property
2169 inRole allows a model to specify that the *Party* acted on behalf of another *Party* while performing a particular role
2170 in an *Occurrence*.

2171

Comment [JB113]: Text updated for PPMN-19/PPMN-83

Comment [JB114]: Updated to address typographical error found by AB review.
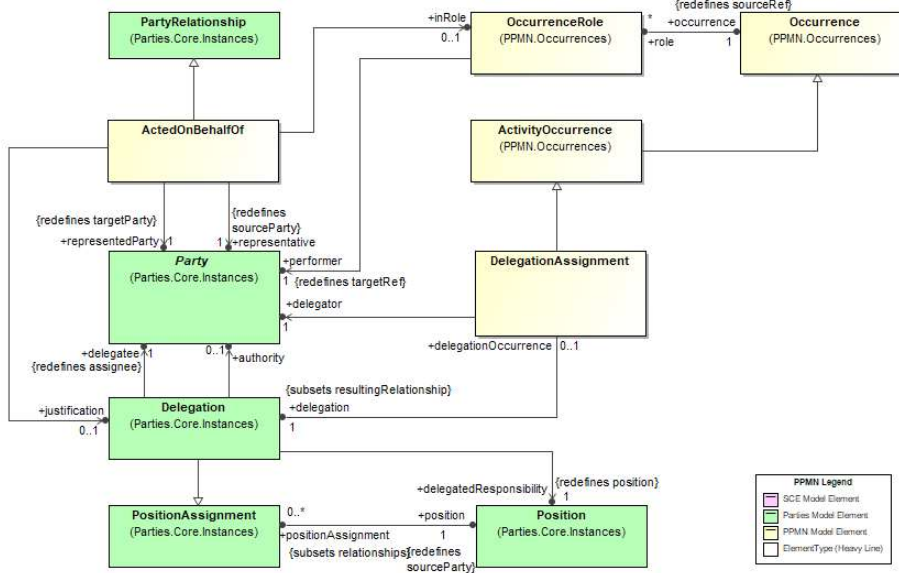
Figure 46: Delegation

8.78.18.8.1   ActedOnBehalfOf

A relationship that indicates that one *Party* represented another *Party* in some way. That action may be justified by some *Delegation* of responsibilities.

Generalizations

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

- *PartyRelationship* (see the section entitled "PartyRelationship" for more information).

Properties

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

Table 74.   ActedOnBehalfOf Attributes and/or Associations

Property/Association	Description
inRole : OccurrenceRole [0..1]	The *OccurrenceRole* in which one *Party* acted on behalf of another *Party*.
justification : Delegation [0..1]	The *Delegation* that provides justification for the `representative` to act on the part of the `representedParty`.
representative : Party [1]	The *Party* representing the `representedParty`.
representedParty : Party [1]	The *Party* on whose part the `representative` acted.

2182

8.7.2 8.8.2 DelegationAssignment

2183

2184 A kind of *ActivityOccurrence* wherein one *Party* delegates a set of responsibilities to another *Party*.

Generalizations

2185

2186 The *DelegationAssignment* element inherits the attributes and/or associations of:

2187 • *ActivityOccurrence* (see the section entitled "ActivityOccurrence" for more information).

Properties

2188

2189 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 *Delegation* that was the result of the *DelegationAssignment*.
delegator : Party [1]	The *Party* responsible for the *DelegationAssignment*.

2190

8.79 8.9 Additional Relationships

2191

2192 In addition to Delegation and Derivation, PPMN includes a number of other types of relationships that are important
2193 to pedigree and/or provenance. These additional relationships are described herein.

2194 **PPMN** includes several other types of relationships that may be important to particular stakeholders in addition to
2195 derivations and delegations. These cover the concepts of attribution, specialization, alternates and general
2196 "informing of".

2197 Attribution is captured through the *AttributedTo* relationship. This element states that an entity of interest was
2198 generated through some unknown activity or action of the *Party*.

2199 ~~The *Specializes* relationship specifies that one element represents a more specific type of thing than the target of the~~
2200 ~~relationship. The *Specializes* relationship will generally between two entities of some type. However, this is not~~
2201 ~~mandated. It may be useful in certain situations to note specialization relationships between *Parties* or *Occurrences*.~~
2202 ~~Note that the `source` and `target` of a *Specialization* must both be of the same general "type". In other words~~
2203 ~~they must both be, for example, *Entities*, or both be *Parties*, or both be *Occurrences*.~~

2204 The *AlternateOf* relationship states that two entities or elements represent the same thing or aspects of the same
2205 thing. The *AlternateOf* relationship will generally between two entities of some type. ~~As with *Specializes*,~~
2206 hHowever~~however~~, this is not always the case. It may be useful in certain situations to note alternate *Parties* or
2207 *Occurrences*. Note that the `source` and `target` of the *AlternateOf* must both be of the same general "type". In
2208 other words they must both be, for example, *Entities*, or both be *Parties*, or both be *Occurrences*.

2209 The *Informed* relationship is used to show that one *Occurrence* provided information or insight to or in some way
2210 affected another *Occurrence*. For example a testing process may inform a redesign of an assembly line for a
2211 manufacturer.

2212

2213

Commented [JB115]: Updated to address typographical error found during AB Review.

Commented [JB116]: Updated to address PPMN-37/PPMN-143
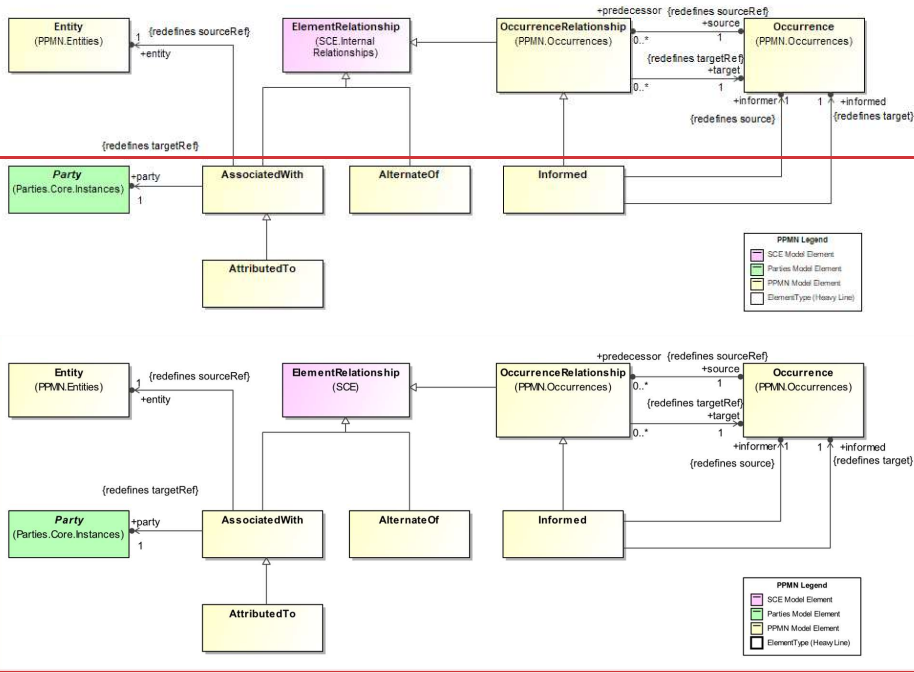
Figure 47: Additional PPMN Relationships

8.79.18.9.1   AlternateOf

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

Generalizations

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

- *ElementRelationship* (see the ~~section entitled "ElementRelationship"~~**SCE** specification for more information).

Properties

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

8.79.28.9.2   AssociatedWith

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

Generalizations

Commented [JB117]: Diagram updated for PPMN-19/PPMN-83.

Commented [JB118]: Updated to address PPMN-37/PPMN-143

Commented [JB119]: Text updated for PPMN-19/PPMN-83

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

- *ElementRelationship* (see the ~~section entitled "ElementRelationship"~~SCE specification for more information).

Properties

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 *Party*.
party : Party [1]	The *Party* to which some entity is associated.

8.79.38.9.3   AttributedTo

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

Generalizations

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

- *AssociatedWith* (see the section entitled "AssociatedWith" for more information).

8.79.48.9.4   Informed

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

Generalizations

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

- *OccurrenceRelationship* (see the section entitled "OccurrenceRelationship" for more information).

Properties

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 *Occurrence* that was informed by the source *Occurrence*.
informer : Occurrence [1]	The *Occurrence* that informed another *Occurrence*.

8.808.10   Packaging

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

2261 The Pedigree and Provenance Metamodel and Notation supports the capture of events that happen in the lifecycle of
2262 entities of interest including creation, evolution, destruction, as well as changes in ownership and custody. In
2263 addition to capturing events that happened in the past, the specification also enables specifying events that are
2264 expected to happen in the future. As stated previously, these elements are loosely referred to as the "instances" and
2265 "types", respectively. The main packaging structures of PPMN support packaging of these elements using
2266 *PPMNInstances* and the *PPMNDefinitions* elements.

2267 *PPMNInstances* are specializations of *PartyInstances* and are designed to group "instances" related to events that
2268 have taken place in the lifecycle of entities of interest. These elements include actual events or *Occurrences*, the
2269 *Entities*, and the *Parties* involved.

2270 *PPMNDefinitions* are specializations of *PartyDefinitions* and are designed to group the **PPMN** "types", i.e. the
2271 elements related to "expected" *Occurrences*. These elements include *OccurrenceTypes*, *EntityTypes*, and *PartyTypes*
2272 among others. ~~*PPMNDefinitions* also reference any profiles that have been applied through the~~ `appliedProfile`
2273 ~~property. Any applied profiles must be contained via the inherited~~ `profile` ~~property.~~

> **Commented [JB122]:** Text updated for PPMN-19/PPMN-83

2274 PPMNInstances and PPMNDefinitions together are included in PPMNModels along with relevant
2275 *PPMNKinds*. ~~*PPMNVocabularies*~~*PPMNKinds*. *PPMNModels* represent the semantics of the model versus the
2276 presentation elements contained in the *PPMNDI* package. *PPMNModels* are specializations of *PartyModels* and so
2277 may include *PartyInstances* and *PartyDefinitions* as well.

> **Commented [JB123]:** Text updated for PPMN-28/PPMN-84

2278 All of these elements are brought together as a complete bundle in the *PPMNModelPackage*. *PPMNModelPackages*
2279 contain both the model elements via the `model` property as well as the presentation elements via the
2280 `presentation` property. *PPMNModelPackages* are a specialization of *PartyModelPackage* and so may contain
2281 all of the *Party*-related elements contained therein.

2282 ~~*ProfilePackages* group elements associated with *AdornmentProfile* definitions so that profiles can be shared~~
2283 ~~between organizations and/or user communities. These elements include *AdornmentProfiles*, *AdornmentDefinitions*,~~
2284 ~~*AttributeTypes*, and *AdornmentAttributeDefinitions*.~~

> **Commented [JB124]:** Text updated for PPMN-72/PPMN-81

2285 All **PPMN** packages and models are specializations of **SCE** *Package* and as such can contain other **SCE** *Packages*
2286 or their specializations. They can also include imports of external elements through ~~throught~~ the *Import* element in
2287 **SCE**.

> **Commented [JB125]:** spelling

> **Commented [JB126]:** Text updated to address PPMN-70/PPMN-135

2288 ~~*ProfilePackages* group elements associated with *AdornmentProfile* definitions so that profiles can be shared~~
2289 ~~between organizations and/or user communities. These elements include *AdornmentProfiles*, *AdornmentDefinitions*,~~
2290 ~~*AttributeTypes*, and *AdornmentAttributeDefinitions*.~~

> **Commented [JB127]:** Text updated for PPMN-19/PPMN-83

> **Commented [JB128]:** Text updated for PPMN-72/PPMN-81

2291

2292

2293
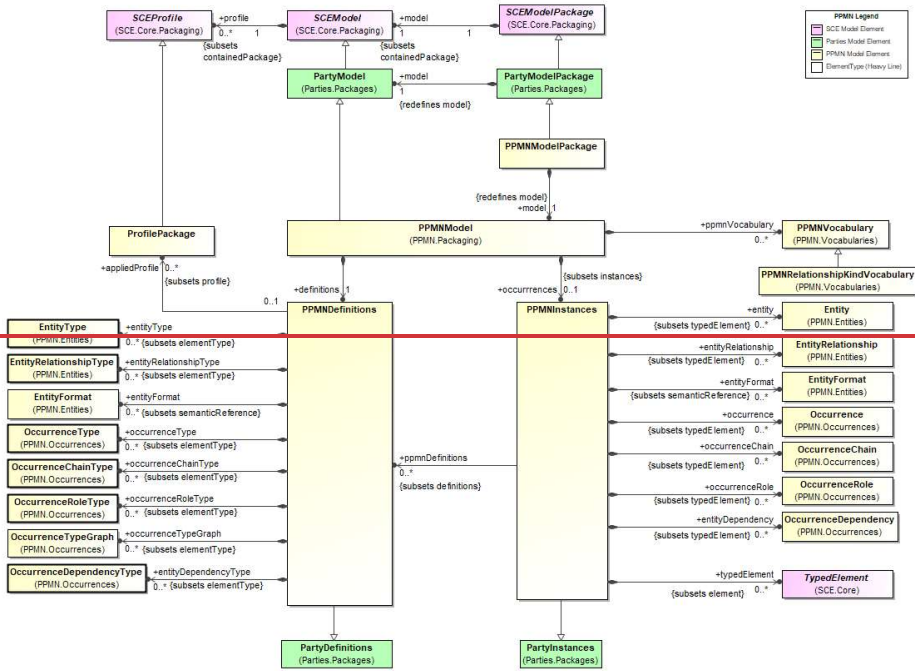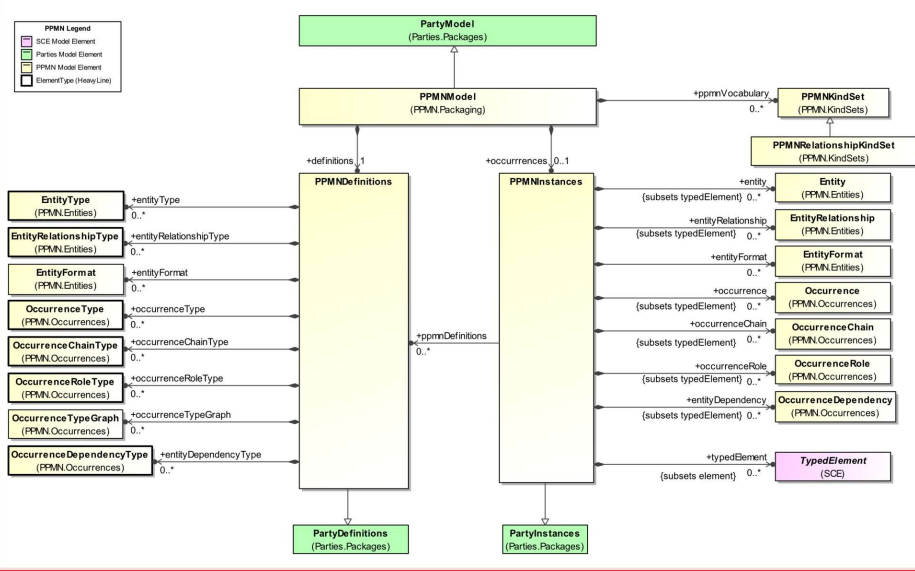
Commented [JB129]: Diagram updated for PPMN-19/PPMN-83.

2294

2295 **Figure 48: PPMN Packaging**

8.80.18.10.1 PPMNDefinitions

2297 A kind of *PartyDefinitions* *PartySCEDefinitions* that is the container for "Type-related" PPMN elements. Type-
2298 related elements include elements such as *OccurrenceChainTypes* and its specializations, *OccurrenceTypes* and its
2299 specializations, and profiles. Type-related elements are contained in *TypePackages*. while profiles are contained in
2300 *ProfilePackages.*

Generalizations

2302 The *PPMNDefinitions* element inherits the attributes and/or associations of:

2303   • *PartyDefinitions* (see the section entitled "PartyDefinitions" for more information).

Properties

2305 The following table presents the additional attributes and/or associations for *PPMNDefinitions*:

Table 78. PPMNDefinitions Attributes and/or Associations

Property/Association	Description
~~**appliedProfile** : ProfilePackage [0..*]~~	~~A set of *ProfilePackages* included in this package that contain any necessary profile definitions.~~
entityDependencyType : OccurrenceDependencyType [0..*]	A list of *EntityDependencyTypes* within the *PPMNModel*.
entityFormat : EntityFormat [0..*]	A list of the *EntityFormats* referenced within the *PPMNDefinitions* package.
entityRelationshipType : EntityRelationshipType [0..*]	A list of *EntityRelationshipTypes* within the *PPMNModel*.
entityType : EntityType [0..*]	A list of *EntitieTypes* within the *PPMNModel*.
occurrenceChainType : OccurrenceChainType [0..*]	A list of *OccurrenceChainTypes* within the *PPMNModel*.
occurrenceRoleType : OccurrenceRoleType [0..*]	A list of *OccurrenceRoleTypes* within the *PPMNModel*.
occurrenceType : OccurrenceType [0..*]	A list of *OccurrenceTypes* within the *PPMNModel*.
occurrenceTypeGraph : OccurrenceTypeGraph [0..*]	A list of *OccurrenceTypeGraphs* within the *PPMNModel*.

2306

8.80.28.10.2 PPMNInstances

2308 **PPMN** information sets are exchanged in bulk through the *OccurrenceSet* element. The *OccurrenceSet* element
2309 provides the outermost container for other **PPMN** elements contained in one or more *PPMNPackages*. The
2310 occurrence chains, occurrences and other "instance-related" elements are contained within one or more
2311 *OccurrenceSets* while "type-related" elements such as *OccurrenceChainTypes*, and *OccurrenceTypes*, and
2312 *PPMNProfiles* if present are contained within definitionsdefinitionsdefinitionsdefinitionsdefinitionsDdefinitions
2313 packages.

Generalizations

2315 The *PPMNInstances* element inherits the attributes and/or associations of:

2316 • *PartyInstances* (see the section entitled "PartyInstances" for more information).

2317 **Properties**

2318 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 *Entities* of interest within the *PPMNModel*.
entityDependency : OccurrenceDependency [0..*]	A list of *EntityDependencies* within the *PPMNModel*.
entityFormat : EntityFormat [0..*]	A list of the *EntityFormats* referenced within the *PPMNInstances* package.
entityRelationship : EntityRelationship [0..*]	A list of *EntityRelationships* within the *PPMNModel*.
occurrence : Occurrence [0..*]	A list of *Occurrences* within the *PPMNModel*.
occurrenceChain : OccurrenceChain [0..*]	A list of *OccurrenceChains* within the *PPMNModel*.
occurrenceRole : OccurrenceRole [0..*]	A list of *OccurrenceRoles* within the *PPMNModel*.
ppmnDefinitions : PPMNDefinitions [0..*]	The property refers to zero or more *PPMNDefinitions* packages that contains the *ElementTypes* that provide a basis for the instances contained in the *PartyInstances* package.
typedElement : TypedElement [0..*]	A list of *TypedElements* within the *PPMNModel*.

2319

8.80.38.10.3 PPMNModel

2321 A *PPMNModel* is the main container for semantic elements of a **PPMN** model including types, instances, and
2322 KindSets.profiles, and KindSetsvocabulariesKindSets. As a specialization of *PartyModel* it also contains Party-
2323 related types, and instances....., profiles, and vocabularies. These elements are separate from the visual elements
2324 included in the PPMNModelPackage.

Commented [JB134]: Text updated for PPMN-19/PPMN-83

2325 **Generalizations**

2326 The *PPMNModel* element inherits the attributes and/or associations of:

2327 • *PartyModel* (see the section entitled "PartyModel" for more information).

2328 **Properties**

2329   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 ~~s and profile~~ elements.
occurrrences : PPMNInstances [0..1]	The packages that contain the elements that represent the definitions of a **PPMN** model. These elements generally include the type ~~s and profile~~ elements.
ppmnVocabulary : PPMNVocabulary [0..*]	The `ppmnVocabulary` is a list of terms (as *KindsKindsKindsKindsKindsSemanticReferenceKinds*) that provide an extensible mechanism to define the elements of enumerations in a *PPMNModel*.

Commented [JB135]: Text updated for PPMN-19/PPMN-83

2330

2331   ~~8.80.4   PPMNModelPackage~~

Commented [JB136]: Text updated for PPMN-19/PPMN-83

2332   ~~A namespace that groups **PPMN** *Elements* comprising the pedigree and provenance information about some set of~~
2333   ~~entities.~~

2334   ~~**Generalizations**~~

2335   ~~The *PPMNModelPackage* element inherits the attributes and/or associations of:~~

2336   • ~~*PartyModelPackage* (see the section entitled "PartyModelPackage" for more information).~~

2337   ~~**Properties**~~

2338   ~~The following table presents the additional attributes and/or associations for *PPMNModelPackage*:~~

~~PPMNModelPackage Attributes and/or Associations~~

~~Property/Association~~	~~Description~~
~~**model** : PPMNModel [1]~~	~~The *PPMNModel* contained within the *PPMNModelPackage*.~~

2339

2340   ~~8.81.0   ProfilePackage~~

Commented [JB137]: Text updated for PPMN-19/PPMN-83

2341   ~~A kind of *PPMNPackage* that comprises **PPMN** profiles that can be applied to other **PPMN** *TypedElements*.~~
2342   ~~*ProfilePackages* provide a mechanism to exchange profile libraries.~~

2343   ~~**Generalizations**~~

2344   ~~The *ProfilePackage* element inherits the attributes and/or associations of:~~

2345   • ~~*SCEProfile* (see the section entitled "SCEProfile" for more information).~~

2346   ~~**Properties**~~

2347   ~~The following table presents the additional attributes and/or associations for *ProfilePackage*:~~

Property/Association	Description
adornmentDefinition : AdornmentDefinition [0..*]	A set of *AdornmentDefinitions* contained within the package.
attributeType : AttributeType [0..*]	A set of *AttributeTypes* contained within the package.
profile : AdornmentProfile [0..*]	A set of *AdornmentProfiles* contained within the package.
profileAttributeDefinition : AdornmentAttributeDefinition [0..*]	A set of *AdornmentAttrributeDefinitions* contained within the package.

