EWIS Interoperability Forum
Test Suite v3.0

2021-06-16

Contacts		
Lothar Klein	Sophie HERAIL	Daniel Ganser
Steinweg 1	CIMPA S.A.S.	Gulfstream Aerospace Corporation
36093 Künzell / Germany	Centreda 1	BTC
	4, Avenue Didier Daurat	171 Crossroads Parkway
	31700 Blagnac, France	Savannah, GA 31407, U.S.A.
	Subcontractor for AIRBUS	
	Operations SAS – IZMA	
lothar.klein@lksoft.com	sophie.herail@airbus.com	dan.ganser@gulfstream.com

Document History

Release	Date	Change
1.0	2019-09-11	Initial Release
2.0	2020-09-09/11	Update & extended for 2nd test round
3.0	2021-06-16	Adding Connectivity3 and -4 test case for 3rd test round. Connectivity1 and 2: - Adding JointType - adding PartTransportFeature for cable, and wire - WireIdentification with definition to PartTransportFeature - fixing CableOccurrenceTerminal replacing attribute LocationGroup by ElementOf

Trademarks

Capital® Harness™ is a trademarks of Mentor Graphics Corporation

CATIA is a trademark of DASSAULT SYSTEMES

AMPHENOL is a trademark of AMPHENOL CORPORATION

Table of Contents

List of Figures

1 Introduction

This document describes the suite of test cases to be used for the EWIS Interoperability Forum. The EWIS-IF is a joint testing forum, organized and facilitated by AFNeT and PDES.

2 Formal Test Syntax

This clause defines a formal syntax for the definition of synthetic test cases in the terms of the Domain Model. Purpose of this syntax is to formulate test cases in a clear, easy readable and unambiguous way. A macro capability allows to define standard patterns once and then apply them again and again. It is intended to convert test cases using this syntax into XML Schematron for the

The formal test syntax allows the definition of patterns of instances of Application Objects (AO); here they are AOs of the Domain Model. The test syntax is used to define the test specifications for both import and export of STEP XML files.

Instances of AOs and other values (string, real, ...) are identified (ID) by a leading "@" followed by a positive number. This number is unique within a particular test case and macro. If the same ID is used several times within a test case or macro, then this means the same AO instance or other value. If the same ID is used in different test cases or macros this does not have any meaning.

Every instance or other value has to have a definition statement. Such a statement starts with the ID, followed by a ":" and then followed by it's type that is defined in the domain model. After this constraints on the attribute values of instances or the value itself can be stated within "("...")".

The order of IDs within a macro is significantly. They should start with @1, @2, @3 … @N. When invoking a macro from another test case or higher level macro, these IDs are replaced with the IDs and values defined within the macro invocation.

3 Macros

3.1 Part_with_PartView

This macro constraints single instances of a Part, a PartVersion, a PartView and a ViewContext to be linked together. Only the name for a Part is constrained as the ID of a Part is often rather system dependent. The ViewContext used as the initialContext for the PartView is constrained for the predefined LifeCycleStage "design".

```
Macro Part_with_PartView (
   @1:Part( Name=@2, Versions[i]=@3 );
   @2:CharacterString
   @3:PartVersion( Views[i]=@4 );
   @4:PartView( InitialContext=@5 );
   @5:ViewContext( LifeCycleStage=PredefinedApplicationDomainEnum(design) );
);
```

3.2 Part_with_ID_and_PartView

This macro is similar to the macro Part_with_PartView but instead of constraining the name of a Part the ID of a part is constraint.

```
Macro Part_with_ID_and_PartView (
   @1:Part( Id=@2, Versions=(@3) );
   @2:Identifier;
   @3:PartVersion( Views=(@4) );
```

```
  @4:PartView( InitialContext=@5 );
  @5:ViewContext( LifeCycleStage=PredefinedApplicationDomainEnum(design) );
);
```

3.3   Part_with_WiringHarnessAssemblyDesign

This macro is similar to the macro Part_with_PartView and constraints single instances of a Part, a PartVersion, a WiringHarnessAssemblyDesign (that is a sub-subytype of PartView) and a ViewContext to be linked together. The Part is constrained for the PartType "wiring_harness".

```
Macro Part_with_WiringHarnessAssemblyDesign (
  @1:Part( Name=@2,
           Versions[i]=@3,
           PartTypes[i]=PartCategoryEnum(wiring_harness) );
  @2:CharacterString
  @3:PartVersion( Views[i]=@4 );
  @4:WiringHarnessAssemblyDesign( InitialContext=@5 );
  @5:ViewContext( LifeCycleStage=PredefinedApplicationDomainEnum(design),
           ApplicationDomain=PredefinedApplicationDomainEnum(electrical)
                             );
);
```

3.4   Harness_design_with_segment_topology

This macro is an extension of the macro Part_with_WiringHarnessAssemblyDesign. In addition to this it adds a constraint for an additional ViewContext with the predefined LifeCycleStage "wiring_harness_segment_topology".

```
Macro Harness_design_with_segment_topology (
  Part_with_WiringHarnessAssemblyDesign(@1,@2,@3,@4,@5);
  @1:Part;
  @2:CharacterString;
  @3:PartVersion;
  @4:WiringHarnessAssemblyDesign( AdditionalContexts[i]=@6 );
  @5:ViewContext;
  @6:ViewContext( ApplicationDomain=PredefinedApplicationDomainEnum(
           wiring_harness_segment_topology) );
);
```

3.5   Undirected_edge

This macro constrains an edge with two vertices so that either one of the Vertices is the EdgeStart and the other Vertices is the EdgeEnd. In STEP all Edges are by default directed, however for the design of the topology of an EWH the direction of an Edge is not relevant (however it might be relevant for the purpose of manufacturing).

```
Macro Undirected_edge (
  @1=Edge( ( EdgeStart=@2, EdgeEnd=@3)
           OR
           ( EdgeStart=@3, EdgeEnd=@2) );
  @2=Vertex;
  @3=Vertex;
);
```

4 Test Case Specifications

4.1 EWH-Assembly1

This test case focuses on a very basic flat assembly structures as it might show up in EWH. This test does not address connectivity or topological information. This test is an extension of the typical assembly structure as provided in the document "Recommended Practices for AP242 Business Object Model XML Assembly Structure".

The following elements are tested:

- Part with PartCategories: discrete_part, raw_material_by_length, wire, cable, connector, lug

- WiringHarnessAssemblyDesign that is a subtype of AssemblyDefinition

- specific kinds of Part Occurrences: SingleOccurrence, QuantifiedOccurrence, WireOccurrence, CableOccurrence

