AP242 ED4 ELECTRICAL WIRE HARNESS (EWS) TUTORIALS

Lothar Klein, LKSoftWare GmbH
Sophie Herail, CIMPA S.A.S.
Thomas Rousselot, AFNeT Services
Pierre Duchier, Airbus

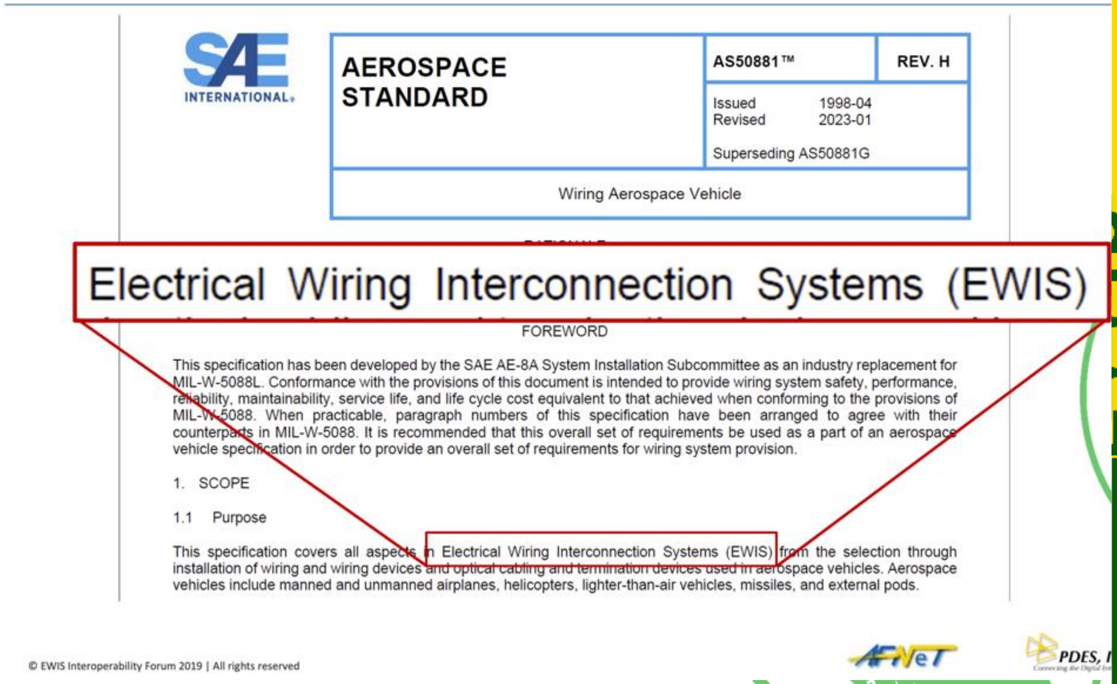
This document is based on material provided in the document AP242_Electrical_Harness_Tutorial_XML.pdf

Version 3.0; 2025-12-16

Table of Content

Introduction

- AP242 supports all major design requirements as defined by the Aerospace standard SAE AS50881 rev. H, that says in clause *1. Scope*: 'This specification covers all aspects in Electrical Wiring Interconnection Systems (EWIS) from the selection through installation of wiring and wiring devices and optical cabling and termination devices used in aerospace vehicles. [...]' "
- See also next page a list of underlying standards for AS50881



SAE INTERNATIONAL

AEROSPACE STANDARD	AS50881™	REV. H
	Issued 1998-04 Revised 2023-01 Superseding AS50881G	
Wiring Aerospace Vehicle		

Electrical Wiring Interconnection Systems (EWIS)

FOREWORD

This specification has been developed by the SAE AE-8A System Installation Subcommittee as an industry replacement for MIL-W-5088L. Conformance with the provisions of this document is intended to provide wiring system safety, performance, reliability, maintainability, service life, and life cycle cost equivalent to that achieved when conforming to the provisions of MIL-W-5088. When practicable, paragraph numbers of this specification have been arranged to agree with their counterparts in MIL-W-5088. It is recommended that this overall set of requirements be used as a part of an aerospace vehicle specification in order to provide an overall set of requirements for wiring system provision.

1. SCOPE

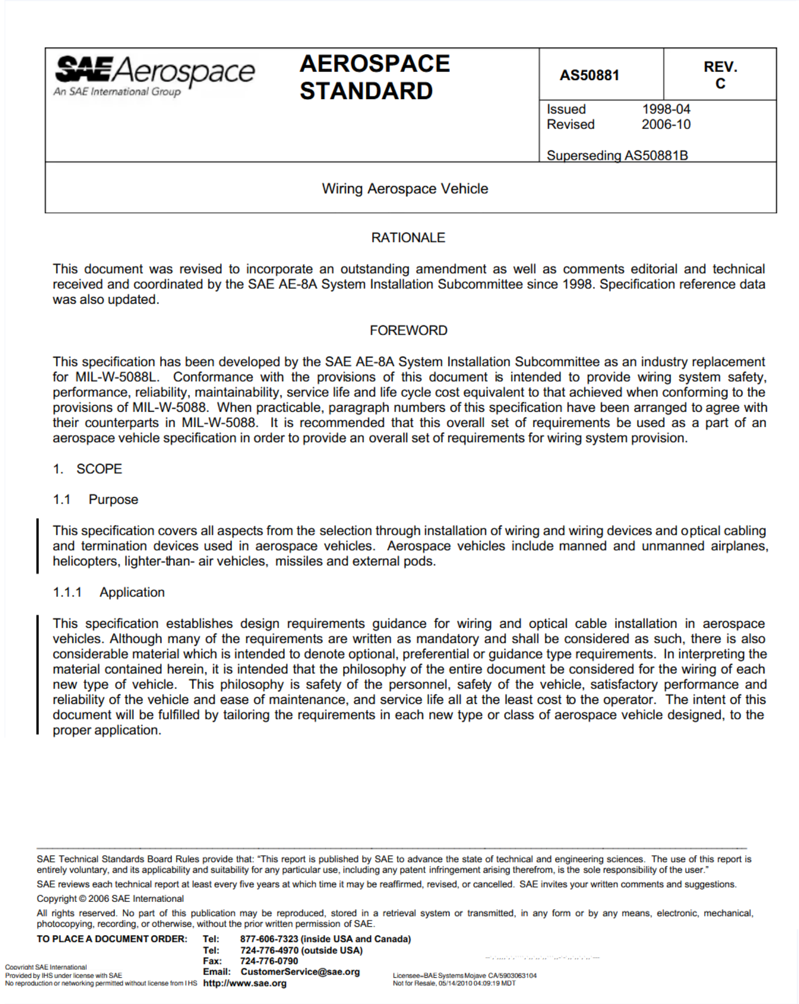
1.1 Purpose

This specification covers all aspects in Electrical Wiring Interconnection Systems (EWIS) from the selection through installation of wiring and wiring devices and optical cabling and termination devices used in aerospace vehicles. Aerospace vehicles include manned and unmanned airplanes, helicopters, lighter-than-air vehicles, missiles, and external pods.

© EWIS Interoperability Forum 2019 | All rights reserved

Introduction

List of underlying standards for AS50881



SAE Aerospace An SAE International Group	AEROSPACE STANDARD	AS50881	REV. C
		Issued 1998-04 Revised 2006-10 Superseding AS50881B	

Wiring Aerospace Vehicle

RATIONALE

This document was revised to incorporate an outstanding amendment as well as comments editorial and technical received and coordinated by the SAE AE-8A System Installation Subcommittee since 1998. Specification reference data was also updated.

FOREWORD

This specification has been developed by the SAE AE-8A System Installation Subcommittee as an industry replacement for MIL-W-5088L. Conformance with the provisions of this document is intended to provide wiring system safety, performance, reliability, maintainability, service life and life cycle cost equivalent to that achieved when conforming to the provisions of MIL-W-5088. When practicable, paragraph numbers of this specification have been arranged to agree with their counterparts in MIL-W-5088. It is recommended that this overall set of requirements be used as a part of an aerospace vehicle specification in order to provide an overall set of requirements for wiring system provision.

1. SCOPE

1.1 Purpose

This specification covers all aspects from the selection through installation of wiring and wiring devices and optical cabling and termination devices used in aerospace vehicles. Aerospace vehicles include manned and unmanned airplanes, helicopters, lighter-than- air vehicles, missiles and external pods.

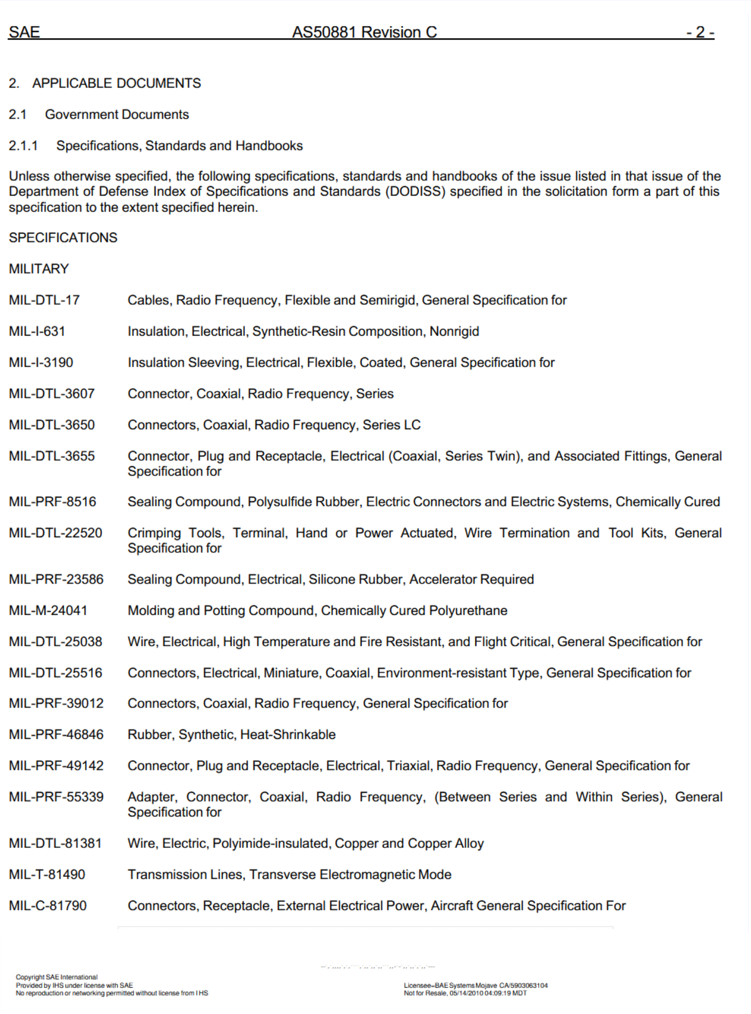
1.1.1 Application

This specification establishes design requirements guidance for wiring and optical cable installation in aerospace vehicles. Although many of the requirements are written as mandatory and shall be considered as such, there is also considerable material which is intended to denote optional, preferential or guidance type requirements. In interpreting the material contained herein, it is intended that the philosophy of the entire document be considered for the wiring of each new type of vehicle. This philosophy is safety of the personnel, safety of the vehicle, satisfactory performance and reliability of the vehicle and ease of maintenance, and service life all at the least cost to the operator. The intent of this document will be fulfilled by tailoring the requirements in each new type or class of aerospace vehicle designed, to the proper application.

SAE Technical Standards Board Rules provide that: "This report is published by SAE to advance the state of technical and engineering sciences. The use of this report is entirely voluntary, and its applicability and suitability for any particular use, including any patent infringement arising therefrom, is the sole responsibility of the user."

SAE reviews each technical report at least every five years at which time it may be reaffirmed, revised, or cancelled. SAE invites your written comments and suggestions.

TO PLACE A DOCUMENT ORDER:
Tel: 877-606-7323 (inside USA and Canada)
Tel: 724-776-4970 (outside USA)
Fax: 724-776-0790
Email: CustomerService@sae.org
http://www.sae.org



SAE AS50881 Revision C - 2 -

2. APPLICABLE DOCUMENTS

2.1 Government Documents

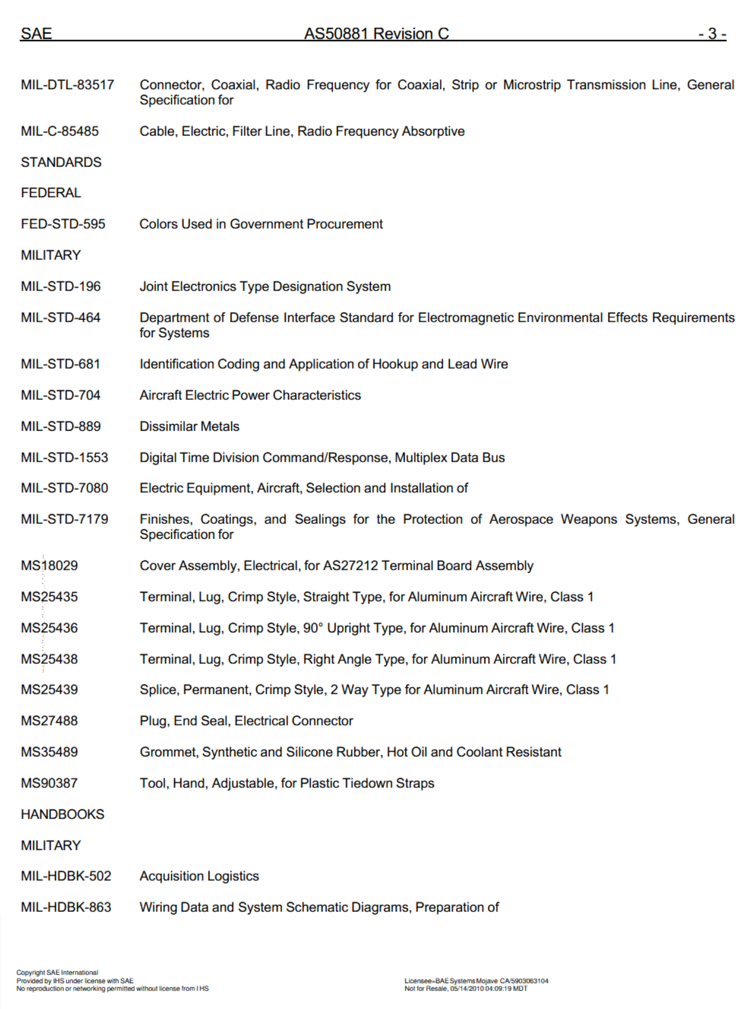
2.1.1 Specifications, Standards and Handbooks

Unless otherwise specified, the following specifications, standards and handbooks of the issue listed in that issue of the Department of Defense Index of Specifications and Standards (DODISS) specified in the solicitation form a part of this specification to the extent specified herein.

SPECIFICATIONS

MILITARY

MIL-DTL-17	Cables, Radio Frequency, Flexible and Semirigid, General Specification for
MIL-I-631	Insulation, Electrical, Synthetic-Resin Composition, Nonrigid
MIL-I-3190	Insulation Sleeving, Electrical, Flexible, Coated, General Specification for
MIL-DTL-3607	Connector, Coaxial, Radio Frequency, Series
MIL-DTL-3650	Connectors, Coaxial, Radio Frequency, Series LC
MIL-DTL-3655	Connector, Plug and Receptacle, Electrical (Coaxial, Series Twin), and Associated Fittings, General Specification for
MIL-PRF-8516	Sealing Compound, Polysulfide Rubber, Electric Connectors and Electric Systems, Chemically Cured
MIL-DTL-22520	Crimping Tools, Terminal, Hand or Power Actuated, Wire Termination and Tool Kits, General Specification for
MIL-PRF-23586	Sealing Compound, Electrical, Silicone Rubber, Accelerator Required
MIL-M-24041	Molding and Potting Compound, Chemically Cured Polyurethane
MIL-DTL-25038	Wire, Electrical, High Temperature and Fire Resistant, and Flight Critical, General Specification for
MIL-DTL-25516	Connectors, Electrical, Miniature, Coaxial, Environment-resistant Type, General Specification for
MIL-PRF-39012	Connectors, Coaxial, Radio Frequency, General Specification for
MIL-PRF-46846	Rubber, Synthetic, Heat-Shrinkable
MIL-PRF-49142	Connector, Plug and Receptacle, Electrical, Triaxial, Radio Frequency, General Specification for
MIL-PRF-55339	Adapter, Connector, Coaxial, Radio Frequency, (Between Series and Within Series), General Specification for
MIL-DTL-81381	Wire, Electric, Polyimide-insulated, Copper and Copper Alloy
MIL-T-81490	Transmission Lines, Transverse Electromagnetic Mode
MIL-C-81790	Connectors, Receptacle, External Electrical Power, Aircraft General Specification For



SAE AS50881 Revision C - 3 -

MIL-DTL-83517	Connector, Coaxial, Radio Frequency for Coaxial, Strip or Microstrip Transmission Line, General Specification for
MIL-C-85485	Cable, Electric, Filter Line, Radio Frequency Absorptive

STANDARDS

FEDERAL

FED-STD-595	Colors Used in Government Procurement

MILITARY

MIL-STD-196	Joint Electronics Type Designation System
MIL-STD-464	Department of Defense Interface Standard for Electromagnetic Environmental Effects Requirements for Systems
MIL-STD-681	Identification Coding and Application of Hookup and Lead Wire
MIL-STD-704	Aircraft Electric Power Characteristics
MIL-STD-889	Dissimilar Metals
MIL-STD-1553	Digital Time Division Command/Response, Multiplex Data Bus
MIL-STD-7080	Electric Equipment, Aircraft, Selection and Installation of
MIL-STD-7179	Finishes, Coatings, and Sealings for the Protection of Aerospace Weapons Systems, General Specification for
MS18029	Cover Assembly, Electrical, for AS27212 Terminal Board Assembly
MS25435	Terminal, Lug, Crimp Style, Straight Type, for Aluminum Aircraft Wire, Class 1
MS25436	Terminal, Lug, Crimp Style, 90° Upright Type, for Aluminum Aircraft Wire, Class 1
MS25438	Terminal, Lug, Crimp Style, Right Angle Type, for Aluminum Aircraft Wire, Class 1
MS25439	Splice, Permanent, Crimp Style, 2 Way Type for Aluminum Aircraft Wire, Class 1
MS27488	Plug, End Seal, Electrical Connector
MS35489	Grommet, Synthetic and Silicone Rubber, Hot Oil and Coolant Resistant
MS90387	Tool, Hand, Adjustable, for Plastic Tiedown Straps

HANDBOOKS

MILITARY

MIL-HDBK-502	Acquisition Logistics
MIL-HDBK-863	Wiring Data and System Schematic Diagrams, Preparation of

Introduction

- This document is based on ISO/IS 10303-242:2025 (4th edition)
- This document focuses on XML implementations based on the
  AP242 ed4 Domain Model (DO-Model)
  as defined in ISO/TS 10303-4442:2025
  Note: Electrical wire harness can also be implemented using the
  MIM (Modular Integrated Model) of AP242 using STEP P21 file format
- All provided files will be made available on https://www.mbx-if.org/home/ewis/
- This tutorial is an extension of the:
  Recommended Practices for
  STEP AP242 Edition 3 Domain Model XML
  Product & Assembly Structure
  Release 3.2; 2024-01-11
  Please refer to that document for all general concepts
- This document covers only the specific extensions of AP242
  for the purpose of electrical wire harness (EWH)
- The EWH Domain Model covers also general electrical systems and so can be used to describe electrical installations for e.g. machines or buildings

Table of Content

- Introduction
- Basic XML structure
- Part, PartCategory & PartVersion
- PartView & ViewContext
- ProductClass & ProductConfiguration
- Occurrence, Assembly & WiringHarnessAssemblyDesign
- ShapeElement
- Geometric & topological models
- Connectors & terminals
- Features, Definitions & AssemblyShapeJoints
- Wire, Cables & TransportFeature
- Shapes/ContactFeatureDefinition & -Elements
- CrossSection of Wire, Cable & HarnessSegment
- Geometry/Topology Associations
- Complex Connectors / Assembly Hierarchy
- External Element References
- ComposedGeometricModel

XML File Format, XML Schema

- Every AP242 XML file has to call out the XML-Schema of AP242. For the 4th edition of AP242 this is defined as a DomainModel in the 5th edition of part ISO/TS 10303-4442
- The “Unit of serialization” (Uos) element is the topmost element that encloses all needed schema definitions.
- This element is crucial for organizing and structuring the data within the XML file, ensuring that all related schema definitions are properly encapsulated and can be referenced correctly during the serialization and deserialization processes.
- All the application objects (AO, entities) not being declared as contained in any other AO are defined as a subtype of cmn:BaseRootObject and show up within the cmn:DataContainer
- Example:

```
<?xml version="1.0" encoding="UTF-8"?>
<n0:Uos
 xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
 xmlns:n0="https://standards.iso.org/iso/ts/10303/-4442/ed-5/tech/xml-schema/domain_model"
 xsi:schemaLocation="https://standards.iso.org/iso/ts/10303/-4442/ed-5/tech/xml-schema/domain_model DomainModel.xsd">
 <Header>
  <Name>HarnessExample.stpx</Name>
  <TimeStamp>2025-03-10T15:09:25Z</TimeStamp>
  <Organization>
   <Name>ISO TC184/SC4 WG12 EWIS team</Name>
  </Organization>
 <Header/>
 <DataContainer xsi:type="n0:AP242DataContainer">
  <!– DATA -->
 </DataContainer>
</n0:Uos>
```

Table of Content

- Introduction
- Basic XML structure
- Part, PartCategory & PartVersion
- PartView & ViewContext
- ProductClass & ProductConfiguration
- Occurrence, Assembly & WiringHarnessAssemblyDesign
- ShapeElement
- Geometric & topological models
- Connectors & terminals
- Features, Definitions & AssemblyShapeJoints
- Wire, Cables & TransportFeature
- Shapes/ContactFeatureDefinition & -Elements
- CrossSection of Wire, Cable & HarnessSegment
- Geometry/Topology Associations
- Complex Connectors / Assembly Hierarchy
- External Element References
- ComposedGeometricModel

Part, PartCategory & PartVersion (1 of 8)

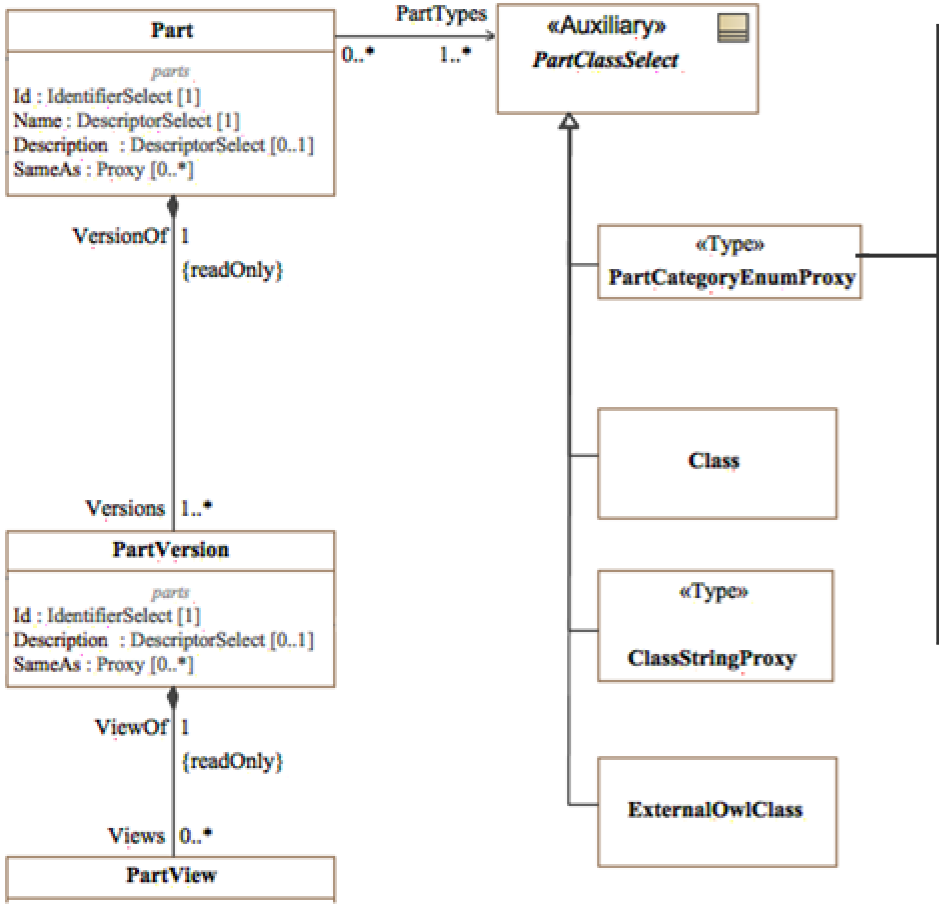
- For the purpose of Rec.Pracs on “XML Product & Assembly Structure” the representation of a Part and a PartVersion may just look like this:

```
<Part uid=”p—0000000017D374A0">
 <Id>
  <Identifier
   uid=”pid--0000000017D374A0—id1”
   id=”as1”
   idRoleRef=“”
   idContextRef=“o—000000178”/>
 </Id>
 <Name>
  <CharacterString>as1</CharacterString>
 </Name>
 …
 <Versions>
 <PartVersion uid="pv--0000000017D374A0—id1">
  …
 <Id id=”A.1"/>
  …
 </PartVersion>
 …
 </Versions>
</Part>
```

- This is not sufficient for an EWH converter as raw materials (cables, wires …) require different handling than other electrical relevant parts (connector, contact, terminal) and other nonelectrical components.

Part, PartCategory & PartVersion (2 of 8)

- The traditional part categories “detail”, “assembly”, “raw_material” are not sufficient for EWH
- New categories on the nature of a part (cable, connector) and how it can be used in an assembly (discrete, raw_material_by_length) are added



EMI_connector_housing,
backshell,
busbar,
cable,
cable_ferrule,
cable_fixing_device,
cable_support_sleeve,
cavity_plug,
conduit,
connector,
connector_cavity,
connector_contact,
connector_housing,
connector_insert,
connector_kit,
contact_member,
corrugated_conduit,
device,
electrified_backshell,
end_cap,
fishnet,
fixing_device,
flat_cable,
grommet,
insulation_piercing_connector,
joint_ferrule,
label,
optical_fibre,
overbraid,
plug,
protective_covering,
seal,
shield,
shield_connector,
shielded_connector,
shrink_boot,
socket,
spacer,
splice,
strain_relief_accessory,
tape,
terminal_block,
terminal_lug,
wire,
wiring_harness

See IEC Electropedia:
http://www.electropedia.org/

Part, PartCategory & PartVersion (3 of 8)

Example: Electrical Wire Harness Part

- The part has categories “wire” and “raw_material_by_length”.

So, a converter importing this file knows:

- The part is a wire (in the meaning of Electropedia. IEC 60050-151)
- The part can only be “used” in some specified lengths

```
<Part uid=”_101000">
 <Id id=”04034-22-9”/>
 <Name>
  <CharacterString>WIRE, ELEC, COMP, SNGL CONDUCTOR, 150 DEG C</CharacterString>
 </Name>
 <PartTypes>
  <PartCategoryEnum>wire</PartCategoryEnum>
  <PartCategoryEnum>raw_material_by_length</PartCategoryEnum>
 </PartTypes>
 <Versions>
  <PartVersion uid=”_101001">
   <Id id=”Version 1"/>
   <Views>
   </Views>
  </PartVersion>
 </Versions>
</Part>
```

Part, PartCategory & PartVersion (4 of 8)

Recommended occurrence controlling types

These part category types defines which type of occurrence to use of any usage of the part:

- If discrete (means one piece) or assembly, we must use either a SingleOccurrence (for one piece) or use a QuantifiedOccurrence with a piece quantity (e.g. 10 pieces)
- If raw_material_by_length, we must use a QuantifiedOccurrence and a length quantity (e.g. 2.3 metre)
- If raw_material_by_area, we must use a QuantifiedOccurrence and an area quantity (e.g. 1.2 square-metre)
- If raw_material_by_volume, we must use a QuantifiedOccurrence and a volume quantity (e.g. 10 milli-litre)

The recommendation is to combine one of the occurrence controlling types with one of the EWH specific types

Typical combinations:

cable wire fishnet grommet shield protective_covering	raw_material_by_length
connector backshell cable_fixing_device fixing_device	discrete
wiring_harness	assembly

Part, PartCategory & PartVersion (5 of 8)

Recommended general part category types throughout STEP

- assembly (is implicitly discrete as well)
- boxed
- collection (is implicitly NOT discrete as you get several pieces)
- tool

Part, PartCategory & PartVersion (6 of 8)

Recommended specific EWH part category types

- **EMI_connector_housing:** connector housing that shields against electromagnetic interference.
- **backshell:** connector accessory that is closing a connector from the back side and guide the wires and cables.
- **busbar:** low-impedance conductor to which several electric circuits can be connected at separate points.

NOTE 1 In many cases, the busbar consists of a bar [IEC ref 60050-151].

- **cable:** assembly of one or more conductors and/or optical fibres, with a protective covering and possibly filling, insulating and protective material [IEV ref 151-12-38].
- **cable_ferrule:** accessory in the form of a short tube to provide cable support or termination for a cable screen [IEV ref 581-27-18].
- **cable_fixing_device:** fixing device for a cable on a structure.
- **cable_support_sleeve:** flexible accessory or a part of a component placed around the cable to minimize flexing of the cable at the point of entry into the component [IEV ref 581-27-23].
- **cavity_plug:** plug for a connector cavity for the purpose of sealing.
- **conduit:** a part of a closed wiring system of generally circular cross section for insulated conductors and/or cables in electrical or communication installations, allowing them to be drawn in and/or replaced [IEV ref 442-02-03].
- **connector:** device providing connection and disconnection to a suitable mating component [IEV ref 151-12-19].

NOTE 2 A connector has one or more contact members.

- **connector_cavity:** cavity in a connector, connector housing or insert intended to receive a connector contact or multi contact.
- **connector_contact:** contact member that is intended to be contained in a connector.
- **connector_housing:** part of a connector into which the connector insert and contacts are assembled [IEV ref 581-27-10].
- **connector_insert:** insulating element designed to support and position contacts in a connector housing [IEV ref 581-27-11].
- **connector_kit:** collection of connector parts that are intended to be assembled together and that contain at least one connector housing and that may contain alternative parts that may or may not be used in the final assembly.
- **contact_member:** conductive element intended to make an electric contact [IEV ref 151-12-16].
- **corrugated_conduit:** a conduit in which the profile is corrugated in the longitudinal section [IEV ref 442-02-06].

NOTE 3 Both annular and helical corrugated conduits are permissible and a combination of both corrugated and plain conduit is possible.

- **device:** material element or assembly of such elements intended to perform a required function [IEV ref 151-11-20].

NOTE 4 A device can form part of a larger device.

- **electrified_backshell:** backshell that is intended to be conductive.
- **end_cap:** device placed on the ends of a cable to prevent the ingress of moisture during storage, transportation and installation [IEV ref 461-20-07].
- **fishnet:** material made from fibers woven or otherwise produced in a grid-like structure with fixed mesh size.
- **fixing_device:** system component specifically designed to secure other components to the wall, ceiling, floor or other structure [IEV ref 442-02-40].
- **flat_cable:** multicore cable having cores or groups of cores arranged in parallel flat formation [IEV ref 461-06-05].
- **grommet:** part of a component or an accessory, used to support and protect the wires or cable at the point of entry; it may also prevent the ingress of moisture or contaminants [IEV ref 581-27-19].
- **joint_ferrule:** through connector (of cables), metallic device for connecting two consecutive lengths of conductor [IEV ref 461-17-04].
- **label:** part that is intended to attach written information to other parts.

Part, PartCategory & PartVersion (7 of 8)

Recommended specific EWH part category types

- **optical_fibre:** waveguide shaped as a filament, made of dielectric materials for guiding optical waves.
- **overbraid:** protective covering (sheath) that is also a braid.
- **plug:** connector attached to a cable [IEV ref 151-12-21].
- **protective_covering:** sheat (North America jacket) : uniform and continuous tubular covering of metallic or non-metallic material, generally extruded [IEV ref 461-05-03].

NOTE 6 The term sheath is only used for metallic coverings in North America, whereas the term jacket is used for non-metallic coverings.

- **seal:** mechanical object that helps join other mechanical objects together by preventing leakage, containing pressure, or excluding contamination.
- **shield:** (of a cable) surrounding earthed metallic layer which serves to confine the electric field within the cable and/or to protect the cable from external electrical influence [IEV ref 461-03-04].

NOTE 7 Metallic sheaths, foils, braids, armours and earthed concentric conductors may also serve as shields.

NOTE 8 In French, the term "blindage" may be used when the main purpose of the screen is the protection from external electrical influence.

- **shield_connector:** or screen connector, device used to make a connection to the screen or shield of a cable for the purpose of continuity or earthing [IEV ref 461-17-12].
- **shielded_connector:** connector designed to prevent the radiation of electromagnetic interference to and from the internal conductor(s) [IEV ref 581-26-19].
- **shrink_boot:** boot that is designed to shrink to a predefined ratio to form a tight, seamless bond around wires and cables.
- **socket:** connector attached to an apparatus, or to a constructional element or alike [IEV ref 151-12-20].

NOTE 9 Contact members of a socket may be socket contacts, pin contacts or both.

- **spacer:** a device which keeps the sub-conductors of a bundle in a given geometrical configuration [IEC ref 60050-466].
- **splice:** connecting device with barrel(s) accommodating conductor(s) with or without additional provision to accommodate and secure the insulation [IEV ref 581-24-19].
- **strain_relief_accessory:** connector accessory to guide and provide strain relief to wires and cables.
- **tape:** sheeting or plastic film of limited width and in long continuous lengths [IEV ref 212-15-03].
- **terminal_block:** part of a component or an accessory, used to support and protect the wires or cable at the point of entry; it may also prevent the ingress of moisture or contaminants [IEV ref 581-26-26].
- **terminal_lug:** metallic device to connect a cable conductor to other electrical equipment [IEV ref 461-17-01].
- **wire:** flexible cylindrical conductor, with or without an insulating covering, the length of which is large with respect to its cross-sectional dimensions [IEV ref 151-12-28].

NOTE 10 The cross-section of a wire may have any shape, but the term "wire" is not generally used for ribbons or tapes.

- **wiring_harness:** assembly with a harness topology, consisting of cables or wires to enable electrical or optical connectivity, grouped together in one or more harness segments, each between two harness nodes in which the cables or wires either switch to other harness nodes or in which the cables and wires ends in connectors, contact members or terminal lugs.

Part, PartCategory & PartVersion (8 of 8)

AP242 part category types that are not recommended to be use for EWH:

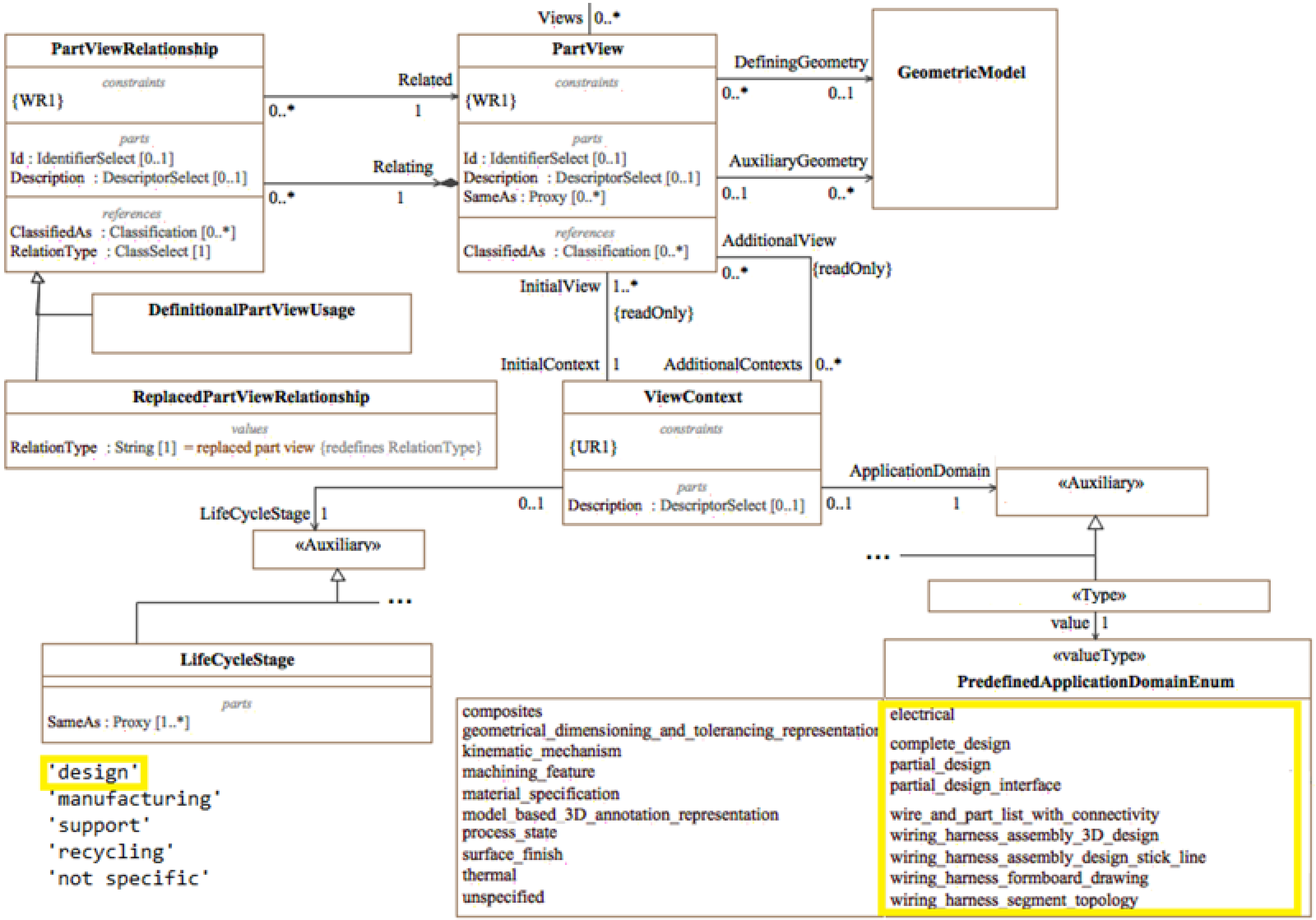
- application_control
- completely_knocked_down
- continuous
- detail (check with the CAx recommended practises)
- in_process
- inseparable_assembly
- prototype
- segulated
- safety
- service

Table of Content

- Introduction
- Basic XML structure
- Part, PartCategory & PartVersion
- PartView & ViewContext
- ProductClass & ProductConfiguration
- Occurrence, Assembly & WiringHarnessAssemblyDesign
- ShapeElement
- Geometric & topological models
- Connectors & terminals
- Features, Definitions & AssemblyShapeJoints
- Wire, Cables & TransportFeature
- Shapes/ContactFeatureDefinition & -Elements
- CrossSection of Wire, Cable & HarnessSegment
- Geometry/Topology Associations
- Complex Connectors / Assembly Hierarchy
- External Element References
- ComposedGeometricModel

PartView & ViewContext (1 of 5)

- New ApplicationDomains for EWH



Views 0..*

PartViewRelationship
constraints
{WR1}
parts
Id : IdentifierSelect [0..1]
Description : DescriptorSelect [0..1]
references
ClassifiedAs : Classification [0..*]
RelationType : ClassSelect [1]

Related 0..* 1

Relating 0..* 1

PartView
constraints
{WR1}
parts
Id : IdentifierSelect [0..1]
Description : DescriptorSelect [0..1]
SameAs : Proxy [0..*]
references
ClassifiedAs : Classification [0..*]

DefiningGeometry 0..* 0..1

AuxiliaryGeometry 0..1 0..*

GeometricModel

AdditionalView 0..* {readOnly}

InitialView 1..* {readOnly}

DefinitionalPartViewUsage

InitialContext 1

AdditionalContexts 0..*

ReplacedPartViewRelationship
values
RelationType : String [1] = replaced part view {redefines RelationType}

ViewContext
constraints
{UR1}
parts
Description : DescriptorSelect [0..1]

ApplicationDomain 0..1 1

«Auxiliary»

LifeCycleStage 1 0..1

«Auxiliary»

...

...