8.1058.11   Primitives

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

PPMN uses the four primitives shown in the figure in addition to other UML primitives.
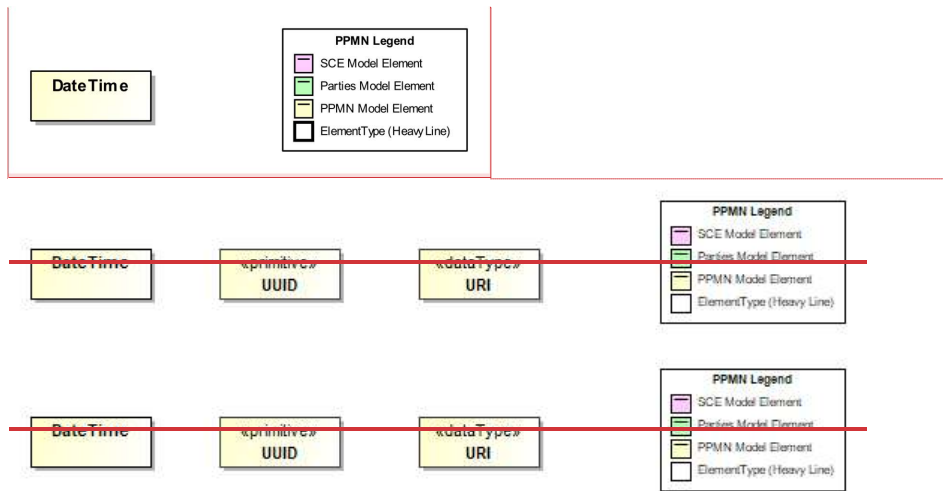
> **Commented [JB138]:** Figure updated to address PPMN-11/PPMN-157

Figure 49:   PPMN Primitives

8.105.18.11.1   DateTime

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

Generalizations

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

Properties

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

8.1068.12   VocabulariesKindSets

PPMNVocabularies 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 ~~The~~ *SemanticReferenceKind* 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.

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

Commented [JB139]: Diagram updated for PPMN-19/PPMN-83, PPMN-28/PPMN-84.
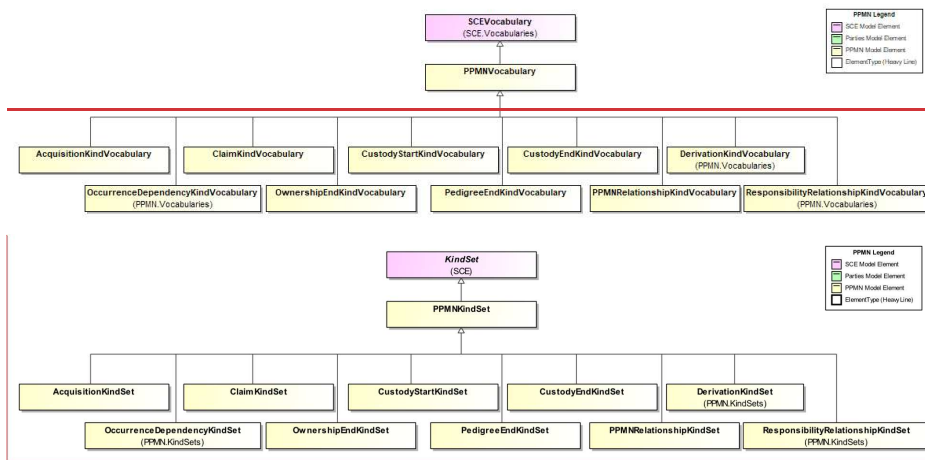
Figure 50:   ~~PPMNVocabulary~~PPMN KindSets

8.106.18.12.1   PPMNVocabularyPPMNKindSet

Commented [JB140]: All text changes in this section due to resolution for issue PPMN-28/PPMN-84.

PPMNVocabularies PPMNKindSet are lists of terms used as possible values for properties within PPMN. The terms are specializations of the *SemanticReference Kind* element that can be used to relate to the term to an external definition or meaning. The terms themselves do not represent the definitions or meanings but provide links to an external source. The ~~vocabulary~~KindSet mechanism is used to support extensibility of the specification.

Generalizations

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

- ~~SCEVocabulary~~ *KindSet* (see the ~~section entitled "SCEVocabulary"~~SCE specification for more information).

Properties

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

8.106.28.12.2   AcquisitionKind~~Vocabulary~~Set

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

Generalizations

The *AcquisitionKind~~Vocabulary~~Set* element inherits the attributes and/or associations of:

- *PPMNVocabulary PPMNKindSet* (see the section entitled ""~~PPMNVocabulary~~PPMNKindSet" for more information).

Properties

The following table presents the additional attributes and/or associations for *Acquisition~~KindVocabulary~~KindSet*:

~~Table 82.~~Table 81.　　　**Acquisition~~KindVocabulary~~KindSet Attributes and/or Associations**

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

~~8.106.3~~8.12.3　　　**Claim~~KindVocabulary~~KindSet**

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

Generalizations

The *Claim~~KindVocabulary~~KindSet* element inherits the attributes and/or associations of:

* *~~PPMNVocabulary~~PPMNKindSet* (see the section entitled "~~PPMNVocabulary~~PPMNKindSet" for more information).

Properties

The following table presents the additional attributes and/or associations for *Claim~~KindVocabulary~~KindSet*:

~~Table 83.~~Table 82.　　　**Claim~~KindVocabulary~~KindSet Attributes and/or Associations**

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

~~8.106.4~~8.12.4　　　**CustodyEnd~~KindVocabulary~~KindSet**

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

Generalizations

The *CustodyEnd~~KindVocabulary~~KindSet* element inherits the attributes and/or associations of:

* *~~PPMNVocabulary~~PPMNKindSet* (see the section entitled "~~PPMNVocabulary~~PPMNKindSet" for more information).

Properties

The following table presents the additional attributes and/or associations for *CustodyEnd~~KindVocabulary~~KindSet*:

Table 84.Table 83. **CustodyEndKindVocabularyKindSet Attributes and/or Associations**

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

8.106.58.12.5 CustodyStartKindVocabularyKindSet

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

Generalizations

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

- *PPMNVocabularyPPMNKindSet* (see the section entitled "PPMNVocabularyPPMNKindSet" for more information).

Properties

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

Table 85.Table 84. **CustodyStartKindVocabularySet Attributes and/or Associations**

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

8.106.68.12.6 DerivationKindVocabularySet

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

Generalizations

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

- *PPMNVocabularyPPMNKindSet* (see the section entitled "PPMNVocabularyPPMNKindSet" for more information).

Properties

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

Table 86.Table 85.   DerivationKind~~Vocabulary~~Set Attributes and/or Associations

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

2429

8.106.78.12.7   OccurrenceDependencyKind~~Vocabulary~~Set

2430

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

Generalizations

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

- *~~PPMNVocabulary~~ PPMNKindSet* (see the section entitled "~~PPMNVocabulary~~PPMNKindSet" for more information).

Properties

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

Table 87.Table 86.   ~~OccurrenceDependencyKindVocabulary~~ OccurrenceDependencyKindSet Attributes and/or Associations

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

2441

8.106.88.12.8   ~~OwnershipEndKindVocabulary~~OwnershipEndKindSet

2442

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

Generalizations

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

- *~~PPMNVocabulary~~ PPMNKindSet* (see the section entitled "~~""""""PPMNVocabulary~~PPMNSKindSet" for more information).

Properties

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

Table 88.Table 87.	OwnershipEndKindVocabulary OwnershipEndKindSet Attributes and/or Associations
Property/Association	**Description**
term : OwnershipEndKind [1..*]	A list of the terms representing valid *OwnershipEndKinds* within a **PPMN** Model.

2452

8.106.98.12.9 PedigreeEndKindVocabularyPedigreeEndKindSet

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

Generalizations

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

- *PPMNVocabulary* *PPMNKindSet* (see the section entitled "PPMNVocabularyPPMNKindSet" for more information).

Properties

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

8.106.108.12.10 PPMNRelationshipKindVocabularyPPMNRelationshipKindSet

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

Generalizations

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

- *PPMNVocabularyPPMNKindSet* (see the section entitled "PPMNVocabularyPPMNKindSet" for more information).

Properties

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

Table 89.Table 88.	PPMNRelationshipKindVocabulary PPMNRelationshipKindSet Attributes and/or Associations
Property/Association	**Description**
term : RelationshipKind [0..*]	A list of the terms representing valid *RelationshipKinds* within a **PPMN** model.

2474

8.106.118.12.11   ResponsibilityRelationshipKindVocabularyResponsibilityRelationshipKindSet

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

Generalizations

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

- *PPMNVocabulary PPMNKindSet* (see the section entitled "PPMNVocabularyPPMNKindSet" for more information).

Properties

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

Table 90.Table 89.   ResponsibilityRelationshipKindVocabulary ResponsibilityRelationshipKindSet Attributes and/or Associations

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

9   PPMN Library

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

9.1   AcquisitionKinds

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

The following figure presents the instances of the *AcquisitionKind* element that are `terms` for the `AcquisitionKindsVocabularyAcquisitionKindSet`:
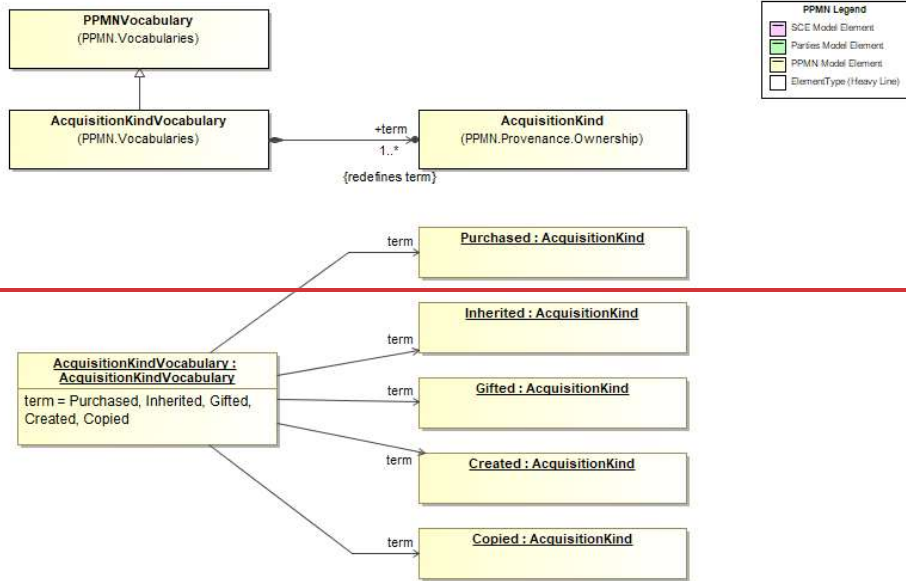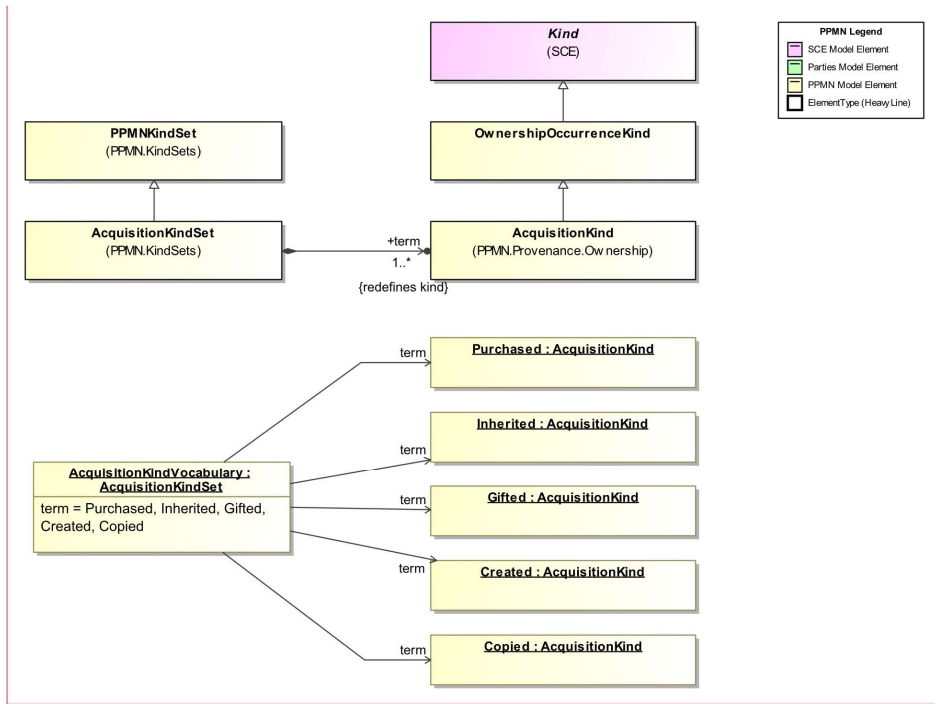
Commented [JB141]: Diagram updated for PPMN-19/PPMN-83, PPMN-28/PPMN-84.

2500

2501 **Figure 51: AcquisitionKinds**

2502 The following table provides a definition of the terms included in the *AcquisitionKinds* ~~Vocabulary~~set.

2503

~~Table 91.~~Table 90.   **AcquisitionKinds ~~Vocabulary~~KindSet**

#	Name	Documentation
1	~~AcquisitionKindVocabulary~~AcquisitionKindSet	A kind of ~~PPMNVocabulary~~ PPMNKindSet that includes terms that specify how a *ChainOfOwnership* 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.

#	Name	Documentation
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.

9.1.1 ~~AcquisitionKindVocabulary~~AcquisitionKindSet

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

9.1.2 Copied

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

9.1.3 Created

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

9.1.4 Gifted

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

9.1.5 Inherited

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

9.1.6 Purchased

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

9.2 ClaimKinds

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

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

PPMN Legend
- SCE Model Element
- Parties Model Element
- PPMN Model Element
- ElementType (Heavy Line)
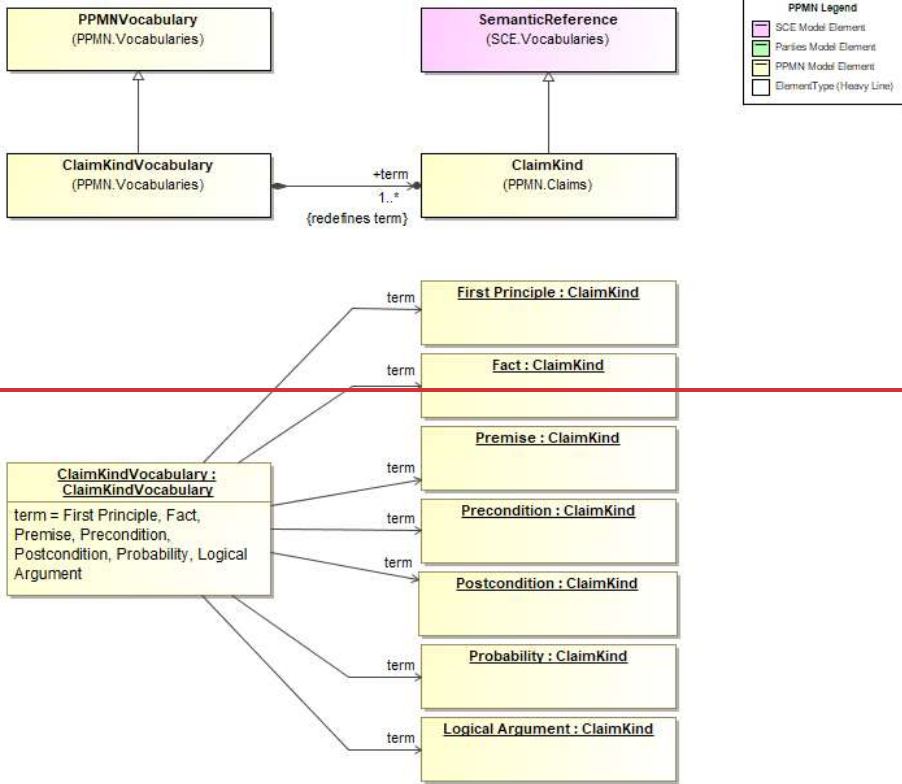
2523
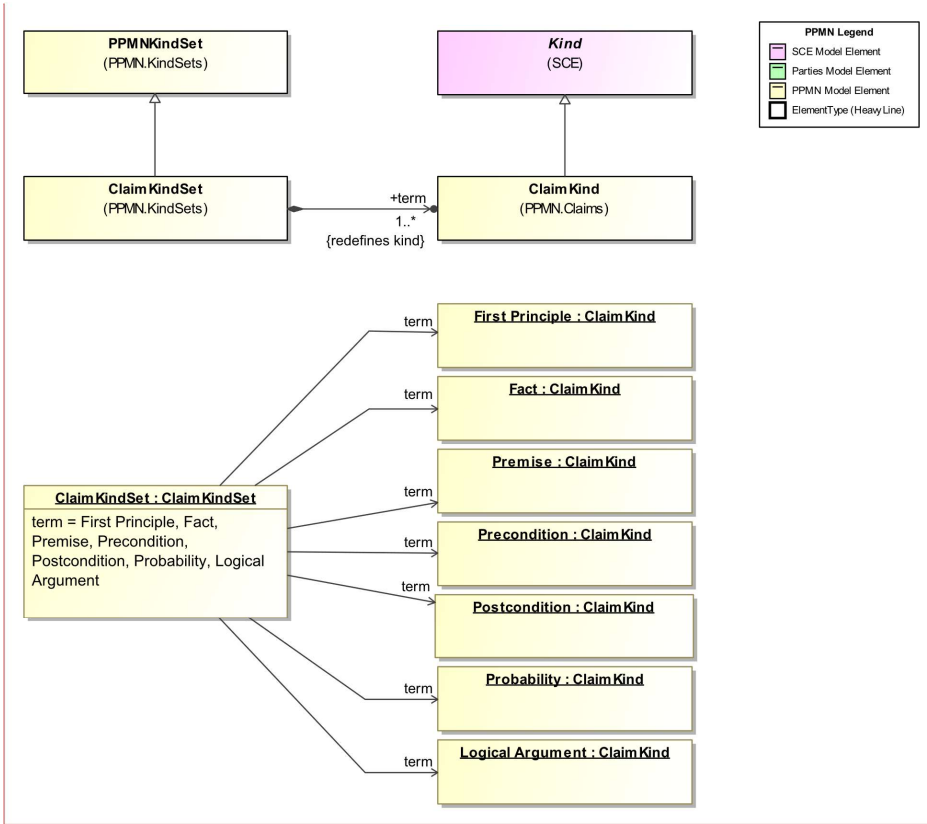
Commented [JB142]: Diagram updated for PPMN-19/PPMN-83, PPMN-28/PPMN-84.

Figure 52:   ClaimKinds

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

~~Table 92.~~Table 91.   **ClaimKinds ~~Vocabulary~~KindSet**

#	Name	Documentation
1	~~ClaimKindVocabulary~~ClaimKindSet	A~~n vocabulary~~ 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.

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

9.2.1 ~~ClaimKindVocabulary~~ClaimKindSet

A~~n vocabulary~~ set of terms that specify the kinds of claims may be made.

9.2.2 Fact

A basic assertion.

9.2.3 First Principle

A foundational assertion that is held as true.

9.2.4 Logical Argument

An assertion that is based on other assertions.

9.2.5 Postcondition

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

9.2.6 Precondition

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

9.2.7 Premise

An assertion that is used in a logical argument.

9.2.8 Probability

An assertion that indicates some degree of truth.

9.3 CustodyEndKinds

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

The following figure presents the instances of the *CustodyEndKind* element that are `terms` for the ~~CustodyEndKindsVocabulary~~`CustodyEndKindSet`:

PPMN Legend
- SCE Model Element
- Parties Model Element
- PPMN Model Element
- ElementType (Heavy Line)
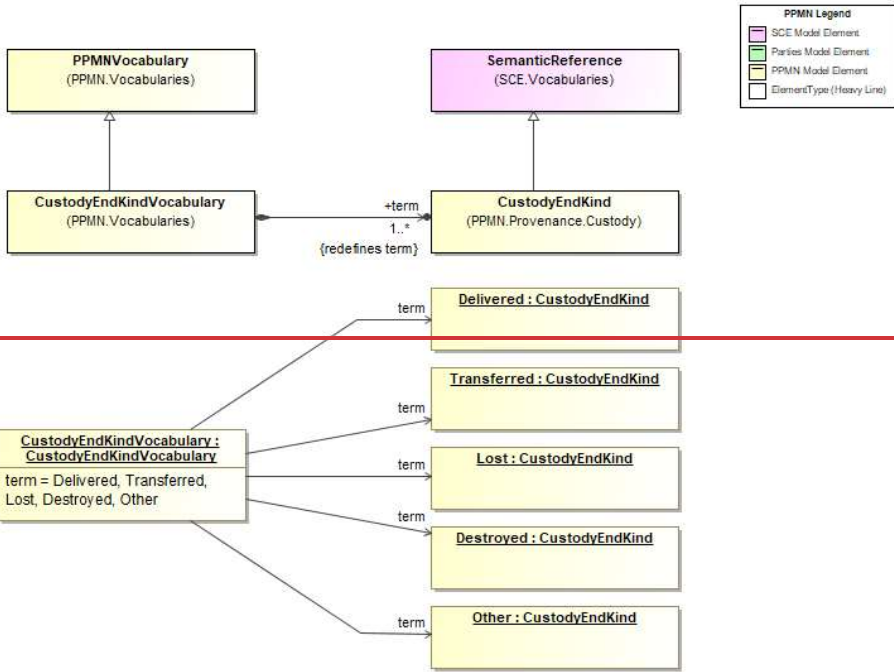
2551
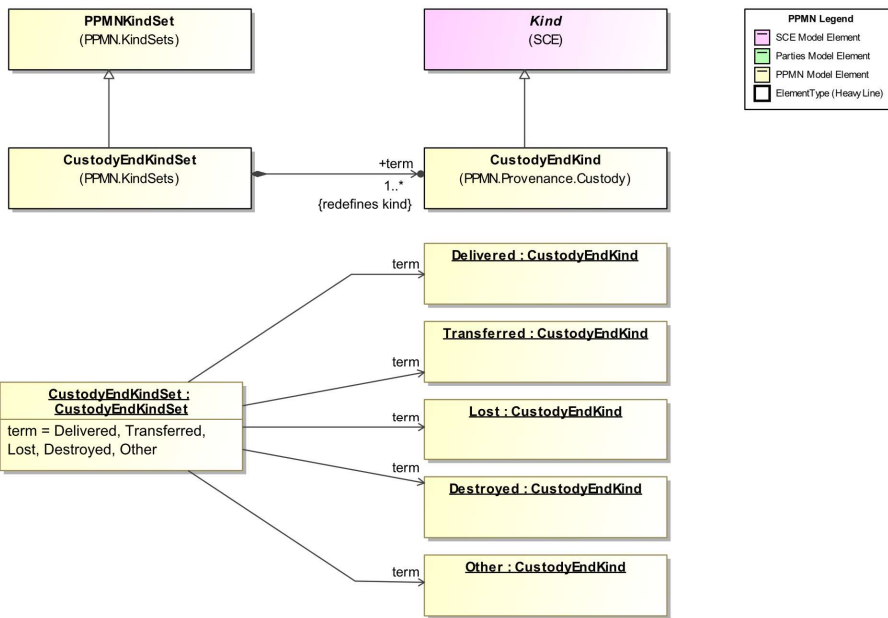
> **Commented [JB143]:** Diagram updated for PPMN-19/PPMN-83, PPMN-28/PPMN-84.