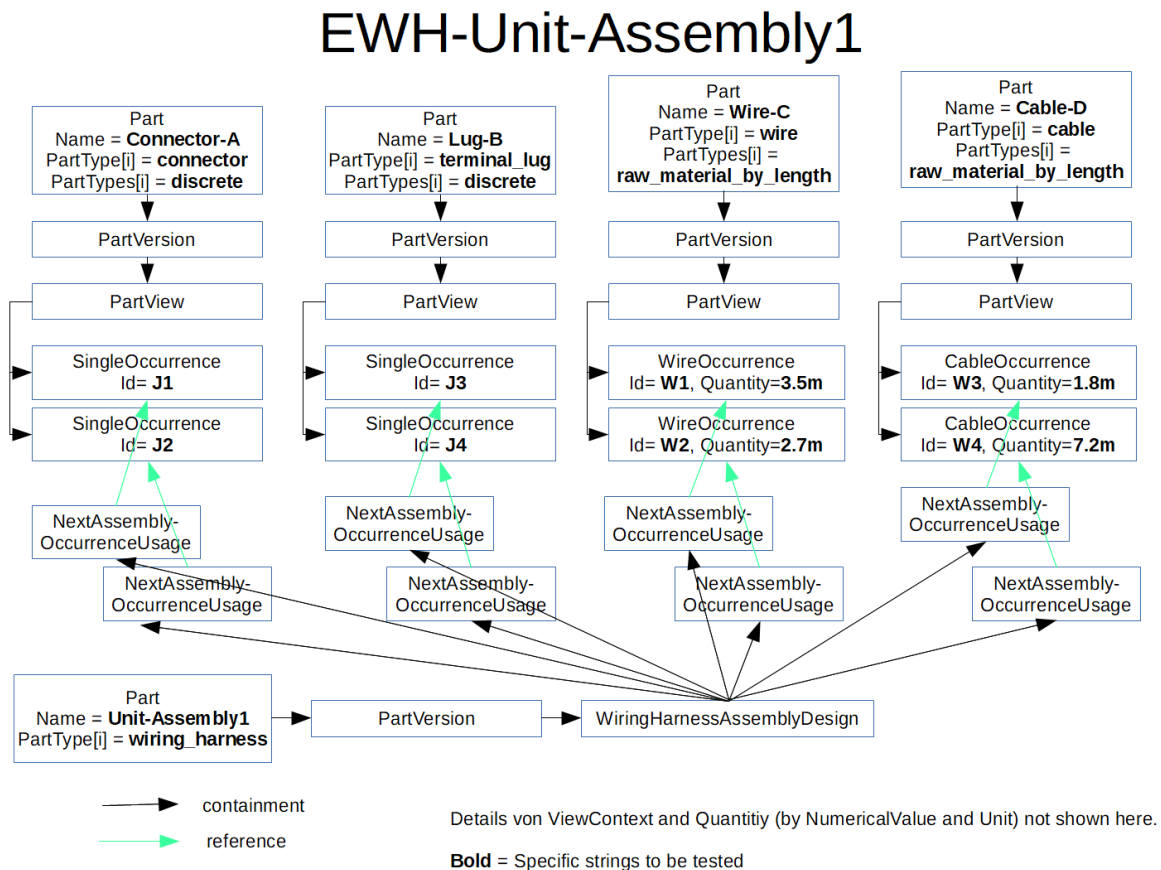
Figure 1: EWH-Assembly1

Formal test-case specification:

```
Test EWH-Assembly1 (

@4:ViewContext;
@5:ViewContext;
@8:Unit( Name=ClassString("metre"), Quantity=ClassString("length") );

@100:Part( PartTypes[i]=PartCategoryEnum(connector), PartTypes[i]=PartCate-
goryEnum(discrete) );
@101:PartVersion;
```

```
@102:PartView;
Part_with_Name_and_PartView( @100, "Connector-A", @101, @102, @4);
@111:SingleOccurrence( Id=IdentifierString("J1"), Definition=@102 );
@121:SingleOccurrence( Id=IdentifierString("J2"), Definition=@102 );


@200:Part( PartTypes[i]=PartCategoryEnum(terminal_lug), PartTypes[i]=Part-
CategoryEnum(discrete) );
@201:PartVersion;
@202:PartView;
Part_with_Name_and_PartView(@200, "Lug-B", @201, @202, @4);
@211:SingleOccurrence( Id=IdentifierString("J3"), Definition=@202 );
@221:SingleOccurrence( Id=IdentifierString("J4"), Definition=@202 );


@300:Part( PartTypes[i]=PartCategoryEnum(wire), PartTypes[i]=PartCatego-
ryEnum(raw_material_by_length) );
@301:PartVersion;
@302:PartView;
Part_with_Name_and_PartView(@300, "Wire-C", @301, @302, @4);
@311:WireOccurrence( Id=IdentifierString("W1"), Definition=@302,
Quantity=@312 );
@312:NumericalValue( Unit=@8, ValueComponent=3.5 );
@321:WireOccurrence( Id=IdentifierString("W2"), Definition=@302,
Quantity=@322 );
@312:NumericalValue( Unit=@8, ValueComponent=2.7 );


@400:Part( PartTypes[i]=PartCategoryEnum(cable), PartTypes[i]=PartCatego-
ryEnum(raw_material_by_length) );
@401:PartVersion;
@402:PartView;
Part_with_Name_and_PartView(@400, "Cable-D", @401, @402, @4);
@411:CableOccurrence( Id=IdentifierString("W3"), Definition=@402,
Quantity=@412 );
@412:NumericalValue( Unit=@8, ValueComponent=1.8 );
@421:CableOccurrence( Id=IdentifierString("W4"), Definition=@402,
Quantity=@422 );
@412:NumericalValue( Unit=@8, ValueComponent=7.2 );


@9000:Part;
@9001:PartVersion;
@9002:WiringHarnessAssemblyDesign;
@9003:ViewContext;
@9004:ViewContext;
part_with_WiringHarnessAssemblyDesign( @9000,"EWH Test-Case
Assembly1",@9001,@9002,@9003,@9004 );


@9101:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@111 );
@9102:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@121 );
@9103:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@211 );
@9104:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@221 );
@9105:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@311 );
@9106:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@321 );
@9107:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@411 );
@9108:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@421 );
```

```
sizeof(Part) = 5;
sizeof(PartVersion) = 5;
sizeof(PartView) = 4;
sizeof(WiringHarnessAssemblyDesign) = 1;
sizeof(NextAssemblyOccurrenceUsage) = 8;
sizeof(SingleOccurrence) = 4;
sizeof(WireOccurrence) = 2;
sizeof(CableOccurrence) = 2;


);
```

4.2 EWH-Topology1

This test case focuses on a very basic topological structure needed for EWH without any other information. The test consists of a flexible topological/geometric representation of the harness, consisting of 6 vertices and 5 edges with length.
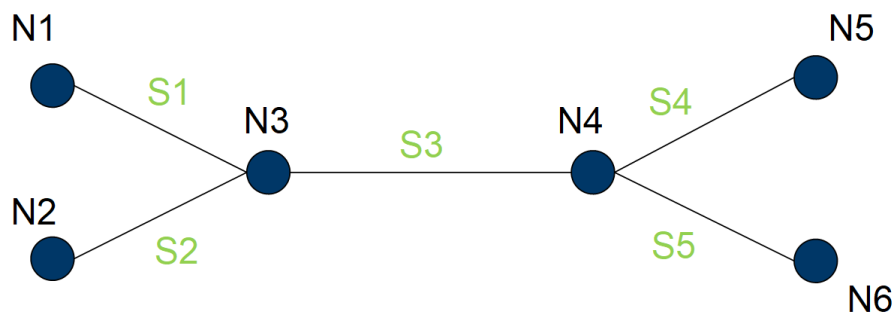
Figure 2: EWH-Topology1