«Type»

value 1

LifeCycleStage
parts
SameAs : Proxy [1..*]

«valueType»
PredefinedApplicationDomainEnum

'design'
'manufacturing'
'support'
'recycling'
'not specific'

composites
geometrical_dimensioning_and_tolerancing_representatio
kinematic_mechanism
machining_feature
material_specification
model_based_3D_annotation_representation
process_state
surface_finish
thermal
unspecified

electrical
complete_design
partial_design
partial_design_interface
wire_and_part_list_with_connectivity
wiring_harness_assembly_3D_design
wiring_harness_assembly_design_stick_line
wiring_harness_formboard_drawing
wiring_harness_segment_topology

PartView & ViewContext (2 of 5)

Example: Part/PartVersion with a WiringHarnessAssemblyDesign View

Below is an example of *Part/PartVersion* with a *WiringHarnessAssemblyDesign* View where the *InitialContext* has the application domain "electrical" with the life-cycle stage "design" and the *AdditionalContext* has the application domain "wiring_harness_segment_topology" for the life-cycle stage "design". XML examples of these components are shown on the next page.

```
<Part uid="_311000"> <!-- Part_H1 -->
  <Id id="Part_H1"/>
  <Name>
    <CharacterString>Electrical Harness example 1</CharacterString>
  </Name>
  <PartTypes>
    <PartCategoryEnum>assembly</PartCategoryEnum>
    <PartCategoryEnum>wiring_harness</PartCategoryEnum>
  </PartTypes>
  <Versions>
    <PartVersion uid="_311001">
      <Id id="/NULL"/>
      <Views>
        <PartView xsi:type="n0:WiringHarnessAssemblyDesign" uid="_311002">
          <InitialContext uidRef="_100102"/>
          <AdditionalContexts>
            <ViewContext uidRef="_100101"/>
            <ViewContext uidRef="_100104"/>
            <ViewContext uidRef="_100105"/>
          </AdditionalContexts>
          …
        </PartView>
      <Views>
    <PartVersion>
  <Versions>
<Part>
```

PartView & ViewContext (3 of 5)

Example: *ViewContexts*

InitialContext

```
<ViewContext uid="_100102">
 <ApplicationDomain>
  <PredefinedApplicationDomainEnum>electrical</PredefinedApplicationDomainEnum>
 </ApplicationDomain>
 <LifeCycleStage>
  <ProxyString>design</ProxyString>
 </LifeCycleStage>
</ViewContext>
```

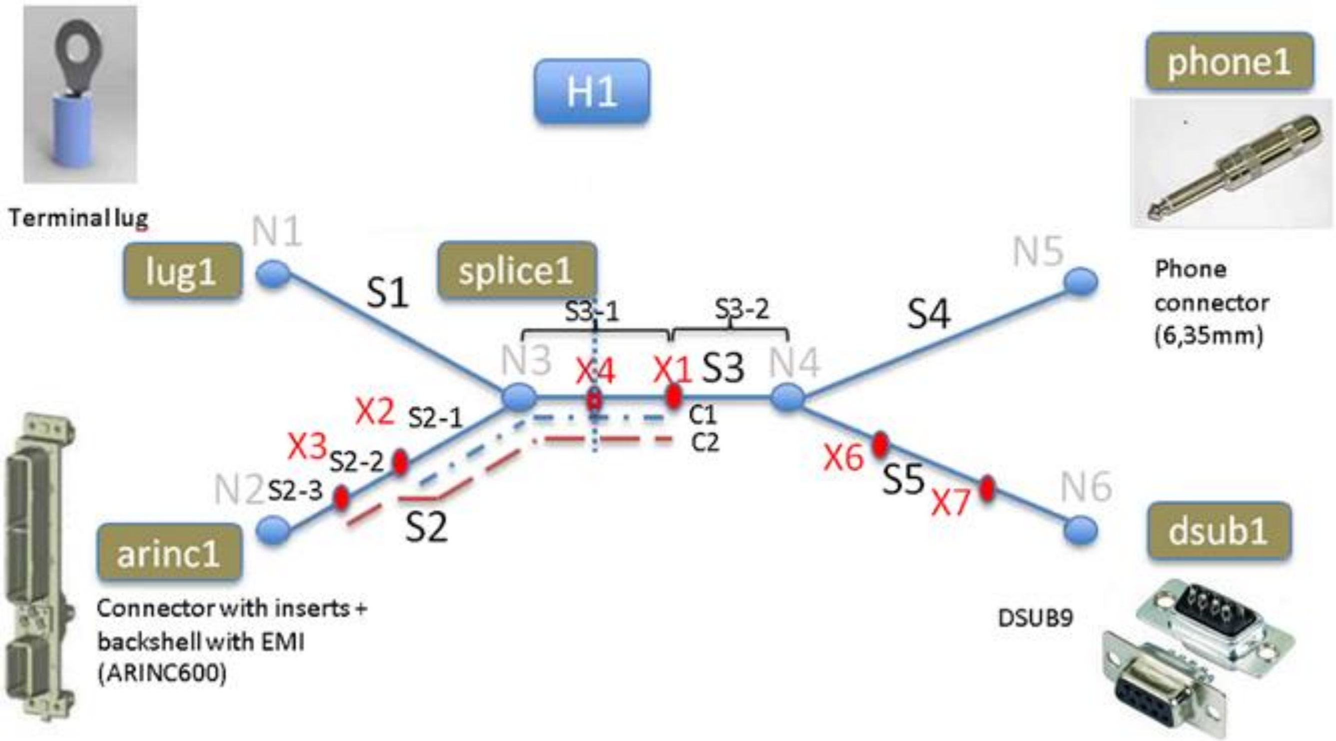
AdditionalContexts

```
<ViewContext uid="_100104">
 <ApplicationDomain>
  <PredefinedApplicationDomainEnum>wiring_harness_segment_topology</PredefinedApplicationDomainEnum>
 </ApplicationDomain>
 <LifeCycleStage>
  <ProxyString>design</ProxyString>
 </LifeCycleStage>
</ViewContext>
<ViewContext uid="_100105">
 <ApplicationDomain>
  <PredefinedApplicationDomainEnum>wire_and_part_list_with_connectivity</PredefinedApplicationDomainEnum>
 </ApplicationDomain>
 <LifeCycleStage>
  <ProxyString>design</ProxyString>
 </LifeCycleStage>
</ViewContext>
```

PartView & ViewContext (4 of 5)

Example: ApplicationDomain wiring_harness_segment_topology

- Consists of harness segments and nodes in a topological defined arrangement
- Consists of all the connectors and splices
- Consists of special partial coatings, shields, fixture places …
- 2D or 3D geometry, wires, cables and connectivity information might not be or only partially provided.

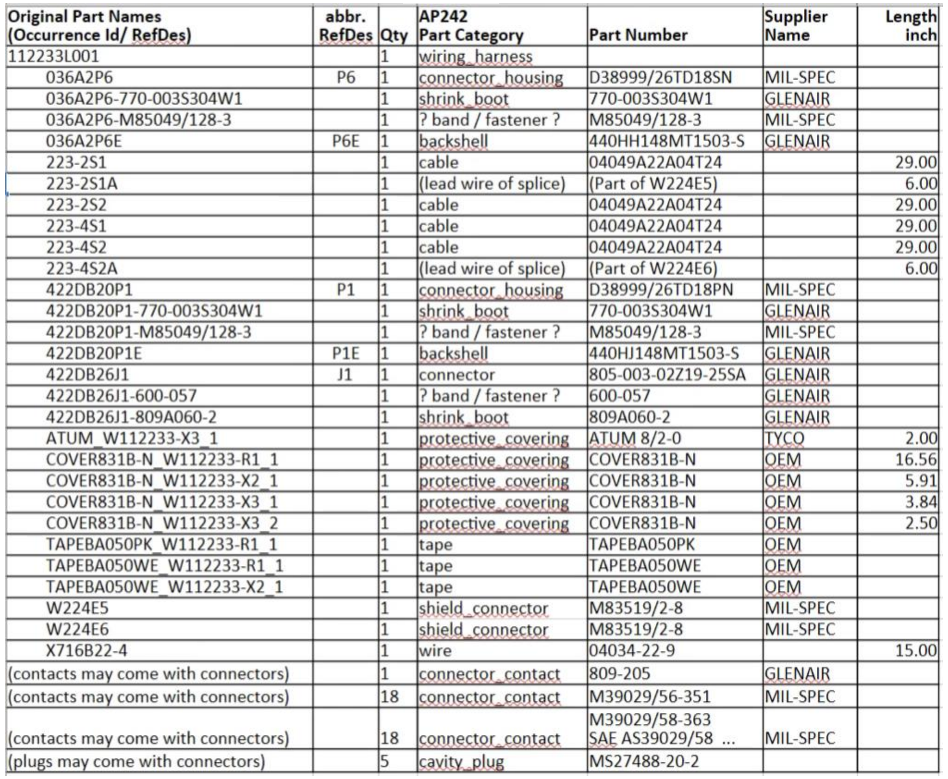


PartView & ViewContext (5 of 5)

Example: ApplicationDomain wire_and_part_list_with_connectivity

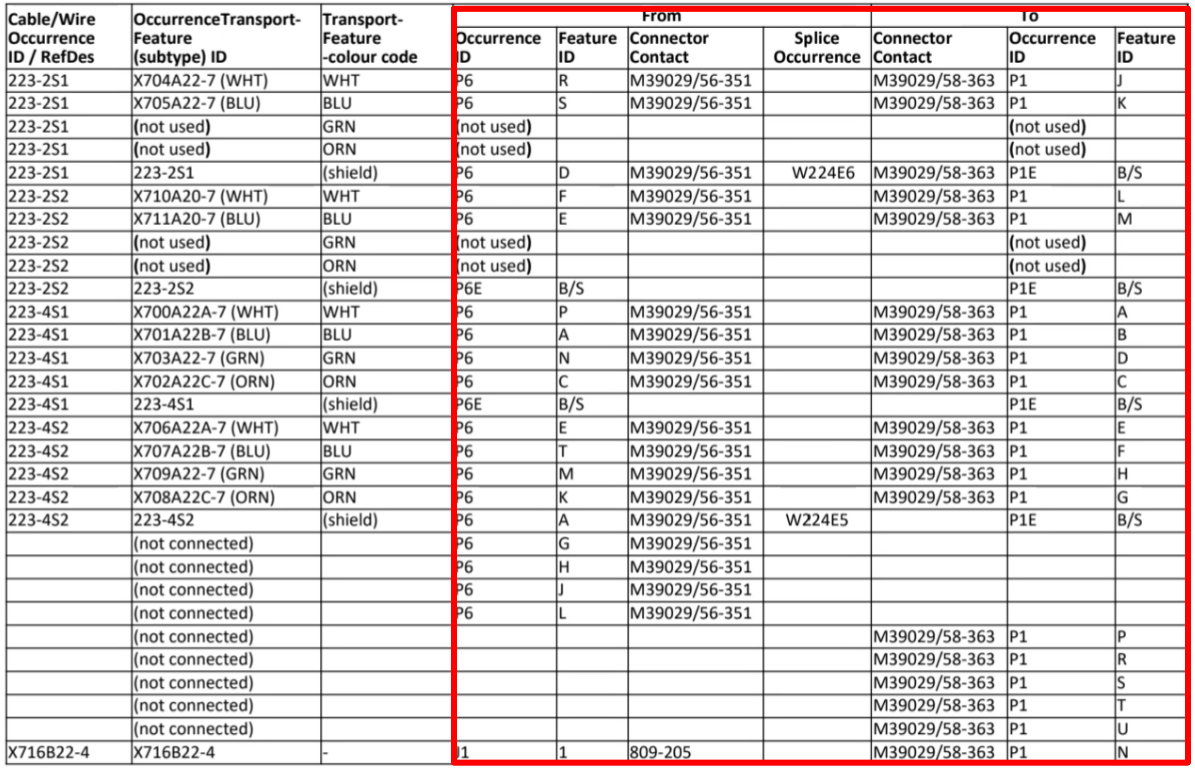
- Captures the connectivity information in a schematics diagram, which wires, and connector/device terminals are connected, but not the lines & symbols the diagram is made of.
- No information which wires are bundled together into a harness segment
- Example consisting of the Bill Of Material (BOM) and connectivity in the form of from/to with indicated wires/cables and splices

BOM



Original Part Names (Occurrence Id/ RefDes)	abbr. RefDes	Qty	AP242 Part Category	Part Number	Supplier Name	Length inch
112233L001		1	wiring_harness			
036A2P6	P6	1	connector_housing	D38999/26TD18SN	MIL-SPEC	
036A2P6-770-003S304W1		1	shrink_boot	770-003S304W1	GLENAIR	
036A2P6-M85049/128-3		1	? band / fastener ?	M85049/128-3	MIL-SPEC	
036A2P6E	P6E	1	backshell	440HH148MT1503-S	GLENAIR	
223-2S1		1	cable	04049A22A04T24		29.00
223-2S1A		1	(lead wire of splice)	(Part of W224E5)		6.00
223-2S2		1	cable	04049A22A04T24		29.00
223-4S1		1	cable	04049A22A04T24		29.00
223-4S2		1	cable	04049A22A04T24		29.00
223-4S2A		1	(lead wire of splice)	(Part of W224E6)		6.00
422DB20P1	P1	1	connector_housing	D38999/26TD18PN	MIL-SPEC	
422DB20P1-770-003S304W1		1	shrink_boot	770-003S304W1	GLENAIR	
422DB20P1-M85049/128-3		1	? band / fastener ?	M85049/128-3	MIL-SPEC	
422DB20P1E	P1E	1	backshell	440HJ148MT1503-S	GLENAIR	
422DB26J1	J1	1	connector	805-003-02Z19-25SA	GLENAIR	
422DB26J1-600-057		1	? band / fastener ?	600-057	GLENAIR	
422DB26J1-809A060-2		1	shrink_boot	809A060-2	GLENAIR	
ATUM_W112233-X3_1		1	protective_covering	ATUM 8/2-0	TYCO	2.00
COVER831B-N_W112233-R1_1		1	protective_covering	COVER831B-N	OEM	16.56
COVER831B-N_W112233-X2_1		1	protective_covering	COVER831B-N	OEM	5.91
COVER831B-N_W112233-X3_1		1	protective_covering	COVER831B-N	OEM	3.84
COVER831B-N_W112233-X3_2		1	protective_covering	COVER831B-N	OEM	2.50
TAPEBA050PK_W112233-R1_1		1	tape	TAPEBA050PK	OEM	
TAPEBA050WE_W112233-R1_1		1	tape	TAPEBA050WE	OEM	
TAPEBA050WE_W112233-X2_1		1	tape	TAPEBA050WE	OEM	
W224E5		1	shield_connector	M83519/2-8	MIL-SPEC	
W224E6		1	shield_connector	M83519/2-8	MIL-SPEC	
X716B22-4		1	wire	04034-22-9		15.00
(contacts may come with connectors)		1	connector_contact	809-205	GLENAIR	
(contacts may come with connectors)		18	connector_contact	M39029/56-351	MIL-SPEC	
(contacts may come with connectors)		18	connector_contact	M39029/58-363 SAE AS39029/58 ...	MIL-SPEC	
(plugs may come with connectors)		5	cavity_plug	MS27488-20-2		

From/to connections



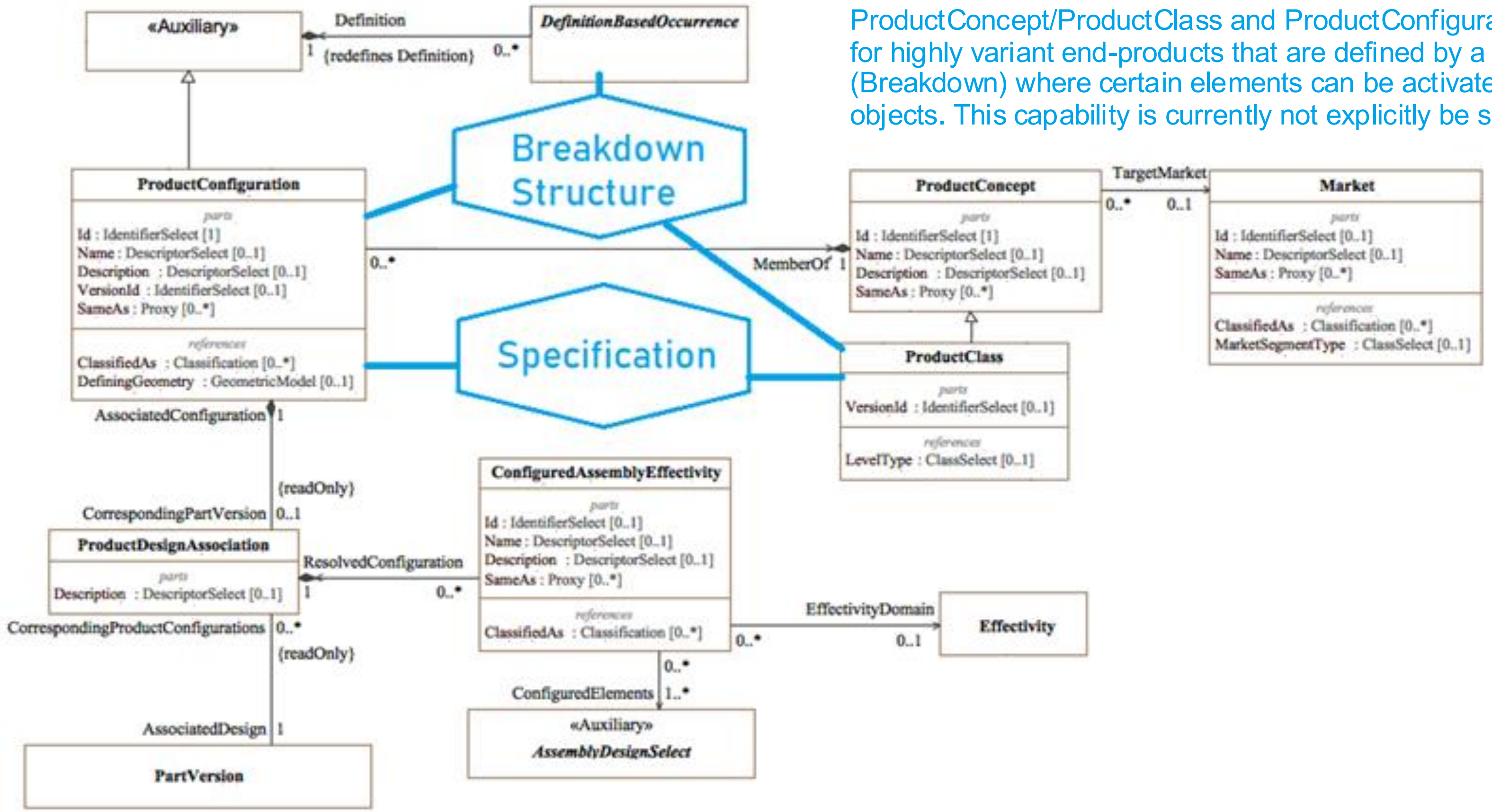
			From				To		
Cable/Wire Occurrence ID / RefDes	OccurrenceTransport-Feature (subtype) ID	Transport-Feature -colour code	Occurrence ID	Feature ID	Connector Contact	Splice Occurrence	Connector Contact	Occurrence ID	Feature ID
223-2S1	X704A22-7 (WHT)	WHT	P6	R	M39029/56-351		M39029/58-363	P1	J
223-2S1	X705A22-7 (BLU)	BLU	P6	S	M39029/56-351		M39029/58-363	P1	K
223-2S1	(not used)	GRN	(not used)					(not used)	
223-2S1	(not used)	ORN	(not used)					(not used)	
223-2S1	223-2S1	(shield)	P6	D	M39029/56-351	W224E6	M39029/58-363	P1E	B/S
223-2S2	X710A20-7 (WHT)	WHT	P6	F	M39029/56-351		M39029/58-363	P1	L
223-2S2	X711A20-7 (BLU)	BLU	P6	E	M39029/56-351		M39029/58-363	P1	M
223-2S2	(not used)	GRN	(not used)					(not used)	
223-2S2	(not used)	ORN	(not used)					(not used)	
223-2S2	223-2S2	(shield)	P6E	B/S				P1E	B/S
223-4S1	X700A22A-7 (WHT)	WHT	P6	P	M39029/56-351		M39029/58-363	P1	A
223-4S1	X701A22B-7 (BLU)	BLU	P6	A	M39029/56-351		M39029/58-363	P1	B
223-4S1	X703A22-7 (GRN)	GRN	P6	N	M39029/56-351		M39029/58-363	P1	D
223-4S1	X702A22C-7 (ORN)	ORN	P6	C	M39029/56-351		M39029/58-363	P1	C
223-4S1	223-4S1	(shield)	P6E	B/S				P1E	B/S
223-4S2	X706A22A-7 (WHT)	WHT	P6	E	M39029/56-351		M39029/58-363	P1	E
223-4S2	X707A22B-7 (BLU)	BLU	P6	T	M39029/56-351		M39029/58-363	P1	F
223-4S2	X709A22-7 (GRN)	GRN	P6	M	M39029/56-351		M39029/58-363	P1	H
223-4S2	X708A22C-7 (ORN)	ORN	P6	K	M39029/56-351		M39029/58-363	P1	G
223-4S2	223-4S2	(shield)	P6	A	M39029/56-351	W224E5		P1E	B/S
	(not connected)		P6	G	M39029/56-351				
	(not connected)		P6	H	M39029/56-351				
	(not connected)		P6	J	M39029/56-351				
	(not connected)		P6	L	M39029/56-351				
	(not connected)						M39029/58-363	P1	P
	(not connected)						M39029/58-363	P1	R
	(not connected)						M39029/58-363	P1	S
	(not connected)						M39029/58-363	P1	T
	(not connected)						M39029/58-363	P1	U
X716B22-4	X716B22-4	-	J1	1	809-205		M39029/58-363	P1	N

Table of Content

- Introduction
- Basic XML structure
- Part, PartCategory & PartVersion
- PartView & ViewContext
- ProductClass & ProductConfiguration
- Occurrence, Assembly & WiringHarnessAssemblyDesign
- ShapeElement
- Geometric & topological models
- Connectors & terminals
- Features, Definitions & AssemblyShapeJoints
- Wire, Cables & TransportFeature
- Shapes/ContactFeatureDefinition & -Elements
- CrossSection of Wire, Cable & HarnessSegment
- Geometry/Topology Associations
- Complex Connectors / Assembly Hierarchy
- External Element References
- ComposedGeometricModel

ProductClass and ProductConfiguration (1 of 2)

- An *Occurrence* might not only be defined by a *PartView* but can also be defined by a *ProductConfiguration* that is a member of a *ProductConcept/ProductClass*. This shall be used when a generic component is used without having a particular Part-Number.
- *ProductConfiguration* & *ProductDesignAssociation* might also be used for end-products such as an EWH with a few variations by a *PartView* that contains a so called 150% assembly structure. Here *ConfiguredAssemblyEffectivity* is used to define which assembly components and *PartShapeElements* are active for a particular *ProductConfiguration*.


«Auxiliary»
Definition
1
{redefines Definition}
0..*
DefinitionBasedOccurrence
Breakdown Structure
Specification
ProductConfiguration
parts
Id : IdentifierSelect [1]
Name : DescriptorSelect [0..1]
Description : DescriptorSelect [0..1]
VersionId : IdentifierSelect [0..1]
SameAs : Proxy [0..*]
references
ClassifiedAs : Classification [0..*]
DefiningGeometry : GeometricModel [0..1]
0..*
MemberOf 1
ProductConcept
parts
Id : IdentifierSelect [1]
Name : DescriptorSelect [0..1]
Description : DescriptorSelect [0..1]
SameAs : Proxy [0..*]
TargetMarket
0..*
0..1
Market
parts
Id : IdentifierSelect [0..1]
Name : DescriptorSelect [0..1]
SameAs : Proxy [0..*]
references
ClassifiedAs : Classification [0..*]
MarketSegmentType : ClassSelect [0..1]
ProductClass
parts
VersionId : IdentifierSelect [0..1]
references
LevelType : ClassSelect [0..1]
AssociatedConfiguration 1
{readOnly}
CorrespondingPartVersion 0..1
ProductDesignAssociation
parts
Description : DescriptorSelect [0..1]
ResolvedConfiguration
1
0..*
CorrespondingProductConfigurations 0..*
{readOnly}
AssociatedDesign 1
PartVersion
ConfiguredAssemblyEffectivity
parts
Id : IdentifierSelect [0..1]
Name : DescriptorSelect [0..1]
Description : DescriptorSelect [0..1]
SameAs : Proxy [0..*]
references
ClassifiedAs : Classification [0..*]
EffectivityDomain
0..*
0..1
Effectivity
0..*
ConfiguredElements 1..*
«Auxiliary»
AssemblyDesignSelect

ProductConcept/ProductClass and ProductConfiguration might also be used for highly variant end-products that are defined by a conceptional product structure (Breakdown) where certain elements can be activated through associated Specification objects. This capability is currently not explicitly be supported for EWH.

ProductClass and ProductConfiguration (2 of 2)

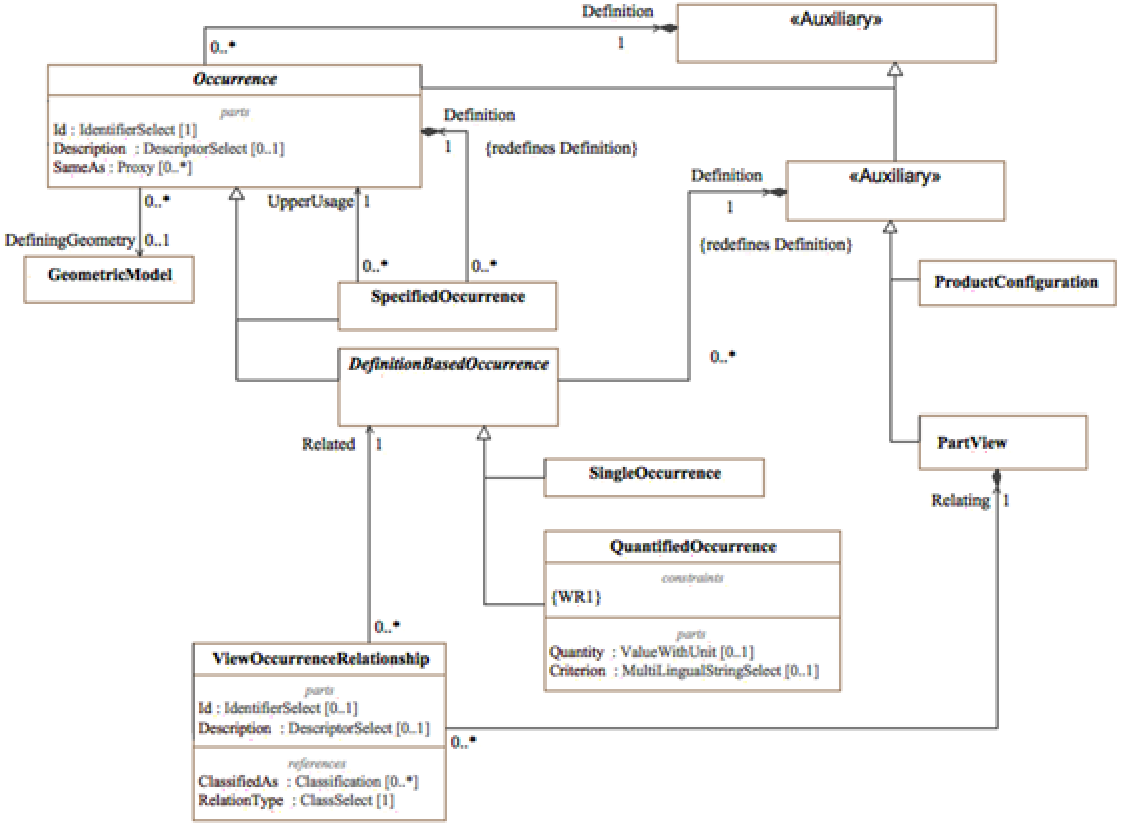
Example: *SingleOccurrence* defined by a *ProductConfiguration*

- A *SingleOccurrence* “battery1” for a standard battery without knowing a particular part number.

```
<ProductConcept xsi:type="n0:ProductClass" uid="_124000">
 <ClassifiedAs>
  <Classification uidRef="_100505"/>
 </ClassifiedAs>
 <Id id="Battery-Std"/>
 <Name>
  <CharacterString>Standard Battery</CharacterString>
 </Name>
 <ProductConfiguration uid="_124001">
  <Id id="Battery-Std"/>
  <Name>
   <CharacterString>Battery,12V,100Ah</CharacterString>
  </Name>
  <Occurrence xsi:type="n0:SingleOccurrence" uid="_224100">
   <Id id="battery1"/>
    …
  </Occurrence>
  …
 </ProductConfiguration>
</ProductConcept>
```

Occurrence, Assembly & WiringHarnessAssemblyDesign (1 of 7)

- *Occurrences* are the "components" of an *Assembly/WringHarnessAssemblyDesign* that are included by a *NextAssemblyOccurrenceUsage* relationship. *Occurrences* are defined by either a *PartView*, a *ProductConfiguration* or indirectly by another *Occurrence*. Several *Occurrences* might share the same definition but will be distinguished by different IDs (reference designations).
- An *Occurrence* is either a *SingleOccurrence*, a *QuantifiedOccurrence*, or in a case of a hierarchical product structure might also be a *SpecifiedOccurrence*.



Occurrence, Assembly & WiringHarnessAssemblyDesign (2 of 7)

Example: *SingleOccurrence*

- Definition of a discrete part that is categorized as terminal lug with two occurrences, “lug3” and “lug4”

```
<Part uid="_103000"> <!-- TERMINAL LUG CRIMP STYLE COPPER INSULATED RING TONGUE -->
 <Id id="MS5036-153"/>
 <Name>
  <LocalizedString lang="en-US">TERMINAL LUG CRIMP STYLE COPPER INSULATED RING TONGUE</LocalizedString>
  <LocalizedString lang="fr-FR">COSSE</LocalizedString>
 </Name>
 <PartTypes>
  <PartCategoryEnum>discrete</PartCategoryEnum>
  <PartCategoryEnum>terminal_lug</PartCategoryEnum>
 </PartTypes>
 <Versions>
  <PartVersion uid="_103001">
   <Id id=”Version 1"/>
   <Views>
     <PartView uid="_103002">
      <DefiningGeometry uidRef="_103090"/>
      <InitialContext uidRef="_100102"/>
      <Occurrence xsi:type="n0:SingleOccurrence" uid="_203205">
       <Id id="lug3"/>
                  </Occurrence>
      <Occurrence xsi:type="n0:SingleOccurrence" uid="_203305">
       <Id id="lug4"/>
                  </Occurrence>
     </PartView>
   </View>
  </PartVersion>
 </Version>
</Part>
```

Occurrence, Assembly & WiringHarnessAssemblyDesign (3 of 7)

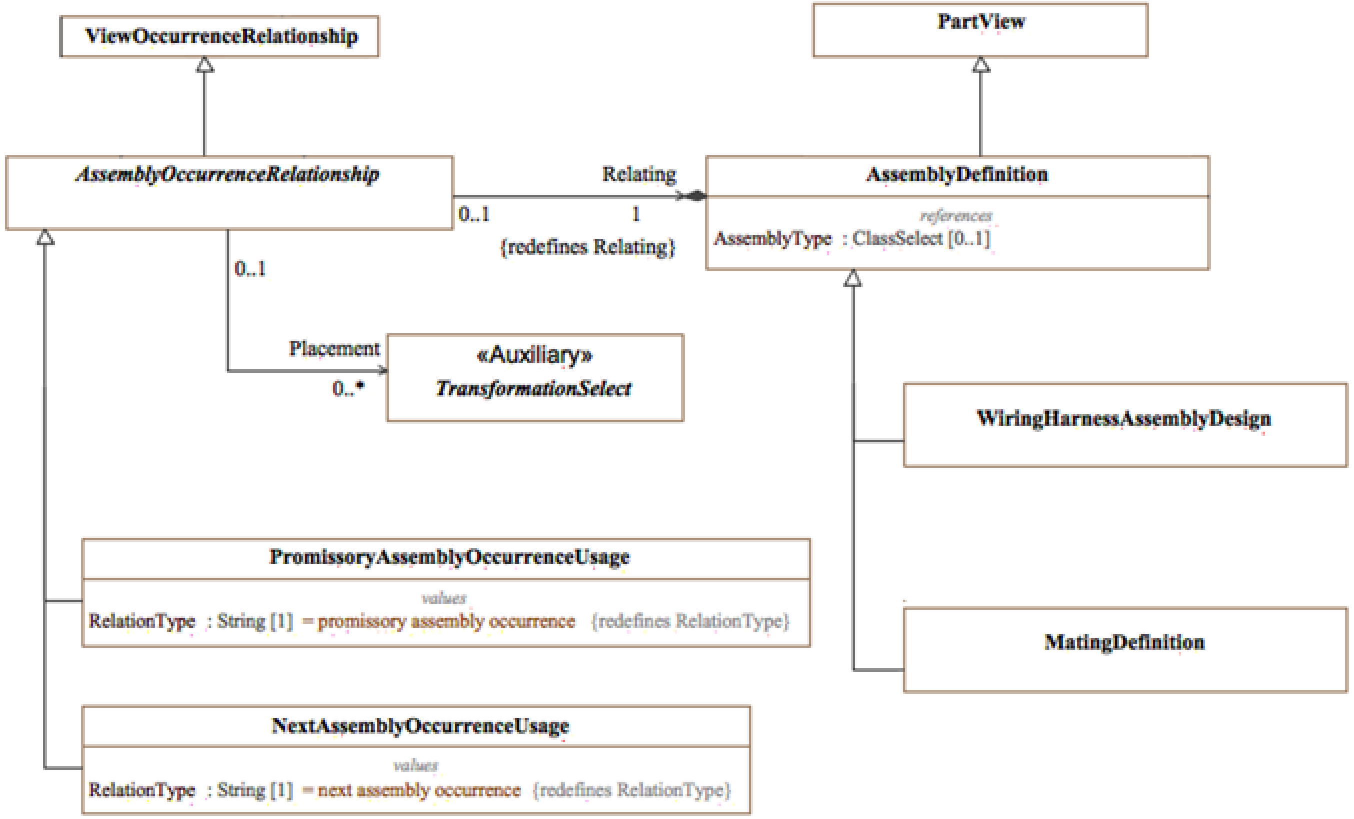
Example: *QuantifiedOccurrence*

- A raw material part that is measured by length and categorized as “protective covering”
- The *QuantifiedOccurrence* “wrap1” of this part has a value of “1.75” and the reference length unit is metre.

```
<Part uid="_121000"> <!-- Wrap -->
  <Id id="SPW-500SP-BK"/>
  <Name>
    <CharacterString>1/2 inch Spiral Wrap - 100 foot spool - Black</CharacterString>
  </Name>
  <PartTypes>
    <PartCategoryEnum>protective_covering</PartCategoryEnum>
    <PartCategoryEnum>raw_material_by_length</PartCategoryEnum>
  </PartTypes>
  <Versions>
    <PartVersion uid="_121001">
      <Id id=”Version 1"/>
      <Views>
        <PartView uid="_121002">
          <DefiningGeometry uidRef="_104890"/>
          <InitialContext uidRef="_100102"/>
          <Occurrence xsi:type="n0:QuantifiedOccurrence" uid="_221005">
            <Id id="wrap1"/>
            <Quantity xsi:type="n0:NumericalValue" uid="_221007">
              <Unit uidRef="_100301"/>
              <ValueComponent>1.75</ValueComponent>
            </Quantity>
          </Occurrence>
        </PartView>
      </View>
    </PartVersion>
  </Version>
</Part>
```

Occurrence, Assembly & WiringHarnessAssemblyDesign (4 of 7)

- An *AssemblyDefinition* is a type of *PartView* that contains *Occurrences* (single or quantified) through either:
  - *NextAssemblyOccurrenceUsage* (default)
  - *PromissoryAssemblyOccurrenceUsage (special, e.g. “Poke Home”)*
- The focus for this tutorial is subtype ***WiringHarnessAssemblyDesign***


ViewOccurrenceRelationship
PartView
AssemblyOccurrenceRelationship
Relating
0..1
1
{redefines Relating}
AssemblyDefinition
references
AssemblyType : ClassSelect [0..1]
0..1
Placement
0..*
«Auxiliary»
TransformationSelect
WiringHarnessAssemblyDesign
PromissoryAssemblyOccurrenceUsage
values
RelationType : String [1] = promissory assembly occurrence {redefines RelationType}
MatingDefinition
NextAssemblyOccurrenceUsage
values
RelationType : String [1] = next assembly occurrence {redefines RelationType}

Occurrence, Assembly & WiringHarnessAssemblyDesign (5 of 7)

Example: Simple/Flat Assembly Structure

- The *QuantifiedOccurrence* “wrap1” and the *SingleOccurrences* “lug3” and “lug4” become members of the “Part_H1” assembly that is a *WiringHarnessAssemblDesign*

```
<Part uid="_311000"> <!-- Wrap -->
 <Id id=”Part_H1"/>
 …
 <Versions>
  <PartVersion uid="_311001">
   …
    <Views>
     <PartView xsi:type="n0:WiringHarnessAssemblyDesign" uid="_311002">
      …
      <ViewOccurrenceRelationship uid="_315032" xsi:type="n0:NextAssemblyOccurrenceUsage">
       <Related uidRef="_221005"/> <!-- wrap1 -->
        …
      </ViewOccurrenceRelationship>
      <ViewOccurrenceRelationship uid="_315047" xsi:type="n0:NextAssemblyOccurrenceUsage">
       <Related uidRef="_203205"/> <!– lug3 -->
        …
      </ViewOccurrenceRelationship>
      <ViewOccurrenceRelationship uid="_315048" xsi:type="n0:NextAssemblyOccurrenceUsage">
       <Related uidRef="_203305"/> <!– lug4 -->
        …
      </ViewOccurrenceRelationship>
      …
     </PartView>
    </View>
  </PartVersion>
 </Versions>
</Part>
```

Occurrence, Assembly & WiringHarnessAssemblyDesign (6 of 7)

PropertyValue with Definition (1 of 2)

Occurrences may be marked by a label using ‘occurrence marking property’ that is defined in **PropertyDefinitionEnum**:

- **occurrence marking property**: Properties that is a visible impression or trace on an Occurrence.
- EXAMPLE: An occurrence of a wire in an electrical wire harness may be marked with a text string on a printed label.
- NOTE: occurrence marking property may used to implement the requirement from IEC 62491 Industrial systems, installations and equipment and industrial products – Labelling of cables and cores

```
<PropertyValueAssignment uid="_202009">
 <AssignedPropertyValues>
  <PropertyValue xsi:type="n0:StringValue" uid="_202011">
   <Definition>
    <PropertyDefinitionString>occurrence marking property</PropertyDefinitionString>
   </Definition>
   <ValueComponent>
    <CharacterString>CABLE 1</CharacterString>
   </ValueComponent>
  </PropertyValue>
 </AssignedPropertyValues>
 <ClassifiedAs>
  <Classification uidRef="Cl__35"/>
 </ClassifiedAs>
</PropertyValueAssignment>
```

Occurrence, Assembly & WiringHarnessAssemblyDesign (7 of 7)

PropertyValue with Definition (2 of 2)

Two values for PropertyDefinitionEnum:

- **minimum bending radius:** Properties that represents the minimum allowed bend radius.
- **occurrence marking property:** Properties that is a visible impression or trace on an **Occurrence**.

EXAMPLE: An occurrence of a wire in an electrical wire harness may be marked with a text string on a printed label.