Figure 53:   CustodyEndKinds

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

~~Table 93.~~Table 92.   **CustodyEndKinds ~~Vocabulary~~KindSet**

#	Name	Documentation
1	~~CustodyEndKindVocabulary~~CustodyEndKindSet	A~~n vocabulary of~~ set terms that specify the kind of *CustodyOccurrence* that results in the end of a *ChainOfCustody*.
2	Delivered	An instance that specifies that an entity was delivered to some other *Party*.
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.

#	Name	Documentation
6	Transferred	An instance that specifies that an entity was transferred to some other *Party*.

9.3.1   ~~CustodyEndKindVocabulary~~**CustodyEndKindSet**

A ~~vocabulary~~ 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*.

9.3.3   Destroyed

An instance that specifies that an entity was destroyed.

9.3.4   Lost

An instance that specifies that an entity was lost.

9.3.5   Other

An instance that specifies that custody of an entity was relinquished in some other way.

9.3.6   Transferred

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

9.4   CustodyStartKinds

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

The following figure presents the instances of the *CustodyStartKind* element that are `terms` for the `CustodyStartKind`~~Sets~~~~Vocabulary~~:
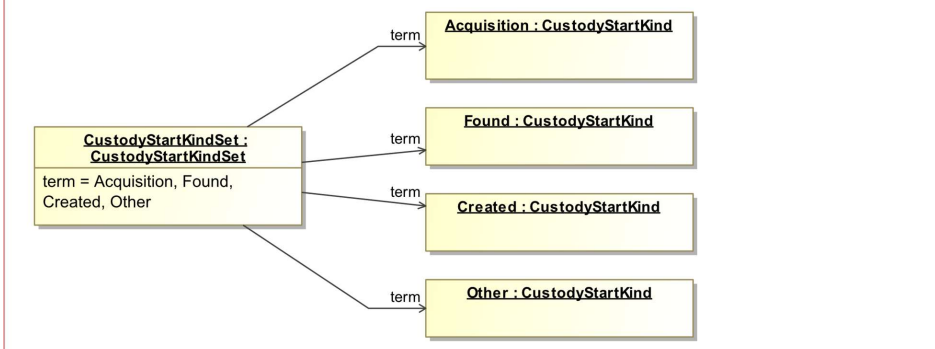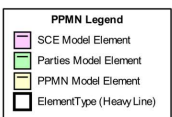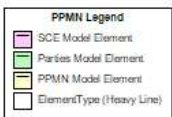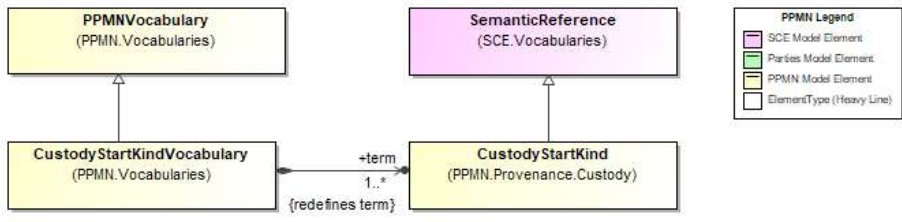
Commented [JB144]: Diagram updated for PPMN-19/PPMN-83, PPMN-28/PPMN-84.

2578 **Figure 54: CustodyStartKinds**

2579 The following table provides a definition of the terms included in the *CustodyStartKinds*
2580 setsetsetsetVocabularyset.

2581

Table 94.Table 93.　　CustodyStartKinds VocabularyKindSet

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

2582 ### 9.4.1　CustodyStartKindVocabularyCustodyStartKindSet

2583 A vocabulary set of terms that specify the kind of *CustodyOccurrence* that results in the start of a *ChainOfCustody*.

2584 ### 9.4.2　Acquisition

2585 An instance that indicates that a *Party* gains custody of an entity through some type of acquisition.

2586 ### 9.4.3　Created

2587 An instance that indicates that a *Party* gains custody of an entity by creation of the entity.

2588 ### 9.4.4　Found

2589 An instance that indicates that a *Party* gains custody of an entity when the entity is found.

2590 ### 9.4.5　Other

2591 An instance that indicates that a *Party* gains custody of an entity by some other event.
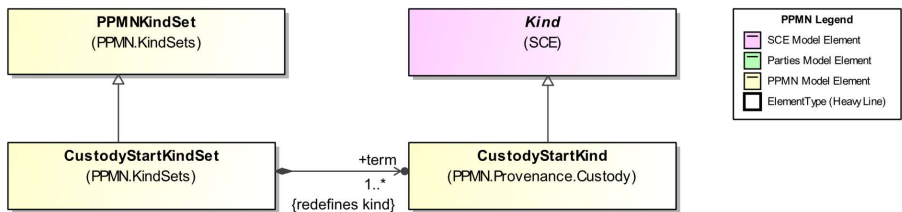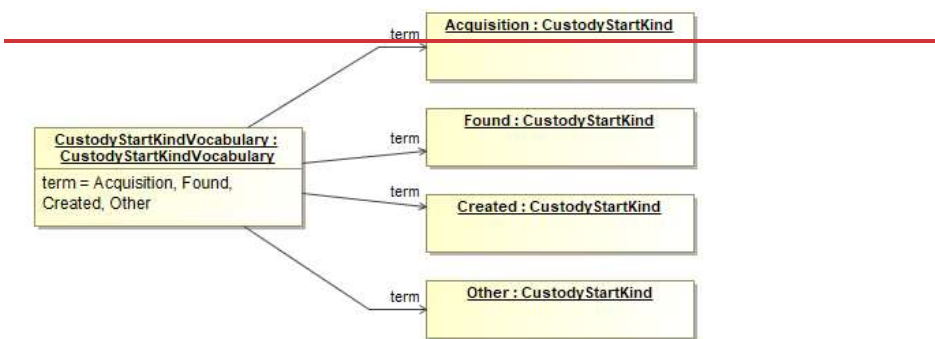
2592 ## 9.5　DerivationKinds

2593 The following table provides a definition of the terms included in the *DerivationKinds* Vocabularyset.
2594

Table 95.Table 94.　　DerivationtKinds VocabularyKindSet

#	Name	Documentation
1	DerivationKindSetsVocabulary	A vocabulary set of terms that specify the kind of derivation that exists between two *Entities*.
2	DerivedFrom	`DerivedFrom` indicates that source *EntityTypes* are derived in some way from `target` *EntityTypes*.
3	DescendantOf	`DescendantOf` indicates that `source` *EntityType* is a descendant of the `target` *EntityType*.

Commented [JB145]: Updated to address PPMN-35/PPMN-104.

#	Name	Documentation
44 ~~34~~	QuotedFrom	`QuotedFrom` indicates that `source` *EntityTypes* are quoted from `target` *EntityTypes*.
55 ~~45~~	RevisionOf	`RevisionOf` indicates that `source` *EntityTypes* are revisions of `target` *EntityTypes*.
66 ~~56~~	SourcedFrom	`SouredFrom` indicates that `source` *EntityTypes* are sourced from from `target` *EntityTypes* which in turn are produced by some party potentially with some special experience or knowledge.

2595   The following figure presents the instances of the *RelationshipKind* element that are `terms` for the
2596   ~~PPMNRelationshipKindsVocabulary~~`PPMNRelationshipKindSet`:
2597

2598
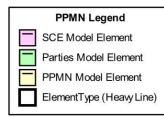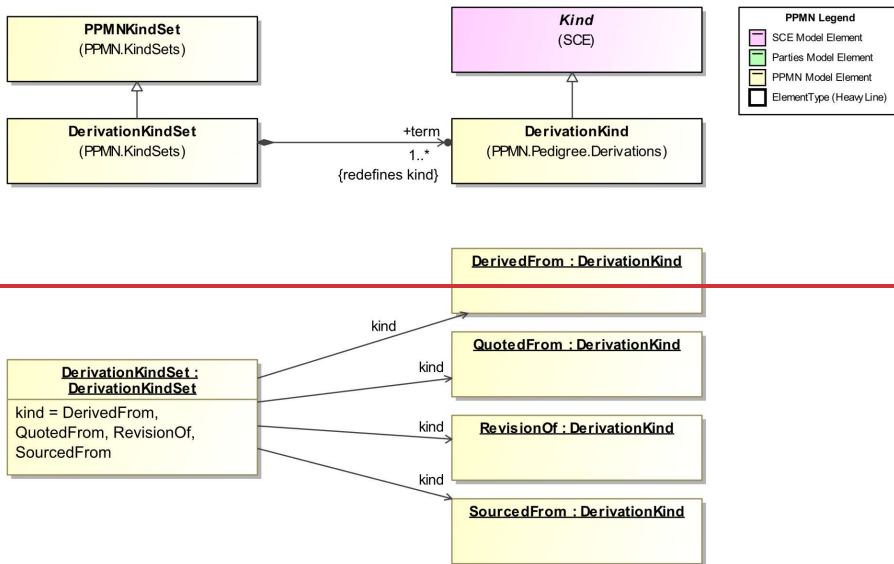
Figure 55:   DerivationKinds

9.5.1   ~~DerivationKindsVocabulary~~DerivationKindSet

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

9.5.2   DerivedFrom

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

9.5.3   DescendantOf

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

~~9.5.3~~9.5.4 QuotedFrom

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

~~9.5.4~~9.5.5 RevisionOf

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

~~9.5.5~~9.5.6 SourcedFrom

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

9.6   OccurrenceDependencyKinds

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

The following figure presents the instances of the *OccurrencDependencyKind* element that are `terms` for the `OccurrenceDependencyKind`~~SetsVocabulary~~:
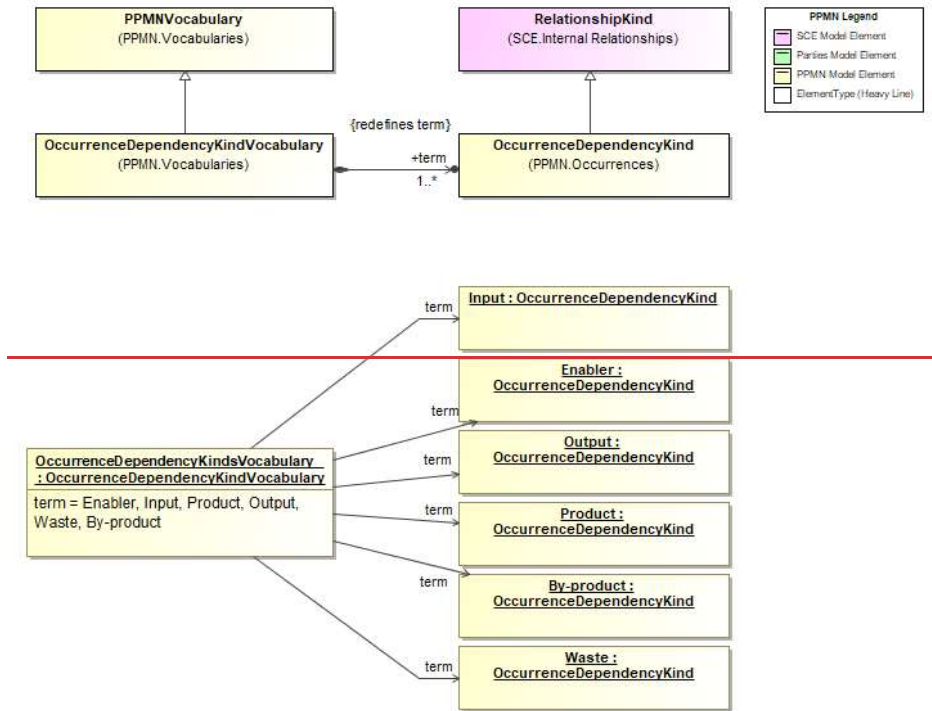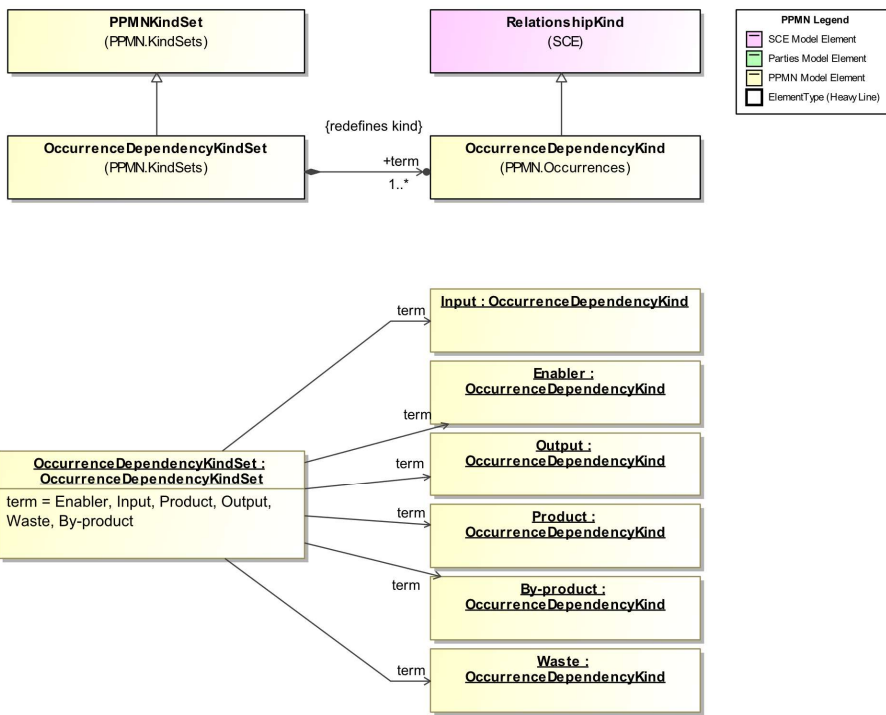
Figure 56: OccurrenceDependencyKinds

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

~~Table 96.~~Table 95.   ~~OccurrenceDependencyKinds~~ OccurrenceDependencyKinds KindSet ~~Vocabulary~~

#	Name	Documentation
1	OccurrenceDependencyKind~~SetsVocabulary~~	A set of terms that specify the kind of *OccurrenceDependency* that exists between two *Occurrences*.
2	By-product	`By-product` indicates that the source *Occurrence* produces or creates the `target` *Entity* as a by-product during the course of the *Occurrence*.

I	Name	Documentation
3	Enabler	`Enabler` indicates that the `source` *Occurrence* uses the `target` *Entity* in some way that enables the *Occurrence*. However, the *Entity* is not used or become a part of any of the products or by-products of the *Occurrence*.
4	Input	`Input` indicates that the `target` *Entity* is an input to the `source` *Occurrence* is an input during the course of the *Occurrence*.
5	Output	`Output` indicates that the `target` *Entity* is an output of some kind of the *Occurrence*..
6	Product	`Product` indicates that the `source` *Occurrence* produces or creates the `target` *Entity* during the course of the *Occurrence*. This is a more specific type of Output.
7	Waste	`Waste` indicates that the `source` *Occurrence* produces or creates the `target` *Entity* as waste during the course of the *Occurrence*. This is a more specific type of Output.

9.6.1 OccurrenceDependencyKindsVocabularyOccurrenceDependencyKindSet

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

9.6.19.6.2 By-product

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

9.6.29.6.3 Enabler

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

9.6.39.6.4 Input

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

9.6.49.6.5 Output

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

9.6.59.6.6 Product

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

9.6.69.6.7 Waste

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

9.7 OwnershipEndKinds

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

The following figure presents the instances of the *OwnershipEndKind* element that are terms for the OwnershipEndKind~~Sets~~~~Vocabulary~~:
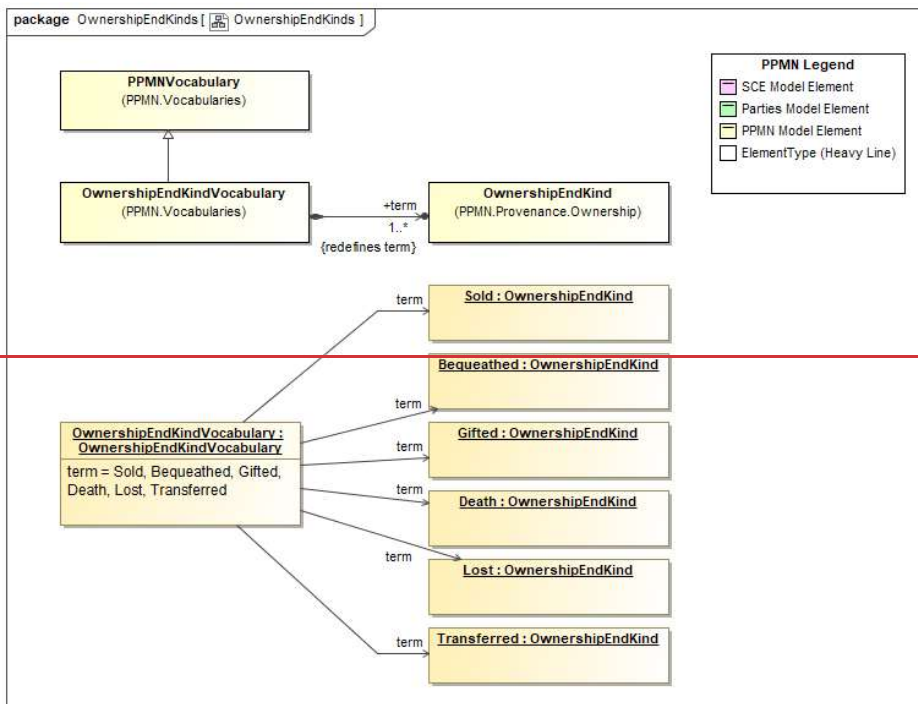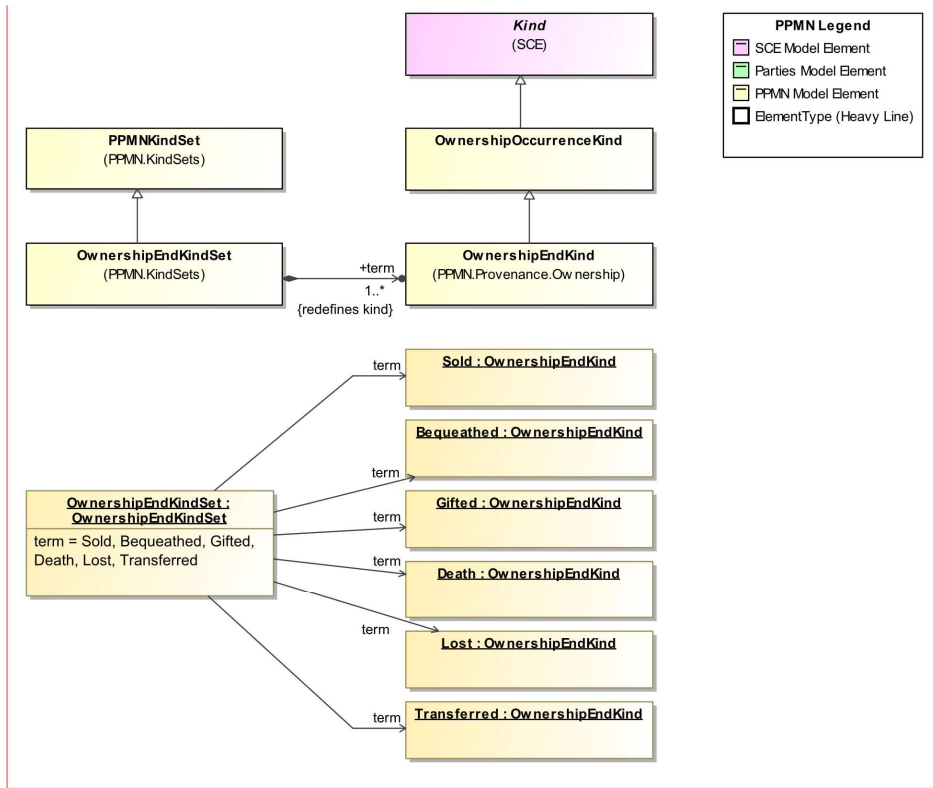
Figure 57:   OwnershipEndKinds

The following table provides a definition of the terms included in the *OwnershipEndKinds* ~~Vocabulary~~set.

~~Table 97.~~Table 96.   **OwnershipEndKinds ~~Vocabulary~~KindSet**

#	Name	Documentation
1	~~OwnershipEndKindVocabulary~~OwnershipEndKindSet	A ~~vocabulary~~ set of terms that specify how the *ChainOfOwnership* 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 *Party*.

#	Name	Documentation
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 *Party*.
7	Transferred	An instance that specifies that ownership of an entity was transferred to some other *Party*.

9.7.1 OwnershipEndKind~~VocabularySet~~

A ~~vocabulary~~set of terms that specify how the *ChainOfOwnership* was ended.

9.7.2 Bequeathed

An instance that specifies that an entity was bequeathed to some other party.

9.7.3 Death

An instance that specifies that an entity died.

9.7.4 Gifted

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

9.7.5 Lost

An instance that specifies that an entity was lost.

9.7.6 Sold

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

9.7.7 Transferred

An instance that specifies that ownership of an entity was transferred to some other *Party*.

9.8 PPMNRelationshipKinds

The *PPMNRelationshiipKinds* library contains instances that represent the standard types of relationships between **PPMN** elements. This library extends the **SCE** RelationshipKinds library of terms to add `Transition`. These elements are instances of **SCE** *RelationshiipKind*. The ~~vocabulary~~set can be extended with additional instances of *RelationshiipKind* or a specialization thereof.