 Formal test-case specification:

```
Test EWH-Topology1 (

@8:Unit( Name=ClassString("metre"), Quantity=ClassString("length") );

@9000:Part;
@9001:PartVersion;
@9002:WiringHarnessAssemblyDesign( Topology=@9901 );
@9003:ViewContext;
@9004:ViewContext;
harness_design_with_segment_topology(@9000,
  "EWH Test-Case Topology1",@9001,@9002,@9003,@9004);

@9900:GeometricCoordinateSpace( Units=@8, DimensionCount=1 );
@9901:EdgeBasedTopologicalRepresentationWithLengthConstraint(
 Items=(@9902), ContextOfItems=@9900 );
@9902:ConnectedEdgeSet( ConnectedEdges=(@9931,@9932,@9933,@9934,@9935) );

@9911:Point();
@9912:Point();
@9913:Point();
@9914:Point();
@9915:Point();
```

```
@9916:Point();

@9921:VertexPoint( name='N1' VertexGeometry=@9911 );
@9922:VertexPoint( name='N2' VertexGeometry=@9912 );
@9923:VertexPoint( name='N3' VertexGeometry=@9913 );
@9924:VertexPoint( name='N4' VertexGeometry=@9914 );
@9925:VertexPoint( name='N5' VertexGeometry=@9915 );
@9926:VertexPoint( name='N6' VertexGeometry=@9916 );

@9931:EdgeBoundedCurveWithLength( name='S1', EdgeGeometry=@9941 );
undirected_edge(@9931, @9921, @9923)
@9932:EdgeBoundedCurveWithLength( name='S2', EdgeGeometry=@9942 );
undirected_edge(@9932, @9922, @9923)
@9933:EdgeBoundedCurveWithLength( name='S3', EdgeGeometry=@9943 );
undirected_edge(@9933, @9923, @9924);
@9934:EdgeBoundedCurveWithLength( name='S4', EdgeGeometry=@9944 );
undirected_edge(@9934, @9924, @9925);
@9935:EdgeBoundedCurveWithLength( name='S5', EdgeGeometry=@9945 );
undirected_edge(@9935,@9924,@9926);

@9941:BoundedCurveWithLength( EdgeLength=PositiveLengthMeasure(2.0) );
@9942:BoundedCurveWithLength( EdgeLength=PositiveLengthMeasure(4.0) );
@9943:BoundedCurveWithLength( EdgeLength=PositiveLengthMeasure(6.0) );
@9944:BoundedCurveWithLength( EdgeLength=PositiveLengthMeasure(8.0) );
@9945:BoundedCurveWithLength( EdgeLength=PositiveLengthMeasure(10.0) );

sizeof(Part) = 1;
sizeof(PartVersion) = 1;
sizeof(WiringHarnessAssemblyDesign) = 1;
sizeof(NextAssemblyOccurrenceUsage) = 0;

sizeof(GeometricCoordinateSpace) = 1;
sizeof(EdgeBasedTopologicalRepresentationWithLengthConstraint) = 1;
sizeof(ConnectedEdgeSet) = 1;
sizeof(BoundedCurveWithLength) = 5;
sizeof(EdgeBoundedCurveWithLength) = 5;
sizeof(VertexPoint) = 6;
sizeof(Point) = 6;
sizeof(CartesianPoint) = 0;
sizeof(PointOnCurve) = 0;

);
```
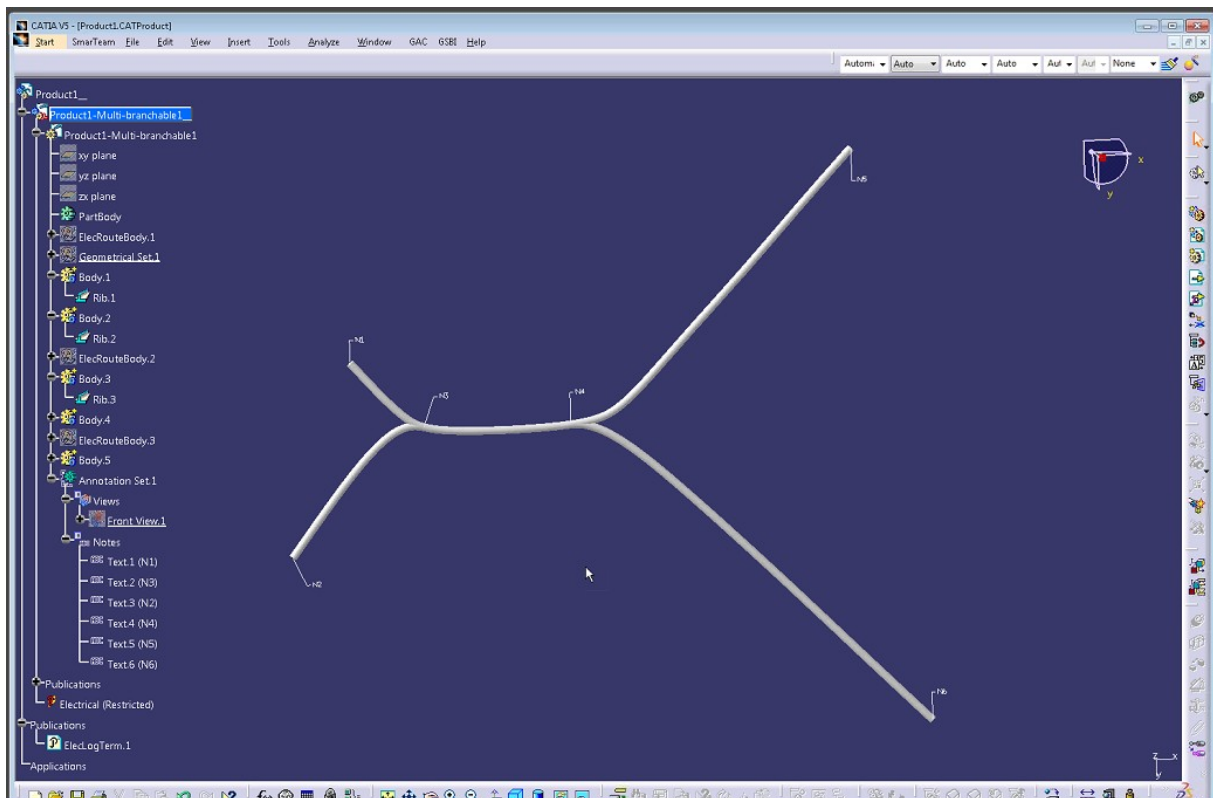
Figure 3: Example in CATIA

Provided test files:

AP242ed2 XML: EWH-UseCase-Topology1.xml

Native CATIA: Topology Test1 - Sample CatiaV5 STP.zip

Mentor/Siemens Capital Harness: Topology Test1 - Sample Capital HX2ML.xml

KBL, generated from Capital Harness:

Topology Test1 - Sample Capital KBL_2.3.kbl

Topology Test1 - Sample Capital KBL2.4.kbl

4.3 EWH-Topology2

This test case is an extension of test case EWH-Topology1 that is merged with a simplified EWH-Assembly1 test case.

- the topology model is extended for Paths, SubEdges and PointOnCurves
  - Path P1 traverses the EdgeBoundedCurveWithLength S1, S3, S4
  - Path P2 traverses the EdgeBoundedCurveWithLength S2, S3, S5
  - Path P3 traverses the SubEdges S2.2, S3.1
  - for the definition of the VertexPoints for the SubEdges, two PointOnCurves are defined in the middle of the underlying BoundedCurveWithLength
  - it is up to the implementations to ensure that the orientations of the Edges in the EdgeList of a Path fits with the orientation of the underlying BoundedCurveWith-

Length. See the attributes *Path.OrientationList* and *EdgeCurve.SameSense* for this purpose.

- the simplified assembly structure consists of
  - a single wire
  - a single cable
  - a protective covering for only a certain region
- two simple 2-pin connectors and two terminal lugs; one at each extremity of the harness topology
- geometry-to-topology association of
  - wire/cable/protection *Occurrences* to *Paths*
  - connectors and terminal lug *Occurrences* to *VertexPoints*
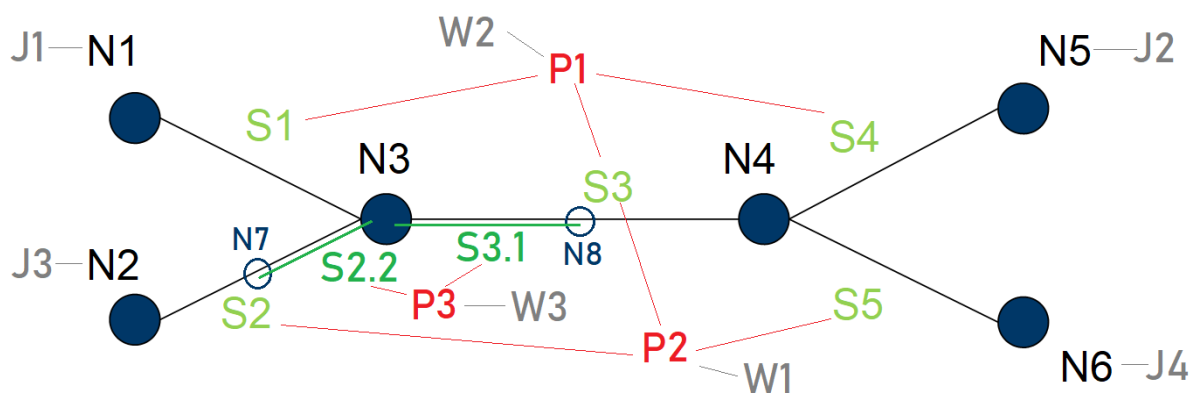
Figure 4: EWH-Topology2

Formal test-case specification:

(Draft; not completed yet; references depends on the available p21 files)

```
Test EWH-Topology2 (

@8:Unit( Name=ClassString("metre"), Quantity=ClassString("length") );

@100:Part( PartTypes[i]=PartCategoryEnum(connector),
        PartTypes[i]=PartCategoryEnum(discrete) );
@101:PartVersion;
@102:PartView( DefiningGeometry=@191 );
Part_with_Name_and_PartView( @100, "Connector-A", @101, @102, @4);
@111:SingleOccurrence( Id=IdentifierString("J1"), Definition=@102 );
@121:SingleOccurrence( Id=IdentifierString("J2"), Definition=@102 );
@190:GeometricCoordinateSpace( DimensionCount=3 );
@191:GeometricModel( items[i]=@192, ContextOfItems=@190 );
@192:AxisPlacement;

@200:Part( PartTypes[i]=PartCategoryEnum(terminal_lug),
        PartTypes[i]=PartCategoryEnum(discrete) );
@201:PartVersion;
@202:PartView( DefiningGeometry=@291 );
Part_with_Name_and_PartView(@200, "Lug-B", @201, @202, @4);
```

```
@211:SingleOccurrence( Id=IdentifierString("J3"), Definition=@202 );
@221:SingleOccurrence( Id=IdentifierString("J4"), Definition=@202 );
@290:GeometricCoordinateSpace( DimensionCount=3 );
@291:GeometricModel( items[i]=@292, ContextOfItems=@290 );
@292:AxisPlacement;

@300:Part( PartTypes[i]=PartCategoryEnum(wire),
           PartTypes[i]=PartCategoryEnum(raw_material_by_length) );
@301:PartVersion;
@302:PartView( DefiningGeometry=@391 );
Part_with_Name_and_PartView(@300, "Wire-C", @301, @302, @4);
@311:WireOccurrence( Id=IdentifierString("W1"),
  Definition=@302, Quantity=@312 );
@312:NumericalValue( Unit=@8, ValueComponent=3.5 );
@390:GeometricCoordinateSpace( DimensionCount=2 );
@391:GeometricModel( name='2D cross section',
  items[i]=@392, ContextOfItems=@390 );
@392:AxisPlacement; # placeholder for 2D centre

@400:Part( PartTypes[i]=PartCategoryEnum(cable),
           PartTypes[i]=PartCategoryEnum(raw_material_by_length) );
@401:PartVersion;
@402:PartView( DefiningGeometry=@491 );
Part_with_Name_and_PartView(@400, "Cable-D", @401, @402, @4);
@411:CableOccurrence( Id=IdentifierString("W2"), Definition=@402,
Quantity=@412 );
@412:NumericalValue( Unit=@8, ValueComponent=1.8 );
@490:GeometricCoordinateSpace( DimensionCount=2 );
@491:GeometricModel( name='cross section', items[i]=@492,
ContextOfItems=@490 );
@492:AxisPlacement;

@500:Part( PartTypes[i]=PartCategoryEnum(protective_covering),
           PartTypes[i]=PartCategoryEnum(raw_material_by_length) );
@501:PartVersion;
@502:PartView( DefiningGeometry=@591 );
Part_with_Name_and_PartView(@500, "Protection-E", @501, @502, @4);
@511:QuantifiedOccurrence( Id=IdentifierString("W3"),
                          Definition=@402, Quantity=@412 );
@512:NumericalValue( Unit=@8, ValueComponent=1.2 );
@590:GeometricCoordinateSpace( DimensionCount=2 );
@591:GeometricModel( name='cross section',
                    items[i]=@592, ContextOfItems=@590 );
@592:AxisPlacement;

@9000:Part;
@9001:PartVersion;
@9002:WiringHarnessAssemblyDesign( Topology=@9901 );
@9003:ViewContext;
@9004:ViewContext;
harness_design_with_segment_topology(@9000,
  "EWH Test-Case Topology2",@9001,@9002,@9003,@9004);
```

```
@9101:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@111,
                                   Placement=(@9111) ); # connector J1
@9102:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@121,
                                   Placement=(@9112) ); # connector J2
@9103:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@211,
                                   Placement=(@9113) ); # terminal lug J3
@9104:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@221,
                                   Placement=(@9114) ); # terminal lug J4
@9105:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@311,
                                   Placement=(@9115 ) ); # wire W1
@9106:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@411,
                                   Placement=(@9116 ) ); # cable W2
@9107:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@511,
                                   Placement=(@9117 ) ); # protection W3


@9111:GeometryToTopologyModelAssociation (
  Relating=@9901, Related=@191, Origin=@192, Target=@9921); # connector J1
@9112:GeometryToTopologyRepresentationAssociation(
  Relating=@9901, Related=@191, Origin=@192, Target=@9925); # connector J2
@9113:GeometryToTopologyRepresentationAssociation(
  Relating=@9901, Related=@291, Origin=@292, Target=@9922); # terminal lug
J3
@9114:GeometryToTopologyRepresentationAssociation(
  Relating=@9901, Related=@291, Origin=@292, Target=@9926); # terminal lug
J4
@9115:GeometryToTopologyRepresentationAssociation(
  Relating=@9901, Related=@391, Origin=@392, Target=@9952); # wire W1
@9116:GeometryToTopologyRepresentationAssociation(
  Relating=@9901, Related=@491, Origin=@492, Target=@9951); # cable W2
@9117:GeometryToTopologyRepresentationAssociation(
  Relating=@9901, Related=@591, Origin=@493, Target=@9953); # protection W3


@9900:GeometricCoordinateSpace( Units=@8, DimensionCount=1 );
@9901:EdgeBasedTopologicalRepresentationWithLengthConstraint(
  Items=(@9902,@9951,@9952,@9953), # the ConnectedEdgeSet + paths
  ContextOfItems=@9900 );
@9902:ConnectedEdgeSet( ConnectedEdges=(@9931,@9932,@9933,@9934,@9935) );
  # only main edges, not sub-edges


@9911:Point();
@9912:Point();
@9913:Point();
@9914:Point();
@9915:Point();
@9916:Point();
@9917:PointOnCurve( BasicCurve=@9942, Parameter=2.0 );
  # in the middle of the basic curve
@9918:PointOnCurve( BasicCurve=@9943, Parameter=3.0 );
  # in the middle of the basic curve


@9921:VertexPoint( name='N1', VertexGeometry=@9911 );
@9922:VertexPoint( name='N2', VertexGeometry=@9912 );
@9923:VertexPoint( name='N3', VertexGeometry=@9913 );
```

```
@9924:VertexPoint( name='N4', VertexGeometry=@9914 );
@9925:VertexPoint( name='N5', VertexGeometry=@9915 );
@9926:VertexPoint( name='N6', VertexGeometry=@9916 );
@9927:VertexPoint( name='N7', VertexGeometry=@9917 );
@9928:VertexPoint( name='N8', VertexGeometry=@9918 );

@9931:EdgeBoundedCurveWithLength( name='S1', EdgeGeometry=@9941 );
undirected_edge(@9931, @9921, @9923)
@9932:EdgeBoundedCurveWithLength( name='S2', EdgeGeometry=@9942 );
undirected_edge(@9932, @9922, @9923)
@9933:EdgeBoundedCurveWithLength( name='S3', EdgeGeometry=@9943 );
undirected_edge(@9933, @9923, @9924);
@9934:EdgeBoundedCurveWithLength( name='S4', EdgeGeometry=@9944 );
undirected_edge(@9934, @9924, @9925);
@9935:EdgeBoundedCurveWithLength( name='S5', EdgeGeometry=@9945 );
undirected_edge(@9935,@9924,@9926);
@9936:SubEdge( name='S2.2', ParentEdge=@9932 );
undirected_edge(@9936,@9927,@9923);
@9937:SubEdge( name='S3.1', ParentEdge=@9933);
undirected_edge(@9937,@9923,@9928);

@9941:BoundedCurveWithLength( EdgeLength=PositiveLengthMeasure(2.0) );
  # for S1
@9942:BoundedCurveWithLength( EdgeLength=PositiveLengthMeasure(4.0) );
  # for S2
@9943:BoundedCurveWithLength( EdgeLength=PositiveLengthMeasure(6.0) );
  # for S3
@9944:BoundedCurveWithLength( EdgeLength=PositiveLengthMeasure(8.0) );
  # for S4
@9945:BoundedCurveWithLength( EdgeLength=PositiveLengthMeasure(10.0) );
  # for S5

# vendors to ensure that the edge are oriented in correct way
@9951:Path( name="P1", EdgeList=(@9931,@9933,@9934) ); # S1+S3+S4
@9952:Path( name="P2", EdgeList=(@9932,@9933,@9935) ); # S2+S3+S5
@9953:Path( name="P3", EdgeList=(@9936,@9937) );       # S2.2+S3.1

sizeof(Part) = 6;
sizeof(PartVersion) = 6;
sizeof(PartView) = 5;
sizeof(WiringHarnessAssemblyDesign) = 1;
sizeof(NextAssemblyOccurrenceUsage) = 7;
sizeof(SingleOccurrence) = 4;
sizeof(WireOccurrence) = 1;
sizeof(CableOccurrence) = 1;

sizeof(GeometricCoordinateSpace) = 1;
sizeof(EdgeBasedTopologicalRepresentationWithLengthConstraint) = 1;
sizeof(ConnectedEdgeSet) = 1;
sizeof(BoundedCurveWithLength) = 5;
sizeof(EdgeBoundedCurveWithLength) = 5;
sizeof(VertexPoint) = 8;
sizeof(Point) = 6;
```

```
sizeof(CartesianPoint) = 0;
sizeof(PointOnCurve) = 2;
sizeof(SubEdge) = 2;
sizeof(Path) = 3;


);
```

4.4 EWH-Topology3

This test case is an extension of the content in EWH-Topology2 for:

- external references into p21 files:

    ○ complete p21 files for discrete parts "Connector-A" and "Lug-B"

    ○ element reference into p21 file for centre-curves and axis-placements

- topology-to-geometry association

This test case is likely to be refined later on as the topic of XML "external element refer-
ences" is new to the community of STEP implementers, and there are no final recommended
practices yet for this area (need common work with CAX-IF and PDM-IF). So even if only a
subset of the below gets implemented would already be a success.