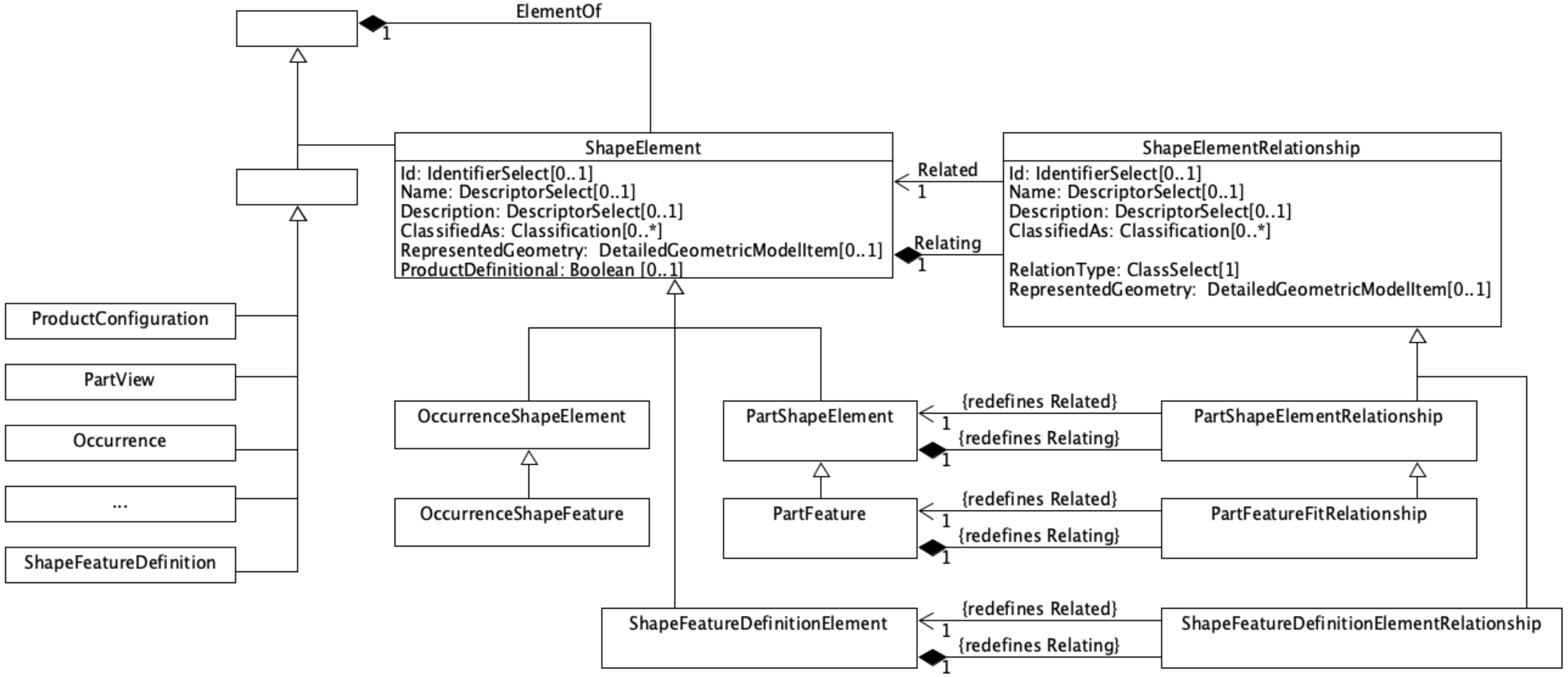
NOTE: occurrence marking property may used to implement the requirement from IEC 62491 Industrial systems, installations and equipment and industrial products – Labelling of cables and cores

Table of Content

- Introduction
- Basic XML structure
- Part, PartCategory & PartVersion
- PartView & ViewContext
- ProductClass & ProductConfiguration
- Occurrence, Assembly & WiringHarnessAssemblyDesign
- ShapeElement
- Geometric & topological models
- Connectors & terminals
- Features, Definitions & AssemblyShapeJoints
- Wire, Cables & TransportFeature
- Shapes/ContactFeatureDefinition & -Elements
- CrossSection of Wire, Cable & HarnessSegment
- Geometry/Topology Associations
- Complex Connectors / Assembly Hierarchy
- External Element References
- ComposedGeometricModel

ShapeElement (1 of 3)

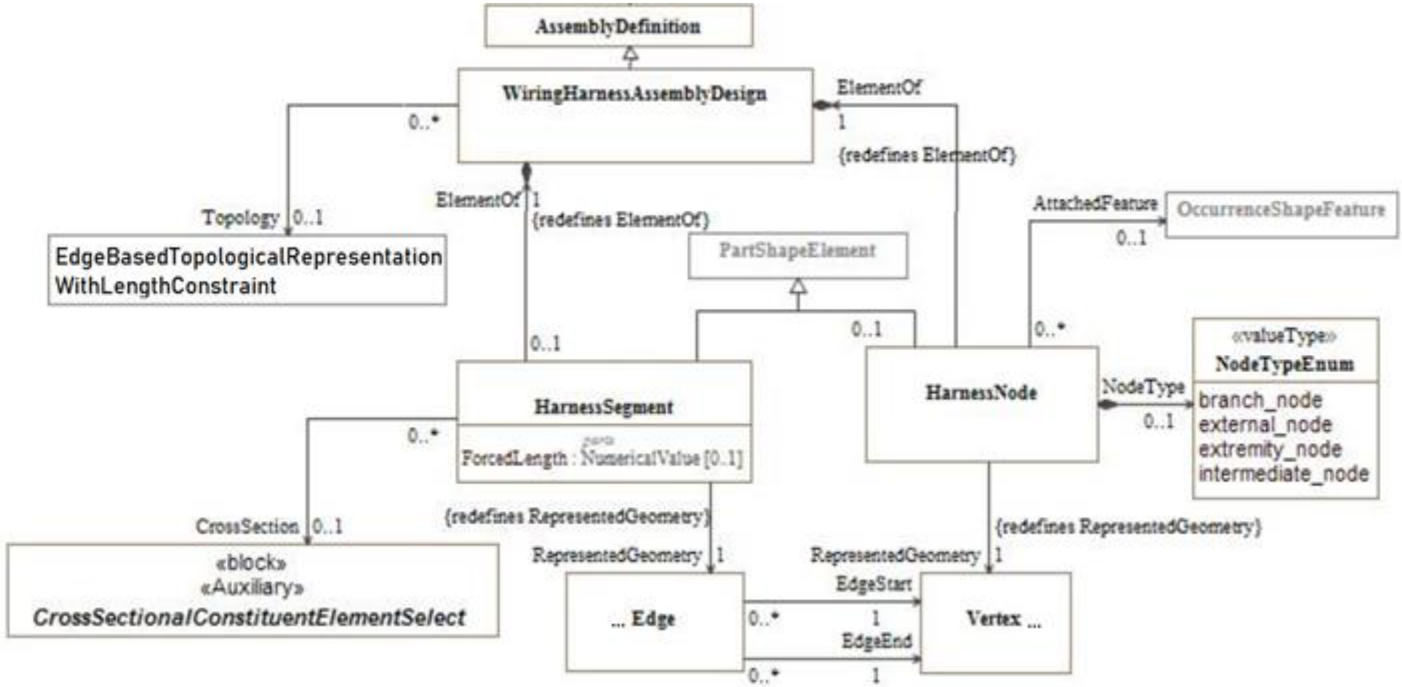
- A *ShapeElement* is the identification of an element of the shape of a *ProductConfiguration*, *PartView*, *Occurrence*, *ShapeFeatureDefinition* or of another *ShapeElement* identifying a sub-*ShapeElement* of the higher-level *ShapeElement*.
  - Depending of the target object (ElementOf attribute) we distinguish between *PartShapeElement*, *OccurrenceShapeElement* and *ShapeFeatureDefinitionElement*
  - A *PartShapeElement* is an element of either a *PartView* or of a *ProductConfiguration*
    - further subtypes are defined for contact and transport features, terminals, assembly joints and constraints and higher-level connectivity
    - some types of *ShapeElements* might be defined by other *ShapeElement* types or by a *ShapeFeatureDefinition*
  - *ShapeFeatureDefinitions* and also *ShapeElements* might be related to each other to define simple and hierarchical complex fit relationships with each other


ElementOf
1
ShapeElement
Id: IdentifierSelect[0..1]
Name: DescriptorSelect[0..1]
Description: DescriptorSelect[0..1]
ClassifiedAs: Classification[0..*]
RepresentedGeometry: DetailedGeometricModelItem[0..1]
ProductDefinitional: Boolean [0..1]
Related
1
Relating
1
ShapeElementRelationship
Id: IdentifierSelect[0..1]
Name: DescriptorSelect[0..1]
Description: DescriptorSelect[0..1]
ClassifiedAs: Classification[0..*]
RelationType: ClassSelect[1]
RepresentedGeometry: DetailedGeometricModelItem[0..1]
ProductConfiguration
PartView
Occurrence
...
ShapeFeatureDefinition
OccurrenceShapeElement
OccurrenceShapeFeature
PartShapeElement
PartFeature
ShapeFeatureDefinitionElement
{redefines Related}
{redefines Relating}
1
PartShapeElementRelationship
PartFeatureFitRelationship
ShapeFeatureDefinitionElementRelationship

ShapeElement (2 of 3)

PartShapeElements for WiringHarnessAssemblyDesign

- A *WiringHarnessAssemblyDesign* is an electrical assembly with a topological representation defined by a *EdgeBasedTopologicalRepresentationWithLengthConstraint*
- *PartShapeElements* of type *HarnessSegment* and *HarnessNode* of an EWH are represented by subtypes of the topological elements *Edge* and *Vertex* respectively
- An *OccurrenceShapeFeature* (e.g. of a connector Occurrence) can be attached to a *HarnessNode*
- The detailed arrangement of the wires, cables, coverings … in a *HarnessSegment* can be defined by a *CrossSectionalConstituentElementSelect*
- *For other kinds of ShapeElement see page 55*



ShapeElement (3 of 3)

Example: *WiringHarnessAssemblyDesign* with *Topology*, *HarnessSegment* and *HarnessNode*

```
<Part uid="_311000"> <!-- Part_H1 -->
 <Id id="Part_H1"/>
 ...
 <Versions>
  <PartVersion uid="_311001">
   <Id><Id/>
   <Views>
    <PartView xsi:type="n0:WiringHarnessAssemblyDesign" uid="_311002">
   ...
    <!--HarnessSegment S2-2-->
    <ShapeElement xsi:type="n0:HarnessSegment" uid="_314001">
     ...
     <RepresentedGeometry uidRef="_321032"/>
     <CrossSection uidRef="_313007"/> <!-- heatshrink1 on braid1 ... -->
    </ShapeElement>
   ...
    <!--HarnessNode N1-->
    <ShapeElement xsi:type="n0:HarnessNode" uid="_314011">
     <RepresentedGeometry uidRef="_321041"/> <!--VertexPoint N1 -->
     <AttachedFeature uidRef="_203008"/> <!--lug1/Internal-->
     <NodeType>extremity_node</NodeType>
    </ShapeElement>
   ...
    <Topology uidRef="_321010" /> <!-- => EdgeBasedTopologicalRepresentationWithLengthConstraint -->
    </PartView>
   <Views>
  <PartVersion>
 <Versions>
<Part>
```

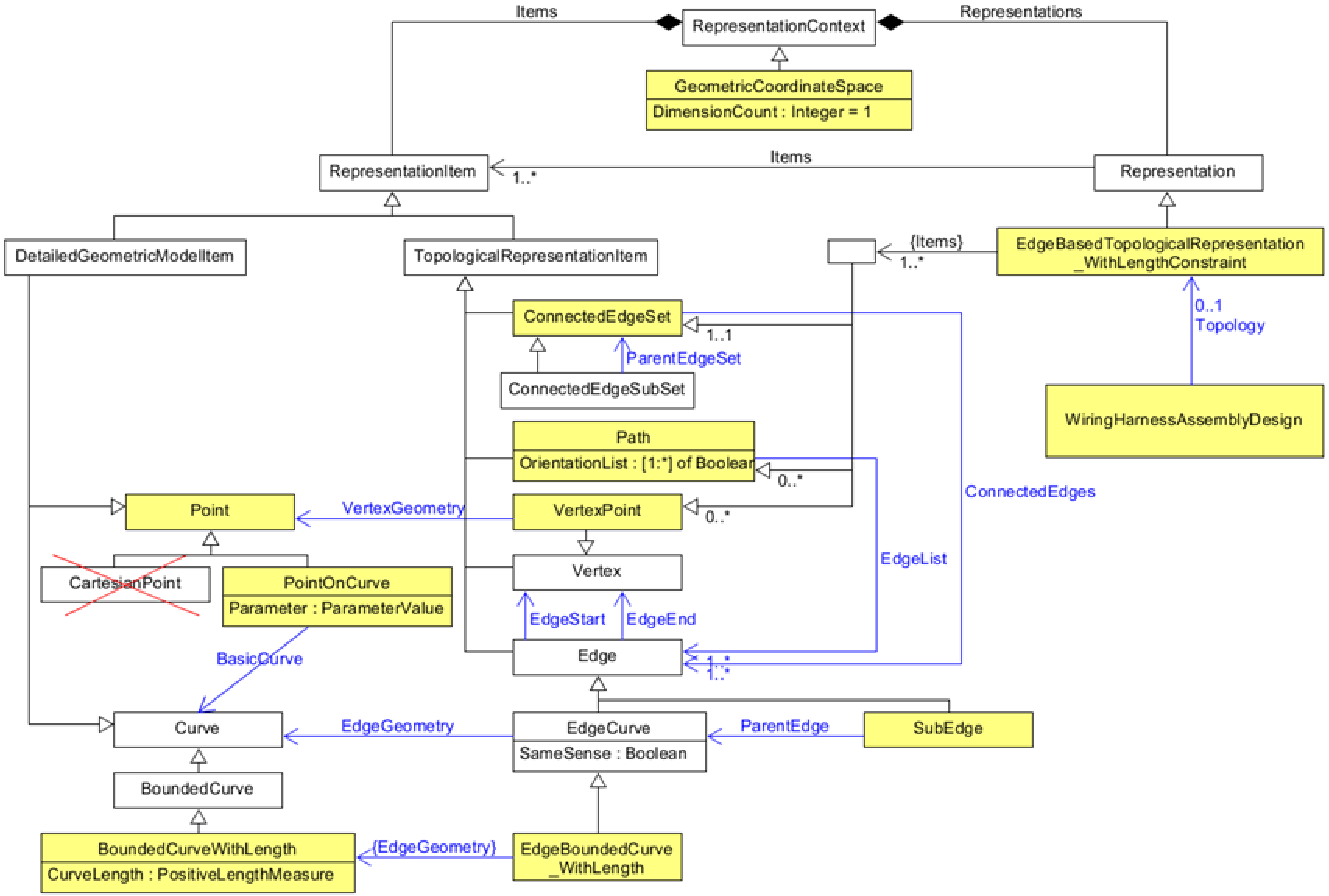
Note: further details on ShapeElement are provided later in this slide-set

Table of Content

- Introduction
- Basic XML structure
- Part, PartCategory & PartVersion
- PartView & ViewContext
- ProductClass & ProductConfiguration
- Occurrence, Assembly & WiringHarnessAssemblyDesign
- ShapeElement
- Geometric & topological models
- Connectors & terminals
- Features, Definitions & AssemblyShapeJoints
- Wire, Cables & TransportFeature
- Shapes/ContactFeatureDefinition & -Elements
- CrossSection of Wire, Cable & HarnessSegment
- Geometry/Topology Associations
- Complex Connectors / Assembly Hierarchy
- External Element References
- ComposedGeometricModel

EWIS Interoperability Forum

Geometric & topological models (1 of 6)



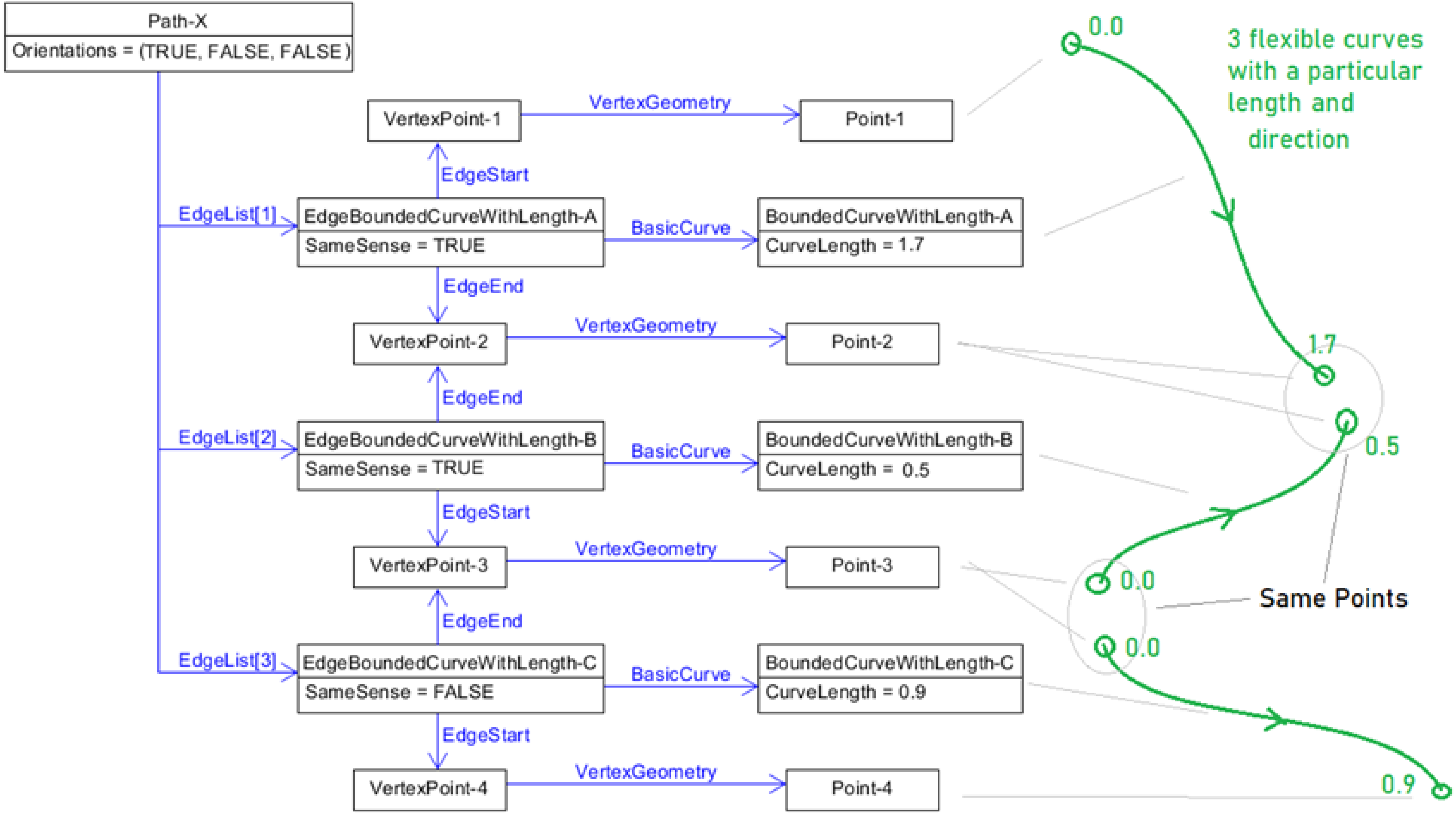
Geometric & topological models (2 of 6)

- An *EdgeBasedTopologicalRepresentationWithLengthConstraint* requires a context that is a *GeometricCoordinateSpace*.
- *GeometricCoordinateSpace* requires to specify a *DimensionCount*. As 2 (=2D) and 3 (=3D) are not suitable and the value must be greater 0, we recommend to use the value 1. But it must be clear that this does NOT mean that this is a 1D geometric coordinate space. There are e.g. no *CartesianPoints* with a single coordinate value (x-axis).
- An *EdgeBasedTopologicalRepresentationWithLengthConstraint* requires exactly one *ConnectedEdgeSet*.
- The *ConnectedEdgeSet* might be *ConnectedEdgeSubSet* of another *ConnectedEdgeSet* of another *EdgeBasedTopologicalRepresentationWithLengthConstraint* of another *WiringHarnessAssemblyDesign*. This extended capability supports the split of big projects, but is out of scope of current tests.
- The connected *Edges* of a *ConnectedEdgeSet* must all be of type *EdgeBoundedCurveWithLength* or in the case of a *ConnectedEdgeSubSet* might also be of type *SubEdge*.
- An *EdgeBoundedCurveWithLength* refers to a *BoundedCurveWithLength* that is a curve where only the length is known, but not any *CartesianPoint* or *Direction*. The start- and end-point of the *BoundedCurveWithLength* is identified by the *VertexPoints* that are referenced as *EdgeStart* and *EdgePoint*. For an *EdgeBoundedCurveWithLength* the *VertexGeometry* must be just *Point*. Subtypes such as *CartesianPoint* or *PointOnCurve* are not allowed.
- Other than the single *ConnectedEdgeSet*, an *EdgeBasedTopologicalRepresentationWithLengthConstraint* might have additional items of type *Path* or *VertexPoint*,
  - these additional *Paths* and *VertexPoints* must all be in the domain of the *ConnectedEdgeSet*.
  - *Paths* are used to define the flexible geometry of *QuantifiedOccurrences* with length such as wires, cables and protections
  - additional VertexPoints are used to define the position of clamps and splices,
    Note that additional VertexPoints that are needed to define SubEdges that are used by Paths (e.g. for protections) do not need to be added as they are founded already through the included Paths.
  - additional VertexPoints must refer to PointOnCurve where the underlying curve is one of the *BoundedCurveWithLength*
  - *Paths may be composed of complete EdgeBoundedCurveWithLengths or a parts of them by using SubEdge*

Geometric & topological models (3 of 6)

Direction control of Edges and Paths

- *the VertexPoints of an EdgeBoundedCurveWithLength corresponds to the start/end of the underlying BoundedCurveWithLengh that are given through its parametric definition*
- *common VertexPoints of several EdgeBoundedCurveWithLength constrain the underlying BoundedCurveWithLength to start/end at exactly the same Point*
- *the SameSens*
- *the Orientatio*



Geometric & topological models (4 of 6)

Example: *ConnectedEdgeWithLengthSet Representation*

- a GeometricCoordinateSpace with DimensonCount 1 contains all *RepresentationItems* and *Representations*
- the *EdgeBasedTopologicalRepresentationWithLengthConstraint* references a *ConnectedEdgeSet*
- … and that references an *EdgeBoundedCurveWithLength*
- … that references a *BoundedCurveWithLength* that has a length of 1.5 m
- *VertexPoints* not shown here

```
<RepresentationContext xsi:type="n0:GeometricCoordinateSpace" uid="_321000">
 <Id id="H1.x Harness topology context"/>
 <Units>
  <Unit uidRef="_100301"/>
 </Units>
 <Representations>
  <Representation xsi:type="n0:EdgeBasedTopologicalRepresentationWithLengthConstraint" uid="_321010">
   <Id id="Topological representation of H1 harness"/>
   <Items>
    <RepresentationItem uidRef="_321020"/>
    …
   </Items>
 </Representations>
 <Items>
  <RepresentationItem xsi:type="n0:ConnectedEdgeSubSet" uid="_321020">
   <ConnectedEdges>
    <Edge uidRef="_321021"/>
    …
   </ConnectedEdges>
  </RepresentationItem>
  <RepresentationItem xsi:type="n0:EdgeBoundedCurveWithLength" uid="_321021">
   <Name>
    <CharacterString>S1</CharacterString>
   </Name>
   <EdgeEnd uidRef="_321041"/> <!--VertexPoint N1 -->
   <EdgeStart uidRef="_321043"/> <!--VertexPoint N3 -->
   <EdgeGeometry uidRef="_341021"/>
   <SameSense>true</SameSense>
  </RepresentationItem>
  <RepresentationItem xsi:type="n0:BoundedCurveWithLength" uid="_341021">
   <CurveLength>1.5</CurveLength>
  </RepresentationItem>
  …
 </Items>
 <DimensionCount>1</DimensionCount>
</RepresentationContext>
```

Geometric & topological models (5 of 6)

Example: Defining SubEdges

- A *SubEdge* references a *VertexPoint* whose geometry is defined by a *PointOnCurve* that is placed in 2m distance from the start of the underlying *BoundedCurveWithLength*
- Limitation: There is no way to define the distance from the end

```
<RepresentationItem xsi:type="n0:SubEdge" uid="_321034">
 <Name>
  <CharacterString>S3-1</CharacterString>
 </Name>
 <EdgeEnd uidRef="_321043"/>
 <EdgeStart uidRef="_321051"/>
 <ParentEdge uidRef="_321023"/>
</RepresentationItem>
…
<RepresentationItem xsi:type="n0:VertexPoint" uid="_321051">
 <Name>
  <CharacterString>X1</CharacterString>
 </Name>
 <VertexGeometry uidRef="_341051"/>
</RepresentationItem>
<RepresentationItem xsi:type="n0:PointOnCurve" uid="_341051">
 <BasicCurve uidRef="_341023"/>
 <Parameter>0.8</Parameter>
</RepresentationItem>
…
<RepresentationItem xsi:type="n0:BoundedCurveWithLength" uid="_341023">
 <CurveLength>2.0</CurveLength>
</RepresentationItem>
```

Geometric & topological models (6 of 6)

Example: H1 Topology with Sub Topologies H1.a and H1.b

- Main *EdgeBoundedCurveWithLength* & *HarnessSegment* S1 … S5
- Main *VertexPoints/Point* & *HarnessNodes* N1 … N6
- Additional *VertexPoints/PointOnCurve* X1 … X7
- Additional *SubEdges* & *HarnessSegments* S2-1 … S3-b

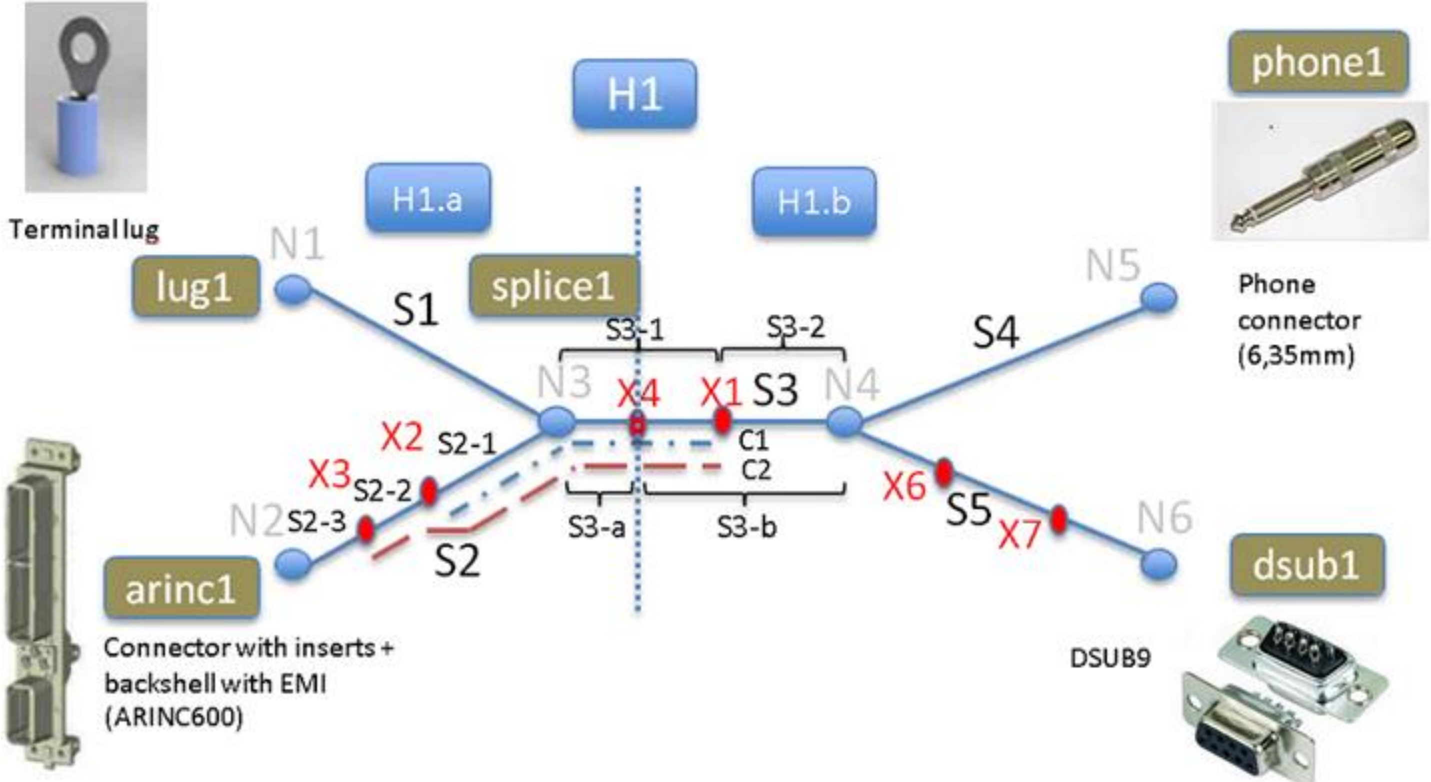


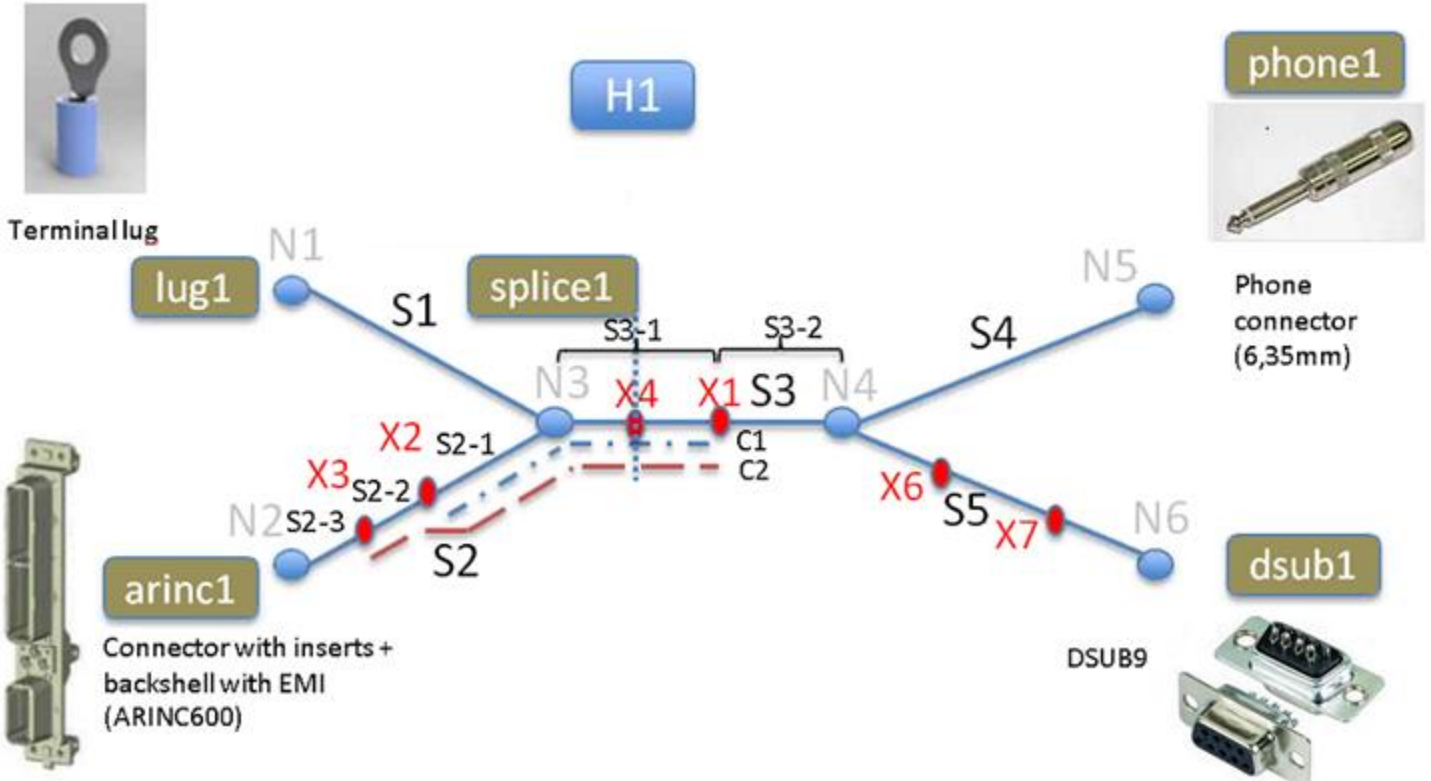
Table of Content

- Introduction
- Basic XML structure
- Part, PartCategory & PartVersion
- PartView & ViewContext
- ProductClass & ProductConfiguration
- Occurrence, Assembly & WiringHarnessAssemblyDesign
- ShapeElement
- Geometric & topological models
- **Connectors & terminals**
- Features, Definitions & AssemblyShapeJoints
- Wire, Cables & TransportFeature
- Shapes/ContactFeatureDefinition & -Elements
- CrossSection of Wire, Cable & HarnessSegment
- Geometry/Topology Associations
- Complex Connectors / Assembly Hierarchy
- External Element References
- ComposedGeometricModel

Connectors & terminals (1 of 9)

Different kind of connectors and terminals to support

- Terminal lugs with a single crimp or screw contact for typically one wire
- Audio connector with 2 (mono) or 3 (stereo) contacts
- Monolytical connectors such as a DSUB9 for welding
- Highly configurable connectors with very many variations such as the ARINC 600



Connectors & terminals (2 of 9)

Kinds of terminals

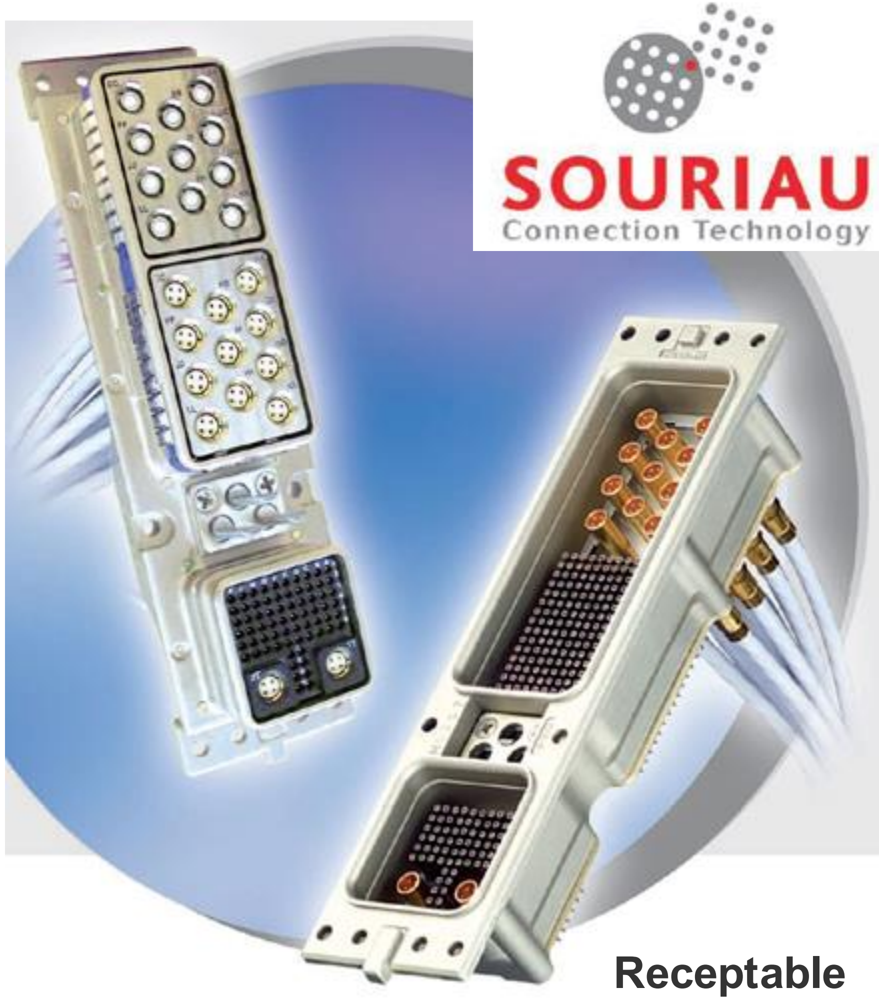
- In many cases a single contact for a connector or a terminal lug has:
  - one “join terminal” to be connected permanently by e.g. a wire or cable,
  - and one “interface terminal” to be connected and disconnected at a next higher level
  - Note: terminal busbars and splices have typically only “join contacts”
- Terminals are joined with other stuff by crimping, welding, screwing, ...
- Contacts are typically available as mating pairs (pin & socket)
- Connecto ug & re

Connectors & terminals (3 of 9)

Typical connector for aircrafts: ARINC 600

- A very modular and highly configurable family of connectors
- available

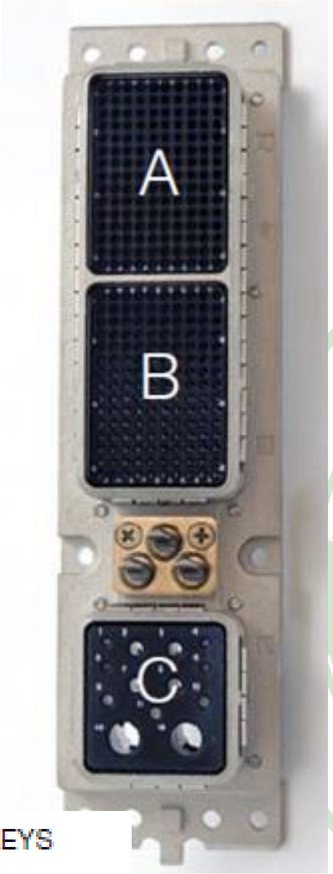
Plug



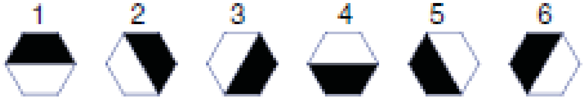
Receptable

Inserts A, B, C

Polarisation keys



LOCATION OF POLARIZATION KEYS
(view from engaging face)

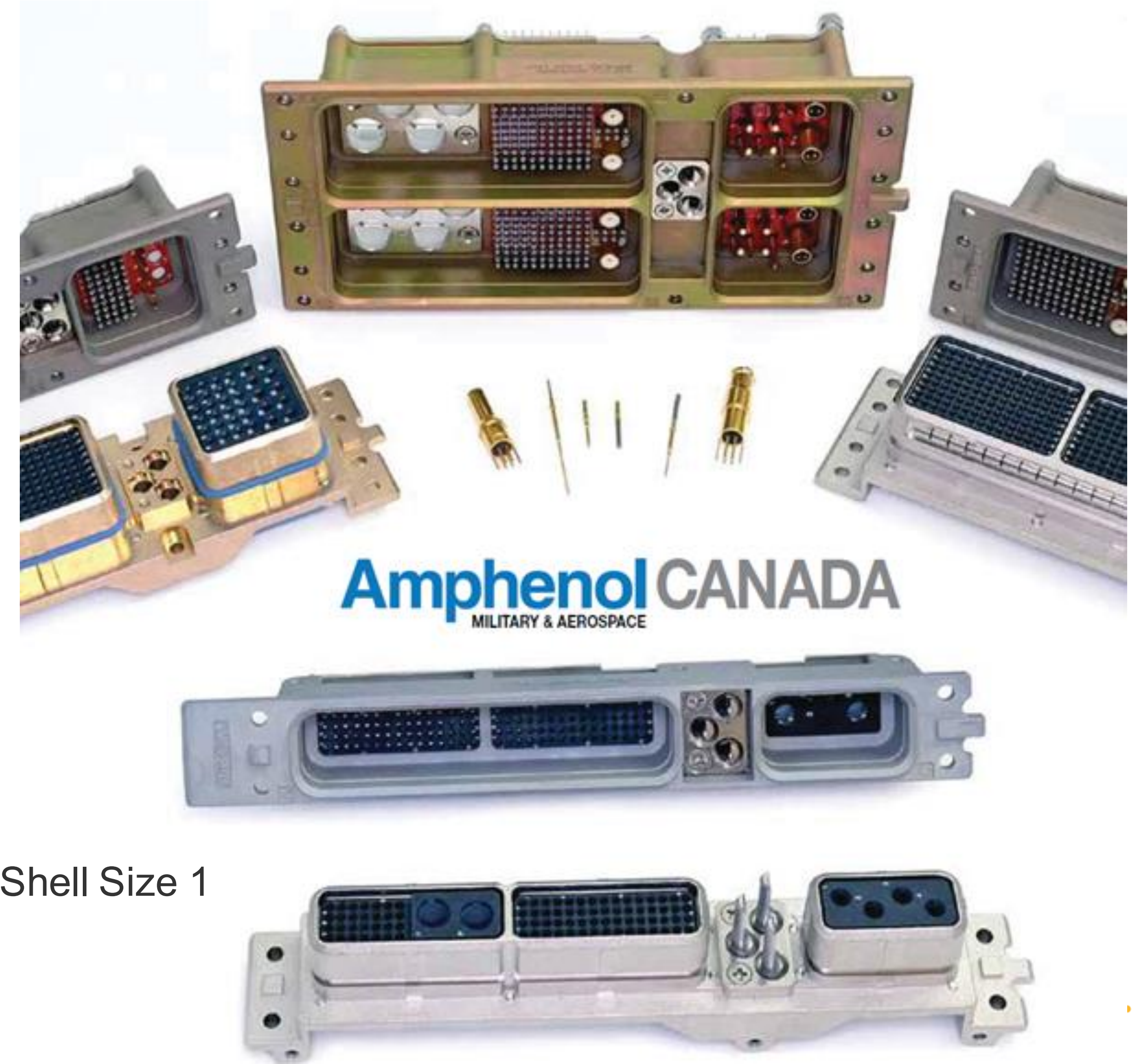


BLACK AREA REPRESENTS KEY POSITION

Shell Size 3

Connectors & terminals (4 of 9)

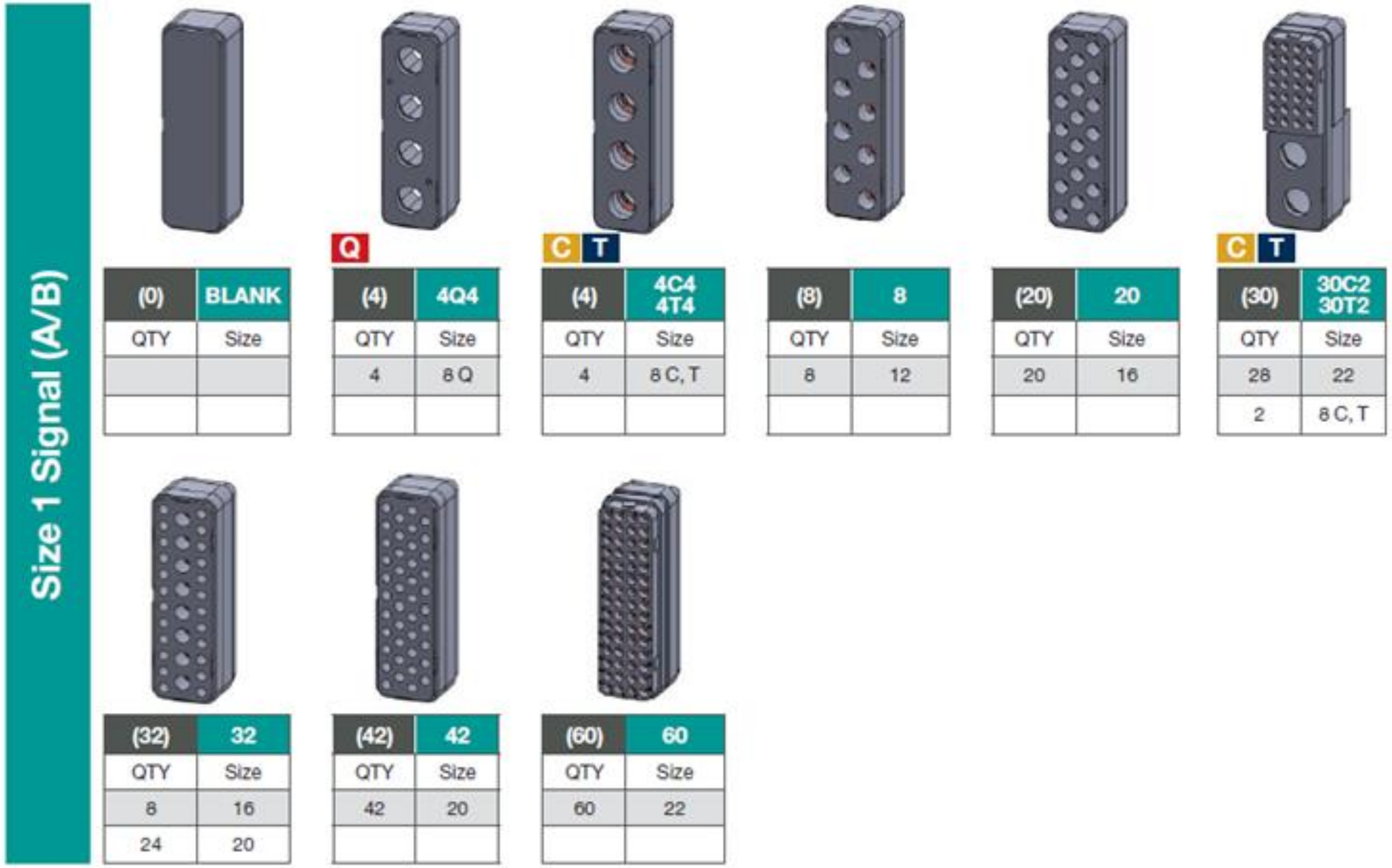
Shell Size 2



Shell Size 1

Connectors & terminals (5 of 9)

Configurable Inserts for Size 1, slot A or B



(0)	BLANK
QTY	Size

(4)	4Q4
QTY	Size
4	8 Q

(4)	4C4 4T4
QTY	Size
4	8 C, T

(8)	8
QTY	Size
8	12

(20)	20
QTY	Size
20	16

(30)	30C2 30T2
QTY	Size
28	22
2	8 C, T

(32)	32
QTY	Size
8	16
24	20

(42)	42
QTY	Size
42	20

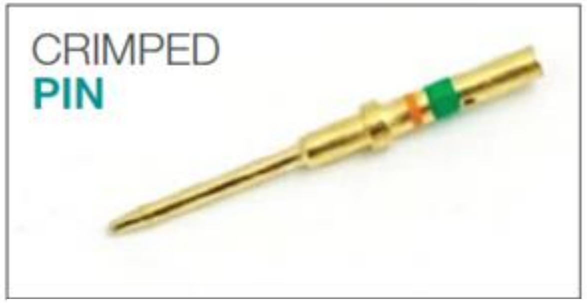
(60)	60
QTY	Size
60	22

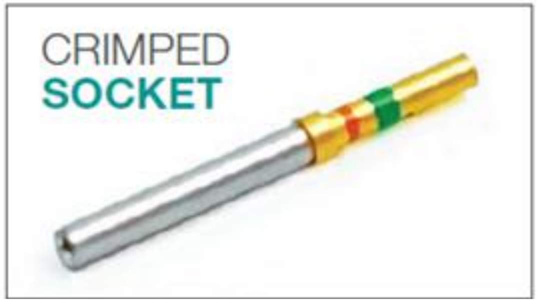
Contact type single, different sizes or:

C COAX T TWINAX OR TRIAX F FIBER Q QUADRAX

Connectors & terminals (6 of 9)

Connector contacts, filler and sealing plugs that go into the cavities of the inserts





Crimped contacts are for joining with wires / cables.

There are also contacs with „PC Tail“ for direct welding with a PCB.