The following figure presents the instances of the *RelationshipKind* element that are `terms` for the PPMNRelationshipKinds~~VocabularySet~~:

PPMN Legend
- SCE Model Element
- Parties Model Element
- PPMN Model Element
- ElementType (Heavy Line)

PPMNVocabulary
(PPMN.Vocabularies)

PPMNRelationshipKindVocabulary
(PPMN.Vocabularies)

+term
0..*
{redefines term}

RelationshipKind
(SCE.Internal Relationships)

Transition : RelationshipKind

Composition : RelationshipKind
(SCELibrary.RelationshipKinds)

Containment : RelationshipKind
(SCELibrary.RelationshipKinds)

PPMNRelationshipKinds :
PPMNRelationshipKindVocabulary
term = Composition, Containment,
Correlation, Dependency,
Miscellaneous, Reference,
Generalization, Transition

Correlation : RelationshipKind
(SCELibrary.RelationshipKinds)

Dependency : RelationshipKind
(SCELibrary.RelationshipKinds)

Miscellaneous : RelationshipKind
(SCELibrary.RelationshipKinds)

Reference : RelationshipKind
(SCELibrary.RelationshipKinds)

Generalization : RelationshipKind
(SCELibrary.RelationshipKinds)

2683

Figure 58: PPMNRelationshipKinds

The following table provides a definition of the terms included in the *PPMNRelationshipKinds* Vocabulary.

~~Table 98.~~Table 97.　　　**PPMNRelationshipKinds ~~Vocabulary~~KindSet**

#	Name	Documentation
1	PPMNRelationshipKinds	A kind of ~~PPMNVocabulary~~ PPMNKindSet that includes terms that specify the kind of relationship between two **PPMN** elements.

#	Name	Documentation
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 *RelationshipKind* 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.

9.8.1   PPMNRelationshipKinds

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

9.8.2   Transition

`Transition` indicates that "flow" or sequencing moves from the `source` element to the `target` element.

9.8.3   Additional Terms from SCE

9.8.3.1   Reference

`Reference` indicates that `source` element references the `target` element.

9.8.3.2   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 *RelationshipKind* instances.

9.8.3.3 Composition

`Composition` indicates that the `source` element is composed of, in part, the `target` element. Other elements could be included in this composition.

9.8.3.4 Dependency

`Dependency` indicates that `target` element is dependent in some way on the `source` element.

9.8.3.5 Containment

`Containment` indicates that the `source` element is a container for the `target` element.

9.8.3.6 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.

9.8.3.7 Generalization

`Generalization` indicates that the `source` element is a generalization of the `target` element (which is based on and extends the `source`).

9.9 ResponsibilityRelationshipKinds

The following figure presents the instances of the *ResponsibilityRelationshipKind* element that are `terms` for the ResponsibilityRelationshipKinds~~Vocabulary~~Set:
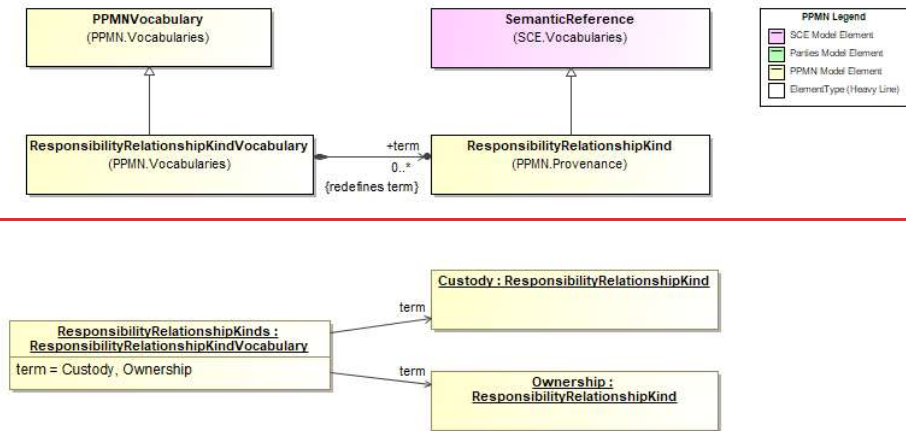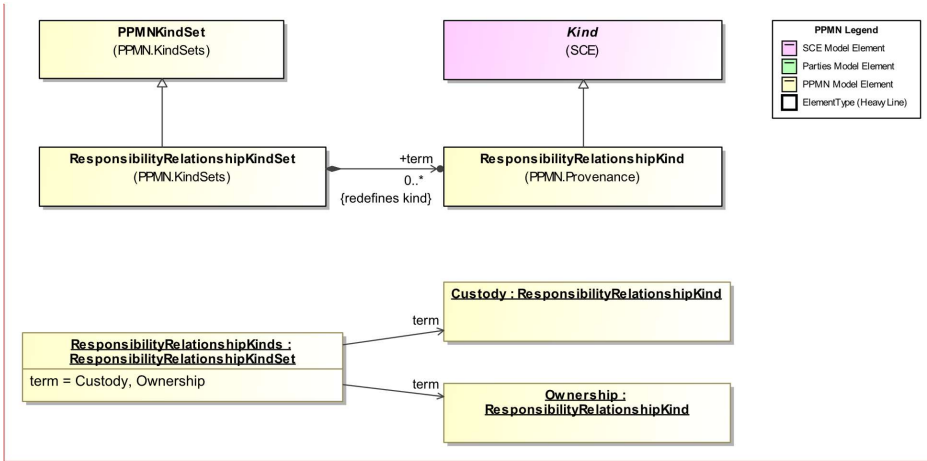
Commented [JB155]: Diagram updated for PPMN-19/PPMN-83, PPMN-28/PPMN-84.

Figure 59: ResponsibilityRelationshipKinds

The following table provides a definition of the terms included in the *PPMNRelationshipKinds* ~~Vocabulary~~set.

~~Table 99.~~Table 98.   ResponsibilityRelationshipKinds ~~Vocabulary~~KindSet

#	Name	Documentation
1	ResponsibilityRelationshipKinds	A kind of ~~PPMNVocabulary~~ PPMNKindSet that includes terms that specify the kind of responsibility a *Party* has with respect to an *Entity*.
2	Custody	`Custody` indicates that the `source` element has custody (immediate charge of or control over) ~~of~~ the `target` element.
3	Ownership	`Ownership` indicates that the `source` element owns the `target` element.

Commented [JB156]: Text updated to address PPMN-43/PPMN-138

9.9.1   ResponsibilityRelationshipKinds

A kind of ~~PPMNVocabulary~~ PPMNKindSet that includes terms that specify the kind of responsibility a *Party* has with respect to an *Entity*.

9.9.2   Custody

`Custody` indicates that the `source` element has custody (immediate charge of or control over) ~~of~~ the `target` element.

Commented [JB157]: Text updated to address PPMN-43/PPMN-138

9.9.3   Ownership

`Ownership` indicates that the `source` element owns the `target` element.

10 Parties Model

This section defines the semantic elements of the **Parties** Metamodel. The main topics are organized into Core Elements, Locations, Packages, Vocabularies, and Primitives.

10.1 Core

The Core elements of the **Parties** metamodel contains elements related to people, organizations, roles, automated systems and the relationships between them. The elements are separated into Instances and Types. The Instances section defines elements that enable modeling specific Parties (i.e., people, organizations, positions and roles and their interrelationships). The Types section defines elements that enable modeling the kinds of Parties that are of interest in some context.

10.1.1 Instances

The Core.Intances section of the **Parties** metamodel contains elements related to people, organizations, roles, automated systems and the relationships between them. These elements enable modeling specific Parties. Elements in the Core.Instances section are generally specializations of **SCE** *TypedElements* and as such may have an ElementType specified. The corresponding types are described below in the Core.Types section.

A *Party* is an abstract concept intended to generalize the notions of *Organization*, *Person*, *Position* or *Non-Human Agent* - essentially things that can be proactive and play a part in a business context. This generalization acknowledges the fact that many of the same business interactions can be defined regardless of the particular type of party involved. For instance, in the sale of a parcel of land, the seller might be a *Person* or an *Organization* or even a *Position* in an *Organization* wherein that *Position* is responsible for handling real estate transactions. Likewise for the buyer. The *Party* pattern captures this notion in a succinct manner that has broad applicability.

2752
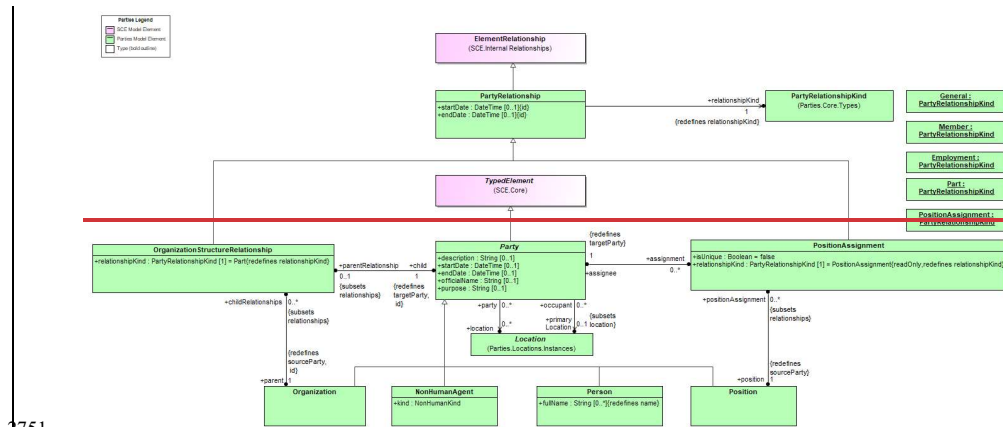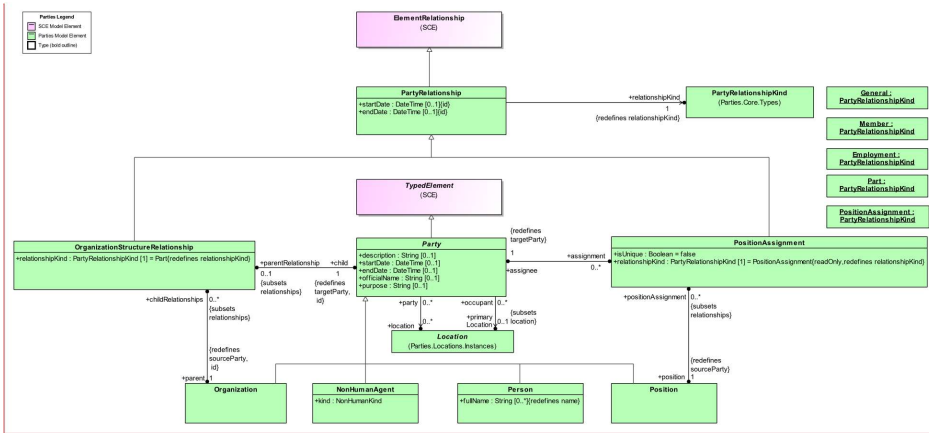
Figure 60: Parties

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*.

2759
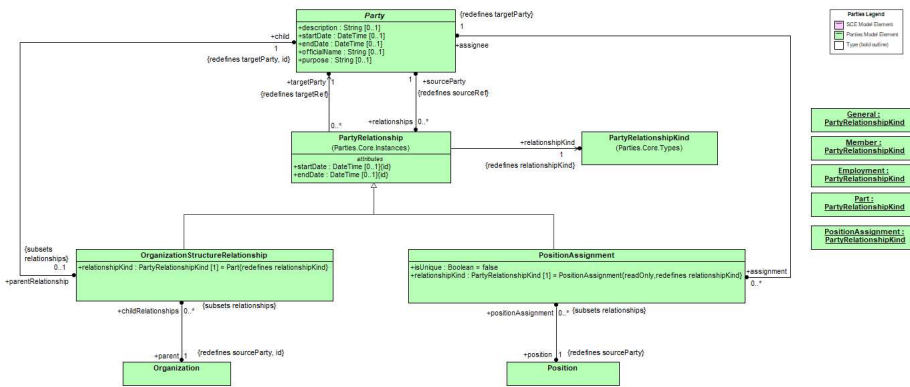
2760

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*.

2764
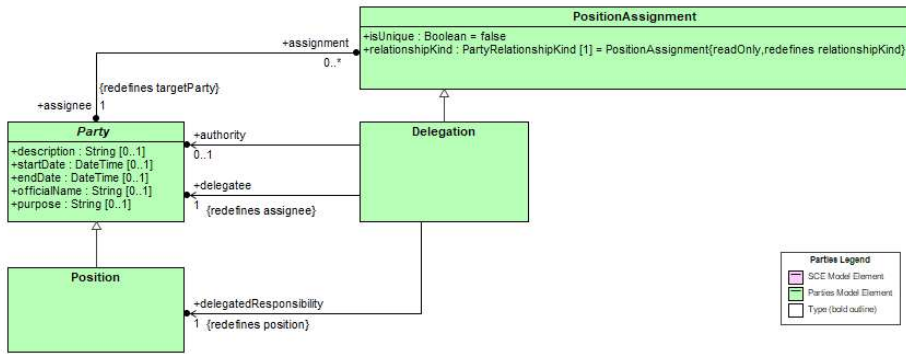
2765

Figure 62:   Delegation

PartyRoles represent a role that a *Party* may play in some context. For instance, in the sale of a parcel of land, the Seller might be a *Person* or an *Organization* or even a *Position* in an *Organization* wherein that *Position* is responsible for handling real estate transactions. Likewise for the buyer. The *PartyRole* captures this notion in a succinct manner that has broad applicability.

2771
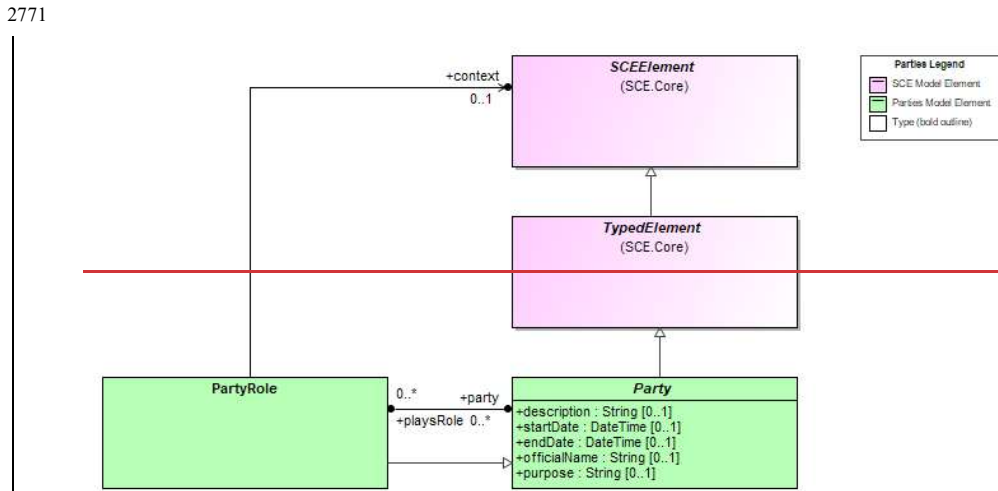
2772
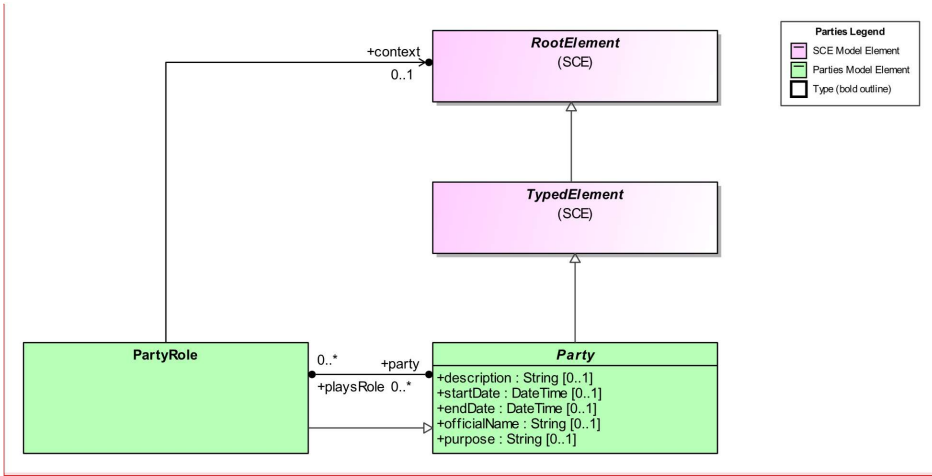
> **Commented [JB159]:** Diagram updated for PPMN-19/PPMN-83

2773

Figure 63:   Party Role

This diagram shows the mapping of *Party* and its specializations to *PartyType* and its specializations.

2776
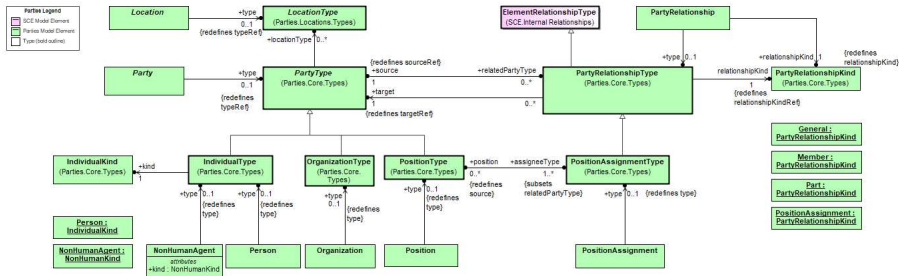
2777

Figure 64:   Parties and Party Types

10.1.1.1 Delegation

A kind of *PositionAssignment* relationship that states that one *Party* has been assigned a set of responsibilities by some authority.

Generalizations

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

- *PositionAssignment* (see the section entitled "PositionAssignment" for more information).

Properties

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

Table 100.Table 99.   Delegation Attributes and/or Associations

Property/Association	Description
authority : Party [0..1]	The *Party* on whose authority the *Delegation* was made.
delegatedResponsibility : Position [1]	The responsibilities, stated as a *Role*, that are being delegated.
delegatee : Party [1]	The *Party* to whom the *Role* was delegated.

2787

2788 **10.1.1.2 NonHumanAgent**

2789 Some type of automated system.

2790 **Generalizations**

2791 The *NonHumanAgent* element inherits the attributes and/or associations of:

2792 • *Party* (see the section entitled "Party" for more information).

2793 **Properties**

2794 The following table presents the additional attributes and/or associations for *NonHumanAgent*:

Table 101.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 *Automation*.

2795

2796 **10.1.1.3 Organization**

2797 *Organization* is used to represent a group of *Parties.* The group may be a company, a department within a company,
2798 a club, a consortium, or some other group.

2799 **Generalizations**

2800 The *Organization* element inherits the attributes and/or associations of:

2801 • *Party* (see the section entitled "Party" for more information).

2802 **Properties**

2803 The following table presents the additional attributes and/or associations for *Organization*:

Table 102.Table 101.   Organization Attributes and/or Associations

Property/Association	Description
childRelationships : OrganizationStructureRelationship [0..*]	A set of relationships to the members of the *Organization*.

type : OrganizationType [0..1]	The class that provides a specification of the *Organization*.

2804

10.1.1.4 OrganizationStructureRelationship

2805

2806 A specialization of PartyRelationship used to indicate internal structural relationships of a *Party*.

Generalizations

2807

2808 The *OrganizationStructureRelationship* element inherits the attributes and/or associations of:

2809 • *PartyRelationship* (see the section entitled "PartyRelationship" for more information).

Properties

2810

2811 The following table presents the additional attributes and/or associations for *OrganizationStructureRelationship*:

~~Table 103.~~Table 102.   **OrganizationStructureRelationship Attributes and/or Associations**

Property/Association	Description
child : Party [1]	The *Party* that is a member of the `organization`.
parent : Organization [1]	The *Organization* in which the *Party* is a member.
relationshipKind : PartyRelationshipKind [1] default: Part	The kind of structural relationship an Organization has with another Party.

2812

10.1.1.5 Party

2813

2814 *Party* is an abstract concept representing a *Person*, *Role*, *Organization*, or other entity involved in some activity,
2815 interaction or endeavor.

Generalizations

2816

2817 The *Party* element inherits the attributes and/or associations of:

2818 • **SCE** *TypedElement* (see the ~~section~~ **SCE** specification for more information).

Commented [JB160]: Text updated for PPMN-19/PPMN-83.

Properties

2819

2820 The following table presents the additional attributes and/or associations for *Party*:

~~Table 104.~~Table 103.   **Party Attributes and/or Associations**

Property/Association	Description
assignment : PositionAssignment [0..*]	A relationship indicating a *Position* to which the *Party* has been assigned.
description : String [0..1]	A textual description of the *Party*.

Property/Association	Description
endDate : DateTime [0..1]	The effective end date of the *Party*.
location : Location [0..*]	The location of the *Party*.
officialName : String [0..1]	The official name of the *Party*.
parentRelationship : OrganizationStructureRelationship [0..1]	A set of relationships to the *Organizations* in which the *Party* has membership.
playsRole : PartyRole [0..*]	The roles played by a *Party*.
primaryLocation : Location [0..1]	The primary location of the *Party*.
purpose : String [0..1]	The purpose of the *Party* with respect to the pedigree and/or provenance context.
relationships : PartyRelationship [0..*]	*PartyRelationships* in which the Party is involved.
startDate : DateTime [0..1]	The effective start date of the *Party*.
type : PartyType [0..1]	The class that provides a specification of the *Party*.

2821

10.1.1.6 PartyRelationship

2822

2823 A kind of *ElementRelationshiip* that indicates a relationship between two *Parties*.

2824 **Generalizations**

2825 The *PartyRelationship* element inherits the attributes and/or associations of:

2826 • *ElementRelationship* (see the ~~section entitled "ElementRelationship"~~**SCE specification** for more
2827 information).

> **Commented [JB161]:** Text updated for PPMN-19/PPMN-83.

2828 **Properties**

2829 The following table presents the additional attributes and/or associations for *PartyRelationship*:

~~Table 105.~~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 Parties.
sourceParty : Party [1]	The source *Party* of the relationship.
startDate : DateTime [0..1]	The effective start date of the relationship.
targetParty : Party [1]	The target *Party* of the relationship.
type : PartyRelationshipType [0..1]	The class that provide a specification of the *PartyRelationship*.

2830

2831 10.1.1.7 PartyRole

2832 A role played by a *Party* in some context. For instance, a Buyer or a Supplier.

Generalizations

2833
2834 The *PartyRole* element inherits the attributes and/or associations of:

2835 • *Party* (see the section entitled "Party" for more information).