Formal test-case specification:

```
Test EWH-Topology3 (


@9:FormatProperty( DataFormat="ISO 10303-242", CharacterCode="ISO 8859-
1" );


@8:Unit( Name=ClassString("metre"), Quantity=ClassString("length") );


@100:Part( PartTypes[i]=PartCategoryEnum(connector),
          PartTypes[i]=PartCategoryEnum(discrete) );
@101:PartVersion;
@102:PartView( DefiningGeometry=@191 );
Part_with_Name_and_PartView( @100, "Connector-A", @101, @102, @4);
@111:SingleOccurrence( Id=IdentifierString("J1"), Definition=@102 );
@121:SingleOccurrence( Id=IdentifierString("J2"), Definition=@102 );
@190:GeometricCoordinateSpace( DimensionCount=3, Items=(@192) );
@191:ExternalGeometricModel( items=(@192), ContextOfItems=@190,
      ExternalFile=@193  ); # was GeometricModel in EWH-Topology2
@192:AxisPlacement( Position=(0.0, 0.0, 0.0) ); # Axis and RefDirection de-
faults
  # alternatively use ExternalRepresentationItem to select placement in p21
file
@193:DigitalFile( FileLocations=@194, FileFormat=@9, exists(Id) ); #
id=file name


@200:Part( PartTypes[i]=PartCategoryEnum(terminal_lug),
          PartTypes[i]=PartCategoryEnum(discrete) );
@201:PartVersion;
@202:PartView( DefiningGeometry=@291 );
Part_with_Name_and_PartView(@200, "Lug-B", @201, @202, @4);
@211:SingleOccurrence( Id=IdentifierString("J3"), Definition=@202 );
@221:SingleOccurrence( Id=IdentifierString("J4"), Definition=@202 );
@290:GeometricCoordinateSpace( DimensionCount=3, Items=(@192) );
```

```
@291:ExternalGeometricModel( Items=(@292), ContextOfItems=@290,
  ExternalFile=@293  ); # was GeometricModel in EWH-Topology2
@292:AxisPlacement( Position=(0.0, 0.0, 0.0) ); # Axis and RefDirection de-
faults
@293:DigitalFile( FileLocations=@294, FileFormat=@9, exists(Id) ); #
id=file name

@300:Part( PartTypes[i]=PartCategoryEnum(wire),
          PartTypes[i]=PartCategoryEnum(raw_material_by_length) );
@301:PartVersion;
@302:PartView( DefiningGeometry=@391 );
Part_with_Name_and_PartView(@300, "Wire-C", @301, @302, @4);
@311:WireOccurrence( Id=IdentifierString("W1"),
  Definition=@302, Quantity=@312 );
@312:NumericalValue( Unit=@8, ValueComponent=3.5 );
@390:GeometricCoordinateSpace( DimensionCount=2 );
@391:GeometricModel( name='2D cross section',
  items[i]=@392, ContextOfItems=@390 );
@392:AxisPlacement; # placeholder for 2D centre

@400:Part( PartTypes[i]=PartCategoryEnum(cable),
          PartTypes[i]=PartCategoryEnum(raw_material_by_length) );
@401:PartVersion;
@402:PartView( DefiningGeometry=@491 );
Part_with_Name_and_PartView(@400, "Cable-D", @401, @402, @4);
@411:CableOccurrence( Id=IdentifierString("W2"), Definition=@402,
Quantity=@412 );
@412:NumericalValue( Unit=@8, ValueComponent=1.8 );
@490:GeometricCoordinateSpace( DimensionCount=2 );
@491:GeometricModel( name='cross section', items[i]=@492,
ContextOfItems=@490 );
@492:AxisPlacement;

@500:Part( PartTypes[i]=PartCategoryEnum(protective_covering),
          PartTypes[i]=PartCategoryEnum(raw_material_by_length) );
@501:PartVersion;
@502:PartView( DefiningGeometry=@591 );
Part_with_Name_and_PartView(@500, "Protection-E", @501, @502, @4);
@511:QuantifiedOccurrence( Id=IdentifierString("W3"),
                          Definition=@402, Quantity=@412 );
@512:NumericalValue( Unit=@8, ValueComponent=1.2 );
@590:GeometricCoordinateSpace( DimensionCount=2 );
@591:GeometricModel( name='cross section',
                 items[i]=@592, ContextOfItems=@590 );
@592:AxisPlacement;

@9000:Part;
@9001:PartVersion;
@9002:WiringHarnessAssemblyDesign(
  Topology=@9901, DefiningGeometry=@9201  );
@9003:ViewContext;
@9004:ViewContext;
harness_design_with_segment_topology(@9000,
  "EWH Test-Case Topology2",@9001,@9002,@9003,@9004);
```

```
@9101:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@111,
  Placement=(@9111,@9801) ); # connector J1
@9102:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@121,
  Placement=(@9112,@9802) ); # connector J2
@9103:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@211,
  Placement=(@9113,@9803) ); # terminal lug J3
@9104:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@221,
  Placement=(@9114,@9804) ); # terminal lug J4
@9105:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@311,
  Placement=(@9115) ); # wire W1
@9106:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@411,
  Placement=(@9116) ); # cable W2
@9107:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@511,
  Placement=(@9117) ); # protection W3


@9111:GeometryToTopologyModelAssociation (
  Relating=@9901, Related=@191, Origin=@192, Target=@9921); # connector J1
@9112:GeometryToTopologyRepresentationAssociation(
  Relating=@9901, Related=@191, Origin=@192, Target=@9925); # connector J2
@9113:GeometryToTopologyRepresentationAssociation(
  Relating=@9901, Related=@291, Origin=@292, Target=@9922); # terminal lug
J3
@9114:GeometryToTopologyRepresentationAssociation(
  Relating=@9901, Related=@291, Origin=@292, Target=@9926); # terminal lug
J4
@9115:GeometryToTopologyRepresentationAssociation(
  Relating=@9901, Related=@391, Origin=@392, Target=@9952); # wire W1
@9116:GeometryToTopologyRepresentationAssociation(
  Relating=@9901, Related=@491, Origin=@492, Target=@9951); # cable W2
@9117:GeometryToTopologyRepresentationAssociation(
  Relating=@9901, Related=@591, Origin=@493, Target=@9953); # protection W3


@9200:GeometricCoordinateSpace( Units=@8, DimensionCount=3 );
@9201:ComposedGeometricModel( ContextOfItems=@9200;
Items=(@9211,@9212,@9213,@9214) );
  # contains the connectors and multi-branchable
@9211=AxisPlacement; # alternativel ExternalRepresentationItem from p21
files
@9212=AxisPlacement;
@9213=AxisPlacement;
@9214=AxisPlacement;
@9801:GeometricRepresentationRelationshipWithPlacementTransformation(
  origin=@192, target=@9211,
  relating=@9201, related=@191, Definitional=TRUE ); # connector J1
@9802:GeometricRepresentationRelationshipWithPlacementTransformation(
  origin=@192, target=@9212,
  relating=@9201, related=@191, Definitional=TRUE ); # connector J2
@9803:GeometricRepresentationRelationshipWithPlacementTransformation(
  origin=@292, target=@9213,
  relating=@9201, related=@292, Definitional=TRUE ); # terminal lug J3
@9804:GeometricRepresentationRelationshipWithPlacementTransformation(
  origin=@292, target=@9214,
  relating=@9201, related=@292, Definitional=TRUE ); # terminal lug J4
```

```
@9808:GeometricRepresentationRelationshipWithSameCoordinateSpace(
  relating=@9201, related=@9211, Definitional=TRUE ); # for stuff in the
multi-branchable


@9210:DigitalFile( FileFormat=@9, FileFormat=@9, exisit(Id) ); # Id=name of
p21 file
@9211:ExternalGeometricModel( items=(@9212,@9213,@9214,@9215),
  ContextOfItems=@9200, ExternalFile=@9210 ); # multi-branchable
@9220:AxisPlacement;
@9221:ExternalRepresentationItem( External=@9231 );
@9222:ExternalRepresentationItem( External=@9232 );
@9223:ExternalRepresentationItem( External=@9233 );
@9224:ExternalRepresentationItem( External=@9234 );
@9225:ExternalRepresentationItem( External=@9235 );
@9226:ExternalRepresentationItem( External=@9236 );
@9227:ExternalRepresentationItem( External=@9237 );


# for the following instance the ID attribute must be set
# corresponding to an anchor or instance-id in the target p21 file
@9231:ExternalEntityInstance( exist(Id), Source=@9210 );
@9232:ExternalEntityInstance( exist(Id), Source=@9210 );
@9223:ExternalEntityInstance( exist(Id), Source=@9210 );
@9224:ExternalEntityInstance( exist(Id), Source=@9210 );
@9225:ExternalEntityInstance( exist(Id), Source=@9210 );
@9226:ExternalEntityInstance( exist(Id), Source=@9210 ); # curve for S2.2
@9227:ExternalEntityInstance( exist(Id), Source=@9210 ); # curve for S3.1


# Alternative for @9226 and @9227
# use PointOnCurve with PARAMETER given in p21 file and
# construct a new curve in XML to associat to

@9299:TopologyToGeometryModelAssociation( Relating=@9201, Related=@9901,
  # order of pairs: connector J1, ... J2, terminal lug J3, ... J4, edges
S1..S5, S2.2, S3.1
  # maybe instead of paths we have to map single EdgeBoundedCurveWithLength
  Origin=(@9921,@9925,@9922,@9926, @9931,@9932,@9933,@9934,@9935,
@9936,@9937),
  Target=(@9211,@9212,@9213,@9214, @9221,@9222,@9223,@9224,@9225,
@9226,@9227) );


@9900:GeometricCoordinateSpace( Units=@8, DimensionCount=1 );
@9901:EdgeBasedTopologicalRepresentationWithLengthConstraint(
  Items=(@9902,@9951,@9952,@9953), # the ConnectedEdgeSet + paths
  ContextOfItems=@9900 );
@9902:ConnectedEdgeSet( ConnectedEdges=(@9931,@9932,@9933,@9934,@9935) );
  # only main edges, not sub-edges


@9911:Point();
@9912:Point();
@9913:Point();
@9914:Point();
@9915:Point();
@9916:Point();
@9917:PointOnCurve( BasicCurve=@9942, Parameter=2.0 );
```

```
  # in the middle of the basic curve
@9918:PointOnCurve( BasicCurve=@9943, Parameter=3.0 );
  # in the middle of the basic curve

@9921:VertexPoint( name='N1', VertexGeometry=@9911 );
@9922:VertexPoint( name='N2', VertexGeometry=@9912 );
@9923:VertexPoint( name='N3', VertexGeometry=@9913 );
@9924:VertexPoint( name='N4', VertexGeometry=@9914 );
@9925:VertexPoint( name='N5', VertexGeometry=@9915 );
@9926:VertexPoint( name='N6', VertexGeometry=@9916 );
@9927:VertexPoint( name='N7', VertexGeometry=@9917 );
@9928:VertexPoint( name='N8', VertexGeometry=@9918 );

@9931:EdgeBoundedCurveWithLength( name='S1', EdgeGeometry=@9941 );
undirected_edge(@9931, @9921, @9923)
@9932:EdgeBoundedCurveWithLength( name='S2', EdgeGeometry=@9942 );
undirected_edge(@9932, @9922, @9923)
@9933:EdgeBoundedCurveWithLength( name='S3', EdgeGeometry=@9943 );
undirected_edge(@9933, @9923, @9924);
@9934:EdgeBoundedCurveWithLength( name='S4', EdgeGeometry=@9944 );
undirected_edge(@9934, @9924, @9925);
@9935:EdgeBoundedCurveWithLength( name='S5', EdgeGeometry=@9945 );
undirected_edge(@9935,@9924,@9926);
@9936:SubEdge( name='S2.2', ParentEdge=@9932 );
undirected_edge(@9936,@9927,@9923);
@9937:SubEdge( name='S3.1', ParentEdge=@9933);
undirected_edge(@9937,@9923,@9928);

@9941:BoundedCurveWithLength( EdgeLength=PositiveLengthMeasure(2.0) );
  # for S1
@9942:BoundedCurveWithLength( EdgeLength=PositiveLengthMeasure(4.0) );
  # for S2
@9943:BoundedCurveWithLength( EdgeLength=PositiveLengthMeasure(6.0) );
  # for S3
@9944:BoundedCurveWithLength( EdgeLength=PositiveLengthMeasure(8.0) );
  # for S4
@9945:BoundedCurveWithLength( EdgeLength=PositiveLengthMeasure(10.0) );
  # for S5

# vendors to ensure that the edge are oriented in correct way
@9951:Path( name="P1", EdgeList=(@9931,@9933,@9934) ); # S1+S3+S4
@9952:Path( name="P2", EdgeList=(@9932,@9933,@9935) ); # S2+S3+S5
@9953:Path( name="P3", EdgeList=(@9936,@9937) );       # S2.2+S3.1

);
```