CONTACT TYPE	SIZE	RECEPTACLE		Plug	
		TYPE	PART NO.	TYPE	PART NO.
Signal	22	SOCKET	AC-782222-301	Pin	AC-772222-301
Power	20	Pin	AC-772020-302	Socket	AC-782020-302
	16		AC-771616-303		AC-781616-303
	12		AC-771212-304		AC-781212-304

FILLER PLUGS

Contact Cavity Size	Amphenol Part Numbers	Color
22	AC-660022-701	Black
20	AC-660020-701	Red
16	AC-660016-701	Blue
16 Fiber	AC-660016F-701	Blue
12	AC-660012-701	Yellow
8 Coax	AC-660008-701	Red
5 Coax (Plug)	AC-660005-701	White
5 Coax (Recept.)	AC-660004-701	White

SEALING PLUGS

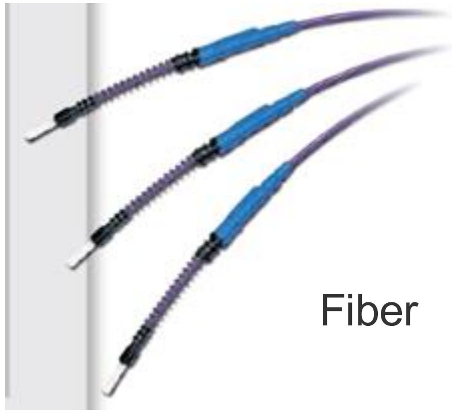
Contact Cavity Size	Amphenol Part Numbers	Color
22	AC-660022-801	Black
20	AC-660020-801	Red
16	AC-660016-801	Blue
12	AC-660012-801	Yellow
8 Coax	AC-660008-8701	Red

Connectors & terminals (7 of 9)

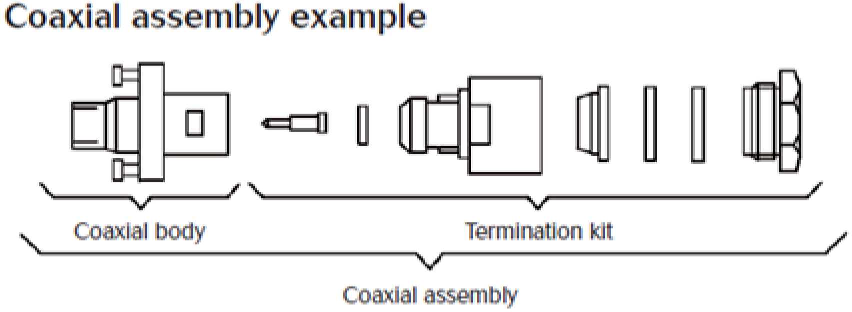
Coaxial, triaxial, quad and fiber contacts

Coax

Quad



Fiber

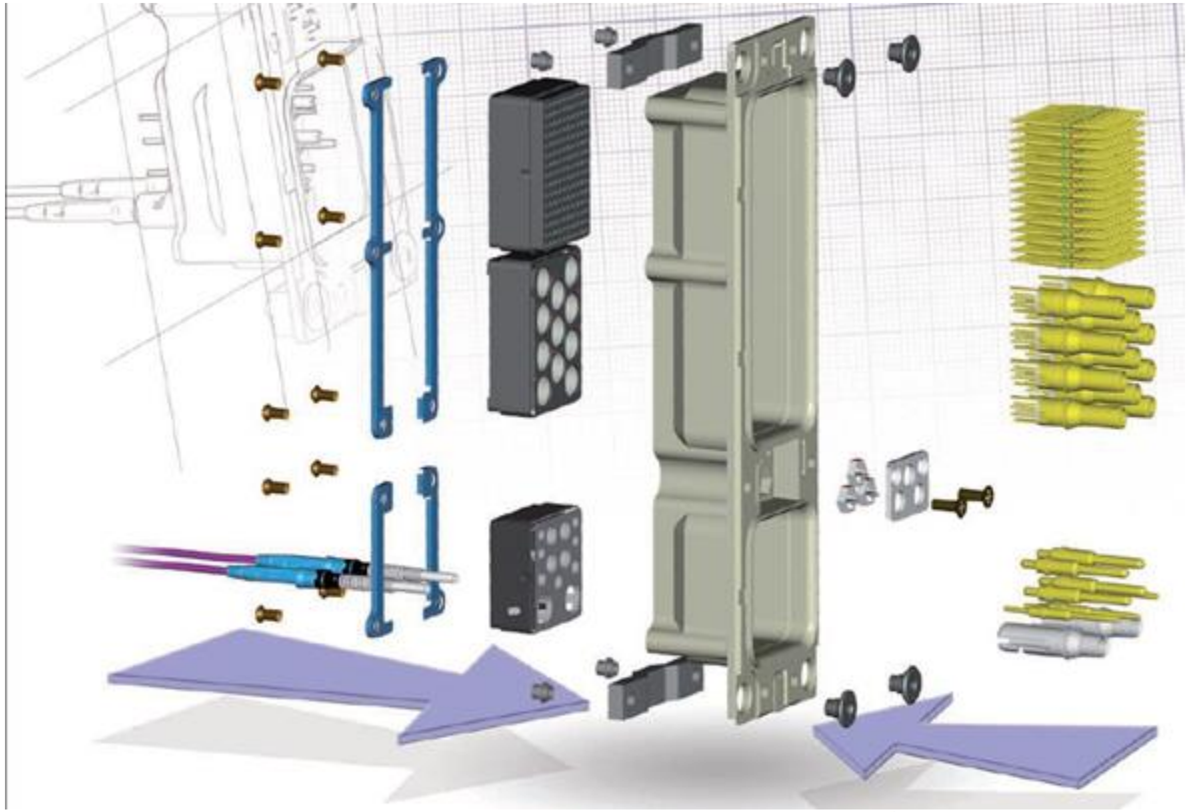


It is up to the use case, customer, and tool supplier to which level of detail AP242 is used. As a minimum all electrical connections have to be clearly identified.

Connectors & terminals (8 of 9)

A single ARINC 600 kit

- It is the job of the harness manufacturer to assemble all this
- Note that crimped contacts have be joined to the wires/cables before inserting them in the cavities of the Inserts.



Connectors & terminals (9 of 9)

For detailed Geometry, Dimensions and Tolerances: see AP242 PMI

- The geometric shape (2D or 3D) with dimensions and tolerances is NOT the focus for EWH but can be done by other capabilities of AP242 (see CAx-IF). For display purposes it is sufficient to reference into externally provided geometry.
- Focus for EWH is the identification of the features and the type of features
  - Mechanical features, e.g. the slots A, B, C and which kind of inserts go into which slot, and which contact go into which cavity
  - Electrical features/terminals and how all is electrical connected by wires & cables

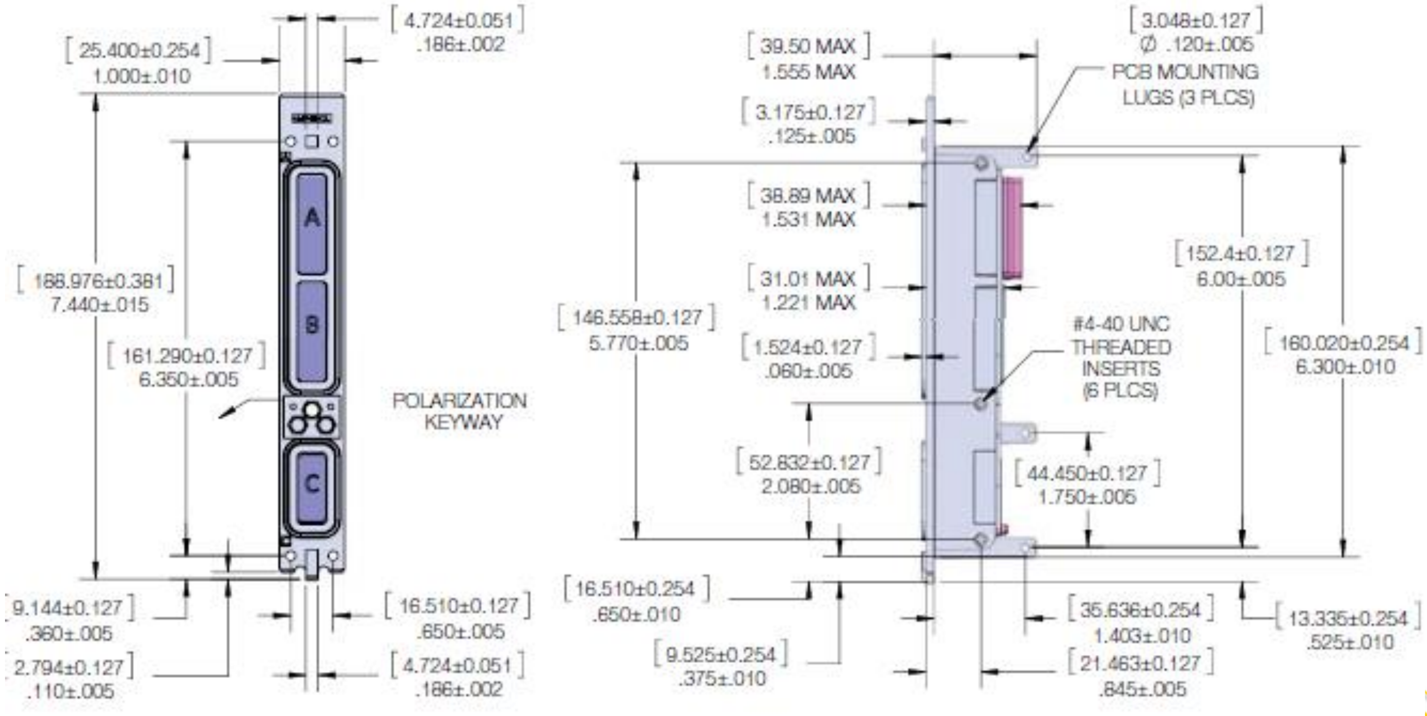


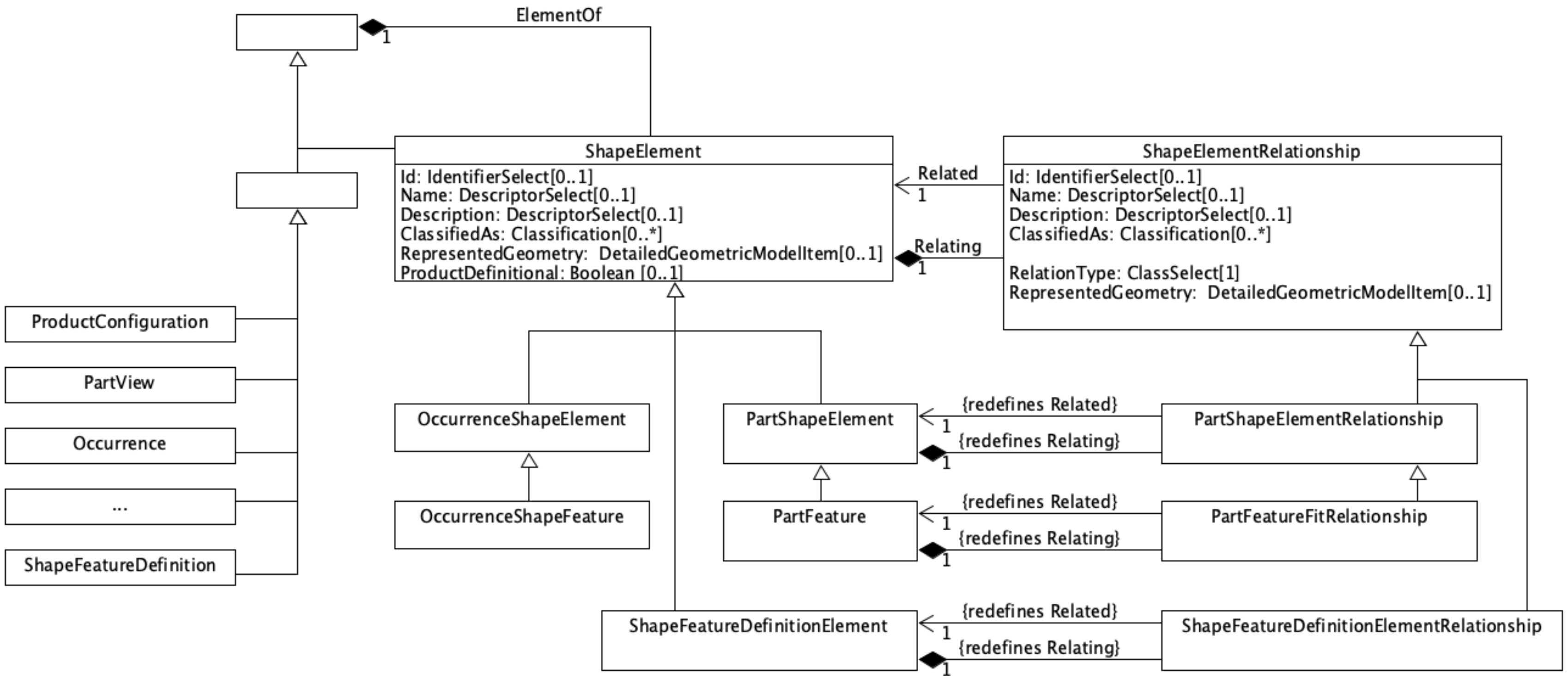
Table of Content

- Introduction
- Basic XML structure
- Part, PartCategory & PartVersion
- PartView & ViewContext
- ProductClass & ProductConfiguration
- Occurrence, Assembly & WiringHarnessAssemblyDesign
- ShapeElement
- Geometric & topological models
- Connectors & terminals
- Features, Definitions & AssemblyShapeJoints
- Wire, Cables & TransportFeature
- Shapes/ContactFeatureDefinition & -Elements
- CrossSection of Wire, Cable & HarnessSegment
- Geometry/Topology Associations
- Complex Connectors / Assembly Hierarchy
- External Element References
- ComposedGeometricModel

Features, Definitions & AssemblyShapeJoints (1 of 8)

ShapeElement in detail

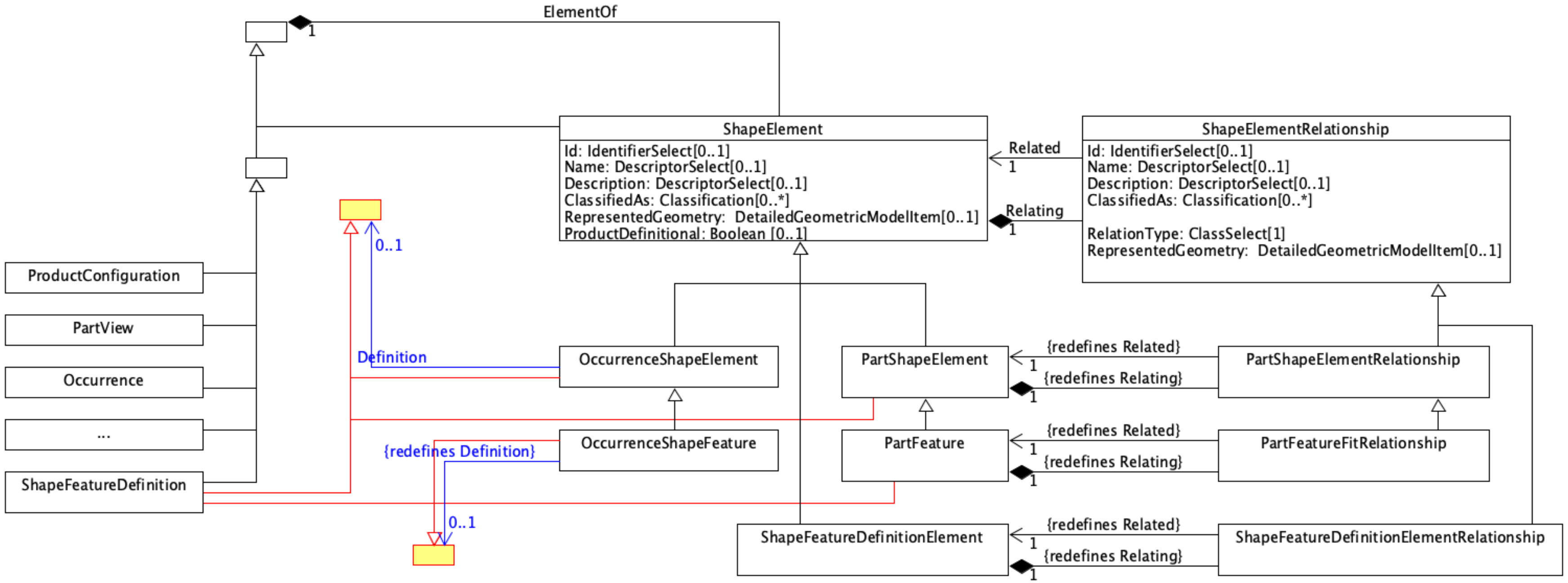
- For EWH, a *ShapeElement* is the identification of an element of the shape of a *ProductConfiguration*, *PartView*, *Occurrence*, *ShapeFeatureDefinition* or of another *ShapeElement* (recursively)
- Some subtypes of *ShapeElement* **might be defined by** a *ShapeFeatureDefinition* or another *ShapeElement*
  - Note: Definition and ParentRelationship attributes are not shown in the diagram
- For all **feature subtypes** the inherited property "*ProductDefinitional*" is true that meaning the features are reachable from the outside
- There are further sub-subtypes of *ShapeElement* for terminals (pins), joins, nets ...


ElementOf
1
ShapeElement
Id: IdentifierSelect[0..1]
Name: DescriptorSelect[0..1]
Description: DescriptorSelect[0..1]
ClassifiedAs: Classification[0..*]
RepresentedGeometry: DetailedGeometricModelItem[0..1]
ProductDefinitional: Boolean [0..1]
Related
1
Relating
1
ShapeElementRelationship
Id: IdentifierSelect[0..1]
Name: DescriptorSelect[0..1]
Description: DescriptorSelect[0..1]
ClassifiedAs: Classification[0..*]
RelationType: ClassSelect[1]
RepresentedGeometry: DetailedGeometricModelItem[0..1]
ProductConfiguration
PartView
Occurrence
...
ShapeFeatureDefinition
OccurrenceShapeElement
OccurrenceShapeFeature
PartShapeElement
PartFeature
{redefines Related}
1
{redefines Relating}
1
PartShapeElementRelationship
{redefines Related}
1
{redefines Relating}
1
PartFeatureFitRelationship
ShapeFeatureDefinitionElement
{redefines Related}
1
{redefines Relating}
1
ShapeFeatureDefinitionElementRelationship

Features, Definitions & AssemblyShapeJoints (2 of 8)

ShapeElement in detail

- An OccurrenceShapeElement might be defined by:
  - PartShapeElement (Note: it is most widely used)
  - ShapeFeatureDefinition (Note: this is a special case)
  - or another OccurrenceShapeElement (Note: recursively, for sub-ShapeElements)
  - in special cases, an OccurrenceShapeElement is defined only for the Occurrence, without any definition.
- The subtype OccurrenceShapeFeature might be defined by:
  - PartFeature (Note: it is most widely used)
  - ShapeFeatureDefinition (Note: this is a special case)
  - or another OccurrenceShapeFeature (Note: recursively, for sub-ShapeElements)
- Further subtypes of OccurrenceShapeFeature redefines the Definition attribute further
- See example usages on later

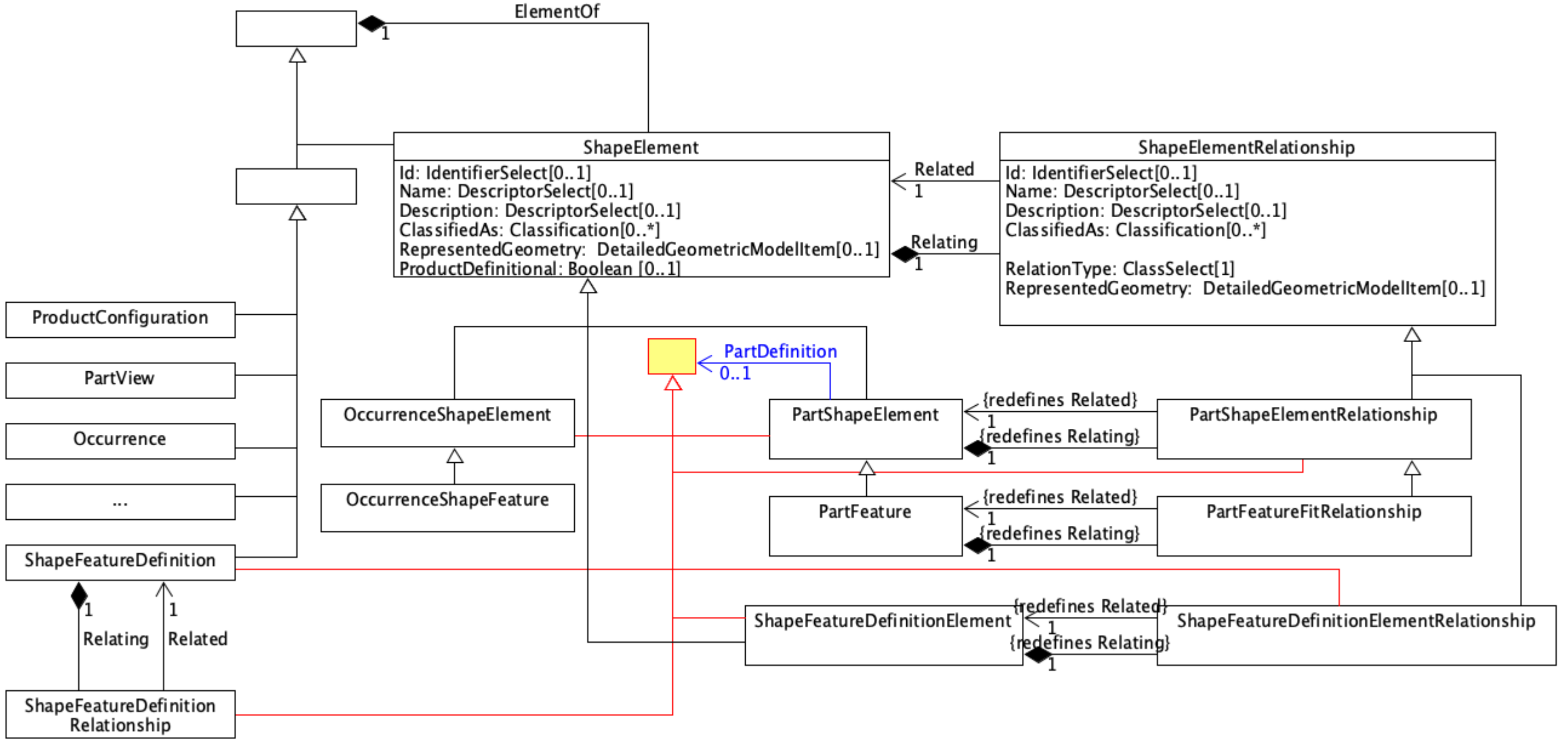

ElementOf
1
ShapeElement
Id: IdentifierSelect[0..1]
Name: DescriptorSelect[0..1]
Description: DescriptorSelect[0..1]
ClassifiedAs: Classification[0..*]
RepresentedGeometry: DetailedGeometricModelItem[0..1]
ProductDefinitional: Boolean [0..1]
Related
1
Relating
1
ShapeElementRelationship
Id: IdentifierSelect[0..1]
Name: DescriptorSelect[0..1]
Description: DescriptorSelect[0..1]
ClassifiedAs: Classification[0..*]
RelationType: ClassSelect[1]
RepresentedGeometry: DetailedGeometricModelItem[0..1]
0..1
ProductConfiguration
PartView
Occurrence
...
ShapeFeatureDefinition
Definition
OccurrenceShapeElement
PartShapeElement
{redefines Related}
1
{redefines Relating}
1
PartShapeElementRelationship
{redefines Definition}
OccurrenceShapeFeature
PartFeature
{redefines Related}
1
{redefines Relating}
1
PartFeatureFitRelationship
0..1
ShapeFeatureDefinitionElement
{redefines Related}
1
{redefines Relating}
1
ShapeFeatureDefinitionElementRelationship

Features, Definitions & AssemblyShapeJoints (3 of 8)

ShapeElement in detail

Definition of PartShapeElement

- A PartShapeElement might be defined by:
  - ShapeFeatureDefinition (Note: this is most widely used)
  - OccurrenceShapeElement (Note: this is used for reflection in a hierarchical assembly)
  - PartShapeElementRelationship (TBD)
  - ShapeFeatureDefinitionElementRelationship (TBD)
  - or another PartShapeElement (Note: recursively, for sub-ShapeElements)
- See example usages on later


ElementOf
1
ShapeElement
Id: IdentifierSelect[0..1]
Name: DescriptorSelect[0..1]
Description: DescriptorSelect[0..1]
ClassifiedAs: Classification[0..*]
RepresentedGeometry: DetailedGeometricModelItem[0..1]
ProductDefinitional: Boolean [0..1]
Related
1
Relating
1
ShapeElementRelationship
Id: IdentifierSelect[0..1]
Name: DescriptorSelect[0..1]
Description: DescriptorSelect[0..1]
ClassifiedAs: Classification[0..*]
RelationType: ClassSelect[1]
RepresentedGeometry: DetailedGeometricModelItem[0..1]
ProductConfiguration
PartView
Occurrence
...
ShapeFeatureDefinition
1
Relating
1
Related
ShapeFeatureDefinition Relationship
OccurrenceShapeElement
OccurrenceShapeFeature
PartDefinition
0..1
PartShapeElement
{redefines Related}
1
{redefines Relating}
1
PartShapeElementRelationship
PartFeature
{redefines Related}
1
{redefines Relating}
1
PartFeatureFitRelationship
ShapeFeatureDefinitionElement
{redefines Related}
1
{redefines Relating}
1
ShapeFeatureDefinitionElementRelationship

Features, Definitions & AssemblyShapeJoints (4 of 8)

ShapeElement in detail

Definition attribute of PartFeatureFitRelationship

- A PartFeatureFitRelationship might be defined by:
  - ShapeFeatureDefinitionFitRelationship (Note: this is most widely used)
  - ShapeFeatureDefinitionElementRelationship (Note: this is a case needed for hierarchical fit relationship)

Parent of ShapeFeatureDefinitionElementRelationship

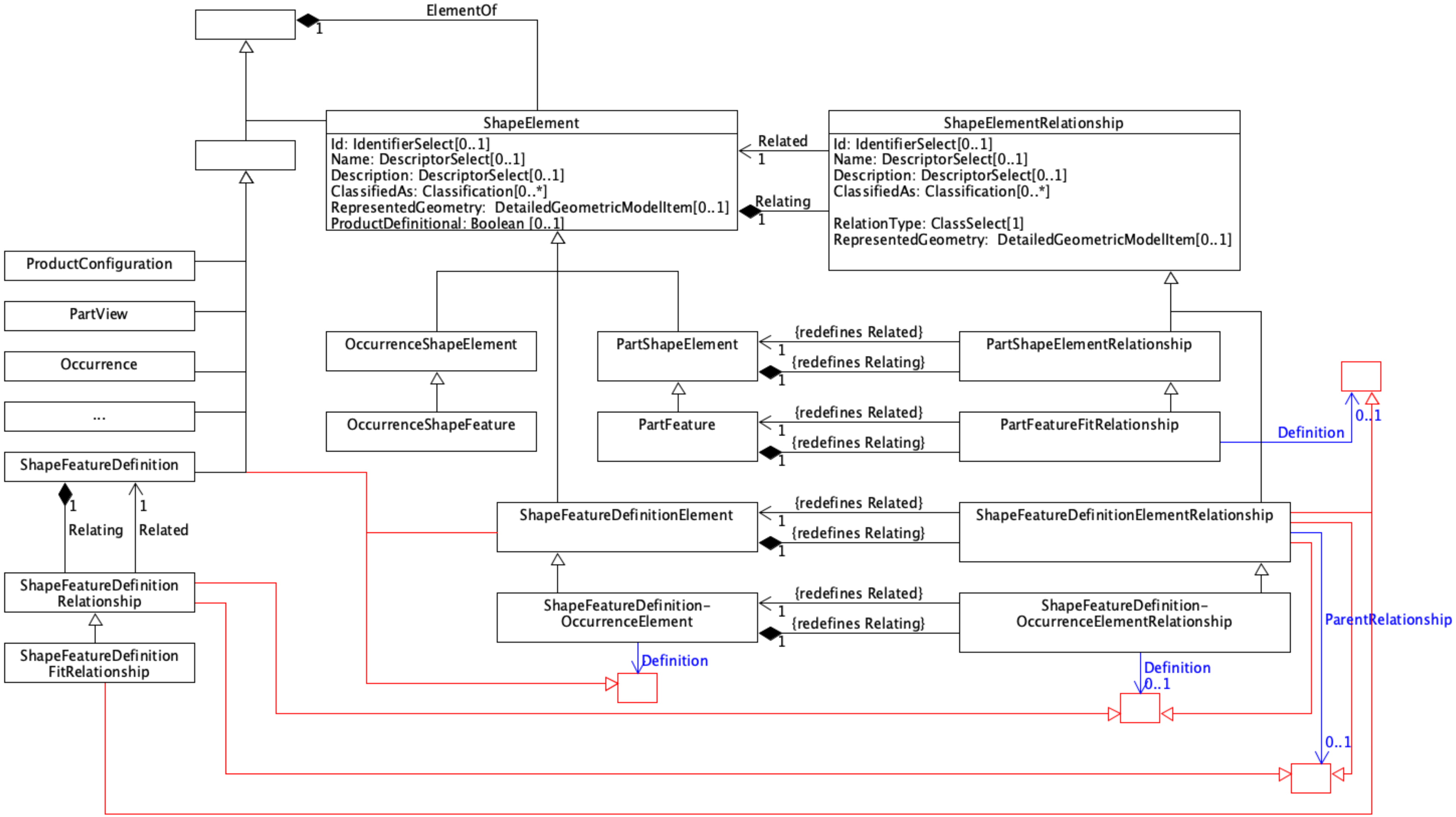
- A ShapeFeatureDefinitionElementRelationship might belong to:
  - ShapeFeatureDefinitionRelationship (Note: this is most widely used)
  - another ShapeFeatureDefinitionElementRelationship (Note: this is a case needed for hierarchical relationship)
- See example usages later

Definition attribute of ShapeFeatureDefinitionOccurrenceElement

- A ShapeFeatureDefinitionOccurrenceElement is a type of ShapeFeatureDefinitionElement that is an Occurrence of another ShapeFeatureDefinition or ShapeFeatureDefinitionElement
- A ShapeFeatureDefinitionOccurrenceElementRelationship is a relationship between two ShapeFeatureDefinitionOccurrenceElements with an optional definition attribute that is either a ShapeFeatureDefinitionRelationship or a ShapeFeatureDefinitionElementRelationship.
- See example usages later

Features, Definitions & AssemblyShapeJoints (5 of 8)

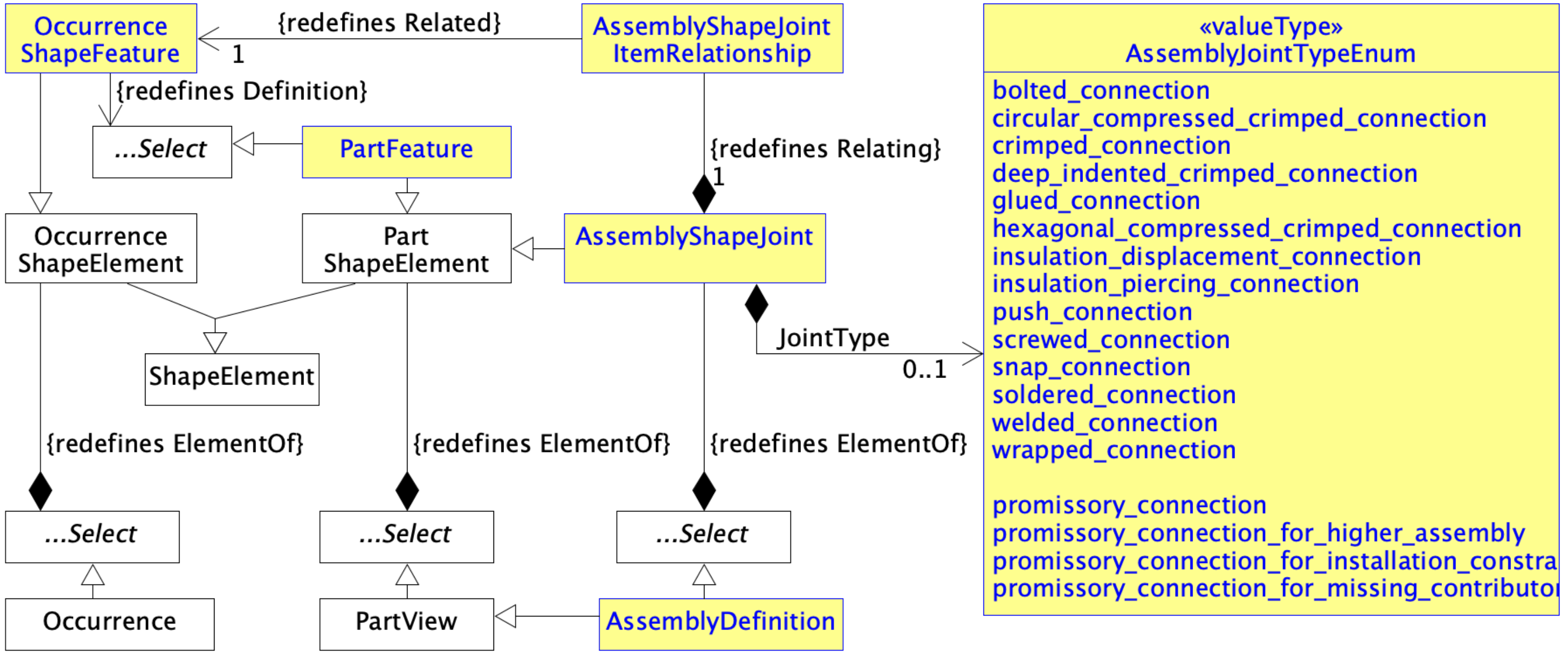
ShapeElement in detail


ElementOf
1
ShapeElement
Id: IdentifierSelect[0..1]
Name: DescriptorSelect[0..1]
Description: DescriptorSelect[0..1]
ClassifiedAs: Classification[0..*]
RepresentedGeometry: DetailedGeometricModelItem[0..1]
ProductDefinitional: Boolean [0..1]
Related
1
Relating
1
ShapeElementRelationship
Id: IdentifierSelect[0..1]
Name: DescriptorSelect[0..1]
Description: DescriptorSelect[0..1]
ClassifiedAs: Classification[0..*]
RelationType: ClassSelect[1]
RepresentedGeometry: DetailedGeometricModelItem[0..1]
ProductConfiguration
PartView
Occurrence
...
ShapeFeatureDefinition
1
Relating
1
Related
ShapeFeatureDefinition Relationship
ShapeFeatureDefinition FitRelationship
OccurrenceShapeElement
OccurrenceShapeFeature
PartShapeElement
PartFeature
{redefines Related}
1
{redefines Relating}
1
PartShapeElementRelationship
{redefines Related}
1
{redefines Relating}
1
PartFeatureFitRelationship
Definition
0..1
ShapeFeatureDefinitionElement
{redefines Related}
1
{redefines Relating}
1
ShapeFeatureDefinitionElementRelationship
ShapeFeatureDefinition-OccurrenceElement
{redefines Related}
1
{redefines Relating}
1
ShapeFeatureDefinition-OccurrenceElementRelationship
Definition
Definition
0..1
ParentRelationship
0..1

Features, Definitions & AssemblyShapeJoints (6 of 8)

AssemblyShapeJoint - Basic

- An *AssemblyShapeJoint* defines a joint between two or more *OccurrenceShapeFeature* within an *AssemblyDefinition*.
- An *AssemblyShapeJoint* may be used for joining mechanical, electrical, optical, or piping features
- An *OccurrenceShapeFeature* is typically defined by a *PartFeature*
- Two or more *OccurrenceShapeFeature* are joined together in an *AssemblyShapeJoint* through *AssemblyShapeJointItemRelationship*
- An optional *JointType* indicates on how an *AssemblyShapeJoint* is realized



«valueType»
AssemblyJointTypeEnum

bolted_connection
circular_compressed_crimped_connection
crimped_connection
deep_indented_crimped_connection
glued_connection
hexagonal_compressed_crimped_connection
insulation_displacement_connection
insulation_piercing_connection
push_connection
screwed_connection
snap_connection
soldered_connection
welded_connection
wrapped_connection

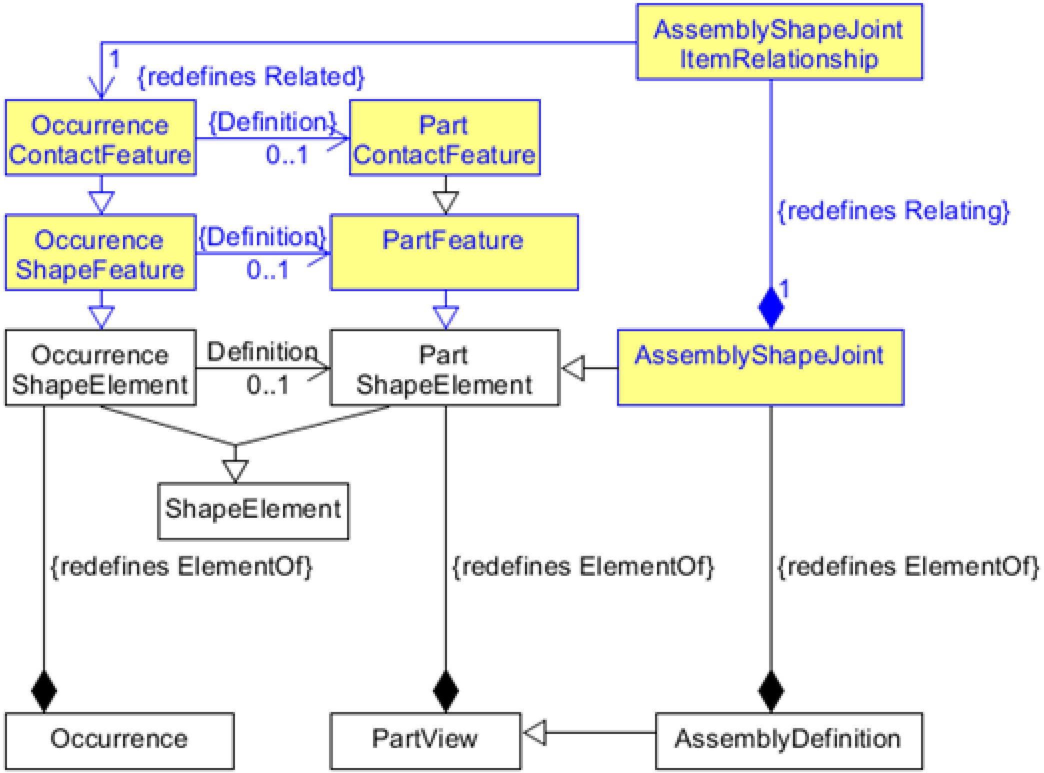
promissory_connection
promissory_connection_for_higher_assembly
promissory_connection_for_installation_constra
promissory_connection_for_missing_contributor

Simplified extract from AP242ed2

Features, Definitions & AssemblyShapeJoints (7 of 8)

Features, Contacts and Joints

- *PartShapeElement and OccurrenceShapeElement* represents specific *ShapeElements* for *Part and Occurrence respectively*
- *PartFeature and OccurrenceShapeFeature* represents specific *ShapeElements* that are on the physical boundary of a *Part and Occurrence respectively*
- *PartContactFeature and OccurrenceContactFeature* represents *ShapeElements* that are intended to be connected with other contact features
- *AssemblyShapeJoint* joins two or more *OccurrenceContactFeature* (or *OccurrenceShapeFeatue*) by *AssemblyShapeJointItemRelationships*



Features, Definitions & AssemblyShapeJoints (8 of 8)

Example: The two Inserts 1 and 2 that are joined into the Slots A and B of a Connector Housing

- SingleOccurrences of Parts are assembled together in a WiringHarnessAssemblyDesign by NextAssemblyOccurrenceUsage
- The PartContactFeatures of a PartView are replicated as OccurrenceContactFeature for the SingleOccurrences
- AssemblyShapeJoints for a WiringHarnessAssemblyDesign join the OccurrenceContactFeatures by AssemblyShapeJointItemRelationships