Properties

2836
2837 The following table presents the additional attributes and/or associations for *PartyRole*:

Table 106.Table 105.   PartyRole Attributes and/or Associations

Property/Association	Description
context : BaseElementBaseElementBaseElementBaseElementBaseElementSCEElementBaseElement [0..1]	The context in which the *Party* plays the role.
party : Party [0..*]	The *Party* that plays the role.
type : PartyRoleType [0..1]	The class that provides a specification of the *PartyRole*.

Commented [JB162]: Text updated for PPMN-19/PPMN-83

2838

2839 10.1.1.8 Person

2840 An individual homo sapiens.

Generalizations

2841
2842 The *Person* element inherits the attributes and/or associations of:

2843 • *Party* (see the section entitled "Party" for more information).

Properties

2844
2845 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 *Person*.
type : IndividualType [0..1]	The class that provides a specification of the *Person*.

2846

2847 **10.1.1.9　Position**

2848 A Position is a formally defined role in an *Organization* filled by some *Person*. *Positions* are often associated with a
2849 set of responsibilities in some context.

2850 Examples of *Positions* include Chief Executive Officer or Technical Staff Member.

2851

2852 **Generalizations**

2853 The *Position* element inherits the attributes and/or associations of:

2854 • *Party* (see the section entitled "Party" for more information).

2855 **Properties**

2856 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 *PositionAssignment* that indicates the *Party* that fills the *Position*.
type : PositionType [0..1]	The class that provides a specification of the *Position*.

2857

2858 **10.1.1.10 PositionAssignment**

2859 *PositionAssignment* indicates a *Party* is assigned to a particular *Position* for a particular period of time.

2860 **Generalizations**

2861 The *PositionAssignment* element inherits the attributes and/or associations of:

2862 • *PartyRelationship* (see the section entitled "PartyRelationship" for more information).

2863 **Properties**

2864 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 *Party* that fills or filled the *Position*.
isUnique : Boolean [] default: false	A boolean stating whether only one *Party* filled a particular Role during that particular date range.

position : Position [1]	The *Position* filled by the noted *Party*.
relationshipKind : PartyRelationshipKind [1] default: PositionAssignment	The kind of relationship between an Organization and a Position within that Organization.
type : PositionAssignmentType [0..1]	The class that provides a specification of the *PositionAssignment*.

2865

10.1.2   Types

2867 The Core.Types section of the **Parties** metamodel contains elements related to the kinds of people, organizations,
2868 roles, automated systems and the relationships between them that are of interest in some context. These elements
2869 enable modeling kinds of Parties rather than particular Parties. Elements in the Core.Types section are generally
2870 specializations of **SCE** *ElementTypes* and as such provide a specification Parties to be created using elements in the
2871 Core.Instances section described above.

2872 *PartyTypes* define the types or classifications of *Parties*. *PartyTypes* provide the ability to specify or "configure"
2873 organizational structures for different kinds of parties such as companies, non-profits, community organizations
2874 organizsations and many others. *PartyType* configurations can be used to provide a constraint mechanism on the
2875 *Parties* created in some context though the Party metamodel does not require their use.

2876 While *PartyType* itself is abstract, the Party metamodel includes the concrete specializations *OrganizationType*,
2877 *IndividualType*, and *PositionType*. These types correspond to the concrete specializations of *Party* where
2878 IndividualType is used as the type for *Person*, *Automation*, and *SoftwareAgent* with the `kind` property set
2879 appropriately,

2880 *PartyRelationshipTypes* capture the possible relationships between *PartyTypes*. *PartyRelationshipTypes* have a
2881 *PartyRelationshipKind* that specifies the kind of relationship: Part, Member, Assignment, or General. (See
2882 *PartyRelationshipKind*.) *PositionAssignmentType* captures the particular relationship type wherein one or more
2883 *PartyTypes* are expected to fill (or be assigned to) a particular *PositionType*.

2884
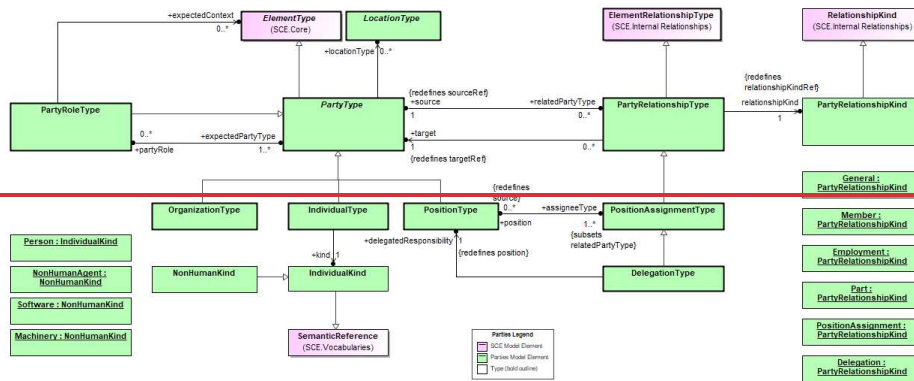
2885
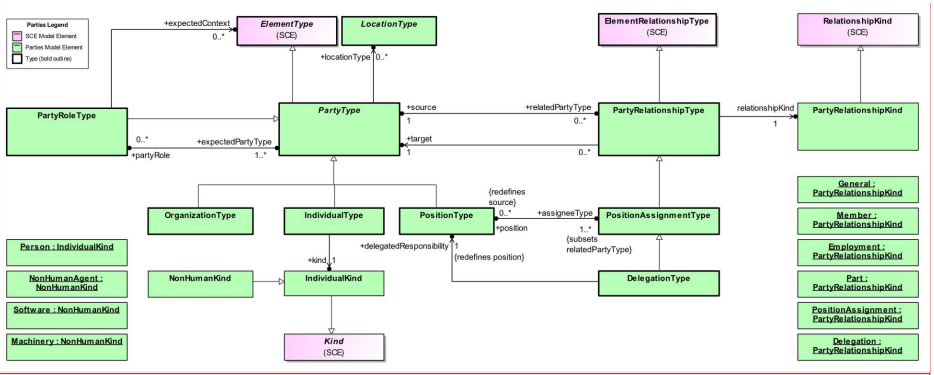
2886

Figure 65: Party Types

PartyRoles define the types or classifications of the roles that may be played by one or more kinds of *Parties* (i.e., *PartyTypes*) in some context. The `expectedPartyType` property specifies which *PartyTypes* are expected to play *PartyRoles* of that *PartyRoleType*.

2891
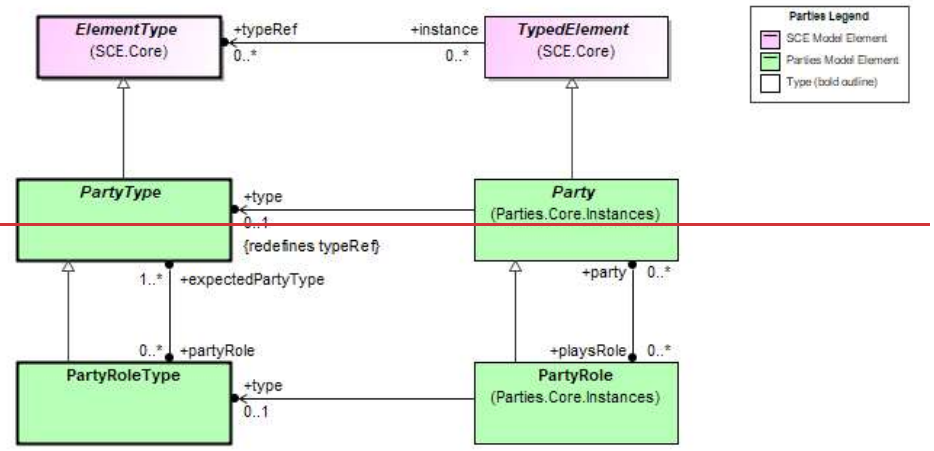
2892
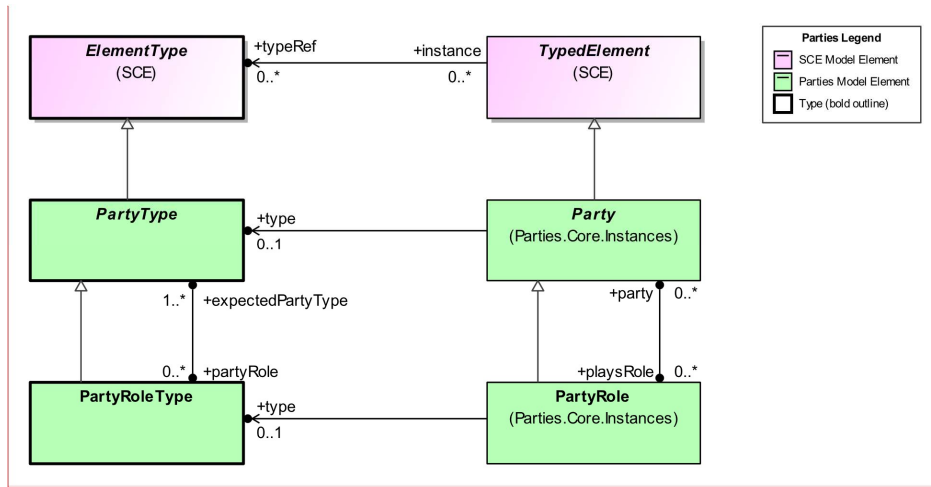
Figure 66:   Party Role Type

Delegation captures the notion that a Party may assign a set of responsibilities to another party. *DelegationType* supports the ability to state that the responsibilities associated with a *PositionType* may be delegated to particular *PartyTypes* on the authority of some *PartyType*.

Figure 67:   Delegation Types

10.1.2.1 DelegationType

DelegationType indicates a particular *PartyType* is delegated responsibility for particular *PositionType* by an authority.

Generalizations

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

- *PositionAssignmentType* (see the section entitled "PositionAssignmentType" for more information).

Properties

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

Table 110.Table 109.　　DelegationType Attributes and/or Associations

Property/Association	Description
authority : PartyType [0..1]	The *PartyType* expected to be the authority by which the delegation approved.
delegatedResponsibility : PositionType [1]	The set of responsibilities as defined by a *PositionType* that may be delegated.
delegatee : PartyType [1]	The *PartyType* to whom the responsibilities are expected to be delegated.

10.1.2.2 IndividualKind

IndividualKind is a kind specialization of *SemanticReferenceKind* that serves as the foundation for terms in a *PartyVocabulary PartyKindSet* that is used to specify the kinds of *IndividualTypes* in a **Parties** model. Instead of being defined a fixed enumerated list, the kinds are defined through instances of *IndividualKind* present in the *IndividualKinds* library. The instances defined in that library SHALL be included in any **Parties** implementation. However, the implementation can allow additional kinds of individuals through the addition of new instances of *IndividualKind* in the *IndividualKinds* library.

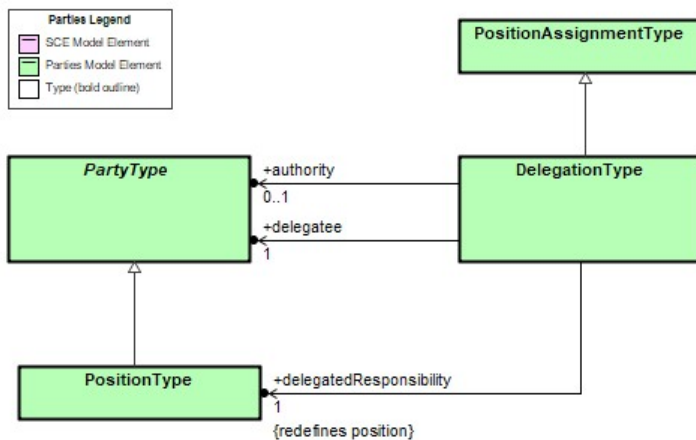**Generalizations**

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

- *SemanticReferenceKind* (see the **SCE** specification section entitled "SemanticReference" for more information).

Properties

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

10.1.2.3 IndividualType

A kind of *PartyType* representing the type or classification of a *Party* of interest that is an individual such as a *Person*, *Automation*, or *SoftwareAgent*.

Generalizations

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

- *PartyType* (see the section entitled "PartyType" for more information).

Properties

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

Table 111.Table 110. **IndividualType Attributes and/or Associations**

Property/Association	Description
kind : IndividualKind [1]	An instance that indicates the kind of individual the *IndividualType* represents.

2932

10.1.2.4   NonHumanKind

NonHumanKind is a kind of *IndividualKind* that serves as the foundation for `terms` in a *PartyVocabulary* that is used to specify the kinds of *NonHumanAgents* in a **Parties** model. Instead of being defined as a fixed enumerated list, the kinds are defined through instances of *NonHumanKind* present in the *IndividualKinds* library. The instances defined in that library SHALL be included in any **Parties** implementation. However, the implementation can allow additional kinds of individuals through the addition of new instances of *NonHumanKind* in the *IndividualKinds* library.

Generalizations

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

- *IndividualKind* (see the section entitled "IndividualKind" for more information).

Properties

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

10.1.2.5   OrganizationType

A kind of *PartyType* that represents the type or classification of an *Organization*.

Generalizations

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

- *PartyType* (see the section entitled "PartyType" for more information).

Properties

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

10.1.2.6   PartyRelationshipKind

PartyRelationshipKind is a specialization of *RelationshipKind* that serves as the foundation for `terms` for a ~~PartiesVocabulary~~ *PartiesKindSet* that is used to specify the kind of relationship that exists between two *PartyTypes* related by a *PartyRelationshipType*. Instead of being defined a fixed enumerated list, the kinds are defined through instances of *PartyRelationshipKind* present in the *PartyRelationshipKinds* library. The instances defined in the **Parties Library** SHALL be included in any **Parties** implementation. However, the implementation can allow additional kinds of relationship types through the addition of new instances of *PartyRelationshipKind* in the *PartyRelationshipKinds* library.

Generalizations

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

- *RelationshipKind* (see the section entitled "RelationshipKind" for more information).

Properties

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

Commented [JB166]: Text updated for PPMN-29/PPMN-85.

10.1.2.7 PartyRelationshipType

A kind of *ElementRelationshiip* that indicates a relationship between two *PartyTypes*.

Generalizations

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

- *ElementRelationshipType* (see the ~~section entitled "ElementRelationshipType"~~**SCE** specification for more information).

> **Commented [JB167]:** Text updated for PPMN-19/PPMN-83.

Properties

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

~~Table 112.~~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 *PartyType* of the relationship.
target : PartyType [1]	The target *PartyType* of the relationship.

10.1.2.8 PartyRoleType

A type or classification of a role that may be played by a particular *PartyType* in some context. For instance, a Buyer or a Supplier.

Generalizations

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

- *PartyType* (see the section entitled "PartyType" for more information).

Properties

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

~~Table 113.~~Table 112.   **PartyRoleType Attributes and/or Associations**

Property/Association	Description
expectedContext : ElementType [0..*]	The context in which instances of the *PartyRoleType* are expected to occur.
expectedPartyType : PartyType [1..*]	The type of *Party* that is expected to play the role specified by the *PartyRoleType*.

10.1.2.9 PartyType

An abstract class representing the type or classification of a *Party* of interest.

Generalizations

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

2987   • **SCE** *ElementType* (see the ~~section~~ SCE specification for more information).

Commented [JB168]: Text updated for PPMN-19/PPMN-83.

2988 **Properties**

2989 The following table presents the additional attributes and/or associations for *PartyType*:

~~Table 114.~~Table 113.   **PartyType Attributes and/or Associations**

Property/Association	Description
locationType : LocationType [0..*]	The type of *Location* at which the instances of the *PartyType* are expected to be located.
partyRole : PartyRoleType [0..*]	The type(s) of roles that *Parties* of type *PartyType* are expected to play.
relatedPartyType : PartyRelationshipType [0..*]	The related *PartyType* of a relationship.

2990

2991 **10.1.2.10 PositionAssignmentType**

2992 *PositionAssignmentType* indicates a particular *PartyType* is expected to fill particular *PositionType*.

2993 **Generalizations**

2994 The *PositionAssignmentType* element inherits the attributes and/or associations of:

2995   • *PartyRelationshipType* (see the section entitled "PartyRelationshipType" for more information).

2996 **Properties**

2997 The following table presents the additional attributes and/or associations for *PositionAssignmentType*:

~~Table 115.~~Table 114.   **PositionAssignmentType Attributes and/or Associations**

Property/Association	Description
kind : [1]	The kind relationship between the *PartyTypes* that is set to *Assignment*
position : PositionType [0..*]	The *PositionType* that will be filled by the *PartyType* referenced by the target of the *PositionTypeAssignment*.

2998

2999 **10.1.2.11 PositionType**

3000 A kind of *PartyType* that represents the type or classification of a *Position*.

3001 **Generalizations**

3002 The *PositionType* element inherits the attributes and/or associations of:

3003   • *PartyType* (see the section entitled "PartyType" for more information).

3004 **Properties**

3005 The following table presents the additional attributes and/or associations for *PositionType*:

Pedigree and Provenance Model and Notation v1.0   173

Property/Association	Description
assigneeType : PositionAssignmentType [1..*]	A *PositionAssignmentType* that indicates the *PartyType* that may fill the *PositionType*.

10.2 Locations

The Locations package contains elements related to physical or virtual locations.

10.2.1 Instances

The Locations.Intances section of the **Parties** metamodel contains elements related to locations and the relationships between them. These elements enable modeling specific locations at which Parties may reside. Elements in the Locations.Instances section are generally specializations of **SCE** *TypedElements* and as such may have an ElementType specified. The corresponding types are described below in the Locations.Types section.

Organizations may deem the location at which an occurrence took place to be of significance. In those situations a Location, either physical or virtual, may be captured in conjunction with an Occurrence.
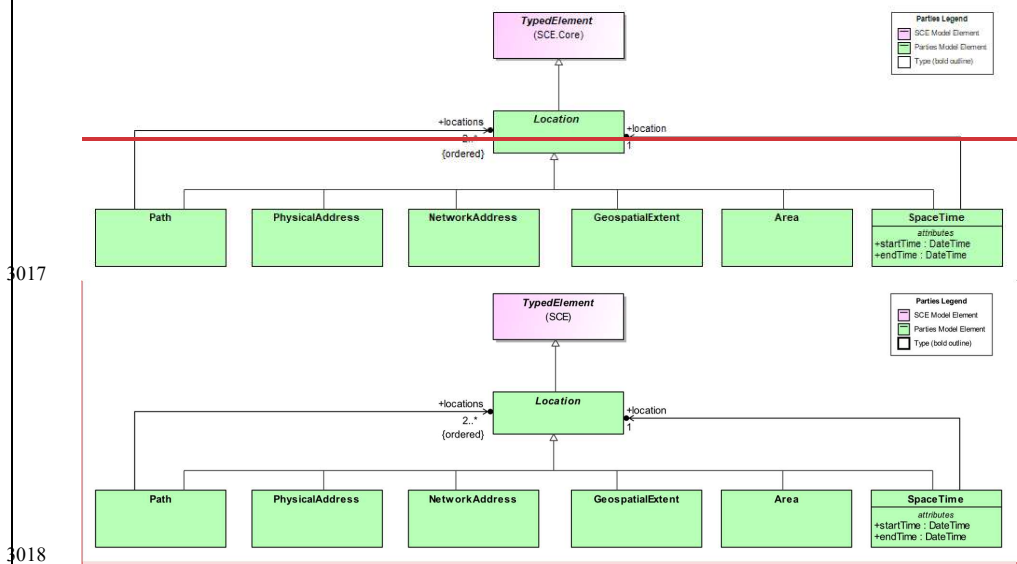
Figure 68:   Locations

10.2.1.1 Area

A kind of location that encompasses some region in the world.

Generalizations

3023   The *Area* element inherits the attributes and/or associations of:

3024   • *Location* (see the section entitled "Location" for more information).

3025   **Properties**

3026   The following table presents the additional attributes and/or associations for *Area*:

~~Table 117.~~Table 116.   Area Attributes and/or Associations

Property/Association	Description
type : AreaType [0..1]	The class that provides a specification of the *Area*.

3027

3028   **10.2.1.2   GeospatialExtent**

3029   A location that is a volume in the world such as a container or a room.

3030   **Generalizations**

3031   The *GeospatialExtent* element inherits the attributes and/or associations of:

3032   • *Location* (see the section entitled "Location" for more information).

3033   **Properties**

3034   The following table presents the additional attributes and/or associations for *GeospatialExtent*:

~~Table 118.~~Table 117.   GeospatialExtent Attributes and/or Associations

Property/Association	Description
type : VolumeType [0..1]	The class that provides a specification of the *GeospatialExtent*.

3035

3036   **10.2.1.3   Location**

3037   A particular place or position.

3038   **Generalizations**

3039   The *Location* element inherits the attributes and/or associations of:

3040   • **SCE** *TypedElement* (see the ~~section~~ **SCE** specification for more information). | **Commented [JB170]:** Text updated for PPMN-19/PPMN-83.

3041   **Properties**

3042   The following table presents the additional attributes and/or associations for *Location*:

~~Table 119.~~Table 118.   Location Attributes and/or Associations

Property/Association	Description
description : String [0..1]	A description of the *Location*.
type : LocationType [0..1]	The class that provides a specification of the *Location*.

3043

10.2.1.4 NetworkAddress

The address of an element or node on a network.

Generalizations

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

- *Location* (see the section entitled "Location" for more information).

Properties

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

Table 120.Table 119.   NetworkAddress Attributes and/or Associations

Property/Association	Description
type : NetworkAddressType [0..1]	The class that provides a specification of the *NetworkAddress*.

10.2.1.5 Path

An ordered collection of *Locations*.

Generalizations

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

- *Location* (see the section entitled "Location" for more information).

Properties

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

Table 121.Table 120.   Path Attributes and/or Associations

Property/Association	Description
locations : Location [2..*]	The locations that specify the *Path*.
type : PathType [0..1]	The class that provides a specification of the *Path*.

10.2.1.6 PhysicalAddress

A physical location in the real world that has an identifiable address.

Generalizations

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

- *Location* (see the section entitled "Location" for more information).

Properties

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

Table 122.Table 121.   PhysicalAddress Attributes and/or Associations

Property/Association	Description
type : PointType [0..1]	The class that provides a specification of the *PhyicalAddress*.

3067

10.2.1.7   SpaceTime

3069   A *Location* at a particular point in time.

Generalizations

3071   The *SpaceTime* element inherits the attributes and/or associations of:

3072   - *Location* (see the section entitled "<u>Location</u>" for more information).