4.5 EWH-Connectivity1

This test case consists of a WiringHarnessAssemblyDesign that is composed of

- a terminal lug "LUG01" that is defined by Part "640903-1" with a single terminal "1"

- a connector "PLUG01" that is defined by Part "RCA123" with terminals "0" and "1"

- a connector "P-CONN01" that is defined by Part "IMC16-2002X" with terminals "1" and "2"

- a cable "CABLE01" that is defined by Part "9962 009100" with two wires, one black and the other white

- a wire "WIRE01" that is defined by Part "83027 001100"

- the two connectors are joint to the two ends of the cable.

- the single wire connects LUG01 with terminal "1" of "PLUG01"
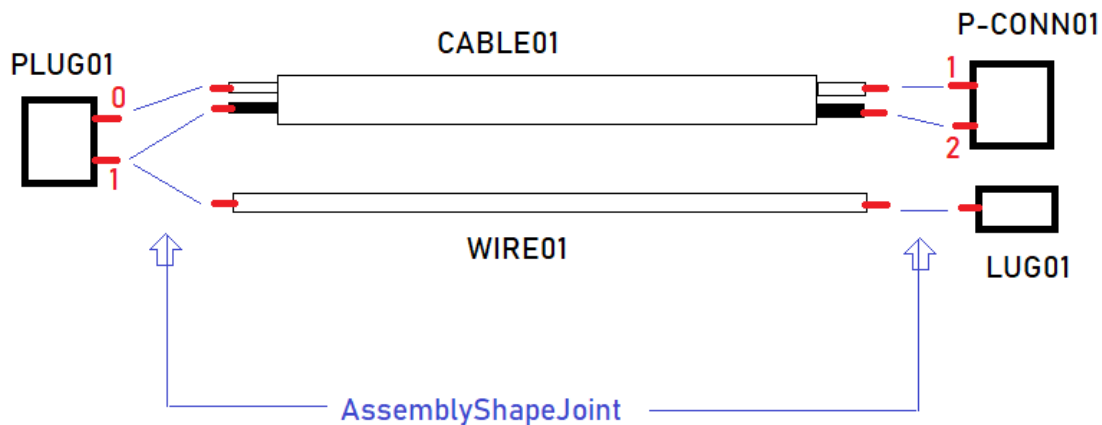
Figure 5: EWH-Connectivity1