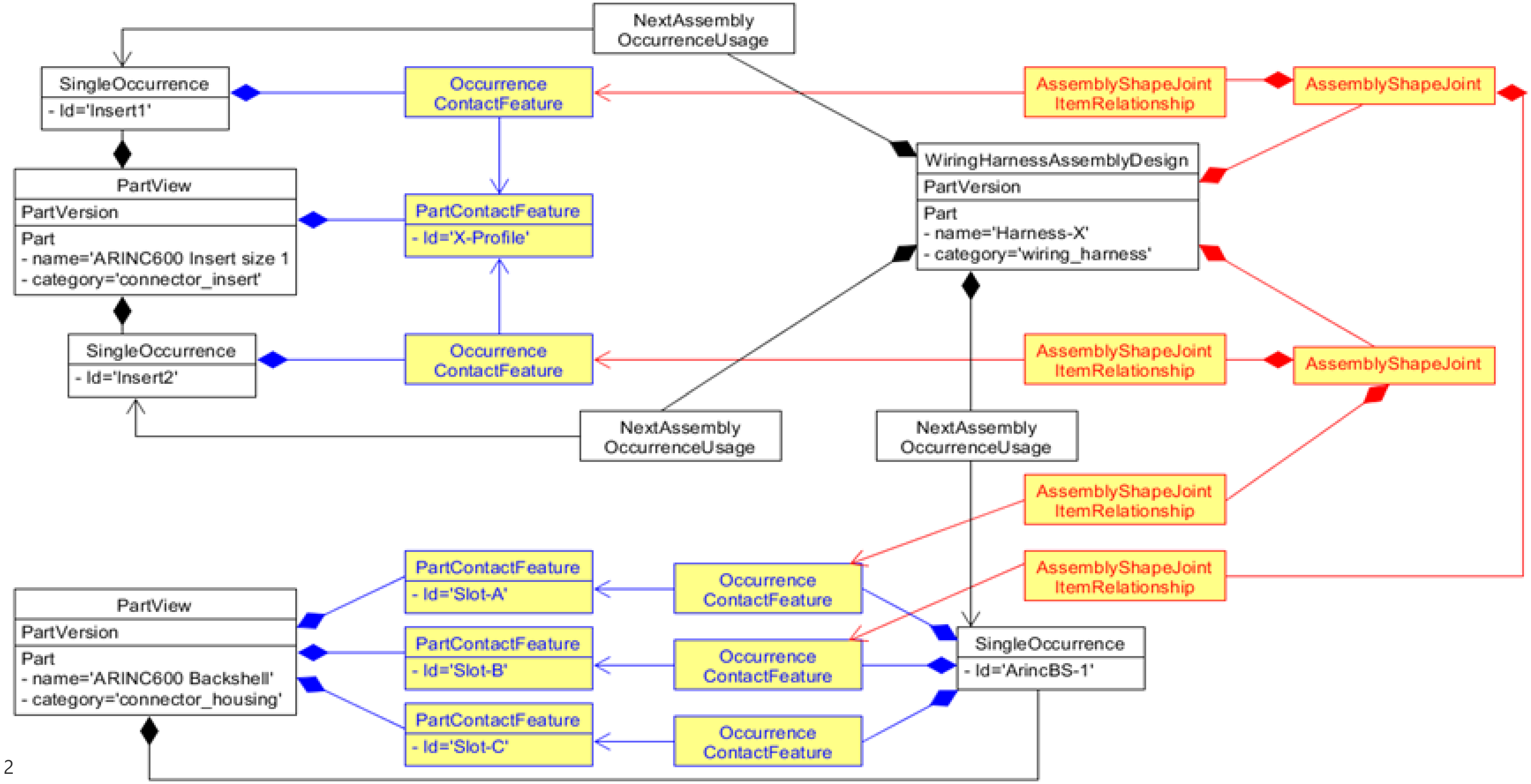

NextAssembly OccurrenceUsage
SingleOccurrence
- Id='Insert1'
Occurrence ContactFeature
AssemblyShapeJoint ItemRelationship
AssemblyShapeJoint
PartView
PartVersion
Part
- name='ARINC600 Insert size 1
- category='connector_insert'
PartContactFeature
- Id='X-Profile'
WiringHarnessAssemblyDesign
PartVersion
Part
- name='Harness-X'
- category='wiring_harness'
SingleOccurrence
- Id='Insert2'
Occurrence ContactFeature
AssemblyShapeJoint ItemRelationship
AssemblyShapeJoint
NextAssembly OccurrenceUsage
NextAssembly OccurrenceUsage
AssemblyShapeJoint ItemRelationship
AssemblyShapeJoint ItemRelationship
PartContactFeature
- Id='Slot-A'
Occurrence ContactFeature
PartView
PartVersion
Part
- name='ARINC600 Backshell'
- category='connector_housing'
PartContactFeature
- Id='Slot-B'
Occurrence ContactFeature
SingleOccurrence
- Id='ArincBS-1'
PartContactFeature
- Id='Slot-C'
Occurrence ContactFeature

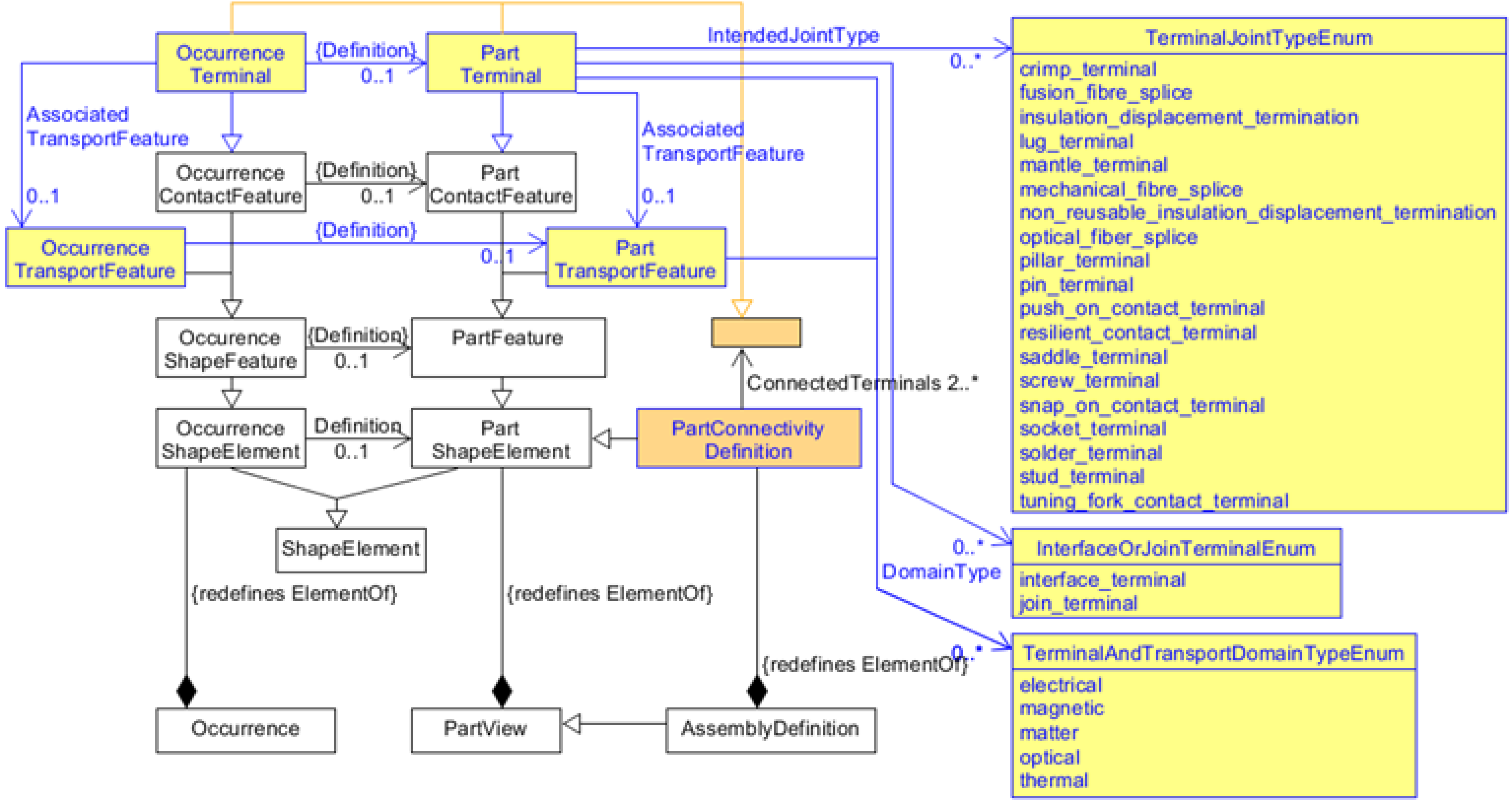
Table of Content

- Introduction
- Basic XML structure
- Part, PartCategory & PartVersion
- PartView & ViewContext
- ProductClass & ProductConfiguration
- Occurrence, Assembly & WiringHarnessAssemblyDesign
- ShapeElement
- Geometric & topological models
- Connectors & terminals
- Features, Definitions & AssemblyShapeJoints
- Wire, Cables & TransportFeature
- Shapes/ContactFeatureDefinition & -Elements
- CrossSection of Wire, Cable & HarnessSegment
- Geometry/Topology Associations
- Complex Connectors / Assembly Hierarchy
- External Element References
- ComposedGeometricModel

Wire, Cables & TransportFeature (1 of 10)

Transport Features, Terminals & ConnectivityDef.

- Electrical energy or information, light, matter is transported by special TransportFeatures for PartViews and Occurrences.
  - In the electrical world this is also called a conductor (e.g. a wire, or a busbar)
  - In the optical area, the TransportFeature might be a fiber
  - In the pneumatic or hydraulic area, the TransportFeature is for matter (e.g. a pipe or a hose)
- A TransportFeature is accessed by Terminals (Part/Occurrence). OccurrenceTerminals are intended to be joined on an assembly level with other terminals by a special manufacturing method; e.g
- PartConnectivityDefinition providing details



Wire, Cables & TransportFeature (2 of 10)

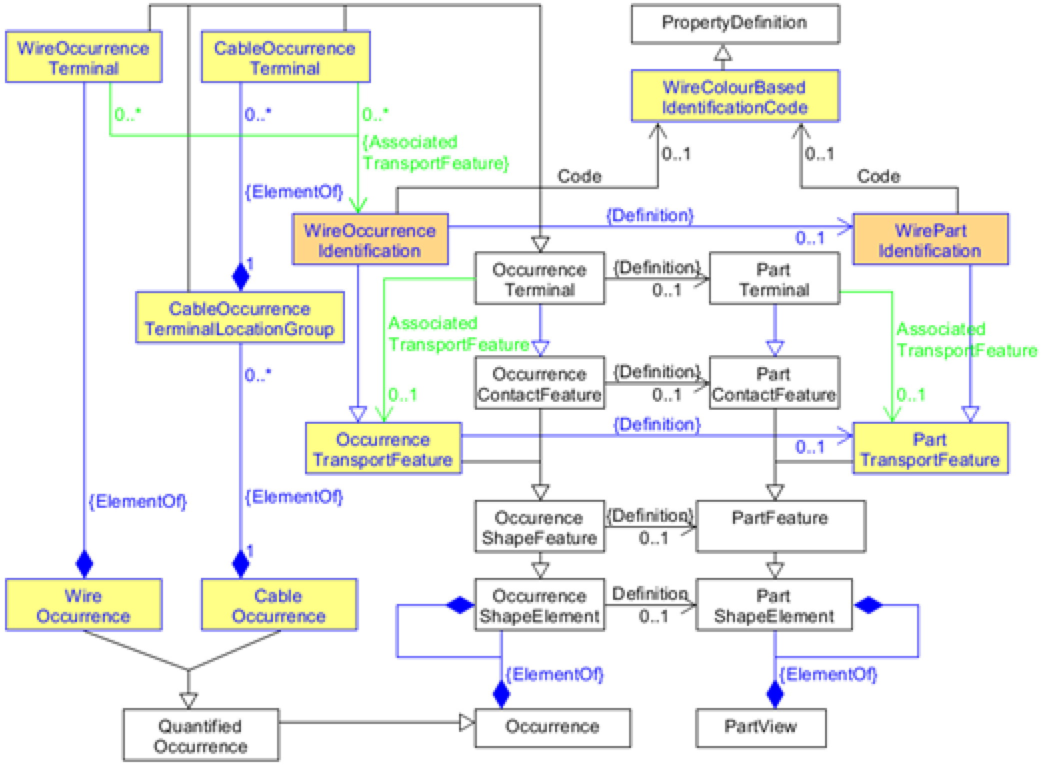
Example: simple connector

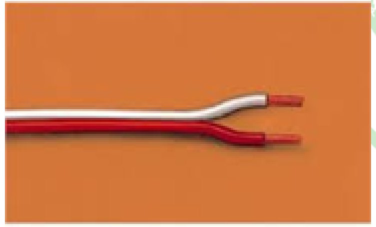
- a Part that is a connector
- ... with two PartTerminals identified as “signal” and “gnd”
- ... that are intended to be joint by crimping (“crimp_terminal”) on the next higher assembly level (“join terminal”)
- a SingleOccurrence of the connector with the ID “phone1”
- ... with two OccurrenceTerminals that are defined by the PartTerminals

```
<Part uid="_117000">
...
 <PartTypes>
  <PartCategoryEnum>connector</PartCategoryEnum>
 </PartTypes>
 <Versions>
  <PartVersion uid="_117001">
   <Id id=”Version 1"/>
   <Views>
    <PartView uid="_117002">
     <InitialContext uidRef="_100102"/>
     <Occurrence xsi:type="n0:SingleOccurrence" uid="_217100">
      <Id id="phone1"/>
      <ShapeElement xsi:type="n0:OccurrenceTerminal" uid="_217102">
       <Definition uidRef="_117004"/>
      </ShapeElement>
      <ShapeElement xsi:type="n0:OccurrenceTerminal" uid="_217104">
       <Definition uidRef="_117009"/>
      </ShapeElement>
     </Occurrence>
     <ShapeElement xsi:type="n0:PartTerminal" uid="_117004">
      <Id id="Join signal"/>
      <IntendedJointType>
       <TerminalJointTypeEnum>crimp_terminal</TerminalJointTypeEnum>
      </IntendedJointType>
      <InterfaceOrJoinTerminal>join_terminal</InterfaceOrJoinTerminal>
     </ShapeElement>
     <ShapeElement xsi:type="n0:PartTerminal" uid="_117009">
      <Id id="Join gnd"/>
      <IntendedJointType>
       <TerminalJointTypeEnum>crimp_terminal</TerminalJointTypeEnum>
      </IntendedJointType>
      <InterfaceOrJoinTerminal>join_terminal</InterfaceOrJoinTerminal>
     </ShapeElement>
    </PartView>
   </Views>
  </PartVersion>
 </Versions>
</Part>
```

Wire, Cables & TransportFeature (3 of 10)

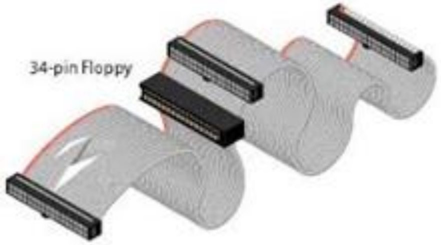
- A wire consists of a single conductor/ (maybe of several strands) and typically an isolation.
- The isolation has to be somehow removed before the conductor/TransportFeature can be accessed by a Wire/CableOccurrenceTerminal (details not covered so far in the standard).
- A cable consists of several “wires”(conductors), and so it is essential to identify the wire in the cable (e.g. by color).
  - This is typically done by WirePartIdentification or for special cases by a WireOccurrenceIdentification
  - Both identifications have a code property to the predefined PropertyDefinition named WireColourBasedIdentificationCode
  - Typical usage: A WireOccurrenceIdentification refers to a WirePartIdentification as Definition.
    A WireOccurrenceIdentification has no colour code, but inherit this from WirePartIdentification
- CableOccurrenceTerminals that are located close together (e.g. same end) are grouped together by CableOccurrenceTerminalLocationGroup





Wire, Cables & TransportFeature (4 of 10)

Often conductors (e.g. Wires, Cables, Busbars) are connected only at the ends but sometimes at any location



ices

Wire, Cables & TransportFeature (5 of 10)

Example: WireOccurrence (1 of 2)

- a Part with the categories “wire” and “raw_material_by_length”
- the WirePartIdentification identifies the single conductor in the wire
  - its code is defined by the WireColourBasedIdentificationCode as “white”
- The cross-sectional structure of the wire can be defined by *CrossSectionalPartShapeElement*. The predefined properties are:
  - MinCrossSectionDiameter, MaxCrossSectionDiameter
  - MinimumBendRadius
  - Whether it is an inner or outer boundary (OuterOrInnerBoundary)

  Note 1: these properties got introduced for determining the values for the cross section of a harness segment

  Note 2: See further details on harness cross sectional structure (onion model)
- *Further user defined properties can be added to CrossSectionalPartShapeElement for:*
  - The CrossSectionShape (e.g. round, square, or flat)
  - Whether the wire is solid (rigid) or stranded (flexible)
  - The cross-sectional area (e.g. 1.5 mm^2)

```
<PropertyDefinition xsi:type="n0:WireColourBasedIdentificationCode" uid="_100201">
 <Id id="white"/>
 <PropertyType>
   <ClassString>wire colour-based identification code</ClassString>
 </PropertyType>
</PropertyDefinition>
```

```
<Part uid="_101000">
…
 <PartTypes>
  <PartCategoryEnum>wire</PartCategoryEnum>
  <PartCategoryEnum>raw_material_by_length</PartCategoryEnum>
 </PartTypes>
 <Versions>
  <PartVersion uid="_101001">
   <Id id="/NULL"/>
   <Views>
    <PartView xsi:type="n0:AssemblyDefinition" uid="_101002">
     <DefiningGeometry uidRef="_104890"/>
     <InitialContext uidRef="_100102"/>
    … <!—See next slide for the Occurrence-->
     <ShapeElement xsi:type="n0:CrossSectionalPartShapeElement" uid="_101020">
      <MinCrossSectionDiameter uid="_1010201" xsi:type="n0:NumericalValue">
       <Unit uidRef="_100301"/>
       <ValueComponent>0.001</ValueComponent>
      </MinCrossSectionDiameter>
      <MinimumBendRadius uid="_1010202" xsi:type="n0:NumericalValue">
       <Unit uidRef="_100301"/>
       <ValueComponent>0.02</ValueComponent>
      </MinimumBendRadius>
      <OuterOrInnerBoundary>outer_boundary</OuterOrInnerBoundary>
     </ShapeElement>
     <ShapeElement xsi:type="n0:WirePartIdentification" uid="_101021">
      <Id id="1"/>
      <Code uidRef="_100201"/>
     </ShapeElement>
    </PartView>
   </Views>
  </PartVersion>
 </Versions>
</Part>
```

Wire, Cables & TransportFeature (6 of 10)

Example: WireOccurrence (2 of 2)

- a WireOccurrence has a particular length, here 1.75m
- the WireOccurrenceIdentification replicates the information from the PartOccurrenceIdentification
- two WireOccurrenceTerminals at the ends of the wire, “end a” and “end b”, are defined both referencing to the same WireOccurrenceIdentification
- additional WireOccurrenceTerminals in the middle of the wire can be defined as needed

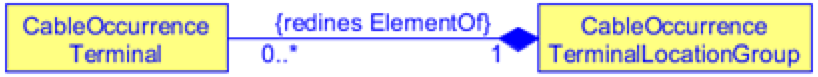
Note: This structure is a bit exhausted but is chosen to be in symmetry with the structure of CableOccurrence

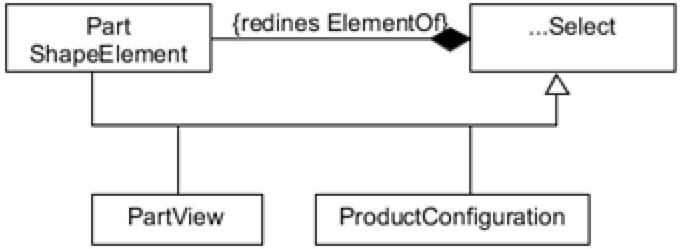
```xml
<Occurrence xsi:type="n0:WireOccurrence" uid="_201004">
 <Id id="wire1"/>
 <ShapeElement xsi:type="n0:WireOccurrenceTerminal" uid="_201006">
  <Name>
   <CharacterString>end a</CharacterString>
  </Name>
  <AssociatedTransportFeature uidRef="_201008"/>
 </ShapeElement>
 <ShapeElement xsi:type="n0:WireOccurrenceTerminal" uid="_201007">
  <Name>
   <CharacterString>end b</CharacterString>
  </Name>
  <AssociatedTransportFeature uidRef="_201008"/>
 </ShapeElement>
 <ShapeElement xsi:type="n0:WireOccurrenceIdentification" uid="_201008">
  <Definition uidRef="_101021"/>
 </ShapeElement>
 <Quantity xsi:type="n0:NumericalValue" uid="_201010">
  <Unit uidRef="_100301"/>
   <ValueComponent>1.75</ValueComponent>
 </Quantity>
</Occurrence>
```

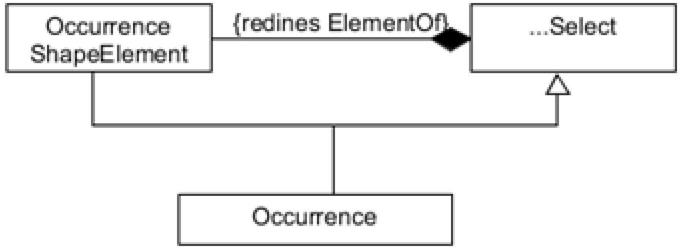
Wire, Cables & TransportFeature (7 of 10)

Groups of ShapeElements - Multi-Terminals

- In most cases a PartShapeElement is an ElementOf a PartView, and an OccurrenceShapeElement is an ElementOf an Occurrence … but not always
- We saw that a CableOccurrenceTerminal is an ElementOf a CableOccurrenceTerminalLocationGroup
- In general, all ShapeElements can also be elements of another ShapeElement. This allows hierarchical grouping of ShapeElements.
- This capability might also be useful to represent e.g. the connectivity of coax, triax, quad connectors, or e.g. Ethernet CAT-5 cable as there are standards on how to connect the detailed conductors.
  (TBD: provide an example for this case)







Wire, Cables & TransportFeature (8 of 10)

Example: CableOccurrenceTerminals (1 of 2)

- two properties of type *"wire colour-based identification code"* with the values "*white*" and "*red*"
- a part with the categories "*cable*" and "*raw_material_by_length*"
- the *PartView* has two conductors, indicated by *WirePartIdentification* that references the *"wire colour-based identification code"* for *"white"* and *"red"*

```
<PropertyDefinition xsi:type="n0:WireColourBasedIdentificationCode" uid="_100201">
 <Id id="white"/>
 <PropertyType>
  <ClassString>wire colour-based identification code</ClassString>
 </PropertyType>
</PropertyDefinition>
<PropertyDefinition xsi:type="n0:WireColourBasedIdentificationCode" uid="_100202">
 <Id id="red"/>
 <PropertyType>
  <ClassString>wire colour-based identification code</ClassString>
 </PropertyType>
</PropertyDefinition>
```

```
<Part uid="_104000">
…
 <PartTypes>
  <PartCategoryEnum>cable</PartCategoryEnum>
  <PartCategoryEnum>raw_material_by_length</PartCategoryEnum>
 </PartTypes>
 <Versions>
  <PartVersion uid="_104001">
   <Id id=”Version 1"/>
   <Views>
    <PartView uid="_104002">
     <DefiningGeometry uidRef="_104890"/>
     <InitialContext uidRef="_100102"/>
     … <!—See next slide for the Occurrence-->
     <ShapeElement xsi:type="n0:WirePartIdentification" uid="_104003">
      <Id id="A"/>
      <Code uidRef="_100201"/>
     </ShapeElement>
     <ShapeElement xsi:type="n0:WirePartIdentification" uid="_104004">
      <Id id="B"/>
      <Code uidRef="_100202"/>
     </ShapeElement>
    </PartView>
   </Views>
  </PartVersion>
 </Versions>
</Part>
```

Wire, Cables & TransportFeature (9 of 10)

Example: CableOccurrenceTerminals (2 of 2)

- *CableOccurrence* has:
    - a particular length (here, 4.25m)
    - two *WireOccurrenceIdentification that reference corresponding PartOccurrcenceIdentification*
    - *two CableOccurrenceTerminalLocationGroup*, one for each end: “end a” and “end b”
    - each *CableOccurrenceTerminalLocationGroup* has two *CableOccurrenceTerminals* that corresponds to the two *WireOccurrenceIdentifications*

```
<Occurrence xsi:type="n0:CableOccurrence" uid="_204006">
 <Id id="cable3"/>
 <ShapeElement xsi:type="n0:CableOccurrenceTerminalLocationGroup" uid="_204010">
  <Name>
   <CharacterString>end a</CharacterString>
  </Name>
 <ShapeElement xsi:type="n0:CableOccurrenceTerminal" uid="_204013">
  <AssociatedTransportFeature uidRef="_204003"/>
 </ShapeElement>
 <ShapeElement xsi:type="n0:CableOccurrenceTerminal" uid="_204014">
  <AssociatedTransportFeature uidRef="_204004"/>
 </ShapeElement>
 </ShapeElement>
 <ShapeElement xsi:type="n0:CableOccurrenceTerminalLocationGroup" uid="_204020">
 <Name>
  <CharacterString>end b</CharacterString>
 </Name>
 <ShapeElement xsi:type="n0:CableOccurrenceTerminal" uid="_204023">
  <AssociatedTransportFeature uidRef="_204003"/>
 </ShapeElement>
 <ShapeElement xsi:type="n0:CableOccurrenceTerminal" uid="_204024">
  <AssociatedTransportFeature uidRef="_204004"/>
 </ShapeElement>
 </ShapeElement>
 <ShapeElement xsi:type="n0:CrossSectionalOccurrenceShapeElement" uid="_204007">
  <Definition uidRef="_104020"/>
 </ShapeElement>
 <ShapeElement xsi:type="n0:WireOccurrenceIdentification" uid="_204003">
  <Definition uidRef="_104003"/>
 </ShapeElement>
 <ShapeElement xsi:type="n0:WireOccurrenceIdentification" uid="_204004">
  <Definition uidRef="_104004"/>
 </ShapeElement>
 <Quantity xsi:type="n0:NumericalValue" uid="_204008">
  <Unit uidRef="_100301"/>
  <ValueComponent>4.25</ValueComponent>
 </Quantity>
</Occurrence>
```

Wire, Cables & TransportFeature (10 of 10)

Example: AssemblyShapeJoint

This example shows how *AssemblyShapeJoint* is used to connect a particular end of a wire in a cable with the terminal of a connector.

- a Part that is an assembly, here *WiringHarnessAssemblyDesign*
- *NextAssemblyOccurrenceUsages* brings the Occurrences “phone1” and “cable3” into the assembly
- a single *AssemblyShapeJoint* realised by crimping (“crimped_connection”)
- the joint is established by *AssemblyShapeJointItemRelationship* between
    - Cable3 (wire A at end_b)
    - Phone1 (terminal “signal”)

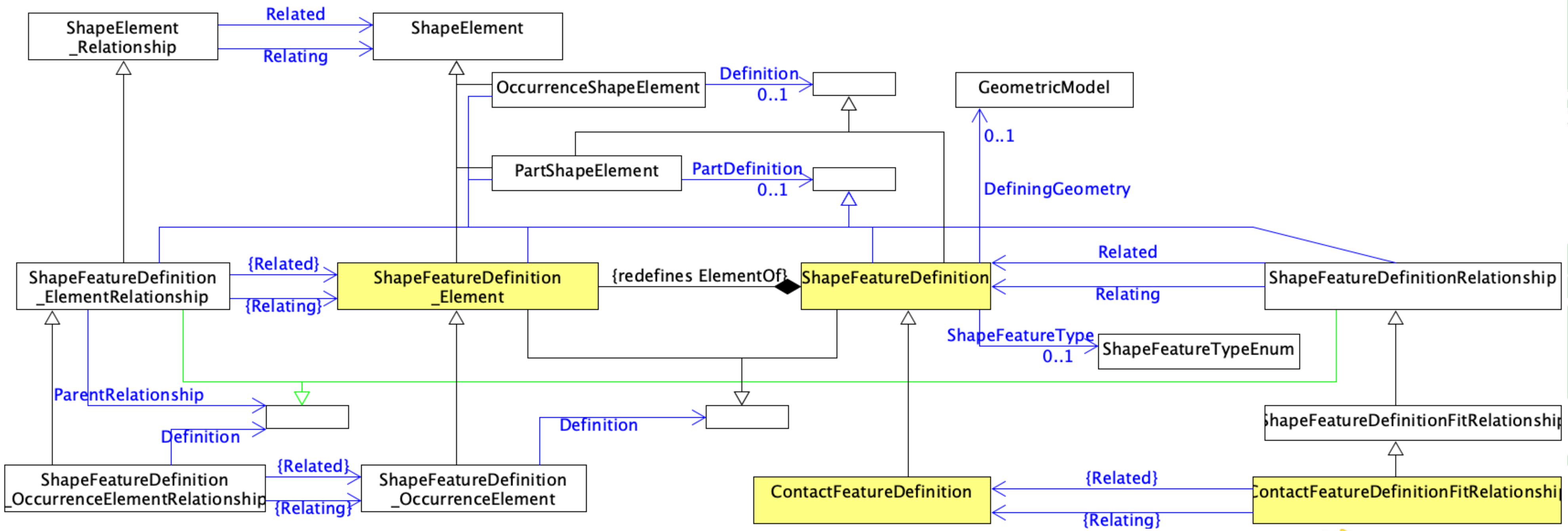
```
<Part uid="_311000"> <!-- Part_H1 -->
...
 <PartTypes>
  <PartCategoryEnum>assembly</PartCategoryEnum>
  <PartCategoryEnum>wiring_harness</PartCategoryEnum>
 </PartTypes>
 <Versions>
  <PartVersion uid="_311001">
  ...
  <Views>
   <PartView xsi:type="n0:WiringHarnessAssemblyDesign" uid="_311002">
    ...
    <!--2 AssemblyJoints on Phone1 with cable3-->
    <ShapeElement xsi:type="n0:AssemblyShapeJoint" uid="_311020">
     <ShapeElementRelationship xsi:type="n0:AssemblyShapeJointItemRelationship" uid="_311120">
      <Related uidRef="_204023"/> <!-- cable3#end b A -->
      <RelationType><ClassString></ClassString></RelationType>
     </ShapeElementRelationship>
     <ShapeElementRelationship xsi:type="n0:AssemblyShapeJointItemRelationship" uid="_311220">
      <Related uidRef="_217102"/> <!-- phone1#Join signal -->
      <RelationType><ClassString></ClassString></RelationType>
     </ShapeElementRelationship>
     <JointType>crimped_connection</JointType>
    </ShapeElement>

    <ViewOccurrenceRelationship uid="_315021" xsi:type="n0:NextAssemblyOccurrenceUsage">
     <Related uidRef="_204006"/> <!--CableOccurrence cable3 (speaker wire)-->
     <RelationType>
      <ClassString>next assembly occurrence</ClassString>
     </RelationType>
    </ViewOccurrenceRelationship>
    <ViewOccurrenceRelationship uid="_315043" xsi:type="n0:NextAssemblyOccurrenceUsage">
     <Related uidRef="_217100"/> <!-- phone1 -->
     <RelationType>
      <ClassString>next assembly occurrence</ClassString>
     </RelationType>
    </ViewOccurrenceRelationship>
    ...
   </PartView>
  </Views>
  </PartVersion>
 </Versions>
</Part>
```

Table of Content

- Introduction
- Basic XML structure
- Part, PartCategory & PartVersion
- PartView & ViewContext
- ProductClass & ProductConfiguration
- Occurrence, Assembly & WiringHarnessAssemblyDesign
- ShapeElement
- Geometric & topological models
- Connectors & terminals
- Features, Definitions & AssemblyShapeJoints
- Wire, Cables & TransportFeature
- Shapes/ContactFeatureDefinition & -Elements
- CrossSection of Wire, Cable & HarnessSegment
- Geometry/Topology Associations
- Complex Connectors / Assembly Hierarchy
- External Element References
- ComposedGeometricModel

Shapes/ContactFeatureDefinition & -Elements (1 of 10)

- a *ShapeFeatureDefinition* allows for the identification of an independent feature with/without a defining *GeometricModel*
- a *ShapeFeatureDefinition* can be used as the definition for a PartFeature. This allows to define the common shape of a feature once, and then use it for many different PartViews
- a ContactFeatureDefinition is a kind of ShapeFeatureDefinition that is intended to be contacted by other corresponding ContactFeatureDefinitions. Mating pairs can be identified by ContactFeatureDefinitionFitRelationship
- a ShapeFeatureDefinitionElement identifies a part of a ShapeFeatureDefinition
- with the subtype ShapeFeatureDefinitionOccurrenceElement it is possible to compose complex ShapeFeatureDefinitions from simpler ones.
- ... (more to follow)



Shapes/ContactFeatureDefinition & -Elements (2 of 10)

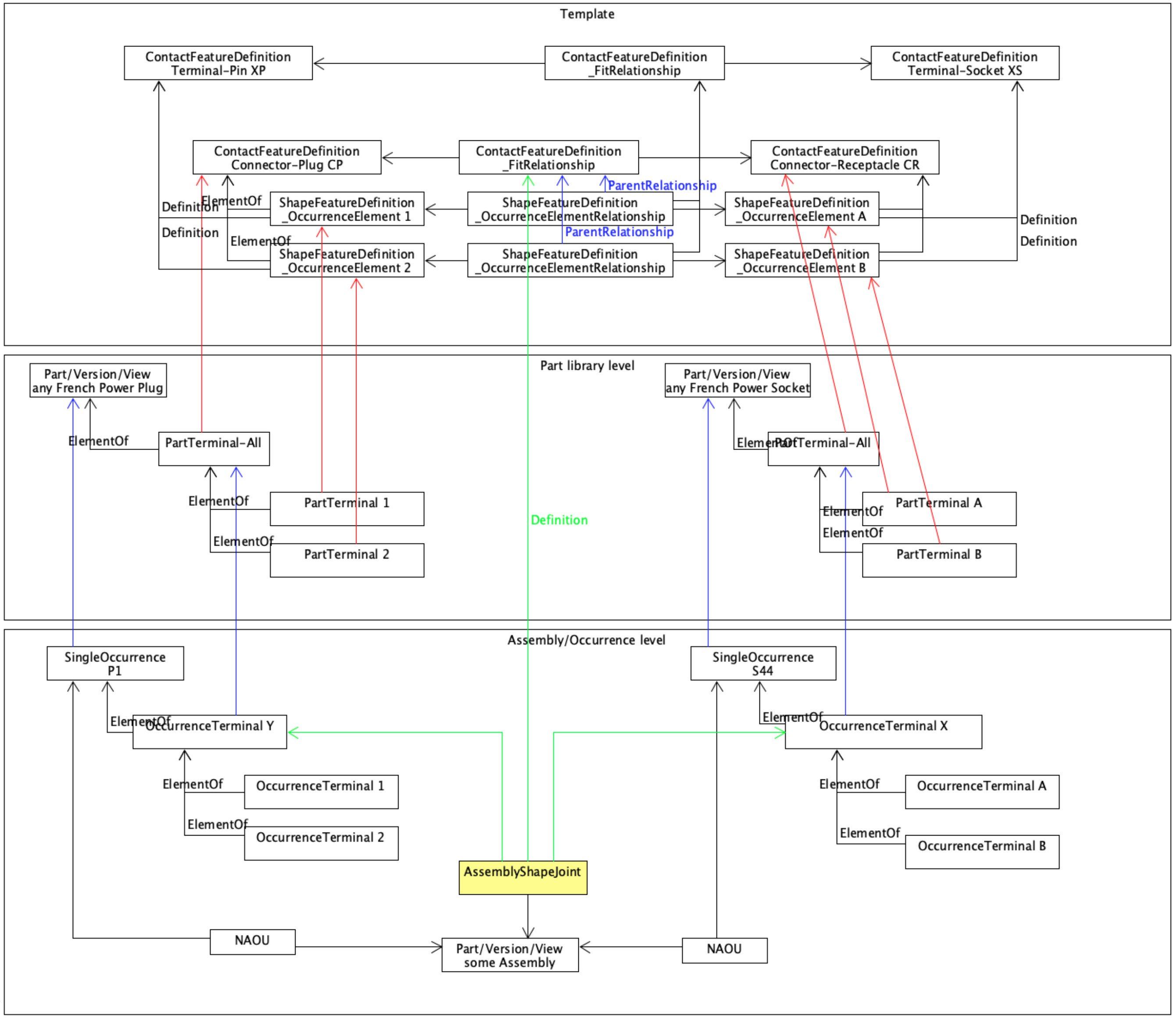
Principle hierarchical usage on ContactFeatureDefinition level –part and -assembly level

The data model can be used to define connectivity on the template level, part library and assembly level

- Level 1: ContactFeatureDefinition. This level might be broken down in several sub-levels.
  - A lower level ContactFeatureDefinition and –FitRelationship are used e.g. a terminal pin and a terminal socket fit together
  - A higher level ContactFeatureDefinition and –FitRelationship are used to see how a plug and a receptacle connector, each with two terminals, fit together by re-using the ContactFeatureDefinition and –FitRelationship from the lower level together
  - Higher level ShapeFeatureDefinitionOccurrenceElement are defined by either lower level ContactFeatureDefinitions (see example), or lower level ShapeFeatureDefinitionOccurrenceElements (not shown here).
  - Higer level ShapeFeatureDefinitionOccurrenceElements might be related by ShapeFeatureDefinitionOccurrenceElementRelationship with a ParentRelationship and a Definition by the lower level
  - The hierarchy on ContactFeatureDefinition can span several levels, e.g. including inserts and contact cavities
- Level 2: Part level (e.g. in a part library)
  - It is sufficient to define the Terminals/PartFeatures and refers to the corresponding ContactFeatureDefinitions and ShapeFeatureDefinitionOccurrenceElements. Implementations can then derive which part fits with which other parts
  - Each connector is represented by a higher-level Terminal/PartContactFeature and lower-level Terminal/PartContactFeature
- Level 3: Occurrence and assembly level
  - It is sufficient to have an AssemblyShapeJoints on the main Terminals/OccurrenceFeatures together with a definition from level 1.
  - In this example e.g. an AssemblyShapeJoint is it sufficient to have a higher level OccurrenceTerminals X and Y
  - A receiving application can deduce that X.Terminal1 is connected with Y.TerminalA and X.Terminal2 is connected with Y.TerminalB. So, there is no need to repeat this information.
- The above pattern greatly simplify the definition of parts and assemblies as the detailed connectivity is predefined in fit patterns
  - On the part/library level, only the underlying ContactFeatureDefinition are referenced
  - On the Assembly/Occurrence level, only a single AssemblyShapeJoint statement for the main Terminals/Features is needed
  - A receiving application can simply deduced the underlying connectivity details for each Terminal by following the explained hierarchy. This typical usages are e.g. for power plug connections and Ethernet/RJ45 connections

Shapes/ContactFeatureDefinition & -Elements (3 of 10)

Principle hierarchical usage on ContactFeatureDefinition level –part and -assembly level


Template
ContactFeatureDefinition Terminal-Pin XP
ContactFeatureDefinition _FitRelationship
ContactFeatureDefinition Terminal-Socket XS
ContactFeatureDefinition Connector-Plug CP
ContactFeatureDefinition _FitRelationship
ContactFeatureDefinition Connector-Receptacle CR
ParentRelationship
Definition
ElementOf
ShapeFeatureDefinition _OccurrenceElement 1
ShapeFeatureDefinition _OccurrenceElementRelationship
ShapeFeatureDefinition _OccurrenceElement A
Definition
ParentRelationship
Definition
Definition
ElementOf
ShapeFeatureDefinition _OccurrenceElement 2
ShapeFeatureDefinition _OccurrenceElementRelationship
ShapeFeatureDefinition _OccurrenceElement B
Part library level
Part/Version/View any French Power Plug
Part/Version/View any French Power Socket
ElementOf
PartTerminal-All
ElementOf
PartTerminal-All
ElementOf
PartTerminal 1
Definition
ElementOf
PartTerminal A
ElementOf
PartTerminal 2
ElementOf
PartTerminal B
Assembly/Occurrence level
SingleOccurrence P1
SingleOccurrence S44
ElementOf
OccurrenceTerminal Y
ElementOf
OccurrenceTerminal X
ElementOf
OccurrenceTerminal 1
ElementOf
OccurrenceTerminal A
ElementOf
OccurrenceTerminal 2
ElementOf
OccurrenceTerminal B
AssemblyShapeJoint
NAOU
Part/Version/View some Assembly
NAOU

Shapes/ContactFeatureDefinition & -Elements (4 of 10)

Example on template level with Deutsch IMC Series connector

This example shows how to use a –FitRelationship between ContactFeatureDefinitions for a size 20 cavity (of some plug) and the contact profile size 20 pin connector contact

- an Organization to identify the company “Deutsch”
- two ContactFeatureDefinitions for size 20 cavity_profile and contact_profile
- the ContactFeatureDefinitions are related by a ContactFeatureDefinitionFitRelationship, this means that the size 20 contact_profile fits into the size 20 cavity_profile
  Note: fit relationship can be applied in either direction
  Note: the fit relationship establishes a mechanical fit
- this information is not essential for exchange of a EWH design, but is useful in a library so that CAD systems can make the right selection
  Note: fit relationships can be used for validation of a design

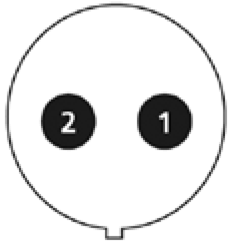
```
<Organization uid="_100"> <!— Deutsch company -->
 <Id id="http://www.deutsch.net"/>
</Organization>

<ShapeFeatureDefinition xsi:type="n0:ContactFeatureDefinition" uid="_101">
 <Id><Identifier uid="_201” id=”IMC Series Size 20 cavity” idContextRef=”_100”></Id> …
 <ShapeFeatureType>cavity_profile</ShapeFeatureType>
</ShapeFeatureDefinition>
<ShapeFeatureDefinition xsi:type="n0:ContactFeatureDefinition" uid="_102">
 <Id><Identifier uid="_201” id=”IMC Series Size 20 pin” idContextRef=”_100”></Id> …
 <ShapeFeatureType>contact_profile</ShapeFeatureType>
 <ShapeFeatureDefinitionRelationship xsi:type="n0:ContactFeatureDefinitionFitRelationship" uid="_103">
  <Related uidRef="_101"/>
 </ShapeFeatureDefinitionRelationship>
</ShapeFeatureDefinition>
```

Shapes/ContactFeatureDefinition & -Elements (5 of 10)

Example on the part level with Deutsch IMC Series connector

This example on a part level shows how a fit relationship can be used to figure out that connector contact fits into either of the cavities of a connector