Properties

3074   The following table presents the additional attributes and/or associations for *SpaceTime*:

Table 123.Table 122.   SpaceTime Attributes and/or Associations

Property/Association	Description
endTime : DateTime []	The ending time of the *SpaceTime*.
location : Location [1]	The location of the *SpaceTime*.
startTime : DateTime []	The starting time of the *SpaceTime*.
type : SpaceTimeType [0..1]	The class that provides a specification of the *SpaceTime*.

3075

10.2.2   Types

3077   The Locations.Types section of the **Parties** metamodel contains elements related to the kinds of locations and the
3078   relationships between them that are of interest in some context. These elements enable modeling kinds of Locations
3079   rather than particular Locations. Elements in the Locations.Types section are generally specializations of **SCE**
3080   *ElementTypes* and as such provide a specification of Locations to be created using elements in the
3081   Locations.Instances section described above.

10.2.2.1   AreaType

3083   A kind of *LocationType* that states that a *Location* is a region or surface in the world.

Generalizations

3085   The *AreaType* element inherits the attributes and/or associations of:

3086   - *LocationType* (see the section entitled "<u>LocationType</u>" for more information).

Properties

3088   The *AreaType* element does not have any additional attributes and/or associations.

10.2.2.2 LocationType

A class representing the type or classification of a *Location.*.

Generalizations

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

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

Properties

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

10.2.2.3 NetworkAddressType

A class that specifies that *Locations* of this type are *NetworkAddresses.*

Generalizations

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

- *LocationType* (see the section entitled "LocationType" for more information).

Properties

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

10.2.2.4 PathType

A kind of *LocationType* that states that a *Location* is a path.

Generalizations

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

- *LocationType* (see the section entitled "LocationType" for more information).

Properties

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

10.2.2.5 PointType

A kind of *LocationType* that states that a *Location* is a specific point in the world.

Generalizations

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

- *LocationType* (see the section entitled "LocationType" for more information).

Properties

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

10.2.2.6 SpaceTimeType

A kind of *LocationType* that states that a *Location* is a *Location* at a particular time.

Generalizations

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

- *LocationType* (see the section entitled "LocationType" for more information).

Properties

3123 The *SpaceTimeType* element does not have any additional attributes and/or associations.

10.2.2.7 VolumeType

3125 A kind of *LocationType* that states that a *Location* is a volume in the world such as a container or room.

Generalizations

3127 The *VolumeType* element inherits the attributes and/or associations of:

3128 • *LocationType* (see the section entitled "LocationType" for more information).

Properties

3130 The *VolumeType* element does not have any additional attributes and/or associations.
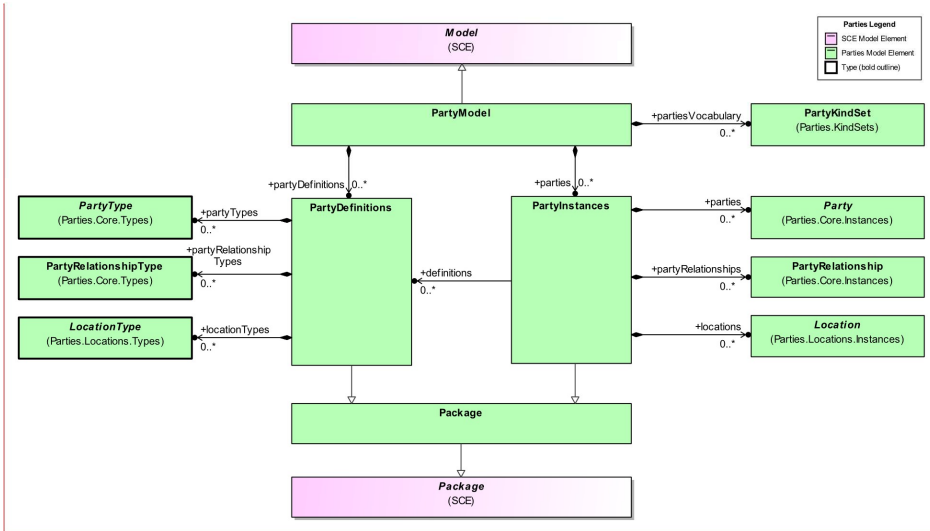
10.3 Packages

3132 The Packages package provides elements to support the packaging of Parties-related elements.

3133 The following figure presents the attributes and associations for the **Parties** packaging elements, including details
3134 about the elements they contain:

3135
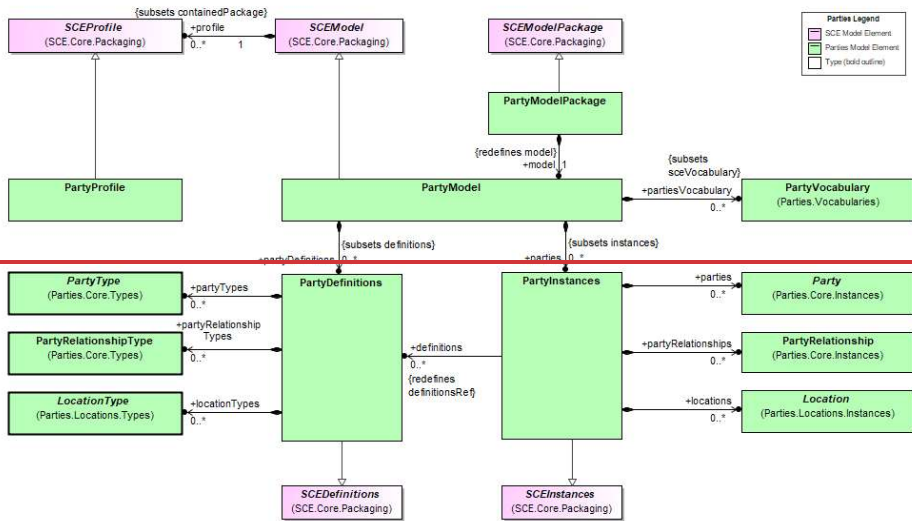
3136

Figure 69:   Party Packages

10.3.1   Package

Package is a kind of **SCE** *Package* that that is concrete. **Parties** *Packages* are a general packaging mechanism that can be used to hold any group of model elements.

Generalizations

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

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

Properties

Package has no additional properties.

10.3.1 10.3.2 PartyDefinitions

PartyDefinitions is a kind of ~~SCEDefinitions~~ *Package* that contains the definitions of PartyTypes that are used to specify types of Party structures.

Generalizations

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

- ~~SCEDefinitions~~ *Package* (see the section entitled
  "PackagePackagePackagePackagePackage~~SCEDefinitions~~Package" for more information).

Properties

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

Table 124.Table 123.   PartyDefinitions Attributes and/or Associations

Property/Association	Description
locationTypes : LocationType [0..*]	The `locationTypes` property references the *LocationTypes* contained within the *PartyDefinitions* package.
partyRelationshipTypes : PartyRelationshipType [0..*]	The `partyRelationshipTypes` property references the *PartyRelationshipTypes* contained within the *PartyDefinitions* package.
partyTypes : PartyType [0..*]	The `partyTypes` property references the *PartyTypes* contained within the *PartyDefinitions* package.

3156

10.3.210.3.3 PartyInstances

3157

3158
3159
PartyInstances is kind of ~~SCEInstances~~ *Package* package that contains *Parties, PartyRelationships,* and their *Locations*.

3160
Generalizations

3161
The *PartyInstances* element inherits the attributes and/or associations of:

3162
3163
- ~~SCEInstances~~ *Package* (see the section entitled "PackagePackagePackagePackagePackage~~SCEInstances~~Package" for more information).

Commented [JB176]: Text updated for PPMN-19/PPMN-83.

3164
Properties

3165
The following table presents the additional attributes and/or associations for *PartyInstances*:

Table 125.Table 124.   PartyInstances Attributes and/or Associations

Property/Association	Description
definitions : PartyDefinitions [0..*]	The property refers to zero or more *SCEDefinitions* packages that contains the *ElementTypes* that provide a basis for the instances contained in the *PartyInstances* package.
locations : Location [0..*]	The `locations` property references the *Location* elements contained within the *PartyInstances* package.
parties : Party [0..*]	The `parties` property references the *Party* elements contained within the *PartyInstances* package.
partyRelationships : PartyRelationship [0..*]	The `partRelationships` property references the *PartyRelationship* elements contained within the *PartyInstances* package.

3166

10.3.310.3.4 PartyModel

PartyModel is kind of **SCE** *Model* that contains definitions of types of *Parties* as well as specifications of *Party* structures themselves.

Generalizations

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

- ~~SCE~~SCE *Model* (see the ~~section entitled "SCEModel"~~SCE specification for more information).

Commented [JB177]: Text updated for PPMN-19/PPMN-83.

Properties

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

~~Table 126.~~Table 125.   PartyModel Attributes and/or Associations

Property/Association	Description
parties : PartyInstances [0..*]	The `parties` property subsets the *SCEModel* `instances` property. It contains a list of all the *PartyInstance* sub-packages contained within a *SCEModel*.
partiesVocabulary : PartyVocabulary [0..*]	The `partiesVocabulary` is a list of terms (as *KindsKindsKindsKindsKindsSemanticReferenceKinds*) that provide an extensible mechanism to define the elements of enumerations in a *PartiesModel*.
partyDefinitions : PartyDefinitions [0..*]	The `partyDefinitions` property subsets the *SCEModel* `definitions` property. It contains a list of all the *PartyDefinitions* sub-packages contained within a *PartyModel*.

~~10.4.0   PartyModelPackage~~

~~The *PartyModelPackage* is a specialization of *SCEModelPackage* and the main package for a Parties model. When the content of that model is serialized, the elements will be contained within a *PartyModelPackage*.~~
~~PartyModelPackage SHALL contain one PartyModel as the `model` and zero or more PartiesDI packages as the presentation.~~

~~Further, as a specialization of *SCEPackage* *PartyModelPackages* may contain other *SCEPackages* and can import other *SCEPackages* as well,~~

~~Generalizations~~

~~The *PartyModelPackage* element inherits the attributes and/or associations of:~~

- ~~*SCEModelPackage* (see the section entitled "SCEModelPackage" for more information).~~

~~Properties~~

~~The following table presents the additional attributes and/or associations for *PartyModelPackage*:~~

3188

3189 ### 10.21.0   PartyProfile

3190 A *PartyProfile* is a kind of *SCEProfile* that comprises profiles that can be applied to elements in a **PartyModel**.
3191 *PartyProfiles* provide a mechanism to exchange profile libraries.

3192 **Generalizations**

3193 The *PartyProfile* element inherits the attributes and/or associations of:

3194   • *SCEProfile* (see the section entitled "SCEProfile" for more information).

3195 **Properties**

3196 The *PartyProfile* element does not have any additional attributes and/or associations.

> **Commented [JB178]:** Text updated for PPMN-19/PPMN-83.

10.2810.4   Primitives

3198 The *Primitives* package provides primitive data elements used by other **Parties** elements.

3199 The following figure presents the primitive elements used in the **Parties** metamodel:

3200

3201
3202 **Figure 70:   Primitives**

3203 ### 10.28.110.4.1   DateTime

3204 A primitive that captures a point in time including a date and the time of day to greatest precision practical.

3205 **Generalizations**

3206 The *DateTime* element does not inherit any attributes or associations of from another element.

3207 **Properties**

3208 The *DateTime* element does not have any additional attributes and/or associations.

3209 ## 10.2910.5   VocabulariesPartyKindSets

3210 *PartyVocabularies PartyKindSets* are sets of terms used within a **Parties** model that are defined by an external
3211 ontology. The terms link to formal definitions for the terms used within the model. The SemanticReferenceThe Kind
3212 element is used to name the term provide a link to the definitions. *PartyVocabularies* are contained within an
3213 *PartiesModel* package.

3214 The following figure presents the elements related to the *PartyVocabulary* *PartyKindSet* section:

3215

3216

3217
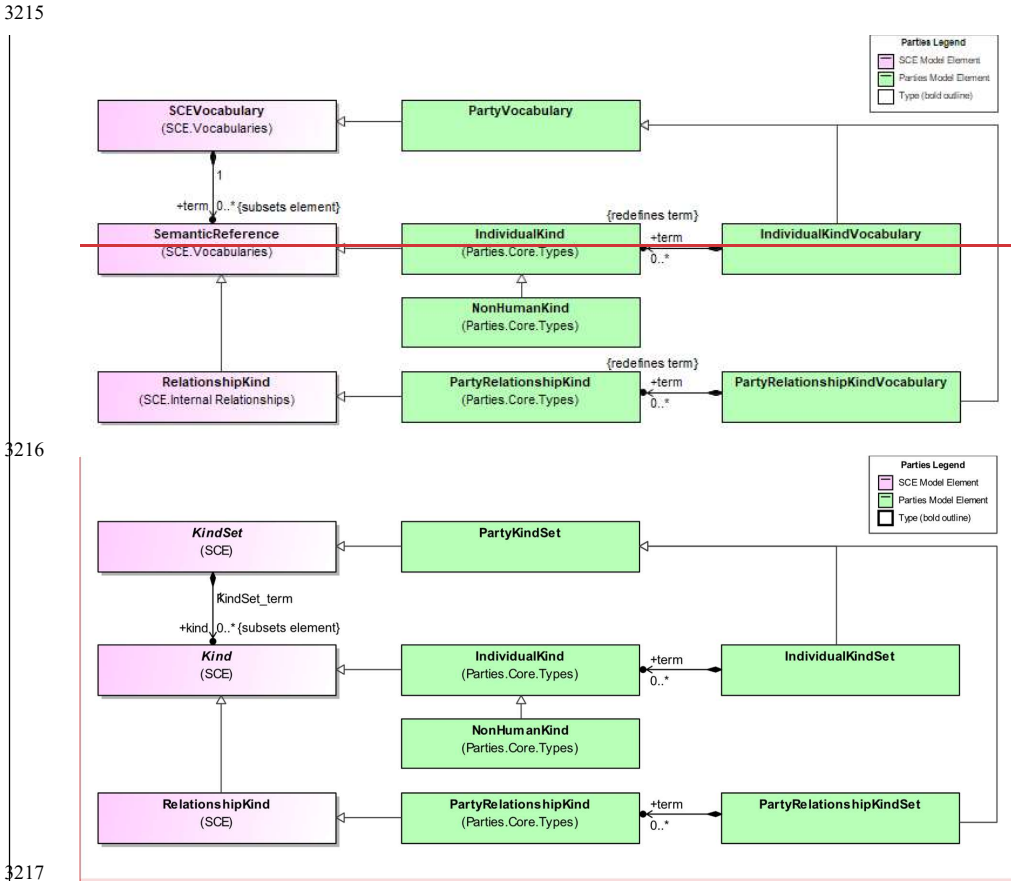
3218 **Figure 71:** ~~PartyVocabularies~~PartyKindSets

3219 ### ~~10.29.1~~10.5.1   ~~PartyVocabulary~~PartyKindSet

3220 A *PartyVocabulary* A *PartyKindSet* is a kind of *SCEVocabulary* that includes a list of terms defined as instances of
3221 the *SemanticReferenceKind* element. As instances of *SemanticReferenceKind*, or a specialization thereof, the
3222 instances can be used to relate the terms to external definitions of the meaning of the term. The terms themselves do
3223 not represent the definitions or meanings but provide links to an external source. The **Parties** model contains two
3224 ~~vocabularies~~KindSets: *PartyRelationshipKinds* and *IndividualKinds.*

3225

3226 **Generalizations**

3227 The *PartyVocabulary* *PartyKindSet* element inherits the attributes and/or associations of:

- *SCEVocabulary KindSet* (see the section entitled "SCEVocabulary"SCE specification for more information).

Commented [JB180]: Text updated for PPMN-19/PPMN-83, and PPMN-29/PPMN-85.

Properties

The *PartyVocabulary PartyKindSet* element does not have any additional attributes and/or associations.

10.29.210.5.2   IndividualKindVocabularySet

A *IndividualKindVocabularySet* is a kind of *PartiesVocabulary PartiesKindSet* that includes a list of terms defined as instances of *IndiviualKind*, itself a *SemanticReferenceKind.* As instances of a specialization of *SemanticReferenceKind*, the instances can be used to relate the terms to external definitions of the meaning of the term. The terms themselves do not represent the definitions or meanings but provide links to an external source.

Generalizations

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

- *PartyVocabulary PartyKindSet* (see the section entitled "PartyVocabularyPartyKindSet" for more information).

Properties

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

Table 127.Table 126.   **IndividualKindVocabularySet Attributes and/or Associations**

Property/Association	Description
term : IndividualKind [0..*]	A list of the terms representing valid IndividualKinds.

10.29.310.5.3   PartyRelationshipKindVocabularySet

A *PartyRelationshipKindVocabularySet* is a kind of *PartiesVocabulary* that includes a list of terms defined as instances of *PartyRelationshipKind*, itself a kind specialization of *SemanticReferenceKind.* As instances of a specialization of *SemanticReferenceKind*, the instances can be used to relate the terms to external definitions of the meaning of the term. The terms themselves do not represent the definitions or meanings but provide links to an external source.

Generalizations

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

- *PartyVocabulary PartyKindSet* (see the section entitled "PartyVocabularyPartyKindSet" for more information).

Properties

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

Table 128.Table 127.   **PartyRelationshipKindVocabularySet Attributes and/or Associations**

Property/Association	Description
term : PartyRelationshipKind [0..*]	A list of the terms representing valid PartyRelationshipKinds.

3256

11   Parties Library

3258   A Library is included in the **Parties** specification to provide standard values that that are intended to be provided by
3259   tools implementing the **Parties** specification. Currently, **Parties** defines the standard values for two vocabularies:
3260   *IndividualKinds* and *PartyRelationshipKinds* (See next sections).

11.1   IndividualKinds

3262   The *IndividualKinds* package contains the instances representing the standard *IndividualKinds* ~~vocabulary~~set. This
3263   ~~vocabulary~~set provides a standard set of terms for the kinds of Individuals that can be instantiated within a Parties
3264   model. These elements include an instance of a *~~PartiesVocabulary~~PartiesKindSet*, `IndividualKinds,`which
3265   represents the ~~vocabulary~~set itself as well as instances of *IndividualKind* representing the kinds of Individuals that
3266   may be instantiated.

3267   The *IndividualKind* element is used to indicate a specific kind IndividualType that is to be created. The instances
3268   defined in this Library SHALL be included in any **Parties** implementation. However, the implementation can allow
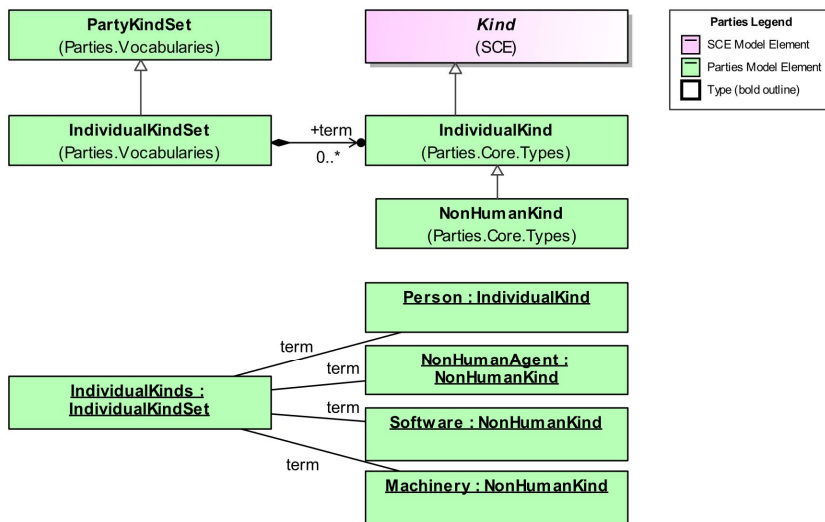3269   additional instances of the class to represent new IndividualTypes.

3270   The following figure presents the instances for the *IndividualKind* element that are `terms` for the
3271   `IndividualKinds` ~~vocabulary~~set.

3272
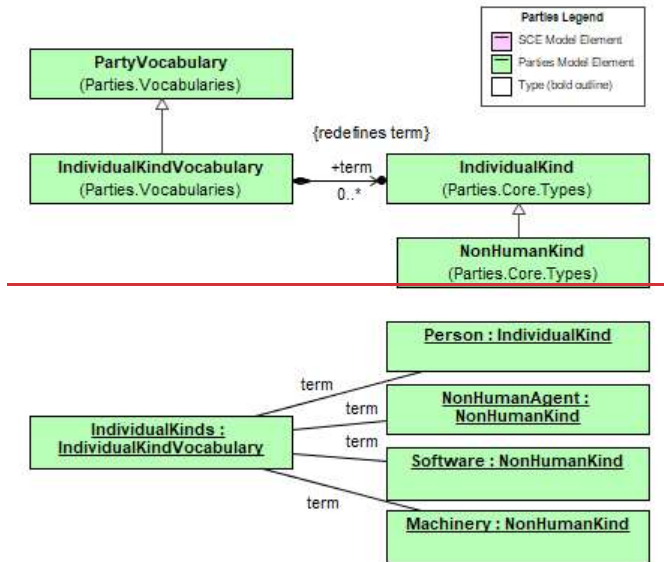
3273

Figure 72: IndividualKinds

The following table provides a definition of the terms included in the *IndividualKinds* ~~Vocabulary~~set.

~~Table 129.~~Table 128.   IndividualKinds ~~Vocabulary~~KindSet

| #	Name	Documentation
1	IndividualKinds	`IndividualKinds` is an instance of *~~PartiesVocabulary~~ PartiesKindSet* that includes terms for the kinds of *PartyRelationships* that may be created in a **Parties** model.
2	Machinery	`Machinery` indicates that the type of NonHumanKind 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.

11.1.1   IndividualKinds

`IndividualKinds` is an instance of *PartiesVocabulary* that includes terms for the kinds of *PartyRelationships* that may be created in a **Parties** model.

11.1.2 Machinery

`Machinery` indicates that the type of NonHumanKind is a machine of some kind.

11.1.3 NonHumanAgent

`NonHumanAgent` indicates that the type of individual is an automated system of some kind.

11.1.4 Person

`Person` indicates that the type of individual is a person.

11.1.5 Software

`Software` indicates that the type of individual is a software module of some kind.

11.2 PartyRelationshipKinds

The *PartyRelationshipKinds* package contains one instance of an **SCE** *SCEVocabularyKindSet*:
`PartyRelationshipKind` which is provided by the **Parties** Library. The purpose of this ~~vocabulary~~set is to provide a set of standard terms for the different types of relationships between Parties. These terms will be represented by instances of the *PartyRelationshipKind* element.