Formal test-case specification:

```
Test EWH-Connectivity1 (

@4:ViewContext;
@5:ViewContext;
@8:Unit( Name=ClassString("metre"), Quantity=ClassString("length") );

@50:Organization( name="MIL ...??")
@51:Organization( name="Deutch Company Ltd ...")
@52:Organization( name="BELDEN company ...")
@60:Identifier( Id=IdentifierString("Standard RCA connector") )

@70:WireColourBasedIdentificationCode( Id="white" );
@71:WireColourBasedIdentificationCode( Id="black" );


# Terminal Lug
@100:Part( PartTypes[i]=PartCategoryEnum(terminal_lug),
  PartTypes[i]=PartCategoryEnum(discrete) );
@101:PartVersion;
@102:PartView;
@103:Identifier( Id=IdentifierString("640903-1"), IdentificationContext=@50
)
Part_with_ID_and_PartView(@100, @103, @101, @102, @4);
@104:PartTerminal( ElementOf=@102, Id="1", DomainType="electrical",
```

```
    InterfaceOrJoinTerminal="join_terminal" );
@111:SingleOccurrence( Id=IdentifierString("LUG01"), Definition=@102 );
@112:OccurrenceTerminal( ElementOf=@111, Definition=@104 );


# Connector with integrated contacts
@200:Part( PartTypes[i]=PartCategoryEnum(connector), PartTypes[i]=PartCate-
goryEnum(discrete) );
@201:PartVersion;
@202:PartView;
@203:Identifier( Id=IdentifierString("RCA123"), IdentificationContext=@60 )
Part_with_ID_and_PartView(@200, @203, @201, @202, @4);
@204:PartTerminal( ElementOf=@202, Id="0", DomainType="electrical",
    InterfaceOrJoinTerminal="join_terminal" );
@205:PartTerminal( ElementOf=@202, Id="1", DomainType="electrical",
    InterfaceOrJoinTerminal="join_terminal" );
@211:SingleOccurrence( Id=IdentifierString("PLUG01"), Definition=@202 );
    @214:OccurrenceTerminal( ElementOf=@211, Definition=@204 );
    @215:OccurrenceTerminal( ElementOf=@211, Definition=@205 );


# Simplified model for Deutch connector with direct PartTerminals
@300:Part( PartTypes[i]=PartCategoryEnum(connector), PartTypes[i]=PartCate-
goryEnum(discrete) );
@301:PartVersion;
@302:PartView;
@303:Identifier( Id=IdentifierString("IMC16-2002X"),
IdentificationContext=@51 )
Part_with_ID_and_PartView(@300, @303, @301, @302, @4);
@306:PartTerminal( ElementOf=@302, Id="1", DomainType="electrical",
    InterfaceOrJoinTerminal="join_terminal" );
@307:PartTerminal( ElementOf=@302, Id="2", DomainType="electrical",
    InterfaceOrJoinTerminal="join_terminal" );
@311:SingleOccurrence( Id=IdentifierString("P-CONN01"), Definition=@302 );
#316:OccurrenceTerminal( ElementOf=@311, Definition=@306 );
#317:OccurrenceTerminal( ElementOf=@311, Definition=@307 );


# Cable
@500:Part( PartTypes[i]=PartCategoryEnum(cable), PartTypes[i]=PartCatego-
ryEnum(raw_material_by_length) );
@501:PartVersion;
@502:PartView;
@503:Identifier( Id=IdentifierString("9962 009100"),
IdentificationContext=@52 )
Part_with_ID_and_PartView(@500, @503, @501, @502, @4);
@504:PartTransportFeature( ElementOf=@503, Id="CABLE01-WHT" );
@505:PartTransportFeature( ElementOf=@503, Id="CABLE01-BLK" );
@511:CableOccurrence( Id=IdentifierString("CABLE01"), Definition=@502,
Quantity=@512 );
    @512:NumericalValue( Unit=@8, ValueComponent=1.8 );
    @513:WireIdentification( ElementOf=@511, Id="CABLE01-WHT",
        Definition=@504, code=@70 );
    @514:WireIdentification( ElementOf=@511, Id="CABLE01-BLK",
        Definition=@505, code=@71 );
    @515:CableOccurrenceTerminalLocationGroup( ElementOf=@511, Name="end
a" );
```

```
        @521:CableOccurrenceTerminal( ElementOf=@515, AssociatedTransportFea-
ture=@513 );
        @523:CableOccurrenceTerminal( ElementOf=@515, AssociatedTransportFea-
ture=@514 );
   @516:CableOccurrenceTerminalLocationGroup( ElementOf=@511, Name="end
b" );
        @522:CableOccurrenceTerminal( ElementOf=@516, AssociatedTransportFea-
ture=@513 );
        @524:CableOccurrenceTerminal( ElementOf=@516, AssociatedTransportFea-
ture=@514 );


# Wire
@600:Part( PartTypes[i]=PartCategoryEnum(wire),
  PartTypes[i]=PartCategoryEnum(raw_material_by_length) );
@601:PartVersion;
@602:PartView;
@603:Identifier( Id=IdentifierString("83027 001100"), IdentificationCon-
text=@52 )
Part_with_ID_and_PartView(@600, @603, @601, @602, @4);
@611:WireOccurrence( Id=IdentifierString("WIRE01"), Definition=@602, Quan-
tity=@612 );
@612:NumericalValue( Unit=@8, ValueComponent=3.5 );
@613=WireIdentification( ElementOf=@611, DomainType="electrical" ... )
@614=WireOccurrenceTerminal( ElementOf=@611,
AssociatedTransportFeature=@613,
  Name="end a" );
@615=WireOccurrenceTerminal( ElementOf=@611,
AssociatedTransportFeature=@613,
  Name="end b" );


# EWH-Assembly
@9000:Part;
@9001:PartVersion;
@9002:WiringHarnessAssemblyDesign;
@9003:ViewContext;
@9004:ViewContext;
part_with_WiringHarnessAssemblyDesign( @9000,
  "EWH Test-Case Connectivity1",@9001,@9002,@9003,@9004 );


@9101:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@111 );
@9102:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@211 );
@9103:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@311 );
@9106:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@511 );
@9107:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@611 );


# connections
@9210:AssemblyShapeJoint( ElementOf=@9002,
JointType="crimped_connection" );
  @9211:AssemblyShapeJointItemRelationship( Relating=@9210, Related=@214 );
# PLUG01 / 0
  @9212:AssemblyShapeJointItemRelationship( Relating=@9210, Related=@521 );
# CABLE01-WHT / end a
@9220:AssemblyShapeJoint( ElementOf=@9002,
JointType="crimped_connection"  );
```

```
  @9221:AssemblyShapeJointItemRelationship( Relating=@9220, Related=@316 );
# P-CONN01 / 01
  @9222:AssemblyShapeJointItemRelationship( Relating=@9220, Related=@522 );
# CABLE01-WHT / end b
@9230:AssemblyShapeJoint( ElementOf=@9002,
JointType="crimped_connection"  );
  @9231:AssemblyShapeJointItemRelationship( Relating=@9230, Related=@215 );
# PLUG01 / 1
  @9232:AssemblyShapeJointItemRelationship( Relating=@9230, Related=@523 );
# CABLE01-BLK / end a
  @9232:AssemblyShapeJointItemRelationship( Relating=@9230, Related=@614 );
# WIRE01 / end a
@9240:AssemblyShapeJoint( ElementOf=@9002,
JointType="crimped_connection"  );
  @9241:AssemblyShapeJointItemRelationship( Relating=@9240, Related=@317 );
# P-CONN01 / 02
  @9242:AssemblyShapeJointItemRelationship( Relating=@9240, Related=@524 );
# CABLE01-BLK / end a
@9250:AssemblyShapeJoint( ElementOf=@9002,
JointType="crimped_connection"  );
  @9251:AssemblyShapeJointItemRelationship( Relating=@9250, Related=@112 );
# LUG01 / 1
  @9252:AssemblyShapeJointItemRelationship( Relating=@9250, Related=@615 );
# WIRE01 / end b

sizeof(Part) = 6;
sizeof(PartVersion) = 6;
sizeof(PartView) = 5;
sizeof(WiringHarnessAssemblyDesign) = 1;
sizeof(NextAssemblyOccurrenceUsage) = 5;
sizeof(SingleOccurrence) = 3;
sizeof(WireOccurrence) = 1;
sizeof(CableOccurrence) = 1;

sizeof(PartTerminal) = 5; # only the join terminals
sizeof(OccurrenceTerminal) = 5; # only the join terminals
sizeof(WireColourBasedIdentificationCode) = 2;
sizeof(WireOccurrenceTerminal) = 2;
sizeof(CableOccurrenceTerminalLocationGroup) = 2;
sizeof(CableOccurrenceTerminal) = 4;
sizeof(AssemblyShapeJoint) = 5;
sizeof(AssemblyShapeJointItemRelationship) = 11;

);
```

4.6 EWH-Connectivity2

This test case is very similar to the test case EWH-Connectivity1. The difference is that the connector "P-CONN01" is now modeled more realistically. There is no direct terminal but instead there are the two cavities "1" and "2" for two separate connector contacts:

- a terminal lug "LUG01" that is defined by Part "640903-1" with a single terminal "1"

- a connector "PLUG01" that is defined by Part "RCA123" with terminals "0" and "1"

- a connector "P-CONN01" that is defined by Part "IMC16-2002X" with cavities "1" and "2"

- two connector contacts "P-CONN01-01" and "P-CONN01-02" that are defined by Part "6860-201-20278" that fits into the cavities of a connector of type "IMC16-2002X". Each of the connector contacts has a single join terminal.