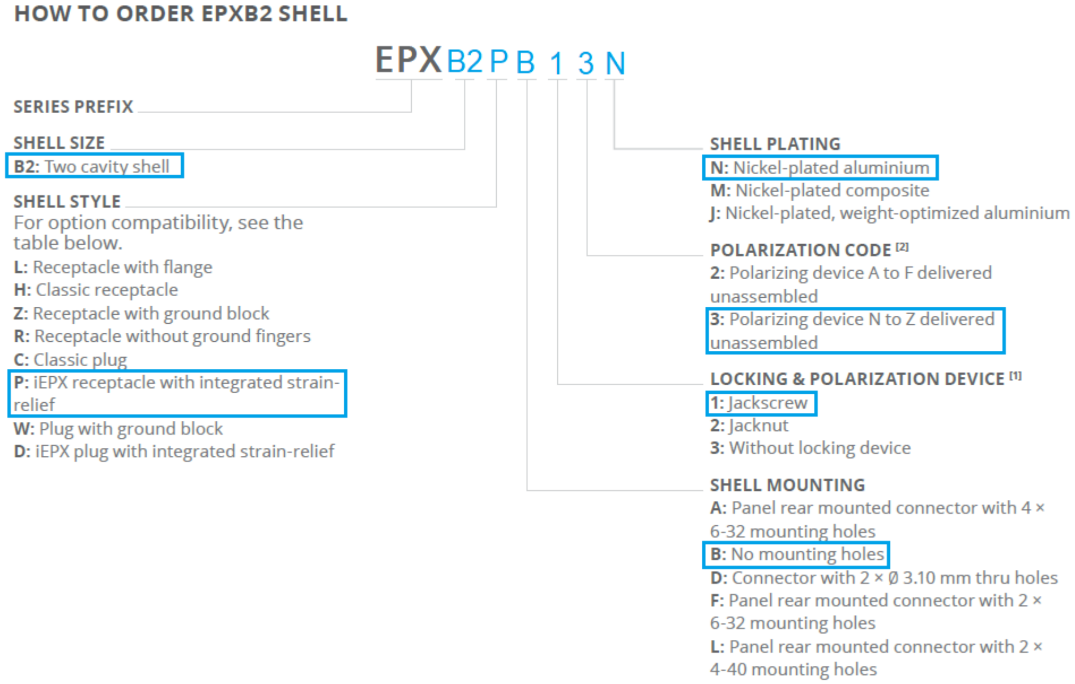
- a *Part* that is a *connector* with two *PartContactFeature* “1” and “2” that are defined by a *ContactFeatureDefinition* that is a cavity*_profile*
- a *Part* that is a *connector_contact* with a *PartContactFeature* “o” that is defined by a *ContactFeatureDefinition* that is a *contact_profile*



```
<Part uid="_200"> <!— Deutsch connector -->
 <Id><Identifier uid="_201” id=”IM16-2002X” idContextRef=”_100”></Id>
 <PartTypes>
  <PartCategoryEnum>connector</PartCategoryEnum>
 </PartTypes>
 <Versions>
  <PartVersion uid="_202">
   <Id id=”Version 1"/>
   <Views>
    <PartView uid="_202">
     <InitialContext uidRef="_100102"/>
     <ShapeElement xsi:type="n0:PartContactFeature" uid="_203">
      <Id id=”1"/>
      <Definition uidRef="_101"/>
     </ShapeElement>
     <ShapeElement xsi:type="n0:PartContactFeature" uid="_204">
      <Id id=”2"/>
      <Definition uidRef="_101"/>
     </ShapeElement>
    </PartView>
   </Views>
  </PartVersion>
 </Versions>
</Part>
<Part uid="_300"> <!— Deutsch connector-contact -->
 <Id><Identifier uid="_201” id=”6860-201-20278” idContextRef=”_100”></Id>
 <PartTypes>
  <PartCategoryEnum>connector_contact</PartCategoryEnum>
 </PartTypes>
 <Versions>
  <PartVersion uid="_302">
   <Id id=”Version 1"/>
   <Views>
    <PartView uid="_303">
     <InitialContext uidRef="_100102"/>
     <ShapeElement xsi:type="n0:PartContactFeature" uid="_304">
      <Id id=”o"/>
      <Definition uidRef="_102"/>
     </ShapeElement>
    </PartView>
   </Views>
  </PartVersion>
 </Versions>
</Part>
```

Shapes/ContactFeatureDefinition & -Elements (6 of 10)

EPXB example from Radiall (EN 4644)



HOW TO ORDER EPXB2 SHELL

EPX B2 P B 1 3 N

SERIES PREFIX

SHELL SIZE
B2: Two cavity shell

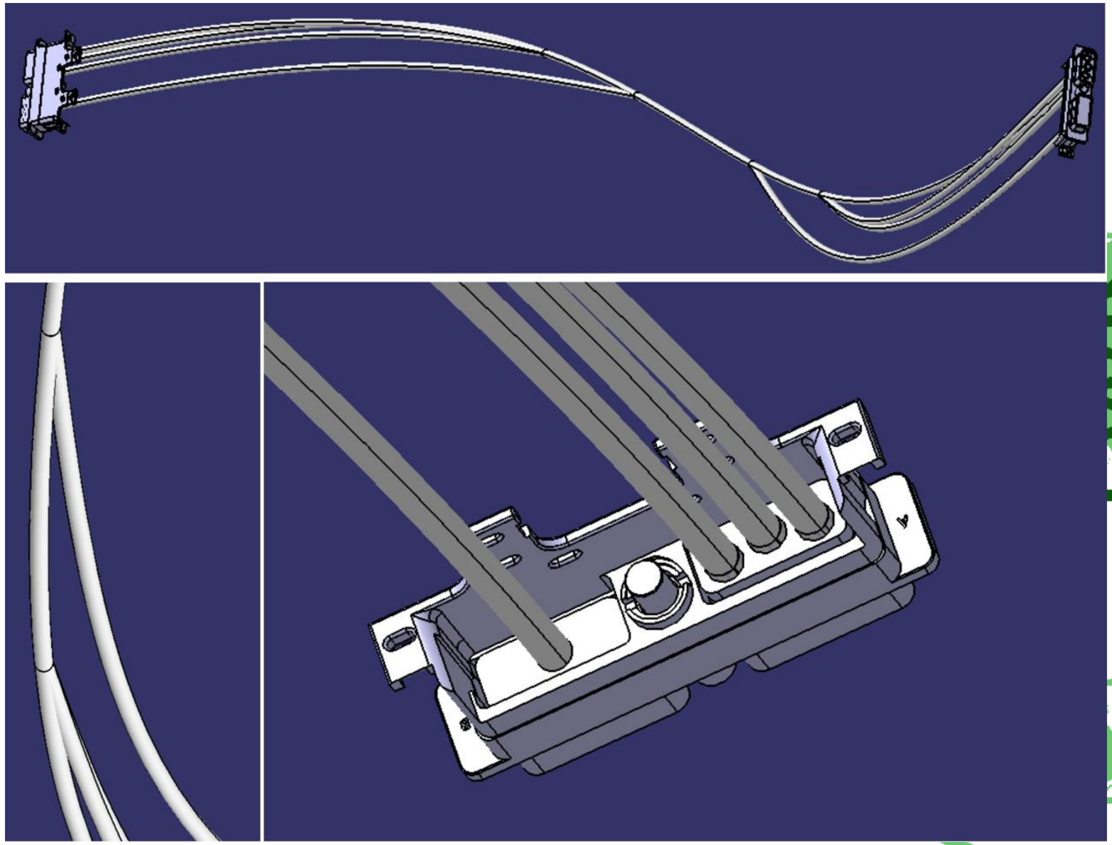
SHELL STYLE
For option compatibility, see the table below.
L: Receptacle with flange
H: Classic receptacle
Z: Receptacle with ground block
R: Receptacle without ground fingers
C: Classic plug
P: iEPX receptacle with integrated strain-relief
W: Plug with ground block
D: iEPX plug with integrated strain-relief

SHELL PLATING
N: Nickel-plated aluminium
M: Nickel-plated composite
J: Nickel-plated, weight-optimized aluminium

POLARIZATION CODE [2]
2: Polarizing device A to F delivered unassembled
3: Polarizing device N to Z delivered unassembled

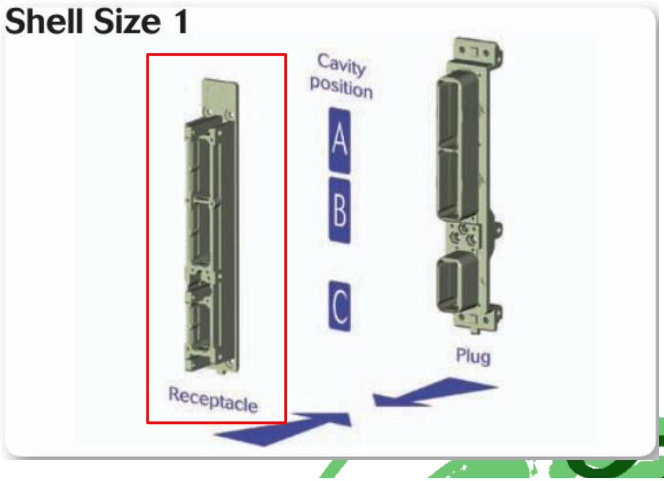
LOCKING & POLARIZATION DEVICE [1]
1: Jackscrew
2: Jacknut
3: Without locking device

SHELL MOUNTING
A: Panel rear mounted connector with 4 × 6-32 mounting holes
B: No mounting holes
D: Connector with 2 × Ø 3.10 mm thru holes
F: Panel rear mounted connector with 2 × 6-32 mounting holes
L: Panel rear mounted connector with 2 × 4-40 mounting holes



Shapes/ContactFeatureDefinition & -Elements (7 of 10)

Relationship between ARINC 600 receptable and plug



```
<Part uid="_125000">
<!-- ARINC 600 shell size 1 rack plug -->
 <Id id="SB6 4 1 M"/>
 <Name>
  <CharacterString>ARINC 600 shell size 1 rack plug</CharacterString>
 </Name>
 <PartTypes>
  <PartCategoryEnum>discrete</PartCategoryEnum>
  <PartCategoryEnum>connector_housing</PartCategoryEnum>
 </PartTypes>
 <Versions>
  <PartVersion uid="_125001">
   <Id/>
   <Views>
    <PartView uid="_125002">
     <InitialContext uidRef="_100102"/>
     <Occurrence xsi:type="n0:SingleOccurrence" uid="_225100">
      <Id id="arinc1 housing"/>
      ...
     </Occurrence>
     <ShapeElement xsi:type="n0:PartContactFeature" uid="_125010">
      ...
      <!-- A-slot-->
      ...
       <!-- B-slot-->
       ...
       <!-- C-slot-->
      <ShapeElement xsi:type="n0:PartContactFeature" uid="_125013">
       <PartDefinition uidRef="_60C030"/>
      </ShapeElement>
      <PartDefinition uidRef="_60C000"/>
     </ShapeElement>
    </PartView>
   </Views>
  </PartVersion>
 </Versions>
</Part>
```

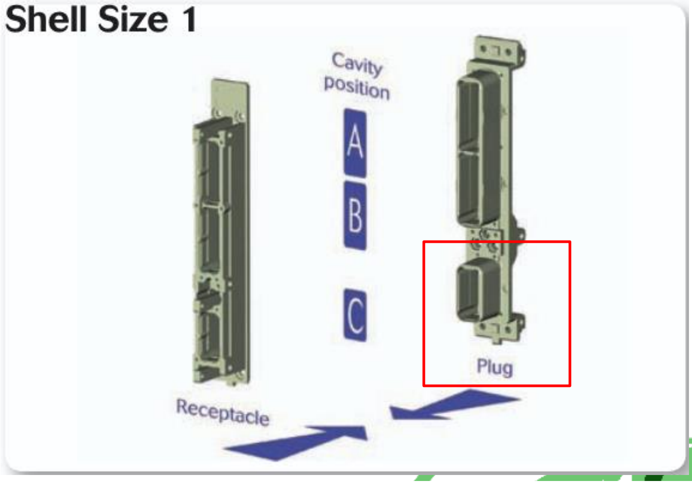
```
<ShapeElement xsi:type="n0:ShapeFeatureDefinitionOccurrenceElement" uid="_60C030">
 <Name>
  <CharacterString>C-slot</CharacterString>
 </Name>
 <Definition uidRef="_604000"/>
</ShapeElement>
```

```
<ShapeFeatureDefinition xsi:type="n0:ContactFeatureDefinition" uid="_604000">
 <Name>
  <CharacterString>arinc600 rack plug C slot profile</CharacterString>
 </Name>
 <ShapeFeatureType>slot_profile</ShapeFeatureType>
 <ShapeFeatureDefinitionRelationship xsi:type="n0:ContactFeatureDefinitionFitRelationship" uid="_604010">
  <Related uidRef="_603000"/>
 </ShapeFeatureDefinitionRelationship>
</ShapeFeatureDefinition>
```

Shapes/ContactFeatureDefinition & -Elements (8 of 10)

Relationship between ARINC 600 receptable and plug

```
<Part uid="_112000">
<!-- ARINC 600 shell 1 C 5W2 insert -->
 <Id id="8660-5W2"/>
 <Name>
  <CharacterString>ARINC 600 shell size 1 C 5W2 insert</CharacterString>
 </Name>
 <PartTypes>
  <PartCategoryEnum>discrete</PartCategoryEnum>
  <PartCategoryEnum>connector_insert</PartCategoryEnum>
 </PartTypes>
 <Versions>
  <PartVersion uid="_112001">
   <Id id="Version 1"/>
   <Views>
    <PartView uid="_112002">
     <InitialContext uidRef="_100102"/>
     <Occurrence xsi:type="n0:SingleOccurrence" uid="_212010">
      <Id id="C-insert"/>
      …
     </Occurrence>
      …
      <!-- size #5-->
      <!-- size 16-->
      <!– C-insert profile-->
     <ShapeElement xsi:type="n0:PartContactFeature" uid="_112090">
      <Id id="Insert profile"/>
      <PartDefinition uidRef="_60B090"/>
     </ShapeElement>
    </PartView>
   </Views>
  </PartVersion>
 </Versions>
</Part>
```



```
<ShapeElement xsi:type="n0:ShapeFeatureDefinitionOccurrenceElement" uid="_60B090">
 <Name>
  <CharacterString>C-insert profile</CharacterString>
 </Name>
 <Definition uidRef="_603000"/>
</ShapeElement>
```

↓

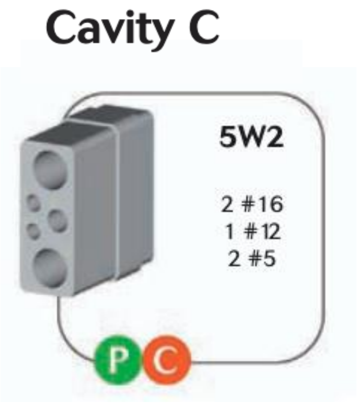
```
<ShapeFeatureDefinition xsi:type="n0:ContactFeatureDefinition" uid="_603000">
 <Name>
  <CharacterString>arinc600 rack plug C insert profile</CharacterString>
 </Name>
 <ShapeFeatureType>insert_profile</ShapeFeatureType>
</ShapeFeatureDefinition>
```

Shapes/ContactFeatureDefinition & -Elements (9 of 10)

Cavity C with contact type 5W2 for ARINC 600

```
<ShapeFeatureDefinition xsi:type="n0:ContactFeatureDefinition" uid="_60B000">
 <Name>
  <CharacterString>arinc600 rack plug C 5W2 insert profile</CharacterString>
 </Name>
 <ShapeElement xsi:type="n0:ShapeFeatureDefinitionOccurrenceElement" uid="_60B010">
  <Name>
   <CharacterString>1</CharacterString> <!-- size #5 -->
  </Name>
  <Definition uidRef="_608000"/>
 </ShapeElement>
 <ShapeElement xsi:type="n0:ShapeFeatureDefinitionOccurrenceElement" uid="_60B020">
  <Name>
   <CharacterString>5</CharacterString> <!-- size #5 -->
  </Name>
  <Definition uidRef="_608000"/>
 </ShapeElement>
 <ShapeElement xsi:type="n0:ShapeFeatureDefinitionOccurrenceElement" uid="_60B030">
  <Name>
   <CharacterString>2</CharacterString> <!-- size #16 -->
  </Name>
  <Definition uidRef="_606000"/>
 </ShapeElement>
 <ShapeElement xsi:type="n0:ShapeFeatureDefinitionOccurrenceElement" uid="_60B040">
  <Name>
   <CharacterString>4</CharacterString> <!-- size #16 -->
  </Name>
  <Definition uidRef="_606000"/>
 </ShapeElement>
 <ShapeElement xsi:type="n0:ShapeFeatureDefinitionOccurrenceElement" uid="_60B050">
  <Name>
   <CharacterString>3</CharacterString> <!-- size #12 -->
  </Name>
  <Definition uidRef="_60A000"/>
 </ShapeElement>
 <ShapeElement xsi:type="n0:ShapeFeatureDefinitionOccurrenceElement" uid="_60B090">
  <Name>
   <CharacterString>C-insert profile</CharacterString>
  </Name>
  <Definition uidRef="_603000"/>
 </ShapeElement>
</ShapeFeatureDefinition>
```

```
<ShapeFeatureDefinition xsi:type="n0:ContactFeatureDefinition" uid="_603000">
 <Name>
  <CharacterString> arinc600 rack plug C insert profile</CharacterString>
 </Name>
```



Shapes/ContactFeatureDefinition & -Elements (10 of 10)

Example with the plug #5 of 5W2 insert

```
<ShapeFeatureDefinition xsi:type="n0:ContactFeatureDefinition" uid="_60B000">
 <Name>
  <CharacterString>arinc600 rack plug C 5W2 insert profile</CharacterString>
 </Name>
 <ShapeElement xsi:type="n0:ShapeFeatureDefinitionOccurrenceElement" uid="_60B010">
  <Name>
   <CharacterString>1</CharacterString> <!-- size #5 -->
  </Name>
  <Definition uidRef="_608000"/>
 </ShapeElement>
 ...
</ShapeFeatureDefinition>
```

```
<ShapeFeatureDefinition xsi:type="n0:ContactFeatureDefinition" uid="_608000">
 <Name>
  <CharacterString>arinc600 rack plug #5 cavity</CharacterString>
 </Name>
 <ShapeFeatureType>cavity_profile</ShapeFeatureType>
 <ShapeFeatureDefinitionRelationship xsi:type="n0:ContactFeatureDefinitionFitRelationship" uid="_608010">
  <Related uidRef="_607100"/>
 </ShapeFeatureDefinitionRelationship>
</ShapeFeatureDefinition>

<ShapeFeatureDefinition xsi:type="n0:ContactFeatureDefinition" uid="_607100">
 <Name>
  <CharacterString>arinc600 rack plug #5 contact mounting</CharacterString>
 </Name>
 <ShapeFeatureType>cavity_plug_or_contact_profile</ShapeFeatureType>
</ShapeFeatureDefinition>
```

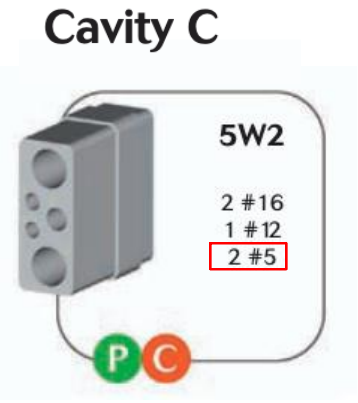
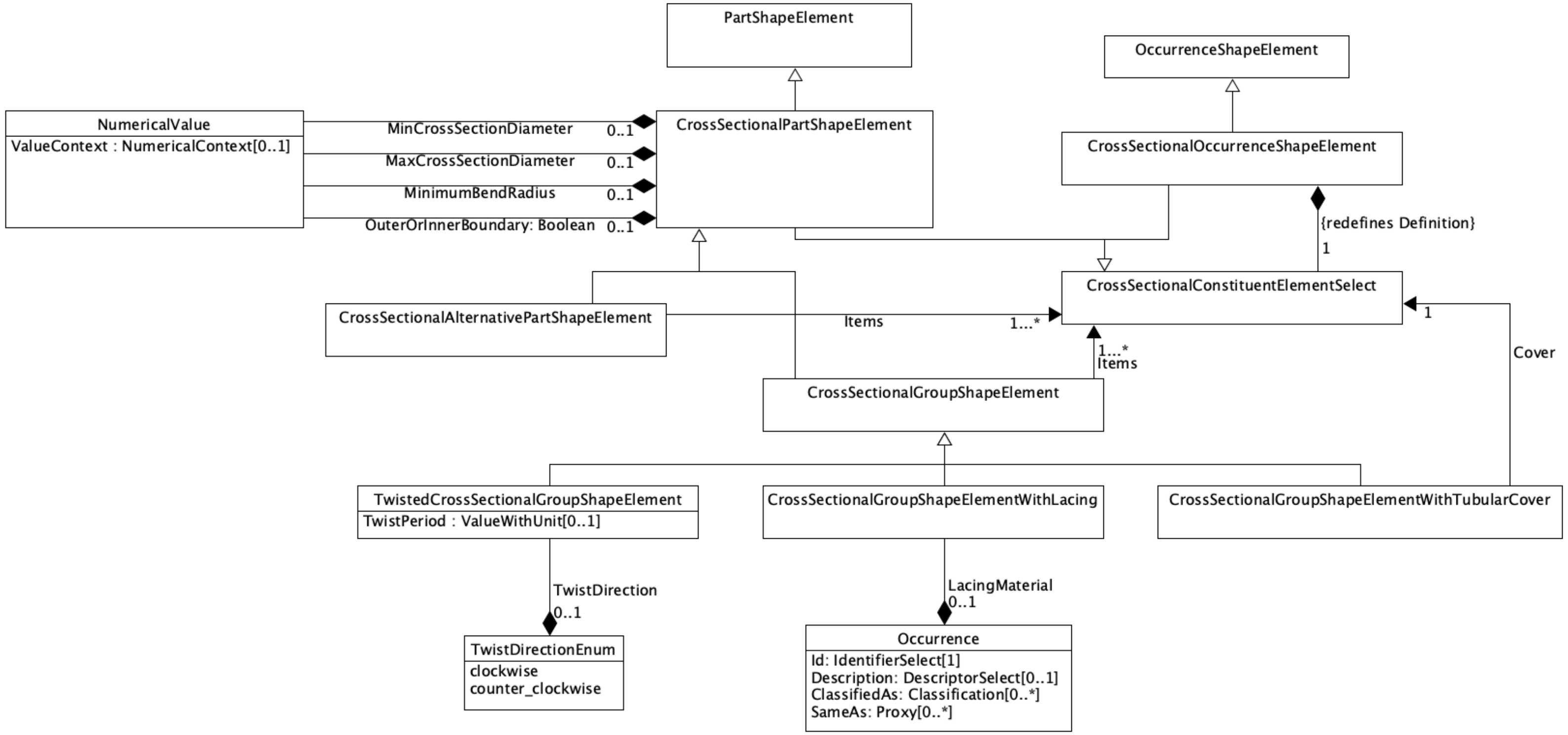


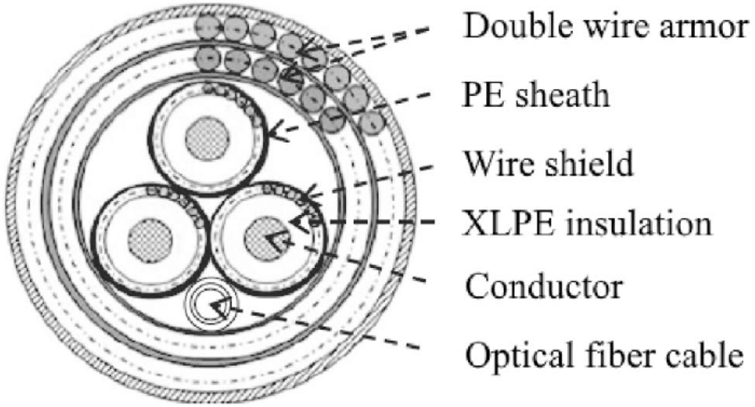
Table of Content

- Introduction
- Basic XML structure
- Part, PartCategory & PartVersion
- PartView & ViewContext
- ProductClass & ProductConfiguration
- Occurrence, Assembly & WiringHarnessAssemblyDesign
- ShapeElement
- Geometric & topological models
- Connectors & terminals
- Features, Definitions & AssemblyShapeJoints
- Wire, Cables & TransportFeature
- Shapes/ContactFeatureDefinition & -Elements
- CrossSection of Wire, Cable & HarnessSegment
- Geometry/Topology Associations
- Complex Connectors / Assembly Hierarchy
- External Element References
- ComposedGeometricModel

CrossSection of Wire, Cable & HarnessSegment (1 of 6)

- The cross section of a harness segment, cables, wires or any protective coverage is perpendicular to the centre line of that object.
- The cross-sectional data model follows the so-called "onion model" with concentric layers and typically several cores.
- A CrossSectionalPartShapeElement represents the cross section of a length extruded part such as a wire, a cable or a protective coverage. It has properties for the minimum and maximum cross-section diameter (MinimumBendRadius) and weather the cross section represents the Outer or one of potentially several Inner shapes (OuterOrInnerBoundary).
- A CrossSectionalGroupShapeElement consists of the combination of one or several items.


PartShapeElement
OccurrenceShapeElement
NumericalValue
ValueContext : NumericalContext[0..1]
MinCrossSectionDiameter 0..1
MaxCrossSectionDiameter 0..1
MinimumBendRadius 0..1
OuterOrInnerBoundary: Boolean 0..1
CrossSectionalPartShapeElement
CrossSectionalOccurrenceShapeElement
{redefines Definition}
1
CrossSectionalConstituentElementSelect
CrossSectionalAlternativePartShapeElement
Items
1...*
1
1...*
Items
Cover
CrossSectionalGroupShapeElement
TwistedCrossSectionalGroupShapeElement
TwistPeriod : ValueWithUnit[0..1]
CrossSectionalGroupShapeElementWithLacing
CrossSectionalGroupShapeElementWithTubularCover
TwistDirection
0..1
LacingMaterial
0..1
TwistDirectionEnum
clockwise
counter_clockwise
Occurrence
Id: IdentifierSelect[1]
Description: DescriptorSelect[0..1]
ClassifiedAs: Classification[0..*]
SameAs: Proxy[0..*]



Source: Rentschler, Manuel & Adam, Frank & Chainho, Paulo & Krügel, Kilian & Vicente, Pedro. (2020). Parametric study of dynamic inter-array cable systems for floating offshore wind turbines. Marine Systems & Ocean Technology. 15. 10.1007/s40868-020-00071-7.

CrossSection of Wire, Cable & HarnessSegment (2 of 6)

Constraints and recommendations:

If a `CrossSectionalPartShapeElement` is an `elementOf` another one, then the flag `OuterOrInnerBoundary` shall be opposite of the one indicated by `elementOf`.

1. A top level `CrossSectionalPartShapeElement` shall always be an outer shape element
2. A `CrossSectionalGroupShapeElement` shall not have `OuterOrInnerBoundary` be set, as it is implicitly always outer
3. The "cover" of a `CrossSectionalGroupShapeElementWithTubularCover` shall always be an inner boundary
4. The resulting shape of an `CrossSectionalGroupShapeElementWithTubularCover` is implicitly given by the outer shape of the "cover"
5. The items of a `CrossSectionalGroupShapeElement` shall be either `CrossSectionalOccurrenceShapeElement` or subtypes of `CrossSectionalPartShapeElement`.
6. The specific type `CrossSectionalPartShapeElement` is not allowed as an item for a `CrossSectionalGroupShapeElement`

CrossSection of Wire, Cable & HarnessSegment (3 of 6)

Implementation example as Part21, using the DomainModel:

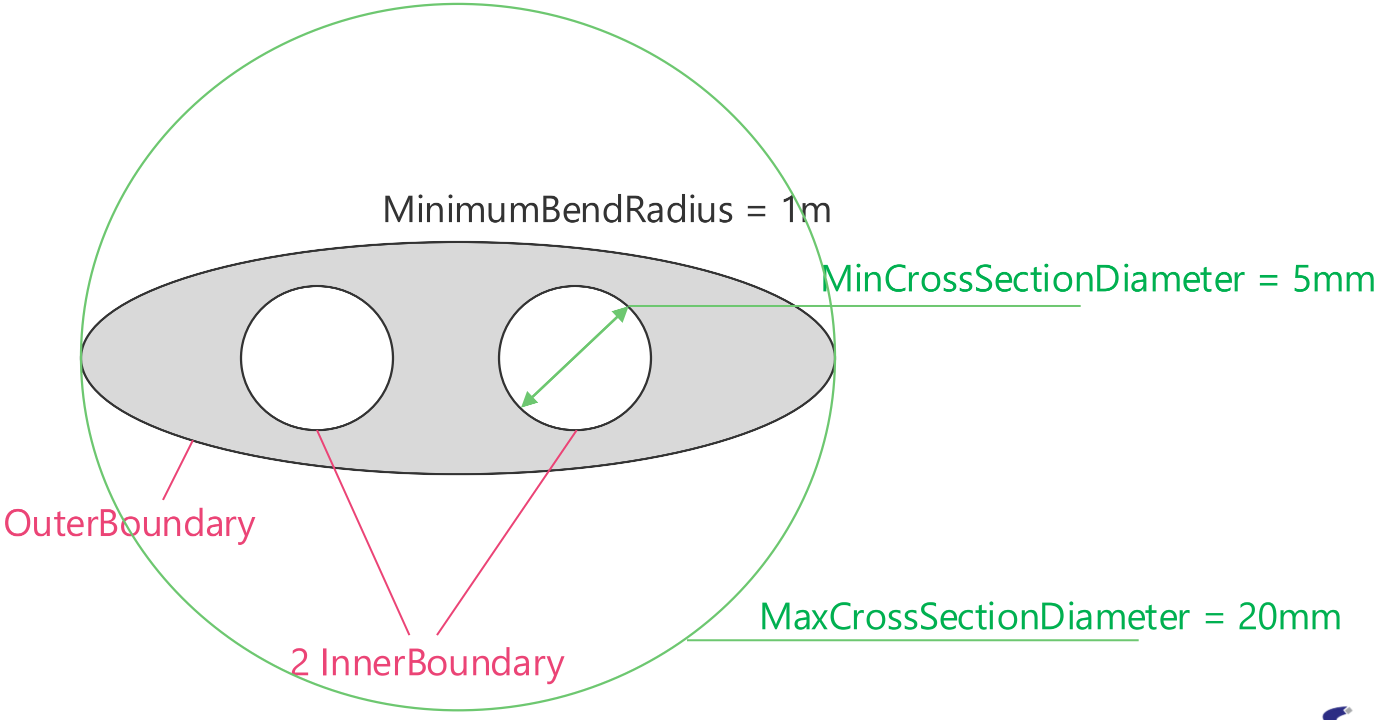
```
#10=PartView(...)
#12=CrossSectionalPartShapeElement(... #10, ... OuterOrInnerBoundary=TRUE )
#13=CrossSectionalPartShapeElement(... #12, ... OuterOrInnerBoundary=FALSE )
#14=CrossSectionalPartShapeElement(... #12, ... OuterOrInnerBoundary=FALSE )

#20=QuantifiedOccurrence( #10, length=3m );
#22=CrossSectionalOccurrenceShapeElement(...#20,  #12)
#23=CrossSectionalOccurrenceShapeElement(...#22,  #13)
#24=CrossSectionalOccurrenceShapeElement(...#22,  #14)

#30=PartView(...): Flat structure, used so far, also for a single part
#32=CrossSectionalPartShapeElement(... #30, ... OuterOrInnerBoundary=TRUE )
#33=CrossSectionalPartShapeElement(... #32, ... OuterOrInnerBoundary=FALSE )

#40=QuantifiedOccurrence( #10, length=3m );
#42=CrossSectionalOccurrenceShapeElement(...#40,  #11)
#43=CrossSectionalOccurrenceShapeElement(...#42,  #12)

===
#100=WiringHarnessAssemblyDesign(...)
#101=NextAssemblyOccurrenceUsage ( #100, #20)
#102=NextAssemblyOccurrenceUsage ( #100, #40)
...

#111=HarnessSegment(... #100, CrossSection=#112 );
#112=CrossSectionalGroupShapeElementWithTubularCover( ... #100,
Items=(#22), Cover=#43)   ## onion model
```

In this example we have two PartViews and a WiringHarnessAssemblyDesign:

- PartView #10 consists of a single outer boundary in which two inner boundaries are nested, #20 being an occurrence of #10
- The OccurrenceShapeElements (#2x) are defined by the corresponding PartShapeElements (#1x)
- The Part #30 has a single outer boundary in which a single inner boundary is nested, #40 being an occurrence of #30
- The OccurrenceShapeElements (#4x) are defined by the corresponding PartShapeElements (#3x)
- The WiringHarnessAssemblyDesign (#100) is an assembly of the occurences #20 and #40
- The HarnessSegment #111 of #100 has a defined cross section given by #112
- The CrossSectionalGroupShapeElementWithTubularCover #112 nest within the inner boundary #43 the outer boundary defined by #22 (here only one, there might be several)
- The outer boundary of #112 can be derived from #42 by following #43

CrossSection of Wire, Cable & HarnessSegment (4 of 6)

- This example consists of 4 CrossSectionalPartShapeElement for the same part
  - One CrossSectionalPartShapeElement for the Outer oval shape (the MaxCrossSectionDiameter is essential to know the minimum size of a hole that this object should fit through). This typically has a minimum bend radius.
  - Two circular Inner CrossSectionalPartShapeElement (the MinCrossSectionDiameter defines how big the object can be placed in them)
  - One CrossSectionalGroupShapeElement that combined the three CrossSectionalPartShapeElement above. The CrossSectionalGroupShapeElement is also a CrossSectionalPartShapeElement.



CrossSection of Wire, Cable & HarnessSegment (5 of 6)

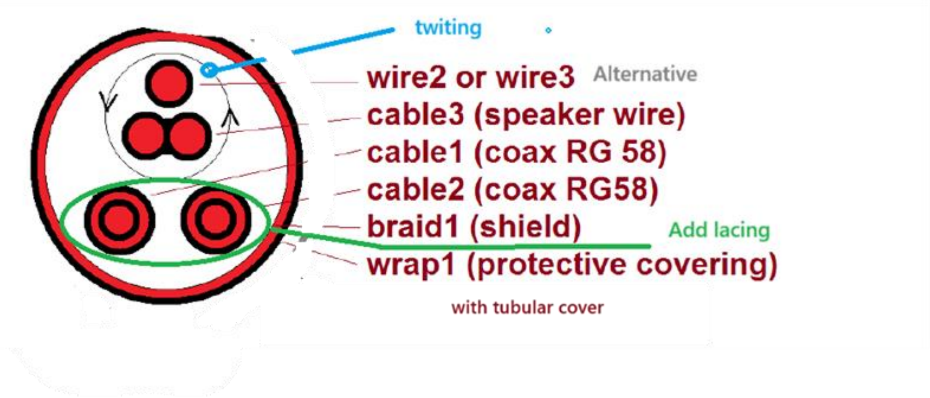
- Here an example with an outer `CrossSectionalPartShapeElement` with properties:
  - `MaxCrossSectionDiameter`: 20 mm
  - `MinimumBendRadius`: 1000 mm
- Two inner `CrossSectionalPartShapeElements` are contained within the outer `CrossSectionalPartShapeElements` with both having the property:
  - `MinCrossSectionDiameter`: 5mm
- The `CrossSectionalOccurenceShapeElements` are defined by corresponding `CrossSectionalPartShapeElements`

```
<ShapeElement xsi:type="n0:CrossSectionalOccurrenceShapeElement" uid="_202007">
 <Definition uidRef="_102020"/> <!-- see below -->
 <!---> optional <-->
 <ShapeElement xsi:type="n0:CrossSectionalOccurrenceShapeElement" uid="_202007_inner1">
  <Definition uidRef="_101020_inner1"/> <!-- see below -->
 </ShapeElement>
 <ShapeElement xsi:type="n0:CrossSectionalOccurrenceShapeElement" uid="_202007_inner2">
  <Definition uidRef="_101020_inner2"/> <!-- see below -->
 </ShapeElement>
</ShapeElement>
<ShapeElement xsi:type="n0:CrossSectionalPartShapeElement" uid="_101020">
 <OuterOrInnerBoundary>outer_boundary</OuterOrInnerBoundary>
 <MaxCrossSectionDiameter xsi:type="n0:NumericalValue" uid="_xxx1">
  <Unit uidRef="_100301"/>
  <ValueComponent>20</ValueComponent>
 </MaxCrossSectionDiameter>
 <MinimumBendRadius xsi:type="n0:NumericalValue" uid="_xxx1.1">
  <Unit uidRef="_100301"/>
  <ValueComponent>1000</ValueComponent> <!-- for 1m -->
 </MinimumBendRadius>
 <ShapeElement xsi:type="n0:CrossSectionalPartShapeElement" uid="_101020_inner1">
  <OuterOrInnerBoundary>inner_boundary</OuterOrInnerBoundary>
  <MinCrossSectionDiameter xsi:type="n0:NumericalValue" uid="_xxx3">
   <Unit uidRef="_100301"/>
   <ValueComponent>5</ValueComponent>
  </MinCrossSectionDiameter>
 </ShapeElement>
 <ShapeElement xsi:type="n0:CrossSectionalPartShapeElement" uid="_101020_inner2">
  <OuterOrInnerBoundary>inner_boundary</OuterOrInnerBoundary>
  <MinCrossSectionDiameter xsi:type="n0:NumericalValue" uid="_xxx5">
   <Unit uidRef="_100301"/>
   <ValueComponent>5</ValueComponent>
  </MinCrossSectionDiameter>
 </ShapeElement>
</ShapeElement>
```

CrossSection of Wire, Cable & HarnessSegment (6 of 6)

This example shows the Hierarchical usage of `CrossSectionalGroupShapeElements`.

- A `CrossSectionalAlternativePartShapeElement` (_313001) defines that the `CrossSectionalOccurrenceShapeElements` (_201109 and _201209) are used alternatively for this group
- A `TwistedCrossSectionalGroupShapeElement` (_313002) contains a `CrossSectionalPartShapeElement`(_313001, see above) and another `CrossSectionalOccurrenceShapeElement` (_204007) and twist these clockwise with a period of 0.05m
- A `CrossSectionalGroupShapeElementWithLacing` is



```xml
<!--wire2 or wire3 (S2&S3)-->
<ShapeElement xsi:type="n0:CrossSectionalAlternativePartShapeElement" uid="_313001">
 <OuterOrInnerBoundary>outer_boundary</OuterOrInnerBoundary>
 <Items>
  <CrossSectionalOccurrenceShapeElement uidRef="_201109"/> <!--wire2/outer-->
  <CrossSectionalOccurrenceShapeElement uidRef="_201209"/> <!--wire3/outer-->
 </Items>
</ShapeElement>
<!--twist (wire2 or wire3) with cable3 (S2&S3)-->
<ShapeElement xsi:type="n0:TwistedCrossSectionalGroupShapeElement" uid="_313002">
 <OuterOrInnerBoundary>outer_boundary</OuterOrInnerBoundary>
 <Items>
  <CrossSectionalPartShapeElement uidRef="_313001"/> <!--wire2 or wire3 -->
  <CrossSectionalOccurrenceShapeElement uidRef="_204007"/> <!--cable3/outer -->
 </Items>
 <TwistDirection>clockwise</TwistDirection>
 <TwistPeriod xsi:type="n0:NumericalValue" uid="_313003">
  <Definition>
   <PropertyDefinitionString>Non sense</PropertyDefinitionString>
  </Definition>
  <Unit uidRef="_100301"/>
  <ValueComponent>0.05</ValueComponent>
 </TwistPeriod>
</ShapeElement>
<!--Lacing of cable1, cable2 & wire3 (S5)-->
<ShapeElement xsi:type="n0:CrossSectionalGroupShapeElementWithLacing" uid="_313008">
 <OuterOrInnerBoundary>outer_boundary</OuterOrInnerBoundary>
 <Items>
  <CrossSectionalOccurrenceShapeElement uidRef="_202007"/><!--cable1/outer-->
  <CrossSectionalOccurrenceShapeElement uidRef="_202107"/><!--cable2/outer-->
 </Items>
</ShapeElement>
<!—Tubular cover -->
<ShapeElement xsi:type="n0:CrossSectionalGroupShapeElementWithTubularCover" uid="_313007">
 <OuterOrInnerBoundary>outer_boundary</OuterOrInnerBoundary>
 <Items>
  <CrossSectionalPartShapeElement uidRef="_ 313002"/> <!– twisting result -->
  <CrossSectionalPartShapeElement uidRef="_ 313008"/> <!– lacing result -->
 </Items>
 <Cover uidRef="_222006"/><!--heatshrink1/inner-->
</ShapeElement>
```

Table of Content

- Introduction
- Basic XML structure
- Part, PartCategory & PartVersion
- PartView & ViewContext
- ProductClass & ProductConfiguration
- Occurrence, Assembly & WiringHarnessAssemblyDesign
- ShapeElement
- Geometric & topological models
- Connectors & terminals
- Features, Definitions & AssemblyShapeJoints
- Wire, Cables & TransportFeature
- Shapes/ContactFeatureDefinition & -Elements
- CrossSection of Wire, Cable & HarnessSegment
- Geometry/Topology Associations
- Complex Connectors / Assembly Hierarchy
- External Element References
- ComposedGeometricModel

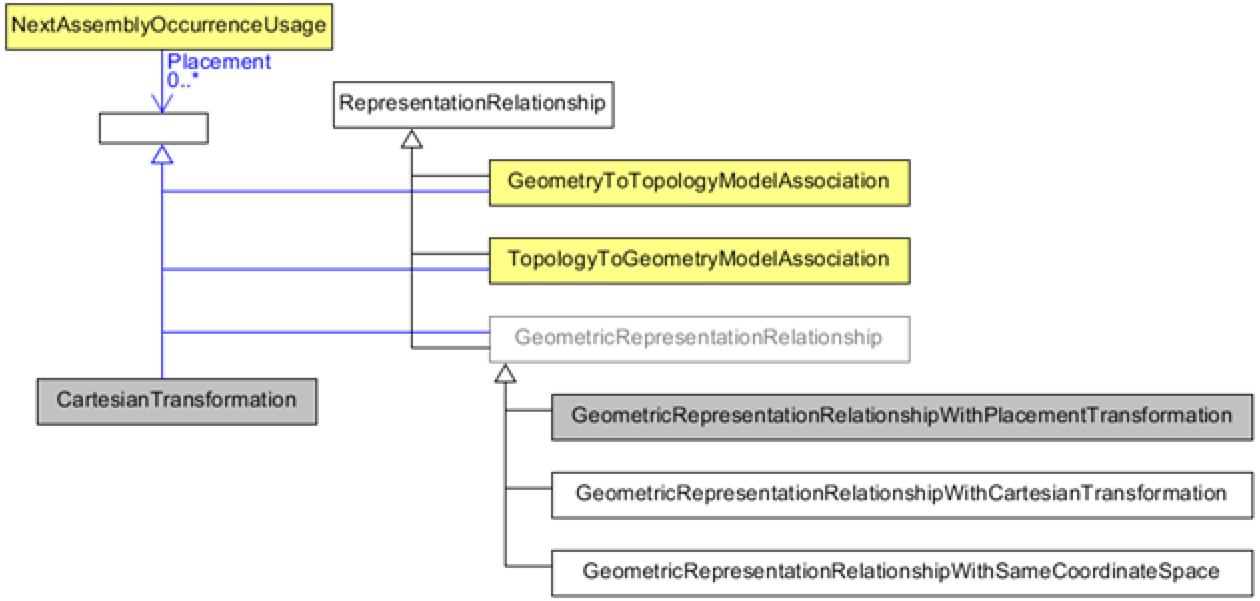
Geometry/Topology Associations (1 of 7)

- The next 5 slides present the associations of geometric/topological models between:
  - piece parts and raw material, given by 2D/3D geometry models
  - electrical wire harness (EWH) assembly, given by topological representations
  - installation model of the EWH, given by either 2D model (e.g. formboard) or 3D model (as installed)

Geometry/Topology Associations (2 of 7)