The instances defined in this Library SHALL be included in any **Parties** implementation. However, the implementation can allow additional instances of the class if required for a particular modeling situation. Specifying the kinds of Party relationships using this instantiation mechanism rather than a fixed enumerated list enables extension of the kinds of relationships that are possible without having to modify the standard.

The following figure presents the instances for the *PartyRelationshipKind* element that are `terms` for the instance (`PartyRelationshipKinds`) of the *~~PartiesVocabulary~~ PartiesKindSet* element.

3301
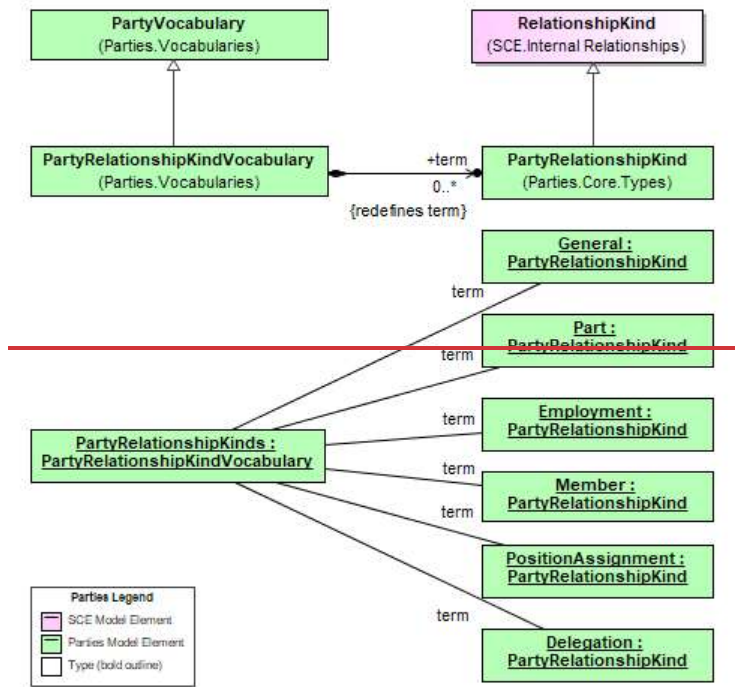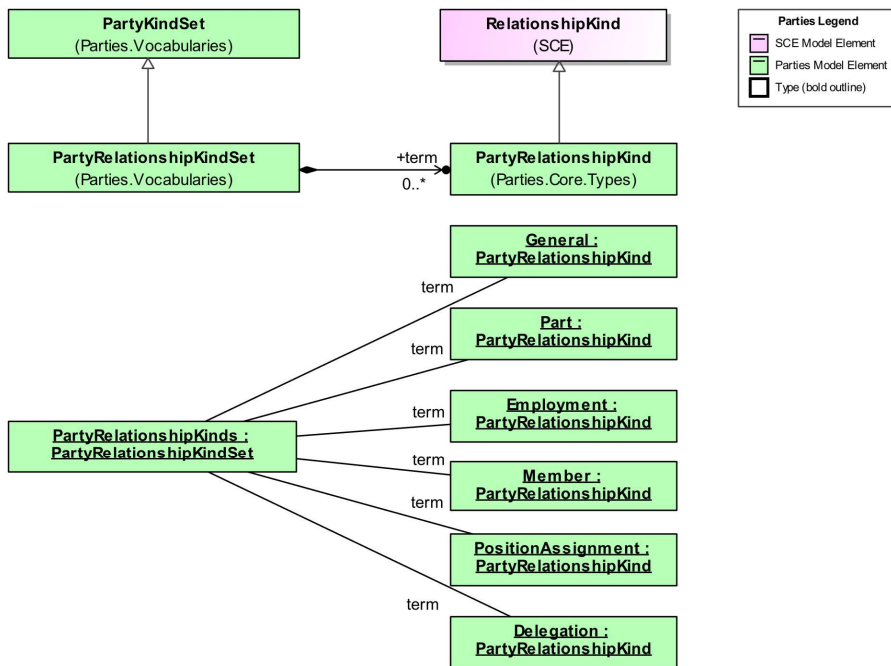
Figure 73: **PartyRelationshipKinds**

The following table provides a definition of the terms included in the *PartyRelationshipKinds* Vocabulary.

Table 130.Table 129.   **PartyRelationshipKinds VocabularySet**

#	Name	Documentation
1	PartyRelationshipKinds	`PartyRelationshipKinds` is an instance of *PartiesVocabulary PartiesKindSet* that includes terms for the kinds of *PartyRelationships* that may be created in a **Parties** model.
2	Delegation	`Delegation` indicates that the `target` element of the *PartyRelationship*, either a *Party* or *PartyType* has been delegated the responsibilities associated with the `source` element, either a *Position* or *PositionType*, respectively.
3	Employment	`Employment` indicates that the `targetParty` element of the *PartyRelationship* is employed by the `sourceParty`.

#	Name	Documentation
4	General	`General` indicates the existence of some general relationship between the `source` element of the *PartyRelationship* is a member of the `target` element.
5	Member	`Member` indicates that the `target` element of the *PartyRelationship* is a member of the `source` element.
6	Part	`Part` indicates that the `target` element of the *PartyRelationship* is a part of the `source` element.
7	PositionAssignment	`Assignment` indicates that the `source` element of the *PartyRelationship*, either a *Party* or *PartyType* is assigned to the `target` element, either a *Position* or *PositionType*, respectively.

11.2.1 PartyRelationshipKinds

PartyRelationshipKinds is an instance of ~~*PartiesVocabulary*~~ *PartiesKindSet* that includes terms for the kinds of *PartyRelationships* that may be created in a **Parties** model.

11.2.2 Delegation

`Delegation` indicates that the `target` element of the *PartyRelationship*, either a *Party* or *PartyType* has been delegated the responsibilities associated with the `source` element, either a *Position* or *PositionType*, respectively.

11.2.3 Employment

`Employment` indicates that the `targetParty` element of the *PartyRelationship* is employed by the `sourceParty`.

11.2.4 General

`General` indicates the existence of some general relationship between the `source` element of the *PartyRelationship* is a member of the `target` element.

11.2.5 Member

`Member` indicates that the `target` element of the *PartyRelationship* is a member of the `source` element.

11.2.6 Part

`Part` indicates that the `target` element of the *PartyRelationship* is a part of the `source` element.

11.2.7 PositionAssignment

`Assignment` indicates that the `source` element of the *PartyRelationship*, either a *Party* or *PartyType* is assigned to the `target` element, either a *Position* or *PositionType*, respectively.

~~SCE Metamodel~~

> **Commented [JB182]:** Original Section 12 SCE Metamodel removed per PPMN-19/PPMN-83.

12   PPMN and Parties Diagram Interchange (PPMN DI and Parties DI)

12.1　Scope

This chapter describes the **PPMN** and **Parties** Diagram Interchange (**PPMN DI** and **Parties DI**, respectively). **PPMN DI** extends the **Parties DI**. The **Parties DI** uses the diagram interchange capabilities provided in **SCE** (see the SCE 1.0 Beta 1 specification (dtc/22-01-04)). The **PPMN DI** is meant to facilitate the interchange of **PPMN** and **Parties** diagrams between tools rather than being used for internal diagram representation by the tools. The simplest interchange approach to ensure the unambiguous rendering of **PPMN** and **Parties** diagrams was chosen. As such, **PPMN DI** does not aim to preserve or interchange any "tool smarts" between the source and target tools (e.g., layout smarts, efficient styling, etc.).

PPMN DI does not ascertain that **PPMN** or **Parties** diagrams are syntactically or semantically correct.

12.2　Diagram Definition and Interchange

PPMN DI and **Parties DI**, through their extension of the **SCE DI** meta-model are defined as a MOF-based meta-models. As such, their instances can be serialized and interchanged using XMI. **PPMN DI** and **Parties DI** are also defined by the **SCEDI** XML schema. Thus, their instances can also be serialized and interchanged using XML.

The **SCE DI** (see the **SCE 1.0 Beta 1** specification) is harmonized with the OMG Diagram Definition (DD) standard version 1.1. The referenced DD contains two main parts: the Diagram Commons (DC) and the Diagram Interchange (DI). The DC defines common types like bounds and points, while the DI provides a framework for defining domain-specific diagram models. As a domain-specific DI, **SCE DI** defines a few new meta-model classes that derive from the abstract classes DI.

The focus of **PPMN DI** and **Parties DI** is the interchange of laid out shapes and edges that constitute **PPMN** and **Parties** diagrams, respectively. Each shape and edge references a particular **PPMN** or **Parties** model element. The referenced model elements are all part of an actual **PPMN** or **Parties** model. As such, **PPMN DI** and **Parties DI** are meant to only contain information that is neither present nor derivable, from the original model whenever possible. Simply put, to render a **PPMN** or **Parties** diagram both the proper **DI** instance(s) (including **PPMN**, **Parties, and SCE DI** instances**)** as well as the referenced **PPMN** and/or **Parties** model instance(s) are REQUIRED.

From the **PPMN DI** perspective, a **PPMN** diagram is a particular snapshot of a **PPMN** model at a certain point in time. Multiple **PPMN** diagrams can be exchanged referencing model elements from the same **PPMN** model. Each diagram may provide an incomplete or partial depiction of the content of the **PPMN** model. The exporting tool is free to decide how many diagrams are exported and the importing tool is free to decide if and how to present the contained diagrams to the user. Similarly for **Parties DI**.

12.3　Notation

As a specification that contains elements that can notated graphically, **PPMN** specifies the depiction for **PPMN** diagram elements, including **Parties** elements and **SCE** *DiagramArtifact* elements.

Serializing a **PPMN** diagram (including those that contain only **Parties** model elements) for interchange requires the specification of a collection of *SCEShape*(s) and *SCEEdge*(s) in the *SCEDiagram*. The *SCEShape*(s) and *SCEEdge*(s) attributes must be populated in such a way as to allow the unambiguous rendering of the **PPMN** diagram by the receiving party. More specifically, the *SCEShape*(s) and *SCEEdge*(s) MUST reference **PPMN (or Parties)** model elements. If no ~~*SCEElement*~~*BaseElement* is referenced or if the reference is invalid, it is expected that this shape or edge will not be depicted.

When rendering a **PPMN** diagram, the correct depiction of an *SCEShape* or *SCEEdge* depends mainly on the referenced model element and its particular attributes and/or references. The purpose of this clause is to: provide a library of the **PPMN** and **Parties** element depictions, and to provide an unambiguous resolution between the referenced model element [*BaseElement*] ~~*SCEElement*~~ and their depiction. Depiction resolution tables are provided below for both *SCEShape* and *SCEEdge*.

12.3.1　Labels

Both *SCEShape* and *SCEEdge* elements may have labels (its name attribute) placed on the shape/edge, or above or below the shape/edge, in any direction or location, depending on the preference of the modeler or modeling tool vendor.

3376 Labels are optional for *SCEShape* and *SCEEdge*. When there is a label, the position of the label is specified by the
3377 bounds of the *SCELabel* of the *SCEShape* or *SCEEdge*. Simply put, label visibility is defined by the presence of the
3378 *SCELabel* element.

3379 The bounds of the *SCELabel* are optional and always relative to the containing *SCEDiagram's* origin point. The
3380 depiction resolution tables provided below exemplify default label positions if no bounds are provided for the
3381 *SCELabel* (for *SCEShape* kinds and *SCEEdge* kinds (see sections above)).

3382 When the *SCELabel* is contained in a *SCEShape*, the text to display is the name of the *BaseElement*.~~SCEElement.~~
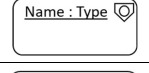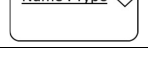
3383 12.3.2 Shape Resolution
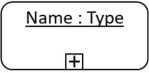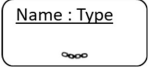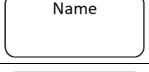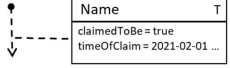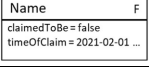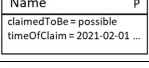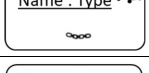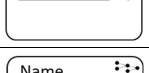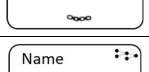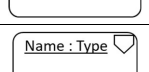
3384 *SCEShape* can be used to represent any of the non-relationship elements from **PPMN** and **Parties** models. These
3385 include elements such as *Entity*, *EntityType*, *Occurrence*, *OccurrenceType*, *Organization*, and
3386 *OrganizationType.*When a *SCEShape* is used to depict a diagram element the actual shape is determined by the
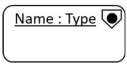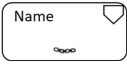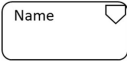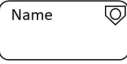3387 referred PPMN or Parties model element.
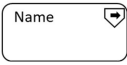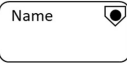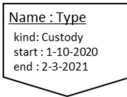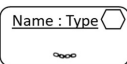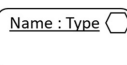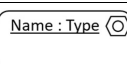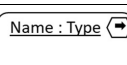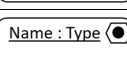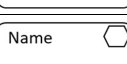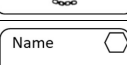
3388 12.3.2.1 Depiction for PPMN Diagram Elements

3389 The following table presents the depiction resolutions for **PPMN** elements:

~~Table 131.~~Table 130.   Depiction Resolution of PPMN Shapes

PPMN Element	PPMN Element Attributes	Depiction
Entity		Name : Type
EntityType		Name
EntitySnapshot		Name : Type ↻
EntityTypeSnapshot		Name ↻
EntityFormat		Name #
Occurrence		Name : Type

Occurrence (with Subchain)		Name : Type ⊞
OccurrenceChain		Name : Type ⚭
OccurrenceType		Name
OccurrenceChainType (with Subchain)		Name ⊞
OccurrenceBranchNode		◇
Claim (as shape)	claimedToBe = true	Name T claimedToBe = true timeOfClaim = 2021-02-01 …
Claim (as shape)	claimedToBe = false	Name F claimedToBe = false timeOfClaim = 2021-02-01 …
Claim (as shape)	claimedToBe = possible	Name P claimedToBe = possible timeOfClaim = 2021-02-01 …
PedigreeChain		Name : Type ⦂∴ ⚭
PedigreeOccurrence		Name : Type ⦂∴
PedigreeChainType		Name ⦂∴ ⚭
PedigreeOccurrenceType		Name ⦂∴
CustodyChain		Name : Type ▽ ⚭
CustodyOccurrence		Name : Type ▽
CustodyOccurrence (Custody Start)	kind = instance of CustodyStartKind	Name : Type ◌
CustodyOccurrence (Custody Transfer)	kind = instance of CustodyTransferKind	Name : Type ➡
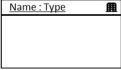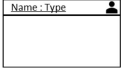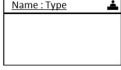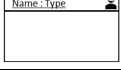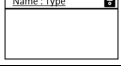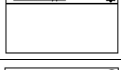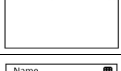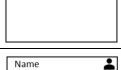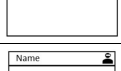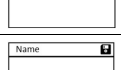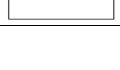
CustodyOccurrence (Custody Start)	kind = instance of CustodyEndKind	Name : Type ◉
CustodyChainType		Name ▽
CustodyOccurrenceType		Name ▽
CustodyOccurrenceType (CustodyStart type)	kind = instance of CustodyStartKind	Name ◌
CustodyOccurrenceType (CustodyTransfer type)	kind = instance of CustodyTransferKind	Name ➡
CustodyOccurrenceType (CustodyEnd type)	kind = instance of CustodyEndKind	Name ◉
Custody (with attributes)		Name : Type kind: Custody start : 1-10-2020 end : 2-3-2021
OwnershipOccurrenceChain		Name : Type ⬡
OwnershipChangeOccurrence		Name : Type ⬡
OwnershipChangeOccurrence (Acquisition)	kind = instance of OwnershipStartKind	Name : Type ◎
OwnershipChangeOccurrence (Ownership Change)	kind = instance of OwnershipTransferKind	Name : Type ➡
OwnershipChangeOccurrence (End of Ownership Chain)	kind = instance of OwnershipEndKind	Name : Type ◉
OwnershipChainType		Name ⬡
OwnershipOccurrenceType		Name ⬡
OwnershipOccurrenceType (Ownership Start)	kind = instance of OwnershipStartKind	Name ◎
OwnershipOccurrenceType (Ownership Transfer)	kind = instance of OwnershipTransferKind	Name ➡
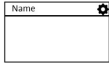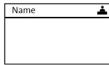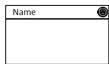
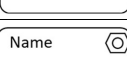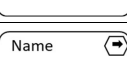|		
OwnershipOccurrenceType (ownership End)	kind = instance of OwnershipEndKind	Name ⦿
Ownership (with attributes)		Name : Type kind: Ownership start : 1-10-2020 end : 2-3-2021

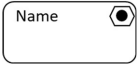3390 **12.3.2.2 Depiction for Parties Diagram Elements**

3391 The following table presents the depiction resolutions for **Parties** elements:

Table 132.Table 131.   Depiction Resolution of Parties Shapes

Parties Element	Parties Element Attributes	Depiction
Organization		Name : Type 🏛
Person		Name : Type 👤
Position		Name : Type 👤
NonHumanAgent		Name : Type 👥
Software		Name : Type 💾
Machinery		Name : Type ⚙
PartyRole		Name : Type ◉
OrganizationType		Name 🏛
IndividualType (Person)	kind = Person	Name 👤
IndividualType (NonHumanAgent)	kind = NonHumanAgent	Name 👥
IndividualType (Software)	kind = Software	Name 💾

IndividualType (Machinery)	kind = Machinery	Name ⚙
PositionType		Name ▲
PartyRoleType		Name ◉
Area		🖼
Path		🖼
PhysicalAddress		📍
NetworkAddress		◎
GeospacialExtent		⬚
SpaceTime		🕐

3392
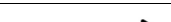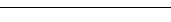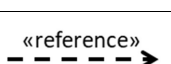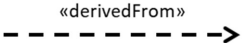
12.3.3   Edge Resolution

3393

3394 *SCEEdge* can be used to represent and of the **PPMN** or **Parties** relationships including relationships such as
3395 *EntityRelationship*, *OccurrenceDependency*, and *PartyRelationship*.

12.3.3.1 Depiction for PPMN Diagram Elements

3396

3397 The following table presents the depiction resolutions for **PPMN** edges:

~~Table 133.~~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	«reference» - - - -▸
DerivedFrom		«derivedFrom» - - - - - -▸

RevisionOf		«revisionOf» — — — — — — — —>
QuotedFrom		«quotedFrom» — — — — — — — —>
SourcedFrom		«sourcedFrom» — — — — — — — —>
DerivationType (DerivedFrom)	kind = DerivedFrom	«derivedFrom» — — — — — — — —>
DerivationType (RevisionOf)	kind = RevisionOf	«revisionOf» — — — — — — — —>
DerivationType (QuotedFrom)	kind = QuotedFrom	«quotedFrom» — — — — — — — —>
DerivationType (SourcedFrom)	kind = SourcedFrom	«sourcedFrom» — — — — — — — —>
OccurrenceRelationship		——————>
OccurrenceDependency	kind = Input	role name ·············> «input»
OccurrenceDependency	kind = Enabler	role name ·············> «enabler»
OccurrenceDependency	kind = Output	role name ·············> «output»
OccurrenceDependency	kind = Product	role name ·············> «product»
OccurrenceDependency	kind = By-product	role name ·············> «by-product»
OccurrenceDependency	kind = Waste	role name ·············> «waste»
OccurrenceDependencyType	kind = Input	role name ·············> «input»

Parties Element	Parties Element Attribute	Depiction
OccurrenceDependencyType	kind = Enabler	role name «enabler»
OccurrenceDependencyType	kind = Output	role name «output»
OccurrenceDependencyType	kind = Product	role 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)		

3398 **12.3.3.2   Depiction for Parties Diagram Elements**

3399 The following table presents the depiction resolutions for **Parties** edges:

Table 134.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 = *not null*	Name : Type ············> «delegation» name
PartyRelationshipType (General)	relationshipKind = Member	name - - - - - - -
PartyRelationshipType (Member)	relationshipKind = Member	name - - - - - -> «member»
PartyRelationshipType (Employment)	relationshipKind = Employment	name - - - - - -> «employment»
PartyRelationshipType (Part)	relationshipKind = Part	name ◇—————
PositionAssignmentType	relationshipKind = PositionAssignment	name ············>
DelegationType (without Authority shown)	relationshipKind = Delegation	name ············> «delegation»
DelegationType (with Authority shown)	relationshipKind = Delegation	Name ············> «delegation» name

3400

Annex A: PROV Traceability

3401

3402

3403   (informative)

3404

3405   A key requirement of **PPMN** is to support all the capabilities available in the W3C PROV specification. This
3406   ANNEX describes the traceability of PPMN elements to elements in W3C PROV. Please note that the model of the
3407   W3C PROV specification presented herein is an interpretation in UML of that specification by the PPMN authors.

3408   This diagram shows the PPMN and W3C PROV concepts related to the primary three PROV elements - Agent,
3409   Entity, and Activity.