Use of the Geometry/Topology Associations for assembly cases

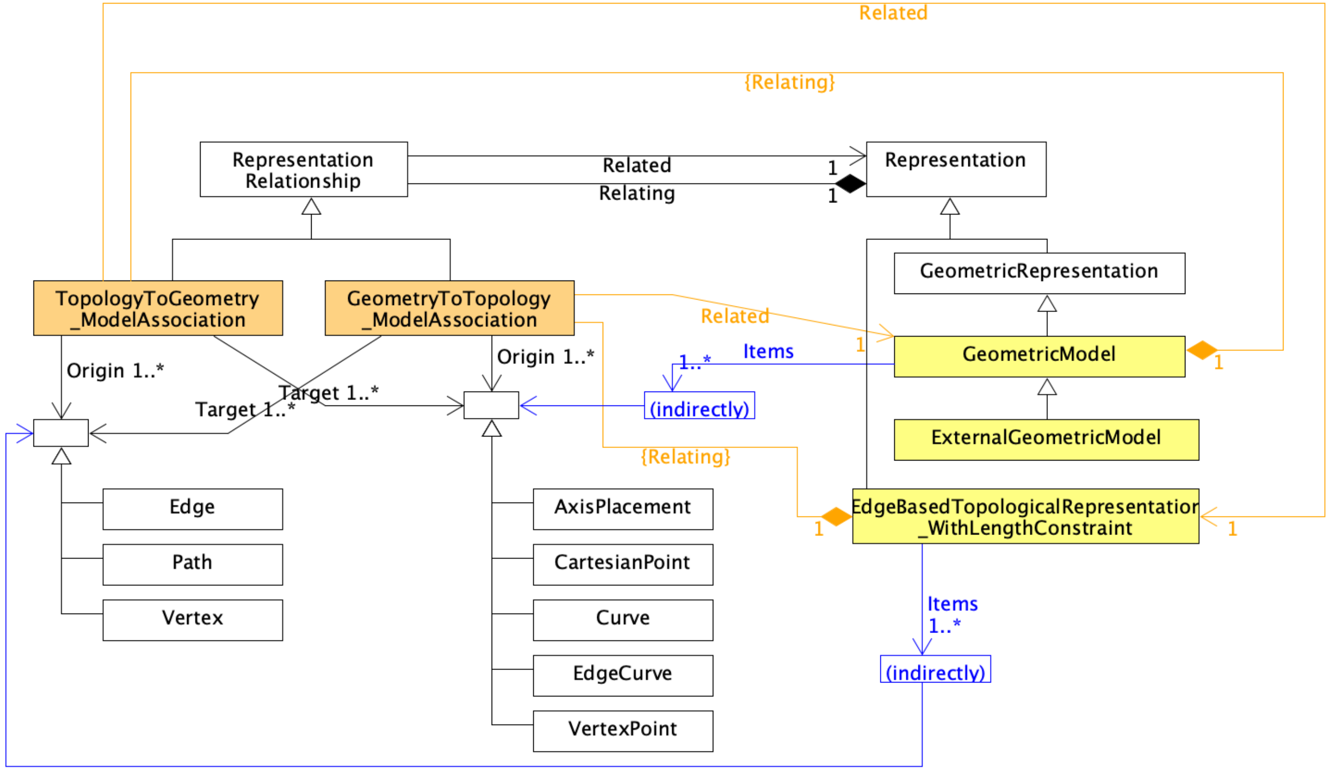
- NextAssemblyOccurrenceUsage is a subtype of AssemblyOccurrenceRelationship, that is essential to define any assembly
- Its attribute Placement can refer to several kinds of RepresentationRelationship or in simple cases directly to CartesianTransformation
- For the purpose of the CAx-IF most often CartesianTransformation or GeometricRepresentationRelationshipWithPlacementTransformation is used/recommended
- For the purpose of EWH the relationship GeometryToTopologyModelAssociation and TopologyToGeometryModelAssociation are need additionally
  - GeometryToTopologyModelAssociation relates the 2D/3D models of assembly components to the EWH topology model
  - TopologyToGeometryModelAssociation relates the topological model of a WiringHarnessAssemblyDesign to a GeometricModel
    - This geometric model might e.g. represent a formboard (2D), or a 3D model as-designed.
    - This geometric model might be an additional model for the wiring harness assembly design or the geometric model of a higher-level assembly (e.g. of the whole vehicle)


NextAssemblyOccurrenceUsage
Placement
0..*
RepresentationRelationship
GeometryToTopologyModelAssociation
TopologyToGeometryModelAssociation
GeometricRepresentationRelationship
CartesianTransformation
GeometricRepresentationRelationshipWithPlacementTransformation
GeometricRepresentationRelationshipWithCartesianTransformation
GeometricRepresentationRelationshipWithSameCoordinateSpace

Geometry/Topology Associations (3 of 7)

Details of GeometryToTopologyModelAssociation & TopologyToGeometryModelAssociation

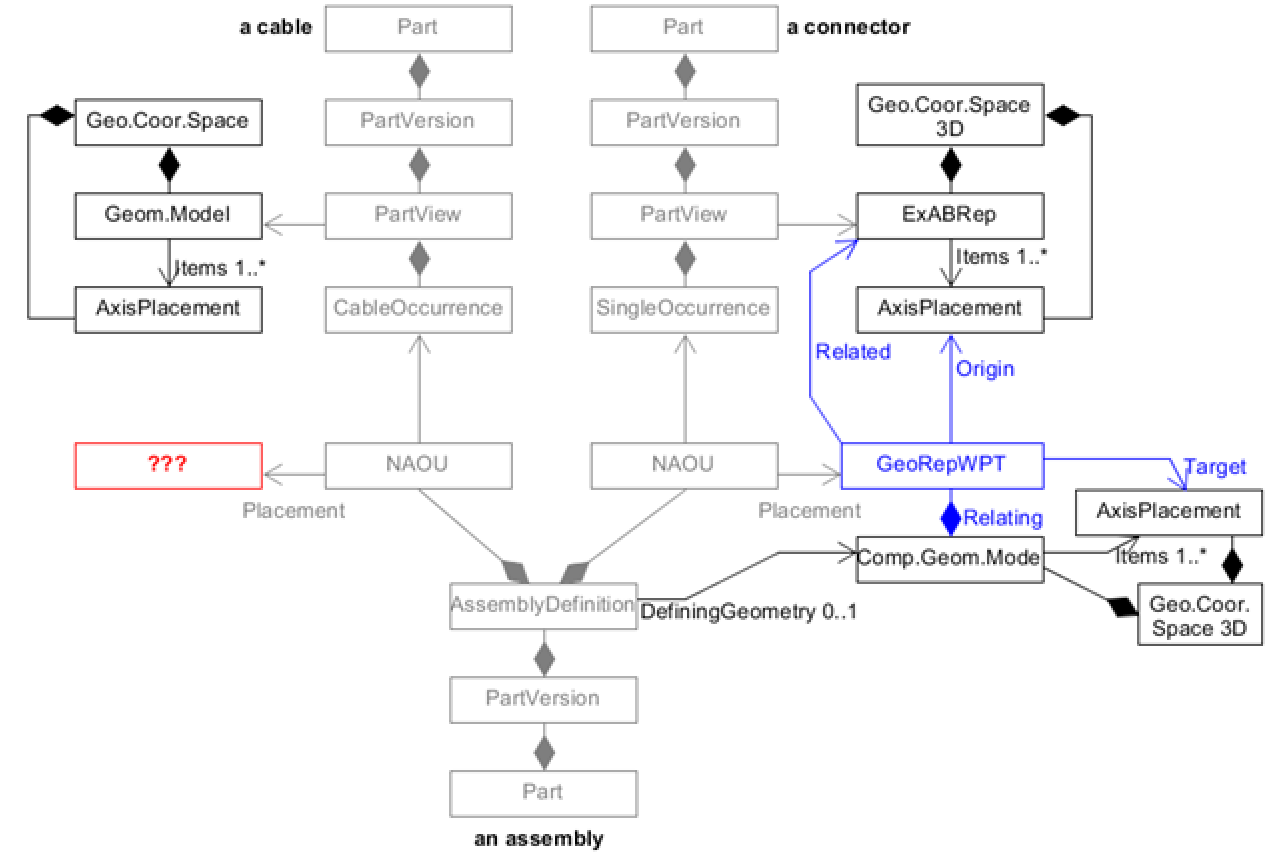
- The GeometryToTopologyModelAssociation maps from an (External)GeometricModel to an EdgeBasedTopologicalRepresentationWithLengthConstraint
- The TopologyToGeometryModelAssociation maps from an EdgeBasedTopologicalRepresentationWithLengthConstraint to an (External)GeometricModel
- The (External)GeometricModels might be 2D or 3D
- The GeometryToTopologyModelAssociation is typically a single item association
  Note: only in special cases of cables and wires that don't have a round cross-section, two items might be needed in this association
- The TopologyToGeometryModelAssociation is typically a multi-item association;
  Note: the list of Origin elements must match to the list of Target elements



Geometry/Topology Associations (4 of 7)

Problem on how to handle geometry of flexible parts

- The instance diagram shows the structure for an assembly consisting of a SingleOccurrence of a connector and a CableOccurrence of a cable
- The geometric model of the connector is placed into the geometric model of the assembly using a GeometricRepresentationRelationshipWithPlacementTransformation
- What to do with the geometric model of the cable?
  The cable is typically flexible while the connector is rigid. So, the flexible cable does not have a geometric model that could be directly placed into the geometric model of an assembly.
- But the cable might have a 2D geometric cross section model that is either round or either a specific shape (e.g. a speaker cable with a cross section like an "8")
- An assembly that is an WiringHarnessAssemblyDesign (WHAD)
  has a topology attribute that refers to EdgeBasedTopologicalRepresentationWithLenghtConstraint (EBTRepWLC) that is able to overcome the indicated problem (???). See next slides.

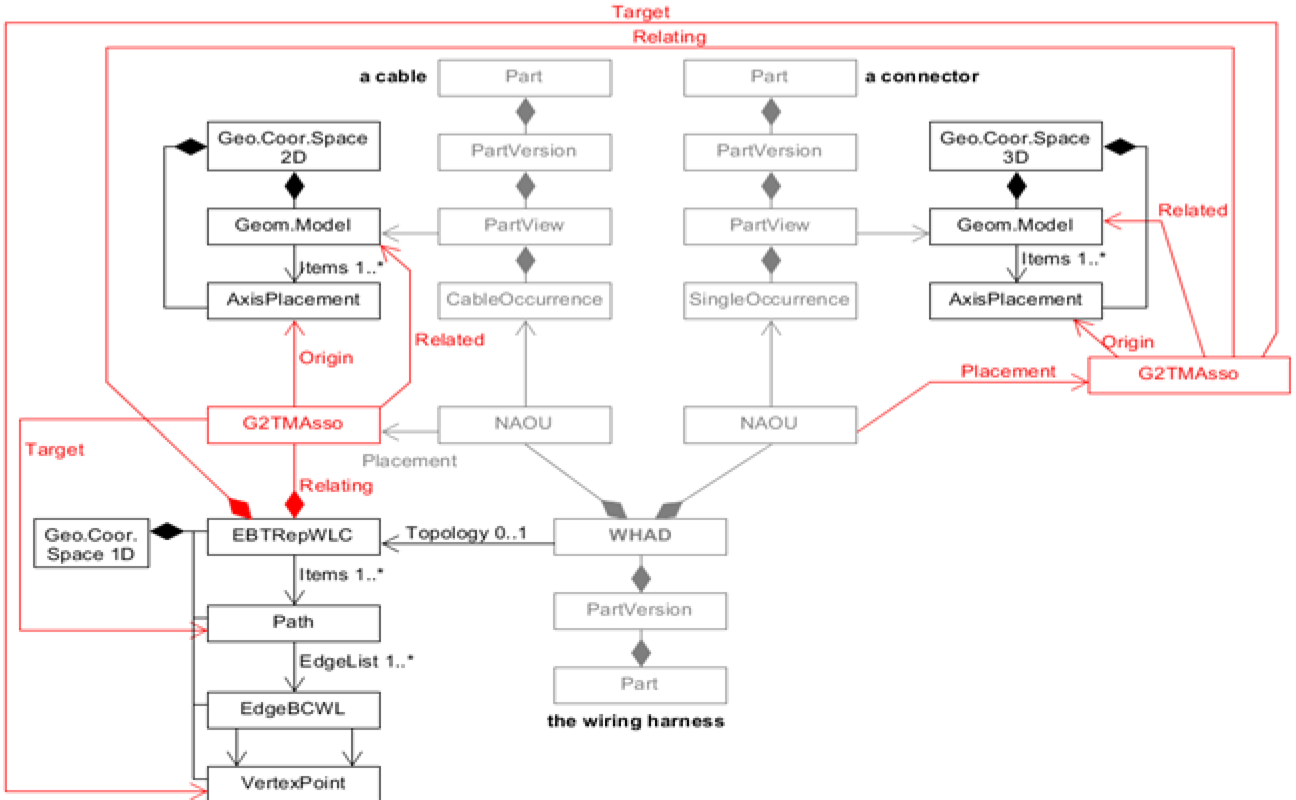


NAOU = NextAssemblyOccurrenceUsage

Geometry/Topology Associations (5 of 7)

Associations of geometric Models to the topological Harness Model

- In red colour we can see two instances of GeometryToTopologyModelAssociation (G2TMAsso)
- Both G2TMAsso relate the GeometricModel (Related) of the parts to the EBTRepWLC (Relating) of the WHAD
- For rigid occurrences (on the right), a single AxisPlacement of the GeometricModel (of the whole part) is used as Origin and associated to a Target VertexPoint of the EBTRepWLC
- For flexible occurrences (on the left), a single AxisPlacement of the GeometricModel (2D cross-section model) is used as Origin and associated either to a Path or an EdgeBoundedCurveWithLenght (EdgeBCWL).
  The interpretation is that this 2D cross section model is constant and swept along the whole Path/Edge.
  Note: the orientation of the cross-section might be controlled by a second AxisPlacement for Origin and a second Path/Edge for the Target



WHAD = WiringHarnessAssemblyDesign
NAOU = NextAssemblyOccurrenceUsage
G2TMAsso = GeometryToTopologyModelAssociation

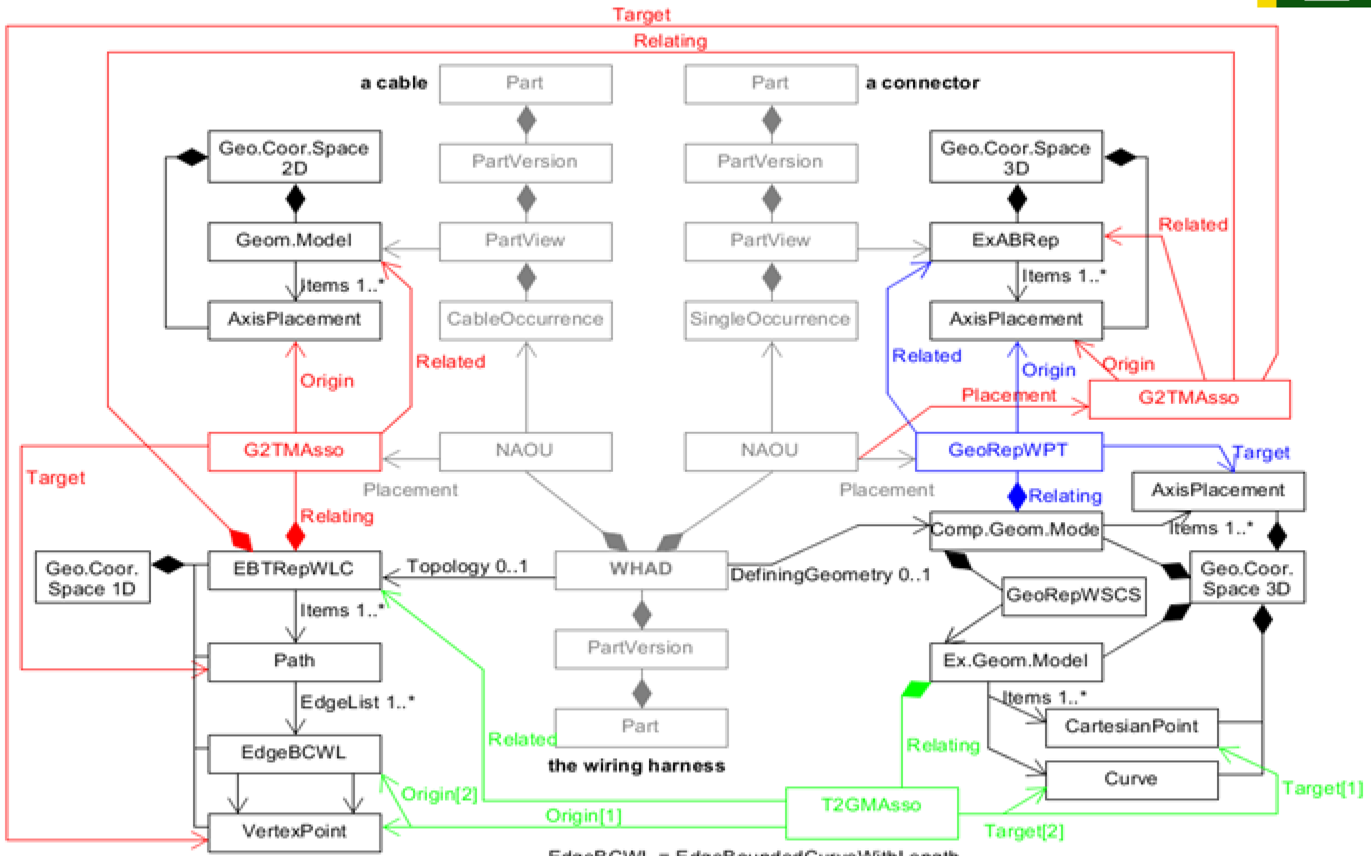
EdgeBCWL = EdgeBoundedCurveWithLength
EBTRepWLC = EdgeBasedTopologicalRepresentationWithLengthConstraint

Geometry/Topology Associations (6 of 7)

A Wiring Harness with all Transformations and Associations

In green colour:

- TopologyToGeometryModelAssociation (T2GMAsso) associates an EBTRepWLC (Related) of a WHAD to a GeometricModel (Relating) of a WHAD (of a higher-level assembly where WHAD is used as a component)
- There are several Origins and several Targets.
  - Origin[1] associates to Target[1], here VertexPoint is associated to CartesianPoint
  - Origin[2] associates to Target[2], here EdgeBCWL is associated to Curve
  - Etc, …

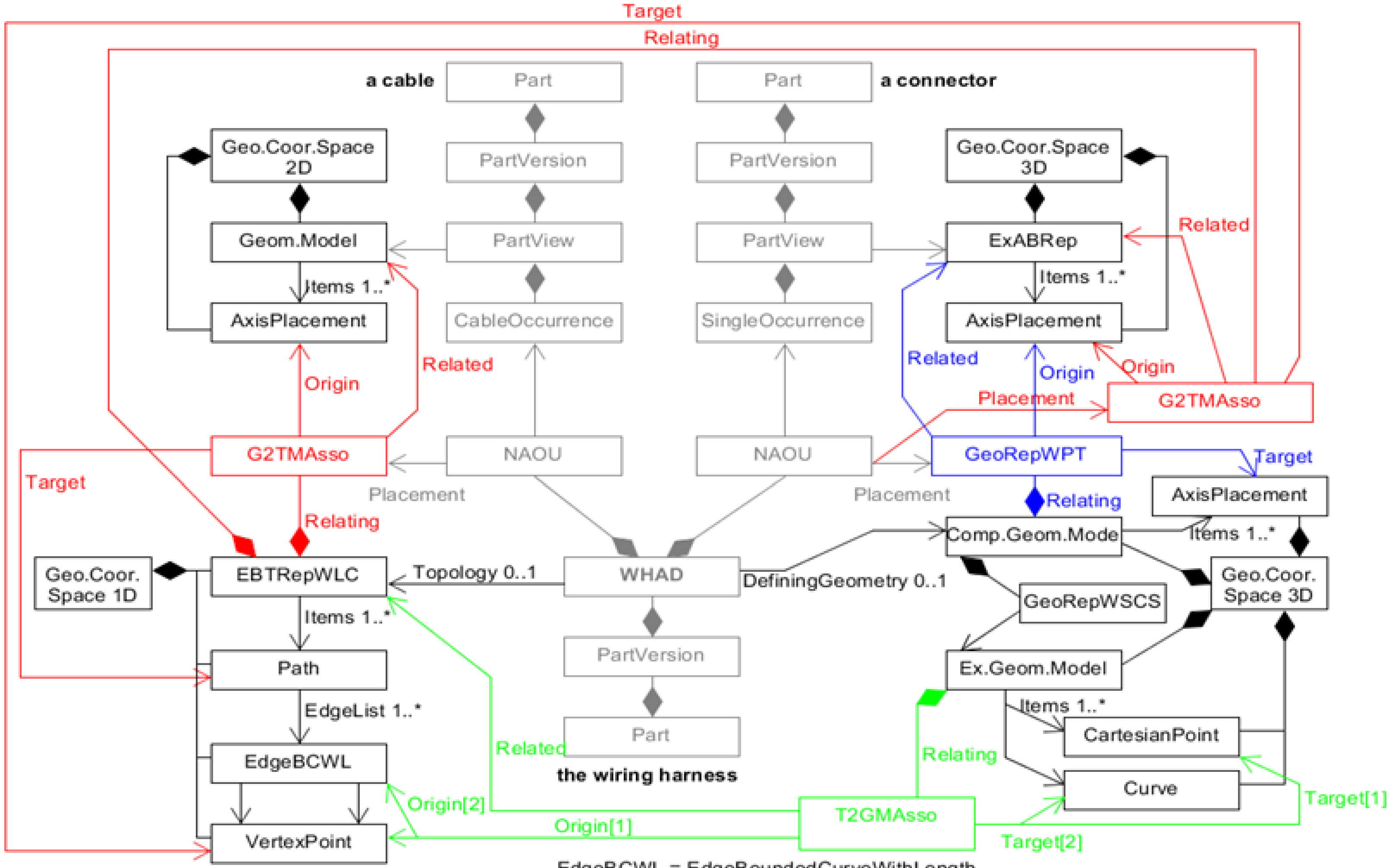


WHAD = WiringHarnessAssemblyDesign
NAOU = NextAssemblyOccurrenceUsage
G2TMAsso = GeometryToTopologyModelAssociation
T2GMAsso = TopologyToGeometryModelAssociation

EdgeBCWL = EdgeBoundedCurveWithLength
ExABRep = ExternalAdvancedBrepShapeRepresentation
EBTRepWLC = EdgeBasedTopologicalRepresentationWithLengthConstraint
GeoRepWPT = GeometricRepresentationRelationshipWithPlacementTransformation
GeoRepWSCS = GeometricRepresentationRelationshipWithSameCoordinateSpace

Geometry/Topology Associations (7 of 7)

A Wiring Harness with all Transformations and Associations



WHAD = WiringHarnessAssemblyDesign
NAOU = NextAssemblyOccurrenceUsage
G2TMAsso = GeometryToTopologyModelAssociation
T2GMAsso = TopologyToGeometryModelAssociation

EdgeBCWL = EdgeBoundedCurveWithLength
ExABRep = ExternalAdvancedBrepShapeRepresentation
EBTRepWLC = EdgeBasedTopologicalRepresentationWithLengthConstraint
GeoRepWPT = GeometricRepresentationRelationshipWithPlacementTransformation
GeoRepWSCS = GeometricRepresentationRelationshipWithSameCoordinateSpace

Table of Content

- Introduction
- Basic XML structure
- Part, PartCategory & PartVersion
- PartView & ViewContext
- ProductClass & ProductConfiguration
- Occurrence, Assembly & WiringHarnessAssemblyDesign
- ShapeElement
- Geometric & topological models
- Connectors & terminals
- Features, Definitions & AssemblyShapeJoints
- Wire, Cables & TransportFeature
- Shapes/ContactFeatureDefinition & -Elements
- CrossSection of Wire, Cable & HarnessSegment
- Geometry/Topology Associations
- Complex Connectors / Assembly Hierarchy
- External Element References
- ComposedGeometricModel

Complex Connectors / Assembly Hierarchy (1 of 6)

- Problem:
  - complex connectors may consist of many components such as a housing, a backshell, many contacts, inserts, strain relief, seals and more
  - often these components are available as a connector kit and the harness manufacturer has to assemble them
  - some CAx systems may represent an assembled connector as a part by its own. But physically these connector assemblies do not exist; they make no sense!
  - the connector components are only assembled together during the assembly process of the whole EWH (not before!). E.g. a contact is first crimped together with a wire before inserting it into a cavity of the connector housing or insert.
  - During the assembly process AssemblyShapeJoints are used in two different ways:
    - electrical joints between the contact terminals and the wire/cable terminals.
      Knowledge of these terminals is essential for the use of EWH
    - mechanical joints between mating features.
      e.g. between the outer feature of a connector contact and
      the inner feature of a cavity into which the connector contact is inserted
  - mechanical contact features are essential for the full description of an EWH-assembly, but CAx-systems may not have this information
  - However mechanical CAD systems used to provide assembly information with transformation. Together with the 3D model a CAM system may calculate the mechanical contact zones between the assembly components (not numerical stable, error prone)

Complex Connectors / Assembly Hierarchy (2 of 6)

- AP242 provides the following alternative solutions so that the needed AssemblyShapeJoints can refer to the right occurrence features / terminals:
- 1) Hierarchical assembly with SpecifiedOccurrences
  - use of dummy sub-assembly parts
  - occurrences of lower level assemblies are reflected to higher assembly levels by SpecifiedOccurrence
  - terminals of lower level occurrences are reflected as terminals of SpecifiedOccurrences
  - AssemblyShapeJoints are joining the terminals of SpecifiedOccurrences with others
  - mechanical features can be skipped; only terminals are essential
- 2) Hierarchical assembly with reflecting lower level terminals to a higher level (New)
  - use of dummy sub-assembly parts
  - no need for SpecifiedOccurrence
  - terminals of lower level occurrences are reflected as terminals of the next higher assembly part (can be used recursively)
  - AssemblyShapeJoints are joining the terminals of sub-assembly occurrences with others
  - mechanical features can be skipped; only terminals are essential
- 3) Flat assembly
  - no need for dummy sub-assembly parts
  - no need for SpecifiedOccurrence
  - mechanical features are essential to know which contact is in which cavity or which insert is in which slot
  - AssemblyShapeJoints are joining the terminal and features of the occurrences
- Flat assembly is most close to the reality and would be the basis for a later Process Plan extension

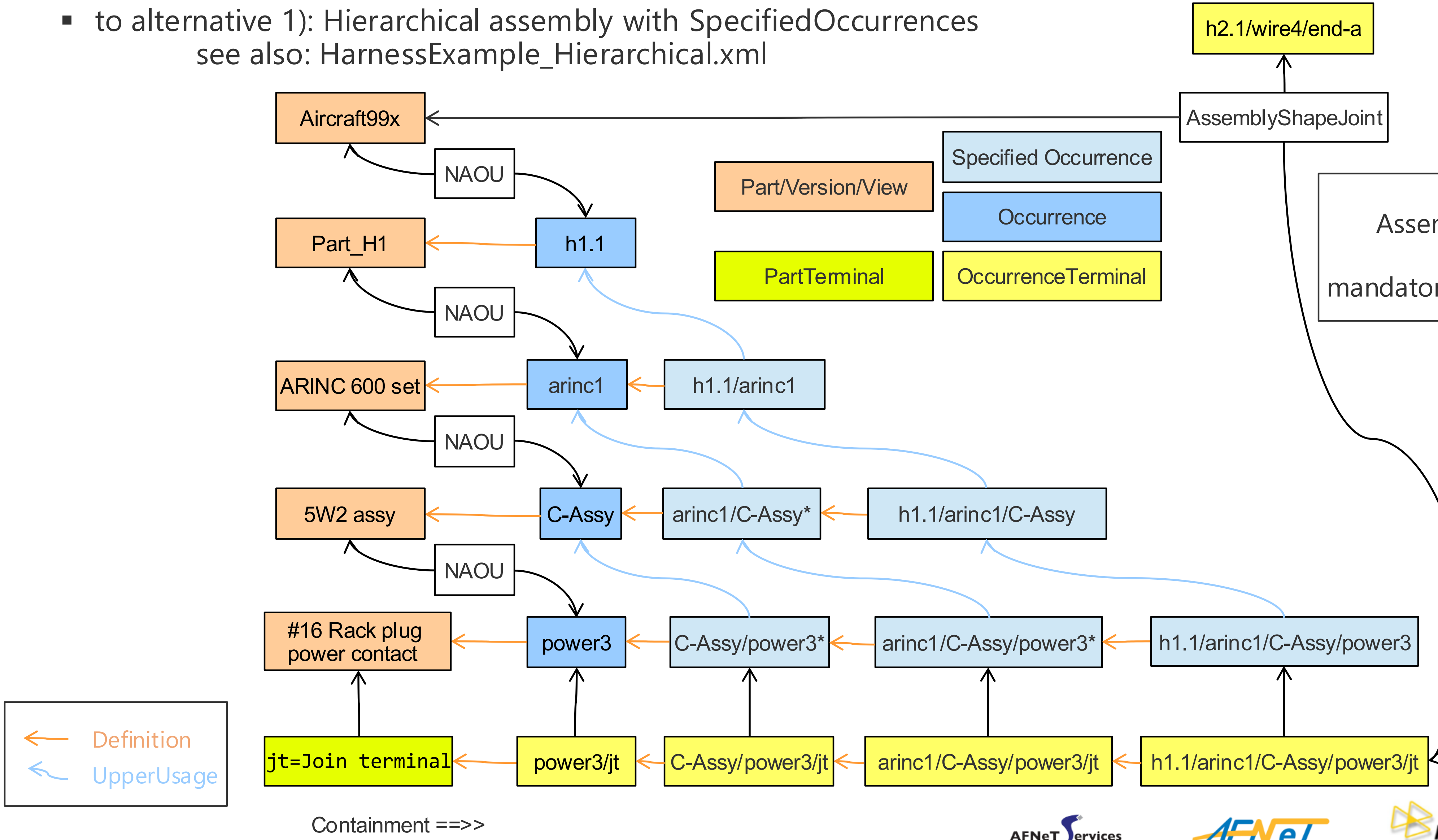
Complex Connectors / Assembly Hierarchy (3 of 6)

1. TestCase: EWH-Connectivity1
   A very simple flat assembly, here the connector contain directly the electrical terminals
2. TestCase: EWH-Connectivity2
   1. A more realistic assembly where the connector has cavities into which ConnectorContacts are to be inserted (mechanical AssemblyShapeJoint).
      But before doing so (process information), the ConnectorContacts have first to be crimped or solded to the wires/cables.
      For tis test case a FLAT ASSEMBLY structure is used, so ConnectorHousing and ConnectorContacts are directly assembled together on the EWH assembly.
      This solution is most exact and represents what is really done !!!
      Problem: typical CAD systems don't have this mechanical information
   2. Hierachical Assembly with Specified Occurrence
      Problem: Ghost assembly of parts that we can't buy, a lot of dummy data to generate (we loose business view)
   3. Hierachical Assembly with Reflection (with avoids Specified Occurrence)
      Problem: still ghost assembly parts needed, but less data to generate (and we still loose business view)
   4. Have the ability of AssemblyGroupComponent (proposal for ED 5 ...)
      Most appropriate as we avoid ghost assemblies
   5. There is a bigger business discrepance of designing an EWH and manufacturing it
      For the DESIGN of of an EWH and how it fits into the higher level system (aircract, vehicle, ...)
      a breakdown structure as defined in ISO/IEC 81346 is more appropriate
      For the manufacturing of an EWH a real detailed ASSEMBLY STRUCTURE contain all needed mechanical part are needed
      BOM (Bill of Material) needs to be complete for all things needed to buy before manufacturing can start

Possible Solutions to avoid the mechanical AssemblyShapeJoint that E-CAD systems typically don't provide:

Complex Connectors / Assembly Hierarchy (4 of 6)

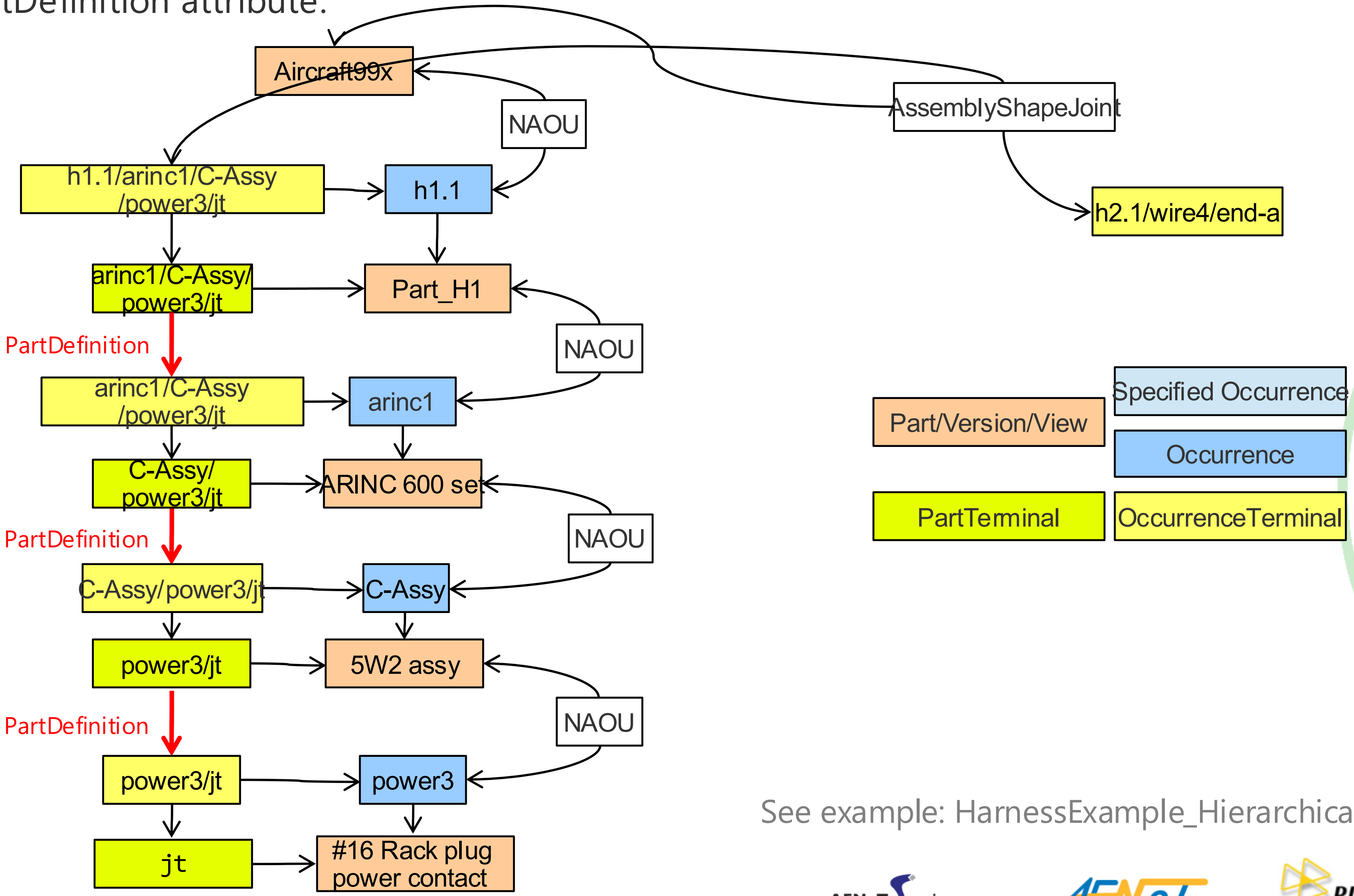
- to alternative 1): Hierarchical assembly with SpecifiedOccurrences
  see also: HarnessExample_Hierarchical.xml



For this particular AssemblyShapeJoint, the three SpecifiedOccurences are mandatory and three are optional*

Complex Connectors / Assembly Hierarchy (5 of 6)

- This alternative 2) is avoiding the use of SpecifiedOccurence.
- This is done by **reflecting** lower-level OccurenceTerminals to PartTerminals of an AssemblyDefinition. This is done by using the PartDefinition attribute.



See example: HarnessExample_HierarchicalReflect.stpx

Complex Connectors / Assembly Hierarchy (6 of 6)

- to alternative 3): Flat assembly
  see also: HarnessExample_Flat.xml

Table of Content

- Introduction
- Basic XML structure
- Part, PartCategory & PartVersion
- PartView & ViewContext
- ProductClass & ProductConfiguration
- Occurrence, Assembly & WiringHarnessAssemblyDesign
- ShapeElement
- Geometric & topological models
- Connectors & terminals
- Features, Definitions & AssemblyShapeJoints
- Wire, Cables & TransportFeature
- Shapes/ContactFeatureDefinition & -Elements
- CrossSection of Wire, Cable & HarnessSegment
- Geometry/Topology Associations
- Complex Connectors / Assembly Hierarchy
- External Models and Element References
- ComposedGeometricModel

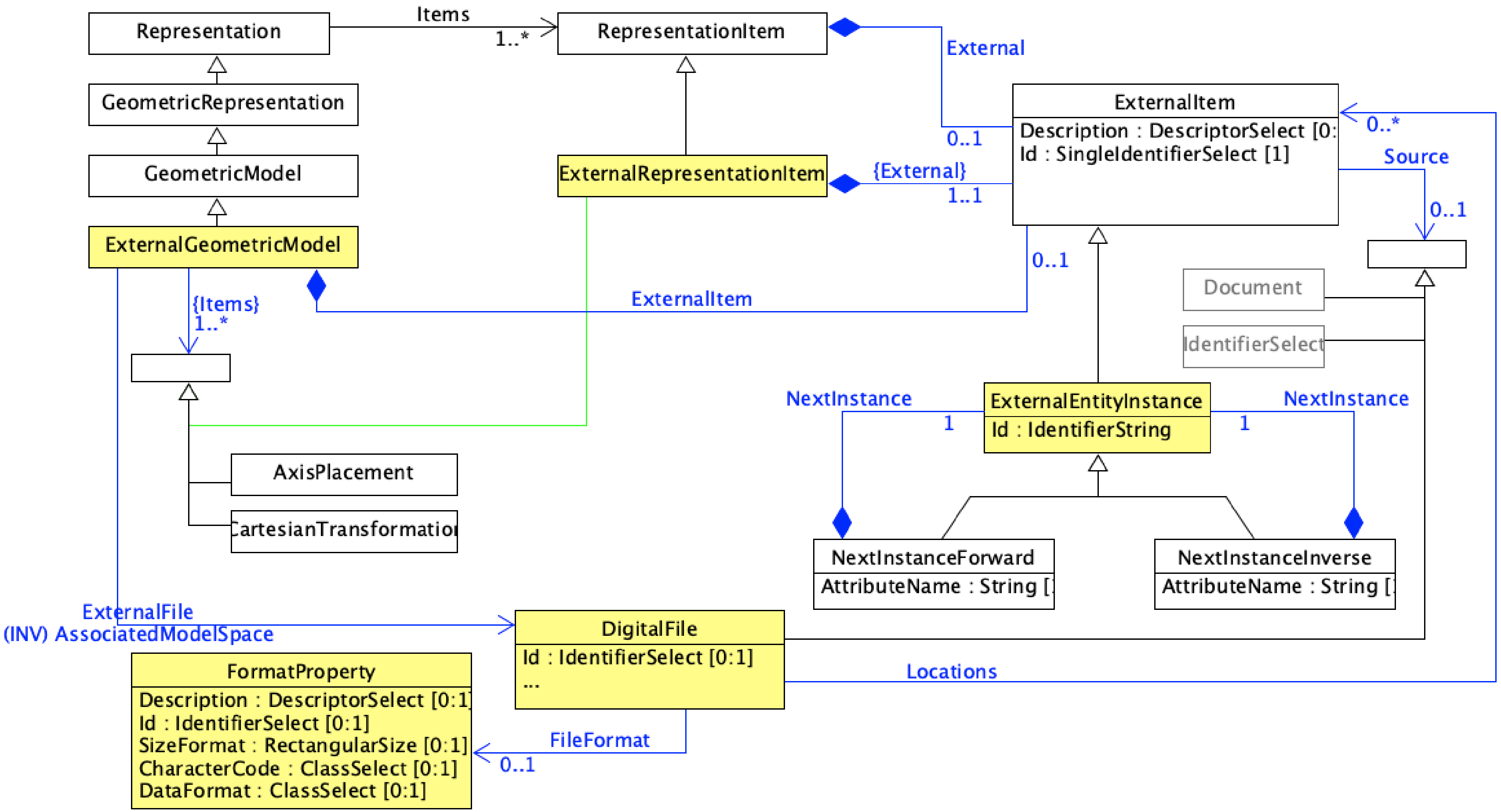
External Models and Element References (1 of 5)

ExternalGeometricModel & External(Element)References

- use ExternalGeometricModel.ExternalItem if there are several Models/Representations in the external files
- use ExternalGeometricModel.Items to refer ExternalRepresentationItems
- for DigitalFile use FileLocations attribute (ed2); don’t use Locations attribute (ed1)
- for external p21 files use ExternalEntityInstance to make clear what is the meaning of the Id
- if available use for the ExternalEntityInstance.Id the external anchor name instead of the instance ID (e.g. #1234)
- NextInstanceForward and NextInstanceInverse allow to follow a path of instance in a p21 file (for later tests)

External Models and Element References (2 of 5)

ExternalGeometricModel & External(Element)References



Representation

Items

1..*

RepresentationItem

External

GeometricRepresentation

GeometricModel

ExternalGeometricModel

ExternalRepresentationItem

{External}

1..1

0..1

ExternalItem

Description : DescriptorSelect [0:

Id : SingleIdentifierSelect [1]

0..*

Source

0..1

0..1

Document

IdentifierSelect

{Items}

1..*

ExternalItem

AxisPlacement

CartesianTransformatio

NextInstance

1

ExternalEntityInstance

Id : IdentifierString

NextInstance

1

NextInstanceForward

AttributeName : String [

NextInstanceInverse

AttributeName : String [

ExternalFile

(INV) AssociatedModelSpace

DigitalFile

Id : IdentifierSelect [0:1]

...