3410
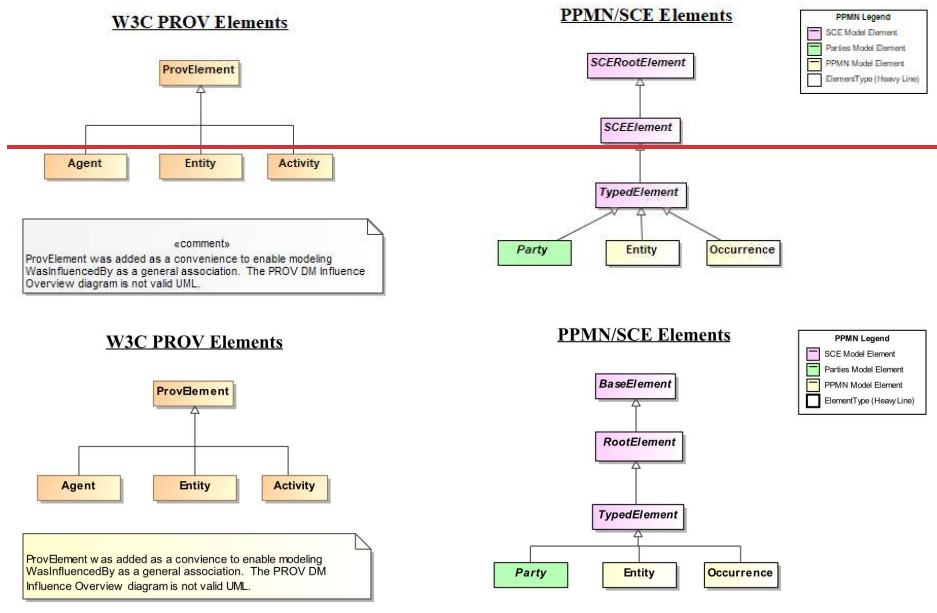
3411

3412

3413   **Figure 74:   PPMN Trace to PROV - Primary PROV Elements**

3414   This diagram shows the PPMN and W3C PROV concepts related to Agents, Responsibility, and Influence.

3415

W3C PROV Elements
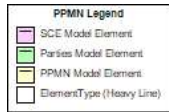
PPMN/SCE Elements
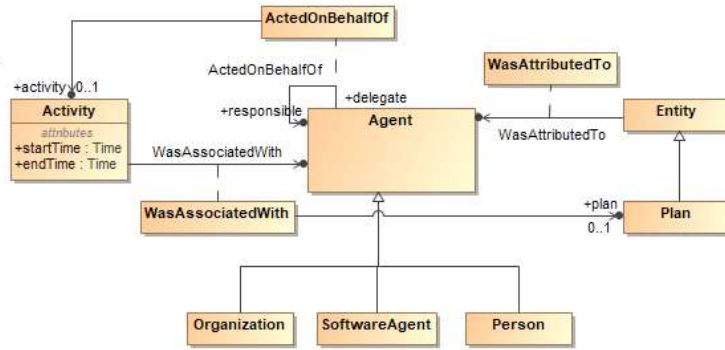
3416

W3C PROV Elements
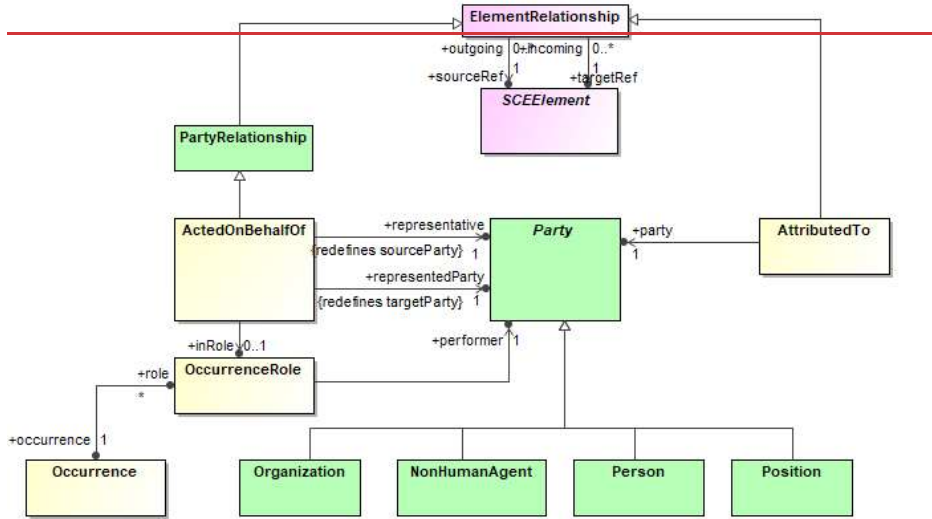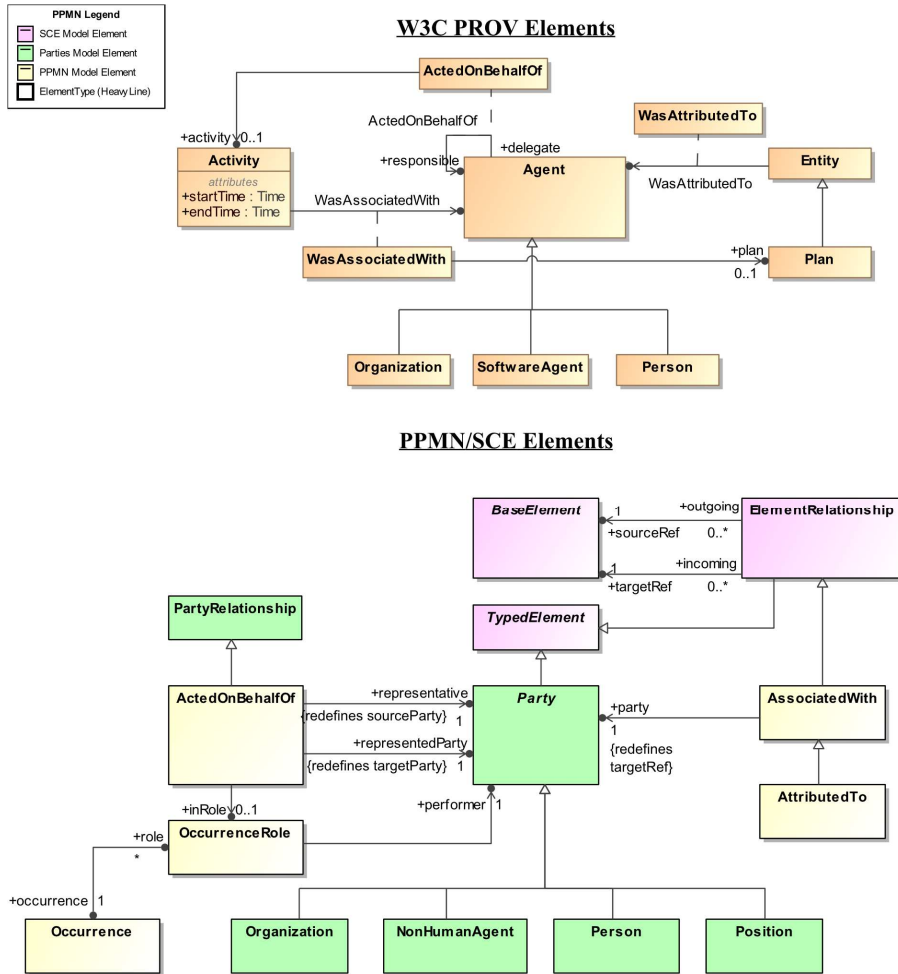
PPMN/SCE Elements

3417

Figure 75:   PPMN Trace to PROV - Agents, Responsibility, and Influence

3419   This diagram shows the PPMN and W3C PROV concepts related to Derivations.

3420

W3C PROV Elements

PPMN Legend
- SCE Model Element
- Parties Model Element
- PPMN Model Element
- ElementType (Heavy Line)

WasQuotedFrom

WasRevisionOf

HadPrimarySource

Entity

WasGeneratedBy

Used

Activity
+startTime : Time
+endTime : Time

-usedEntity

HadPrimarySource

-generatedEntity

WasDerivedFrom

WasRevisionOf

WasQuotedFrom

WasDerivedFrom

+usage

Used
+time : Time

0..1

+generation

WasGeneratedBy
+time : Time

0..1

PPMN/SCE Elements

+entityDependency

OccurrenceDependency

+inputEntity

0..*

*

+target 1

+source

Entity

1

+target

EntityRelationship

0..*

+resultingEntityRelationship

0..1

ActivityOccurrence

1

1

0..*

+occurrence

+derivedEntity 1

+derivati
onSourc
e

DerivedFrom

+derivations

0..1

PedigreeOccurrence

0..*

+pedigreeOccurrence

**DerivedFrom :
DerivationKind**

**DescendantOf :
DerivationKind**

**QuotedFrom :
DerivationKind**

**RevisionOf :
DerivationKind**

**SourcedFrom :
DerivationKind**

W3C PROV Elements

PPMN/SCE Elements

3422

Figure 76:   PPMN Trace to PROV - Derivations

This diagram shows the PPMN and W3C PROV concepts related to Entities and their relationships to Activities (or Occurrences in PPMN).

3426
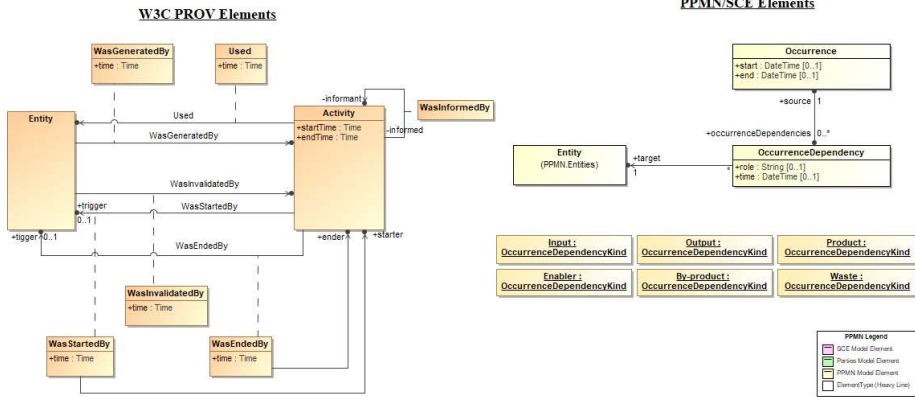
3427

Figure 77:   PPMN Trace to PROV - Entities and Activities

3429   This diagram shows the PPMN and W3C PROV concepts related to Influence.

3430
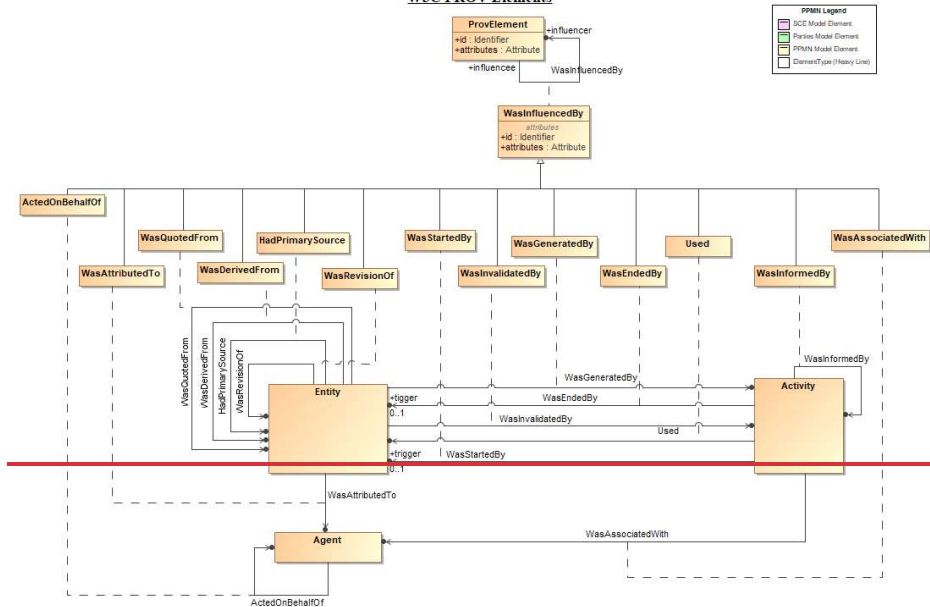
PPMN/SCE Elements
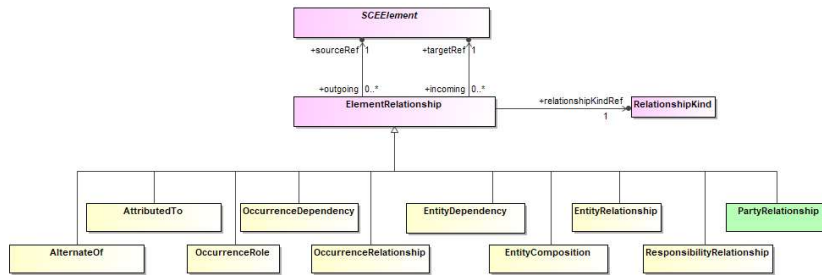
3431
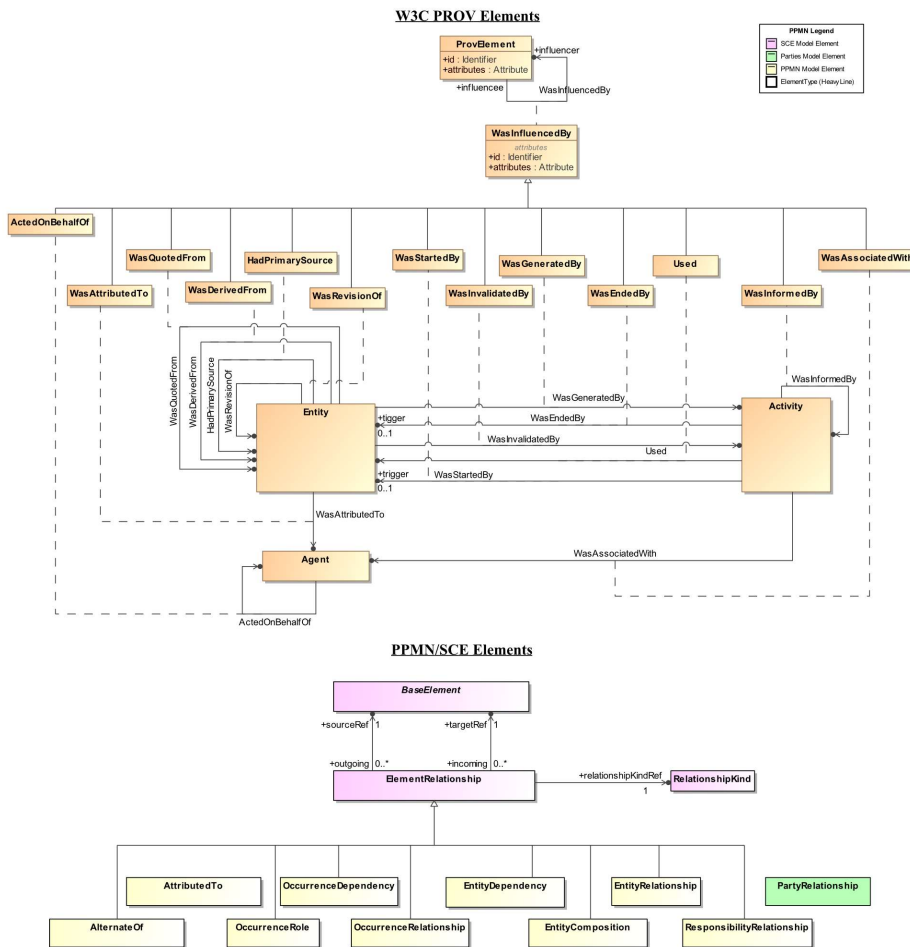
Figure 78:   PPMN Trace to PROV - Influence

This diagram shows the PPMN and W3C PROV concepts related to the core PROV elements.

W3C PROV Elements

PPMN/SCE Elements
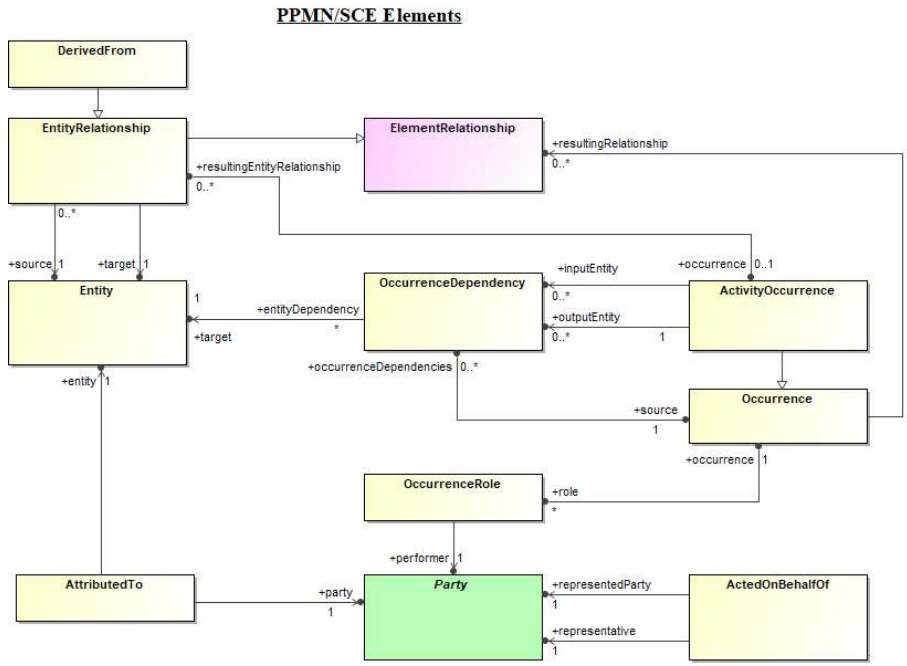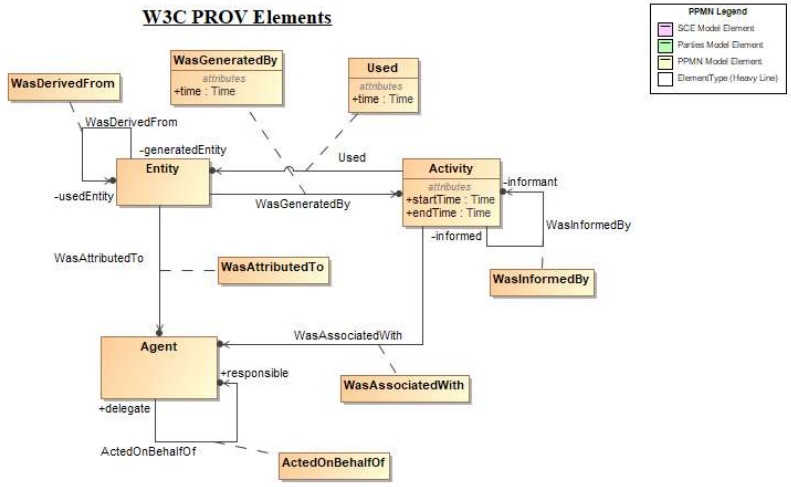
3436

Figure 79:   PPMN Trace to PROV - PROV Core Structures

3438   ~~This traceability matrix shows the traceability of PPMN elements to W3C PROV elements related to Elements,~~

3439 Entities, and Extensions.

Table 135.   PPMN to PROV Traceability Matrix - Elements, Entities and Extensions

3440

3441 This traceability matrix shows the traceability of PPMN elements to W3C PROV elements related to Activities.

Table 136.   PPMN to PROV Traceability Matrix - Activities

3442

3443 This traceability matrix shows the traceability of PPMN elements to W3C PROV elements related to Pedigree.

~~Table 137.   PPMN to PROV Traceability Matrix - Pedigree~~

Legend
↗ Trace

	ActedOnBehalfOf[delegate->Agent]	Activity	Agent	AlternateOf[Entity->Entity]	Attribute	Bundle	Collection	EmptyCollection	Entity	HadPrimarySource[Entity->Entit]	Identifier	Location	Namespace	Organization	Person	Plan	ProvElement	QualifiedName	Role	SoftwareAgent	SpecializationOf[Entity->Entity]	Attribution	Delegation	Derivation	Generation	Usage	Time	Used[Activity->Entity]	Value	WasAssociatedWith[Activity->A]	WasAttributedTo[Entity->Agent]	WasDerivedFrom[GeneratedEntity]	WasEndedBy[Activity->trigger(E]	WasGeneratedBy[Entity->Activit]	WasInfluencedBy[Influencee:Prov]	WasInformedBy[informed:Activit]	WasInvalidatedBy[Entity->Activ]	WasQuotedFrom[Entity->Entity]	WasRevisionOf[Entity->Entity]	WasStartedBy[Activity->trigger]
Pedigree									1													1	1									3							3	3
Derivations									1													1										3							3	3
DerivationKind	3																															↗							↗	↗
DerivationType	3																															↗							↗	↗
DerivedFrom	2																				1			↗																
DescendantOf																																								
QuotedFrom	1																																					↗		
RevisionOf	1																																						↗	
SourcedFrom	1							↗																																
Pedigree Occurrences																						1																		
CreationOccurrenceType																																								
EntityPedigree																																								
EntityPedigreeType																																								
PedigreeKind																																								
PedigreeOccurrenceChain																																								
PedigreeOccurrence	1																				1			↗																
PedigreeOccurrenceChainType																																								
PedigreeOccurrenceType																																								
PedigreeTypeGraph																																								

~~This traceability matrix shows the traceability of PPMN elements to W3C PROV elements related to Provenance.~~

Table 138.   PPMN to PROV Traceability Matrix - Provenance

3446

3447 This traceability matrix shows the traceability of PPMN elements to W3C PROV elements related to Delegation,
3448 Derivation, Primitives and other relationships.

Table 139. **PPMN to PROV Traceability Matrix - Delegations, Derivations, Primitives and Other Relationships**

3449

3450 This traceability matrix shows the traceability of PPMN elements to W3C PROV elements related to Parties and

3451 Locations.

Table 140.   PPMN to PROV Traceability Matrix - Parties and Locations

Table 140.   PPMN to PROV Traceability Matrix - Parties and Locations

3452

3453   TBD

Table 141.   SCE to PROV Traceability Matrix

SCE \ PROV-DM [PROV-DM/trunk #9]	ActedOnBehalfOf[delegate:Agent...]	Activity	Agent	AlternateOf[Entity -> Entity]	Attribute	Bundle	Collection	EmptyCollection	Entity	HadPrimarySource[Entity -> Enti...]	Identifier	Location	Namespace	Organization	Person	Plan	ProvElement	QualifiedName	Role	SoftwareAgent	SpecializationOf[Entity -> Entity]	Attribution	Delegation	Derivation	Generation	Usage	Time	Used[Activity -> Entity]	Value	WasAssociatedWith	WasAttributedTo	WasDerivedFrom	WasEndedBy	WasGeneratedBy	WasInfluencedBy	WasInformedBy	WasInvalidatedBy	WasQuotedFrom	WasRevisionOf	WasStartedBy
□ SCE									1		1						1	1								1			1											
□ Annotations																																								
Annotation																																								
Attachment																																								
Category																																								
Documentation																																								
□ Core									1		1						1	1								1														
ElementType																																								
□ Packaging												1																												
SCEDefinitions																																								
SCEInstances																																								
SCEModel																																								
SCEModelPackage	1											↗																												
SCEPackage																																								
SCEProfile																																								
SCEElement	4											↗					↗	↗								↗														
SCERootElement																																								
TypedElement																																								
□ External Relationships																																								
ExternalRelationship																																								
Import																																								
□ Internal Relationships																																								
ElementRelationship																																								
ElementRelationshipType																																								
RelationshipKind																																								
□ Models																										1														
Association																																								
Group																																								
ModelArtifact																																								
TextAnnotation	1																				1					↗														
□ Vocabularies																																								
SCEVocabulary																																								
SemanticReference																																								

Legend — ↗ Trace

3454
3455
3456
3457