Locations

FormatProperty

Description : DescriptorSelect [0:1]

Id : IdentifierSelect [0:1]

SizeFormat : RectangularSize [0:1]

CharacterCode : ClassSelect [0:1]

DataFormat : ClassSelect [0:1]

FileFormat

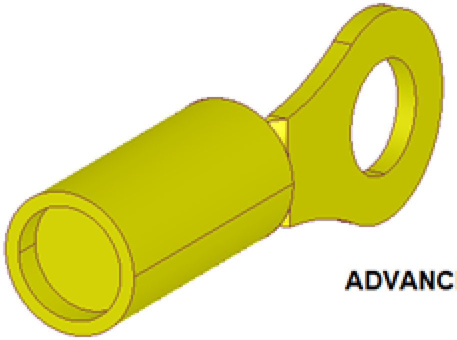
0..1

External Models and Element References (3 of 5)

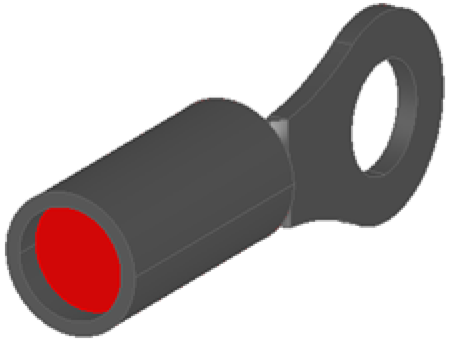
Example: Terminal Lug with external Geometry (1/3)

- PartTerminals are associated to AdvancedFaces of the GeometricModel (ABRep) of a Part

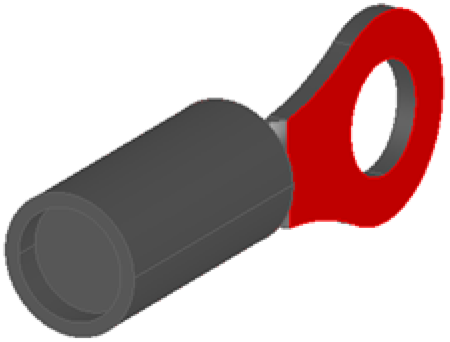
File: c-51864-1-af-3d.stp



ADVANCED_BREP_SHAPE_REPRESENTATION #1023



Advanced_face #940

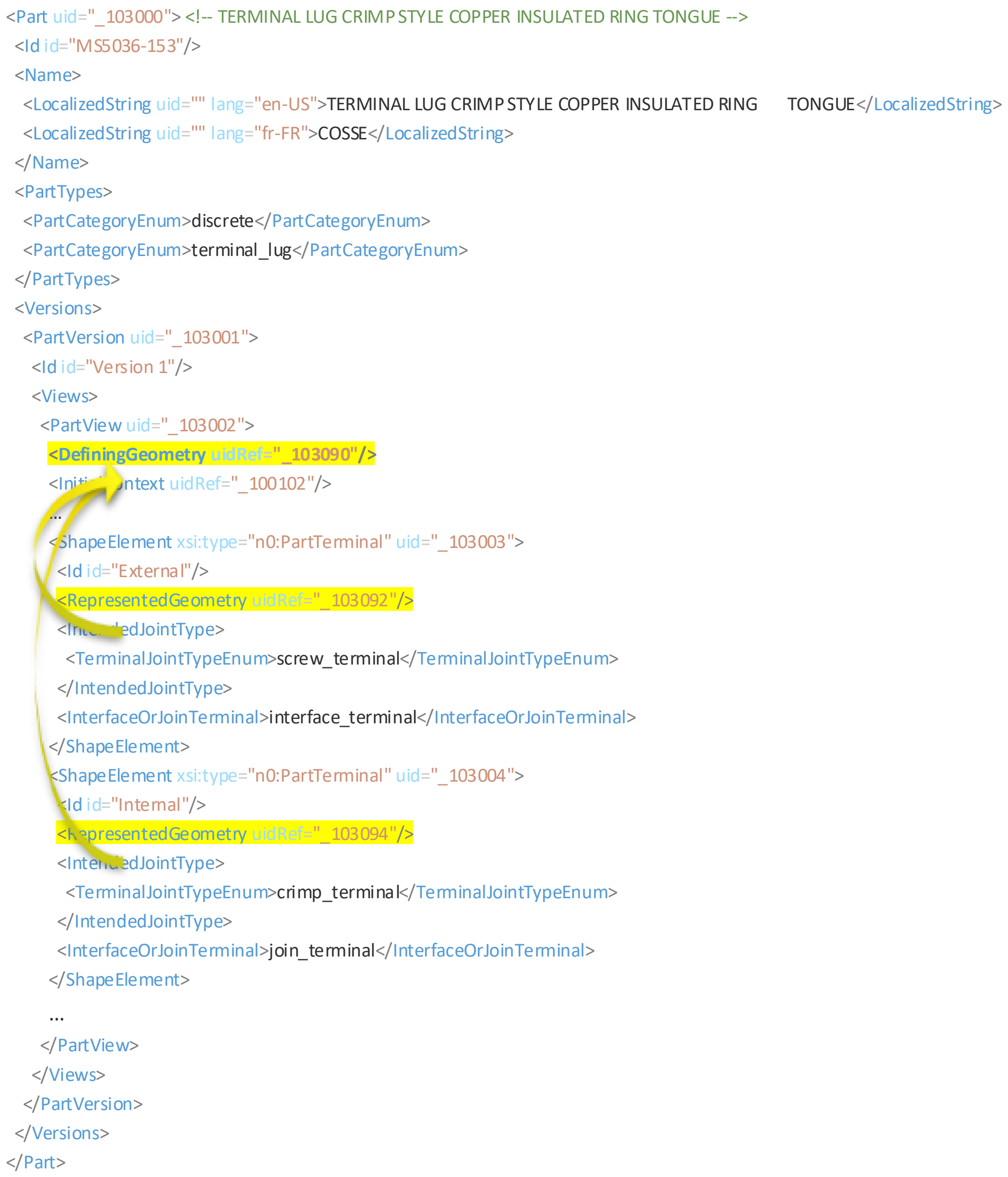


Advanced_face #521

External Models and Element References (4 of 5)

Example: Terminal Lug with external Geometry (2/3)

- A part with an external geometric model and 2 features/terminals
- The represented geometry of the *PartTerminals* are within the *DefiningGeometry* of the Part



```
<Part uid="_103000"> <!-- TERMINAL LUG CRIMP STYLE COPPER INSULATED RING TONGUE -->
 <Id id="MS5036-153"/>
 <Name>
  <LocalizedString uid="" lang="en-US">TERMINAL LUG CRIMP STYLE COPPER INSULATED RING     TONGUE</LocalizedString>
  <LocalizedString uid="" lang="fr-FR">COSSE</LocalizedString>
 </Name>
 <PartTypes>
  <PartCategoryEnum>discrete</PartCategoryEnum>
  <PartCategoryEnum>terminal_lug</PartCategoryEnum>
 </PartTypes>
 <Versions>
  <PartVersion uid="_103001">
   <Id id="Version 1"/>
   <Views>
    <PartView uid="_103002">
     <DefiningGeometry uidRef="_103090"/>
     <InitialContext uidRef="_100102"/>
     ...
     <ShapeElement xsi:type="n0:PartTerminal" uid="_103003">
      <Id id="External"/>
      <RepresentedGeometry uidRef="_103092"/>
      <IntendedJointType>
       <TerminalJointTypeEnum>screw_terminal</TerminalJointTypeEnum>
      </IntendedJointType>
      <InterfaceOrJoinTerminal>interface_terminal</InterfaceOrJoinTerminal>
     </ShapeElement>
     <ShapeElement xsi:type="n0:PartTerminal" uid="_103004">
      <Id id="Internal"/>
      <RepresentedGeometry uidRef="_103094"/>
      <IntendedJointType>
       <TerminalJointTypeEnum>crimp_terminal</TerminalJointTypeEnum>
      </IntendedJointType>
      <InterfaceOrJoinTerminal>join_terminal</InterfaceOrJoinTerminal>
     </ShapeElement>

     ...
    </PartView>
   </Views>
  </PartVersion>
 </Versions>
</Part>
```

External Models and Element References (5 of 5)

Example: Terminal Lug with external Geometry (3/3)

An ExternalGeometricModel (here subtype for an ABREP) that references into items of a p21 file.

```xml
<!--Geometry for terminal lug-->
<RepresentationContext xsi:type="n0:GeometricCoordinateSpace" uid="_103091">
 <Id id="/NULL"/>
 <Representations>
  <Representation xsi:type="n0:ExternalAdvancedBrepShapeRepresentation" uid="_103090">
   <Id id="c-51864-1-af-3d.stp"/>
   <Items>
    <RepresentationItem uidRef="_103092"/>
    <RepresentationItem uidRef="_103094"/>
    <RepresentationItem uidRef="_103096"/>
   </Items>
   <ExternalFile uidRef="_103080"/>
   <ExternalItem xsi:type="n0:ExternalEntityInstance" uid="_103097">
    <Id id="#1023"/>
   </ExternalItem>
  </Representation>
 </Representations>
 <Items>
  <RepresentationItem xsi:type="n0:ExternalRepresentationItem" uid="_103092">
   <External xsi:type="n0:ExternalEntityInstance" uid="_103093">
    <Id id="#521"/>
   </External>
  </RepresentationItem>
  <RepresentationItem xsi:type="n0:ExternalRepresentationItem" uid="_103094">
   <External xsi:type="n0:ExternalEntityInstance" uid="_103095">
    <Id id="#940"/>
   </External>
  </RepresentationItem>
  <RepresentationItem xsi:type="n0:AxisPlacement" uid="_103096">
   <Position>0.0 0.0 0.0</Position>
  </RepresentationItem>
 </Items>
 <DimensionCount>3</DimensionCount>
</RepresentationContext>
```

```xml
<!— terminal lug geometric representation ->
<File xsi:type="n0:DigitalFile" uid="_103080">
 <FileFormat uidRef="_100300"/>
 <FileLocations>
  <FileLocationIdentification uid="_103081">
   <SourceId>c-51864-1-af-3d.stp</SourceId>
   <SourceType>file</SourceType>
  </FileLocationIdentification>
 </FileLocations>
</File>
```

Example of a p21 ed3 file with Anchors

- Use of anchors in a p21 file requires implementation level 4
  see FILE_DESCRIPTION below
- Recommendations:
  - use anchors only for entity instances
  - continue to use "syntactical conformance class" 1 for "internal mapping"
  - for the anchor names use the name given in the source system
    (in CATIA v5 called “publication, in NX called “port”)
    Ideally a source system would use UUIDs to achieve global unique and persistent anchor names
  - centre line curves for harness segments best contained in a GEOMETRICALLY_BOUNDED_WIREFRAME_SHAPE_REPRESENTATION

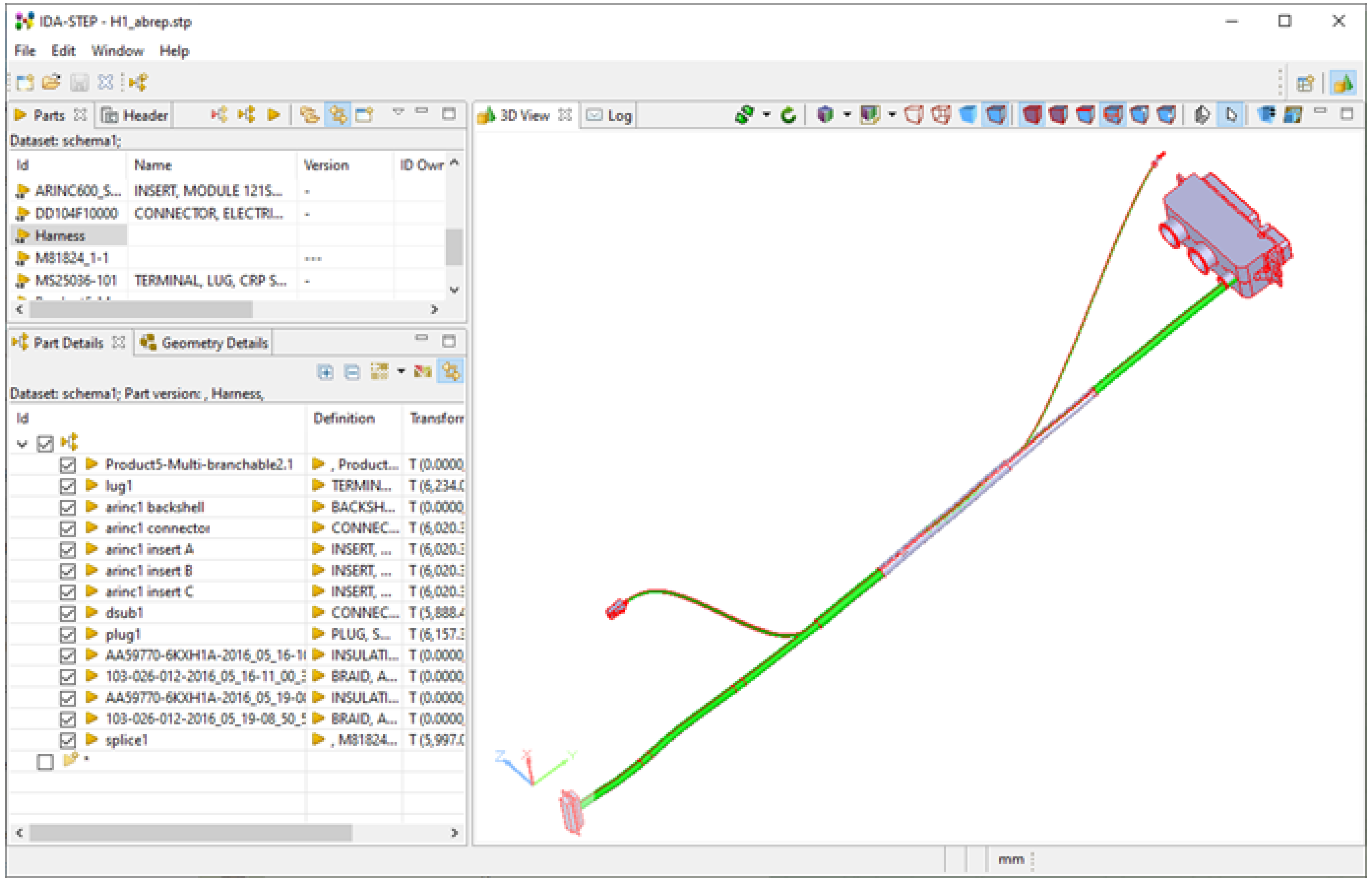
```
ISO-10303-21;
HEADER;
FILE_DESCRIPTION(...,'4;1');
FILE_NAME('star1.p21', ... );
FILE_SCHEMA(('CONFIG_CONTROL_DESIGN'));
ENDSEC;
ANCHOR;
<placement1> = #1011;
<placement2> = #1012;
<curve1> = #1021;
<2871d0c8-9f87-4349-aab0-7832e53fa25a> = #1022;        <== example of a UUID
ENDSEC;

DATA;
...
#1000=(GEOMETRIC_REPRESENTATION_CONTEXT(3)...)
#1001=GEOMETRICALLY_BOUNDED_WIREFRAME_SHAPE_REPRESENTATION('',(#1011,#1012,#1020),#1000);
#1011=AXIS2_PLACEMENT_3D(...) ;
#1012=AXIS2_PLACEMENT_3D(...) ;
#1020=GEOMETRIC_CURVE_SET('',(#1021,#1022, ...)) ;
#1021=B_SPLINE_CURVE_WITH_KNOTS(...) ;
#1022=B_SPLINE_CURVE_WITH_KNOTS(...) ;
...
ENDSEC;
END-ISO-10303-21;
```

Table of Content

- Introduction
- Basic XML structure
- Part, PartCategory & PartVersion
- PartView & ViewContext
- ProductClass & ProductConfiguration
- Occurrence, Assembly & WiringHarnessAssemblyDesign
- ShapeElement
- Geometric & topological models
- Connectors & terminals
- Features, Definitions & AssemblyShapeJoints
- Wire, Cables & TransportFeature
- Shapes/ContactFeatureDefinition & -Elements
- CrossSection of Wire, Cable & HarnessSegment
- Geometry/Topology Associations
- Complex Connectors / Assembly Hierarchy
- External Models and Element References
- ComposedGeometricModel

ComposedGeometricModel (1/6)

- The files HarnessExample_Hierarchical.xml and HarnessExample_HierarchicalReflect.xml reference into the STEP p21 file H1_abrep.stp
- The p21 file contains a main assembly part with the name Harness with an assembly component for the dummy part Product5-Multi-branchable2
- Here we show how it is possible to reference into centre lines for the harness segments using a ComposedGeometricModel for the WiringHarnessAssemblyDesign



Dataset: schema1;

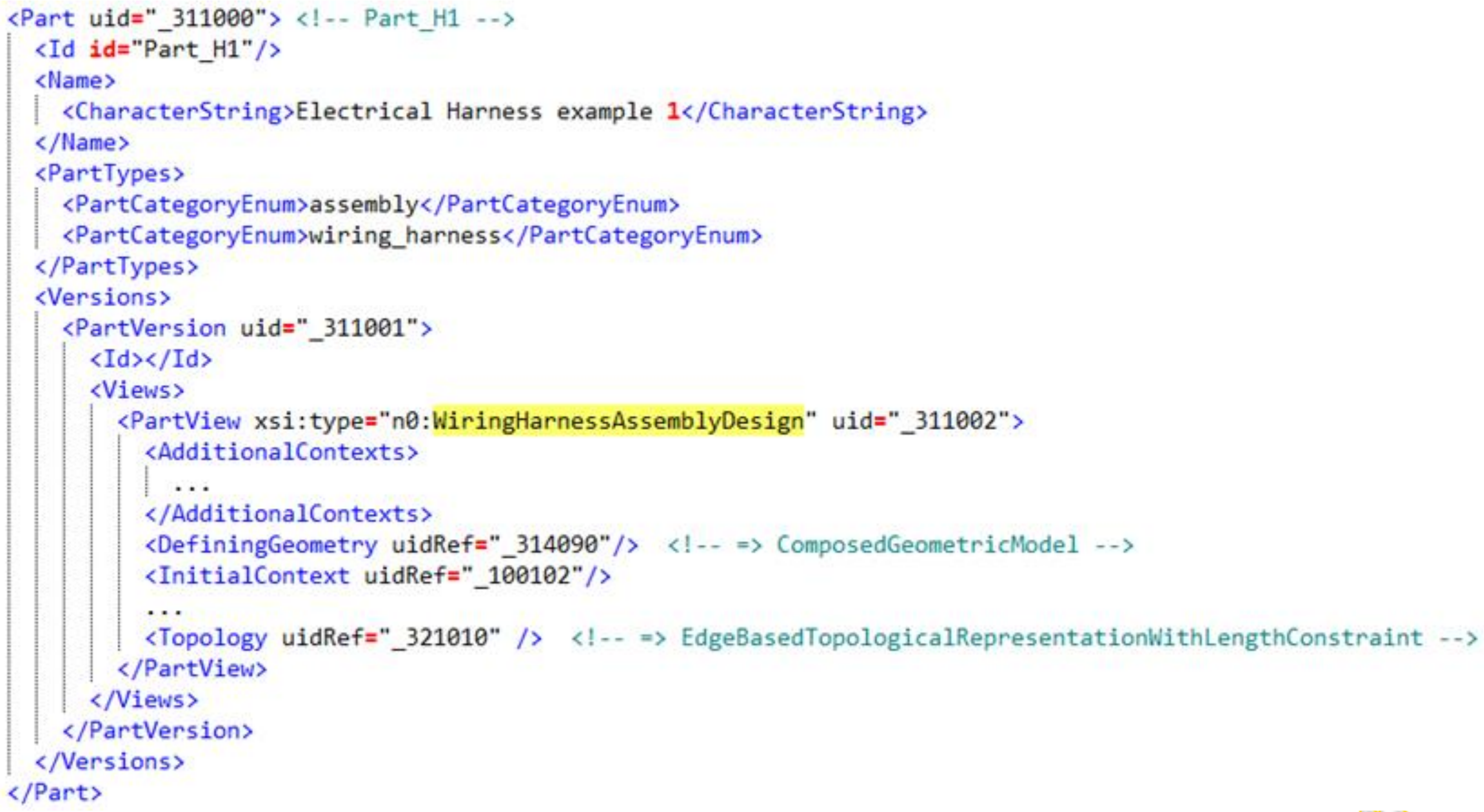
Id	Name	Version	ID Owr
ARINC600_S...	INSERT, MODULE 1215...	-	
DD104F10000	CONNECTOR, ELECTRI...	-	
Harness			
M81824_1-1		...	
MS25036-101	TERMINAL, LUG, CRP S...	-	

Dataset: schema1; Part version: , Harness,

Id	Definition	Transforr
☑		
☑ Product5-Multi-branchable2.1	, Product...	T (0.0000
☑ lug1	TERMIN...	T (6,234.(
☑ arinc1 backshell	BACKSH...	T (0.0000
☑ arinc1 connector	CONNEC...	T (6,020.:
☑ arinc1 insert A	INSERT, ...	T (6,020.:
☑ arinc1 insert B	INSERT, ...	T (6,020.:
☑ arinc1 insert C	INSERT, ...	T (6,020.:
☑ dsub1	CONNEC...	T (5,888.4
☑ plug1	PLUG, S...	T (6,157.:
☑ AA59770-6KXH1A-2016_05_16-1(	INSULATI...	T (0.0000
☑ 103-026-012-2016_05_16-11_00_:	BRAID, A...	T (0.0000
☑ AA59770-6KXH1A-2016_05_19-0(	INSULATI...	T (0.0000
☑ 103-026-012-2016_05_19-08_50_5	BRAID, A...	T (0.0000
☑ splice1	, M81824...	T (5,997.(
□ *		

ComposedGeometricModel (2/6)

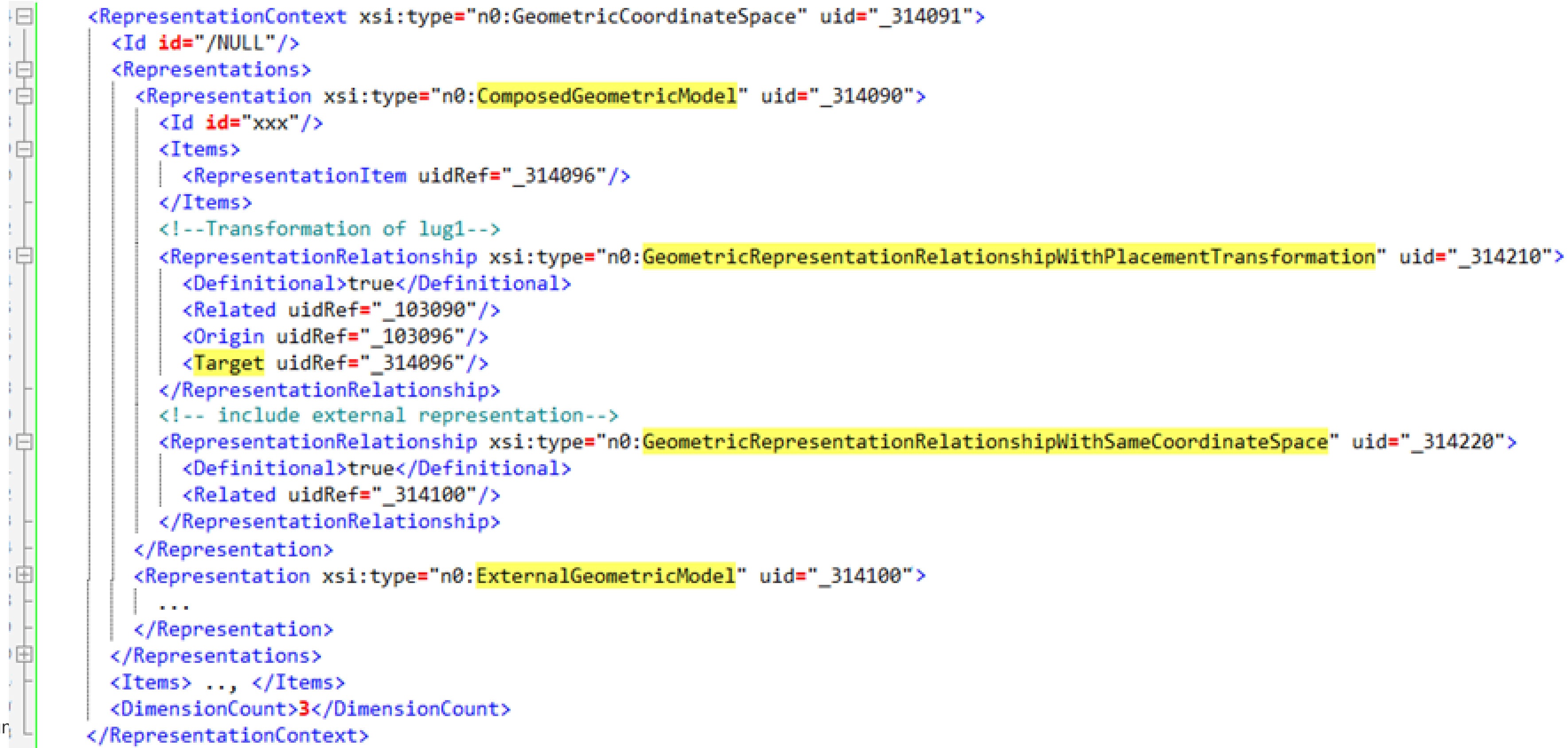
- The ***WiringHarnessAssemblyDesign*** is:
  referring the *ComposedGeometricModel*
  as *DefiningGeometry*
  refetring the *EdgeBasedTopologicalRepresentationWithLengthConstraint*
  as *Topology*



```
<Part uid="_311000"> <!-- Part_H1 -->
  <Id id="Part_H1"/>
  <Name>
    <CharacterString>Electrical Harness example 1</CharacterString>
  </Name>
  <PartTypes>
    <PartCategoryEnum>assembly</PartCategoryEnum>
    <PartCategoryEnum>wiring_harness</PartCategoryEnum>
  </PartTypes>
  <Versions>
    <PartVersion uid="_311001">
      <Id></Id>
      <Views>
        <PartView xsi:type="n0:WiringHarnessAssemblyDesign" uid="_311002">
          <AdditionalContexts>
            ...
          </AdditionalContexts>
          <DefiningGeometry uidRef="_314090"/>  <!-- => ComposedGeometricModel -->
          <InitialContext uidRef="_100102"/>
          ...
          <Topology uidRef="_321010" />  <!-- => EdgeBasedTopologicalRepresentationWithLengthConstraint -->
        </PartView>
      </Views>
    </PartVersion>
  </Versions>
</Part>
```

ComposedGeometricModel (3/6)

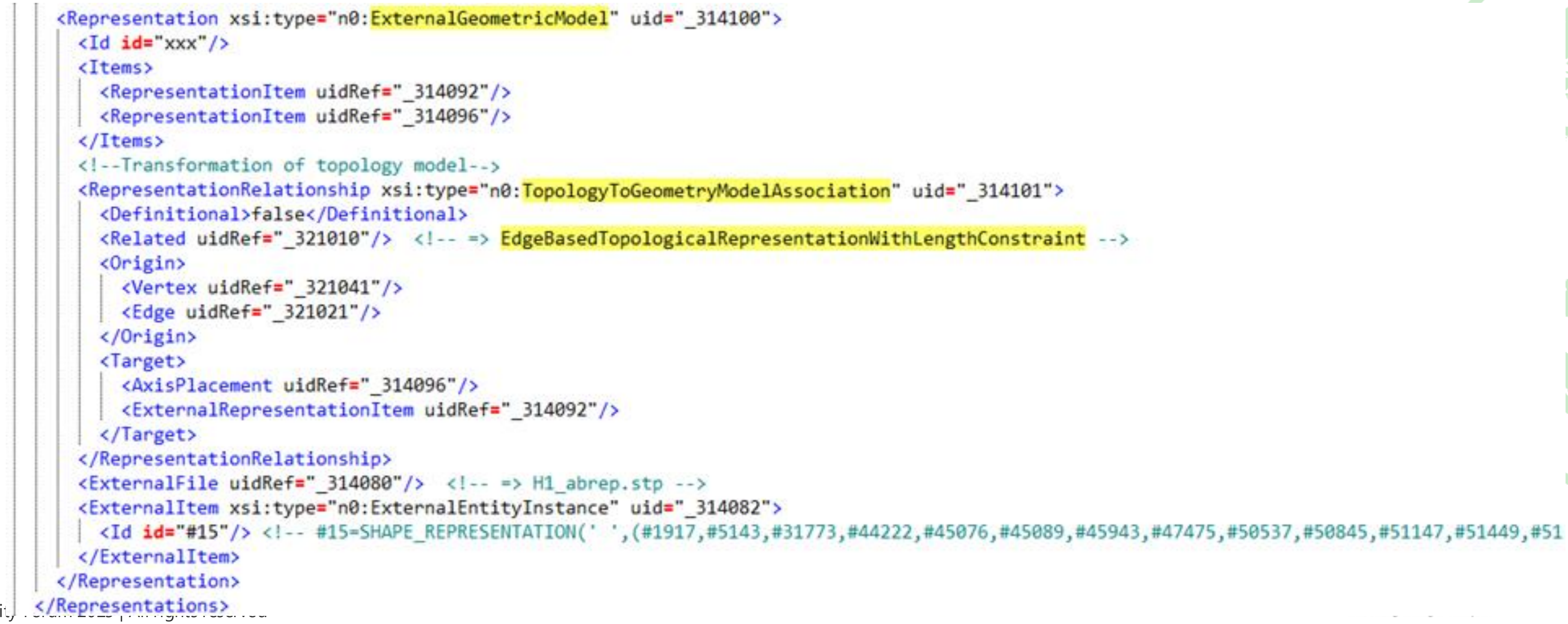
- The DefiningGeometry of a WiringHarnessAssemblyDesign might be a ComposedGeometricModel that composes the simplified geometry of the harness segments and the geometry of the connectors.
- The geometric models of connectors are typically defined in their own GeometricCoordinateSpace and brought into the ComposedGeometricModel by a GeometricRepresentationRelationshipWithPlacementTransformation.
- The ExternalGeometricModel of the harness segments are typically defined in the same GeometricCoordinateSpace and thus brought into the ComposedGeometricModel by a GeometricRepresentationRelationshipWithSameCoordinateSpace.
- The Target of the placement transformations for the connectors has to fit with the geometry of the harness segments and is therefore defined in the ExternalGeometricModel of the harness segment or of another higher one.



```
<RepresentationContext xsi:type="n0:GeometricCoordinateSpace" uid="_314091">
  <Id id="/NULL"/>
  <Representations>
    <Representation xsi:type="n0:ComposedGeometricModel" uid="_314090">
      <Id id="xxx"/>
      <Items>
        <RepresentationItem uidRef="_314096"/>
      </Items>
      <!--Transformation of lug1-->
      <RepresentationRelationship xsi:type="n0:GeometricRepresentationRelationshipWithPlacementTransformation" uid="_314210">
        <Definitional>true</Definitional>
        <Related uidRef="_103090"/>
        <Origin uidRef="_103096"/>
        <Target uidRef="_314096"/>
      </RepresentationRelationship>
      <!-- include external representation-->
      <RepresentationRelationship xsi:type="n0:GeometricRepresentationRelationshipWithSameCoordinateSpace" uid="_314220">
        <Definitional>true</Definitional>
        <Related uidRef="_314100"/>
      </RepresentationRelationship>
    </Representation>
    <Representation xsi:type="n0:ExternalGeometricModel" uid="_314100">
      ...
    </Representation>
  </Representations>
  <Items> .., </Items>
  <DimensionCount>3</DimensionCount>
</RepresentationContext>
```

ComposedGeometricModel (4/6)

- Within the ***ExternalGeometricModel*** (that is part of the *ComposedGeometricModel*) we find a ***TopologyToGeometryModelAssociation*** that is associating *Items* from ***EdgeBasedTopologicalRepresentationWithLengthConstraint*** to items of the *ExternalGeometricModel*
- Note that the number and order of the items under ***Origin*** and ***Target*** fit as the association is pairwise (1st with 1st, 2nd with 2nd, ...).
- ***ExternalFile*** refers to the external p21 file that contains the geometry
- ***ExternalItem*** / ***ExternalEntityInstance*** refers to the SHAPE_REPRESENTATION instance in that file

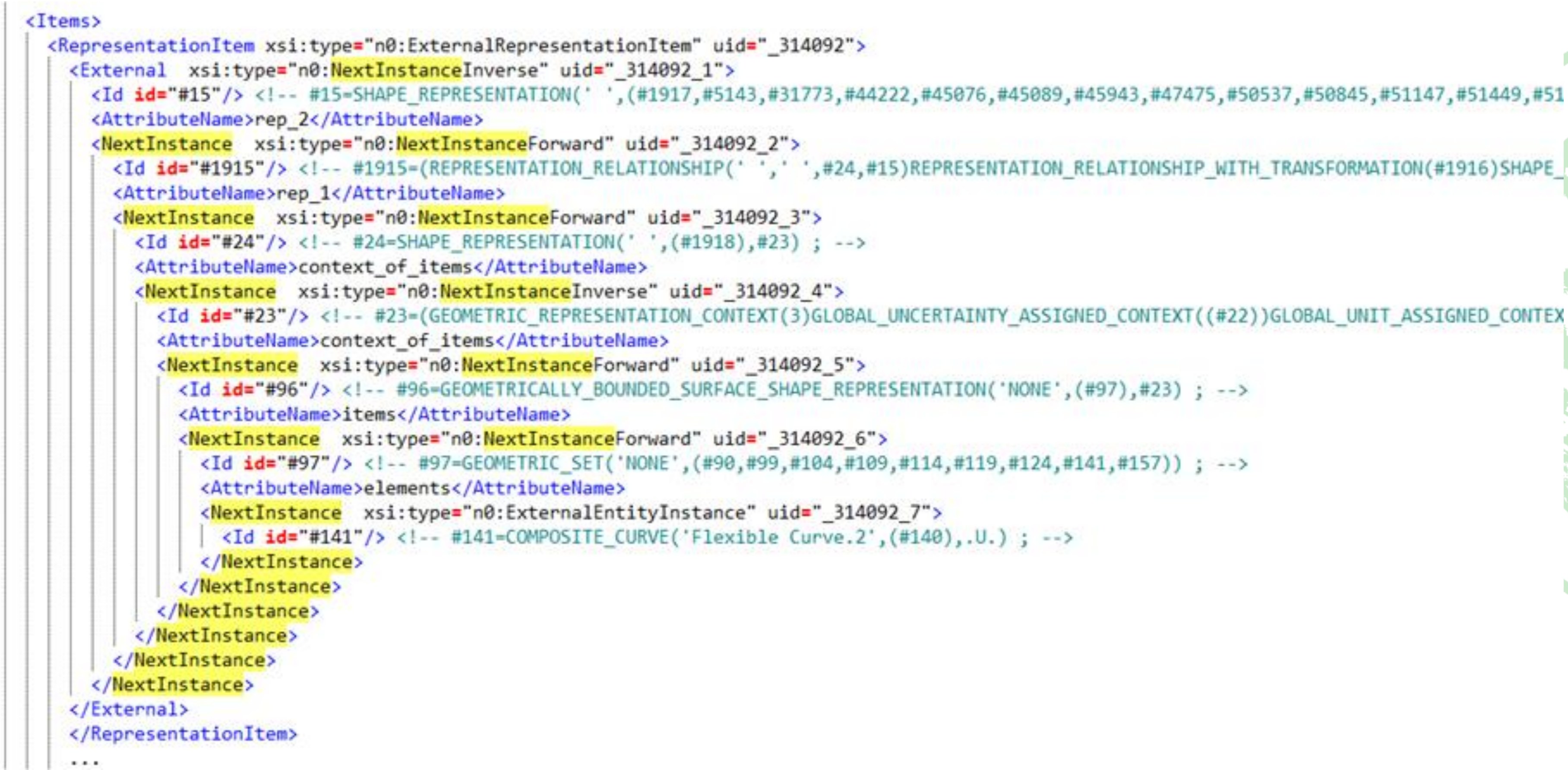


```
<Representation xsi:type="n0:ExternalGeometricModel" uid="_314100">
  <Id id="xxx"/>
  <Items>
    <RepresentationItem uidRef="_314092"/>
    <RepresentationItem uidRef="_314096"/>
  </Items>
  <!--Transformation of topology model-->
  <RepresentationRelationship xsi:type="n0:TopologyToGeometryModelAssociation" uid="_314101">
    <Definitional>false</Definitional>
    <Related uidRef="_321010"/>  <!-- => EdgeBasedTopologicalRepresentationWithLengthConstraint -->
    <Origin>
      <Vertex uidRef="_321041"/>
      <Edge uidRef="_321021"/>
    </Origin>
    <Target>
      <AxisPlacement uidRef="_314096"/>
      <ExternalRepresentationItem uidRef="_314092"/>
    </Target>
  </RepresentationRelationship>
  <ExternalFile uidRef="_314080"/>  <!-- => H1_abrep.stp -->
  <ExternalItem xsi:type="n0:ExternalEntityInstance" uid="_314082">
    <Id id="#15"/> <!-- #15=SHAPE_REPRESENTATION(' ',(#1917,#5143,#31773,#44222,#45076,#45089,#45943,#47475,#50537,#50845,#51147,#51449,#51
  </ExternalItem>
</Representation>
</Representations>
```

ComposedGeometricModel (5/6)

- Depending on how the p21 file is structured, it may be needed to traverse through a p21 file to find the right entity instance. This can be done with ***NextInstanceForward*** and ***NextInstanceInverse***.

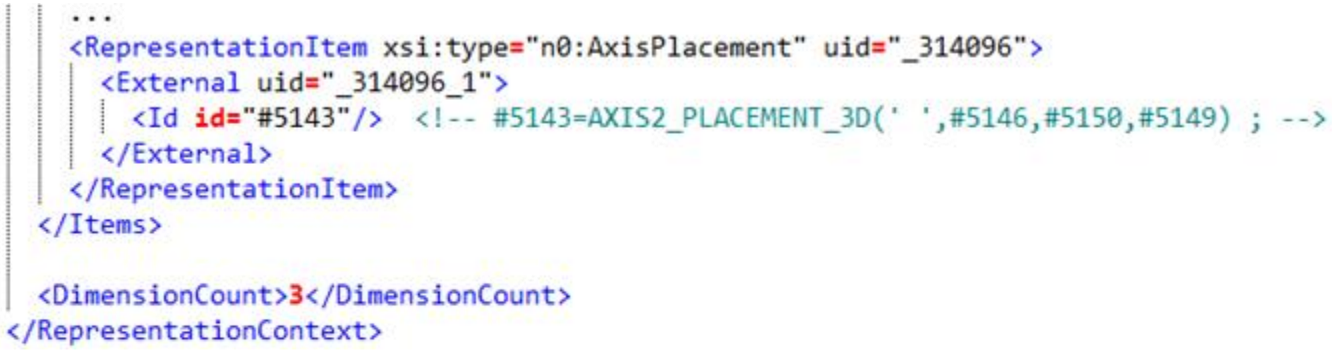
Example:
1) we start from instance #15, a *SHAPE_REPRESENTATION* and follow inverse the attribute ***rep_2*** of
2) instance #1915, a *REPRESENTATION_RELATIONSHIP_WITH_TRANSFORMATION* where the transformation has to be taken care! From there we follow attribute ***rep_1***
3) and reach instance #23, a *GEOMETRIC_REPRESENTATION_CONTEXT* and follow inverse the attribute ***context_of_items*** of
4) instance #96, a *GEOMETRICALLY_BOUNDED_SURFACE_SHAPE_REPRESENTATION* and follow the attribute ***items***
5) we reach instance #97, a *GEOMETRIC_CURVE* set and follow the attribute ***elements***
6) and finally the **ExternalEntityInstance** #141, a *COMPOSITE_CURVE* is reached



```
<Items>
  <RepresentationItem xsi:type="n0:ExternalRepresentationItem" uid="_314092">
    <External  xsi:type="n0:NextInstanceInverse" uid="_314092_1">
      <Id id="#15"/> <!-- #15=SHAPE_REPRESENTATION(' ',(#1917,#5143,#31773,#44222,#45076,#45089,#45943,#47475,#50537,#50845,#51147,#51449,#51
      <AttributeName>rep_2</AttributeName>
      <NextInstance  xsi:type="n0:NextInstanceForward" uid="_314092_2">
        <Id id="#1915"/> <!-- #1915=(REPRESENTATION_RELATIONSHIP(' ',' ',#24,#15)REPRESENTATION_RELATIONSHIP_WITH_TRANSFORMATION(#1916)SHAPE_
        <AttributeName>rep_1</AttributeName>
        <NextInstance  xsi:type="n0:NextInstanceForward" uid="_314092_3">
          <Id id="#24"/> <!-- #24=SHAPE_REPRESENTATION(' ',(#1918),#23) ; -->
          <AttributeName>context_of_items</AttributeName>
          <NextInstance  xsi:type="n0:NextInstanceInverse" uid="_314092_4">
            <Id id="#23"/> <!-- #23=(GEOMETRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNCERTAINTY_ASSIGNED_CONTEXT((#22))GLOBAL_UNIT_ASSIGNED_CONTEX
            <AttributeName>context_of_items</AttributeName>
            <NextInstance  xsi:type="n0:NextInstanceForward" uid="_314092_5">
              <Id id="#96"/> <!-- #96=GEOMETRICALLY_BOUNDED_SURFACE_SHAPE_REPRESENTATION('NONE',(#97),#23) ; -->
              <AttributeName>items</AttributeName>
              <NextInstance  xsi:type="n0:NextInstanceForward" uid="_314092_6">
                <Id id="#97"/> <!-- #97=GEOMETRIC_SET('NONE',(#90,#99,#104,#109,#114,#119,#124,#141,#157)) ; -->
                <AttributeName>elements</AttributeName>
                <NextInstance  xsi:type="n0:ExternalEntityInstance" uid="_314092_7">
                  <Id id="#141"/> <!-- #141=COMPOSITE_CURVE('Flexible Curve.2',(#140),.U.) ; -->
                </NextInstance>
              </NextInstance>
            </NextInstance>
          </NextInstance>
        </NextInstance>
      </NextInstance>
    </External>
    </RepresentationItem>
    ...
```

ComposedGeometricModel (6/6)

- We can also reference an ***AxisPlacement*** that is defined in a p21 by, e.g. an AXIS2_PLACEMENT_3D.
- This is done by populating the attribute ***External***.
- When the attribute *External* is used the *AxisPlacement* attributes ***Axis***, ***Position*** and ***RefDirection*** must not be used as the placement information is taken from the external p21 file only



```
    ...
    <RepresentationItem xsi:type="n0:AxisPlacement" uid="_314096">
      <External uid="_314096_1">
        <Id id="#5143"/>  <!-- #5143=AXIS2_PLACEMENT_3D(' ',#5146,#5150,#5149) ; -->
      </External>
    </RepresentationItem>
  </Items>

  <DimensionCount>3</DimensionCount>
</RepresentationContext>
```