- a cable "CABLE01" that is defined by Part "9962 009100" with two wires, one black and the other white

- a wire "WIRE01" that is defined by Part "83027 001100"

- connector "PLUG01" is joint to one ends of the cable, and the two connector contacts are joint to the other end. The connector contacts are then inserted into connector "PLUG01"

- the single wire connects LUG01 with terminal "1" of "PLUG01"

Initial input data from users to this test:

Part Number	Occurrence (REFDES)	Terminals	Description	Images
640903-1	LUG01	1	MIL standard Receptacle (similar to Lug)	
RCA123	PLUG01	0 / 1	Standard RCA plug (Or Cinch) https://en.wikipedia.org/wiki/RCA_connector	
IMC16-2002X	P-CONN01		Deutch waterproof connector with two cavities	
6860-201-20278	P-CONN01-01 / P-CONN01-02		Deutch Plug Contact	
9962 009100			BELDEN Cable	
83027 001100			BELDEN Wire	

Table 1: Original part list for connectivity test

From	From Pin	Wire Name	Material	To	To Pin
PLUG01	0	CABLE01-WHT	9962 009100	P-CONN01	P-CONN01-01
PLUG01	1	CABLE01-BLK	9962 009100	P-CONN01	P-CONN01-02
PLUG01	1	WIRE01	83027 001100	LUG01	

Table 2: Original wire list for connectivity test

Adaptions on the original input data to be used by AP242-EWH:

- the cavities of the connector "IMC16-2002X" are not numbered, but they are in the Deutsch documentation indicated with "1" and "2". It is essential to not mix them up;

- no terminals are defined for the connector contact "6860-201-20278", but of course there is an implicit join-terminal (for crimping) and an interface-terminal for the external connection (the later one is not covered here);

- there is no explicit information which connector contact P-CONN01-01/-02 goes into the cavities 1/2 of the connector. This can only be derived from the naming. For AP242-EWH it is essential to state which connector-contact is inserted into which cavity of the connector (by AssemblyShapeJoint);

- the wire list indicates two connections onto the PLUG01/1 pin. For AP242-EWH this is handled by a triple AssemblyShapeJoint of PLUG01/1 with the cable and single wire terminals.
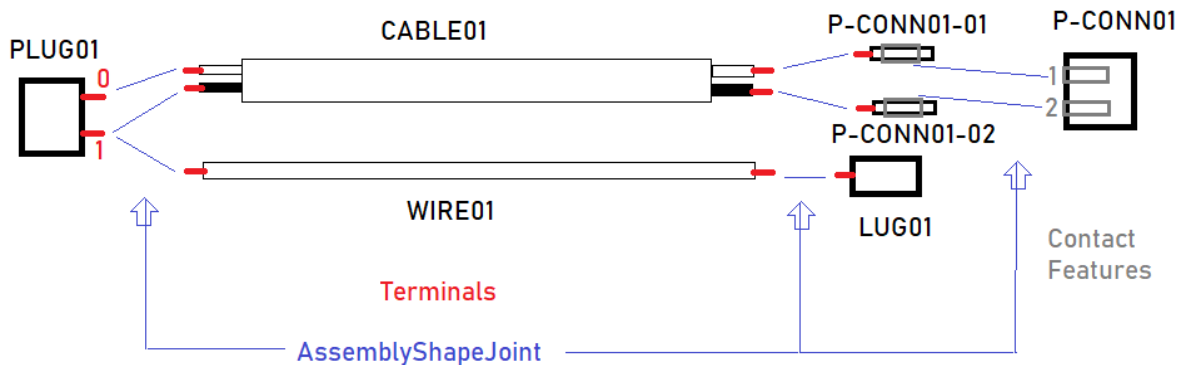
Figure 6: EWH-Connectivity2

For the AssemblyShapeJoints the following JointTypes are to be used:

- JointType="soldered_connection" for the electrical joints on PLUG01 and LUG01;

- JointType="crimped_connection" for the electrical connections of P-CONN01-01/-02 with the cable;

- JointType="snap_connection" for the mechanical connection of P-CONN01-01/-02 with P-CONN01.

Formal test-case specification:

```
Test EWH-Connectivity2 (

@4:ViewContext;
@5:ViewContext;
@8:Unit( Name=ClassString("metre"), Quantity=ClassString("length") );
```

```
@50:Organization( name="MIL ...??")
@51:Organization( name="Deutch Company Ltd ...")
@52:Organization( name="BELDEN company ...")
@60:Identifier( Id=IdentifierString("Standard RCA connector") )

@70:WireColourBasedIdentificationCode( Id="white" );
@71:WireColourBasedIdentificationCode( Id="black" );


# Terminal Lug
@100:Part( PartTypes[i]=PartCategoryEnum(terminal_lug),
  PartTypes[i]=PartCategoryEnum(discrete) );
@101:PartVersion;
@102:PartView;
@103:Identifier( Id=IdentifierString("640903-1"),
  IdentificationContext=@50 )
Part_with_ID_and_PartView(@100, @103, @101, @102, @4);
@104:PartTerminal( ElementOf=@102, Id="1", DomainType="electrical",
    InterfaceOrJoinTerminal="join_terminal" );
@111:SingleOccurrence( Id=IdentifierString("LUG01"), Definition=@102 );
  @112:OccurrenceTerminal( ElementOf=@111, Definition=@104 );


# Connector with integrated contacts
@200:Part( PartTypes[i]=PartCategoryEnum(connector),
  PartTypes[i]=PartCategoryEnum(discrete) );
@201:PartVersion;
@202:PartView;
@203:Identifier( Id=IdentifierString("RCA123"),
  IdentificationContext=@60 )
Part_with_ID_and_PartView( @200, "", @201, @202, @4);
@204:PartTerminal( ElementOf=@202, Id="0", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal" ); # or left, right m GND ?
@205:PartTerminal( ElementOf=@202, Id="1", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal" );
@211:SingleOccurrence( Id=IdentifierString("PLUG01"), Definition=@202 );
  @214:OccurrenceTerminal( ElementOf=@211, Definition=@204 );
  @215:OccurrenceTerminal( ElementOf=@211, Definition=@205 );


# Realistic model for Deutsch connector with cavities
@300:Part( PartTypes[i]=PartCategoryEnum(connector),
  PartTypes[i]=PartCategoryEnum(discrete) );
@301:PartVersion;
@302:PartView;
@303:Identifier( Id=IdentifierString("IMC16-2002X"),
  IdentificationContext=@51 )
Part_with_ID_and_PartView( @300, @303, @301, @302, @4);
@306:PartContactFeature( ElementOf=@302, Id="1", Definition=@1001 );
@307:PartContactFeature( ElementOf=@302, Id="2", Definition=@1001 );
@311:SingleOccurrence( Id=IdentifierString("P-CONN01"), Definition=@302 );
  #316:OccurrenceContactFeature( ElementOf=@311, Definition=@306 );
  #317:OccurrenceContactFeature( ElementOf=@311, Definition=@307 );
```

```
# Contact for Deutsch connector
@400:Part( PartTypes[i]=PartCategoryEnum(connector_contact),
  PartTypes[i]=PartCategoryEnum(discrete) );
@401:PartVersion;
@402:PartView;
@403:Identifier( Id=IdentifierString("6860-201-20278"),
  IdentificationContext=@51 )
Part_with_ID_and_PartView( @400, @403, @401, @402, @4);
@406:PartTerminal( ElementOf=@402, Id="j", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal" );
@407:PartContactFeature( ElementOf=@402, Id="o", Definition=@1002 );
@411:SingleOccurrence( Id=IdentifierString("P-CONN01-01"),
Definition=@402 );
  @412:OccurrenceTerminal( ElementOf=@411, Definition=@406 );
  @413:OccurrenceContactFeature( ElementOf=@411, Definition=@407 );
@421:SingleOccurrence( Id=IdentifierString("P-CONN01-02"),
Definition=@402 );
  @422:OccurrenceTerminal( ElementOf=@421, Definition=@406 );
  @423:OccurrenceContactFeature( ElementOf=@421, Definition=@407 );


# Cable
@500:Part( PartTypes[i]=PartCategoryEnum(cable),
  PartTypes[i]=PartCategoryEnum(raw_material_by_length) );
@501:PartVersion;
@502:PartView;
@503:Identifier( Id=IdentifierString("9962 009100"),
IdentificationContext=@52 )
Part_with_Name_and_PartView(@500, "Cable-D", @501, @502, @4);
@504:PartTransportFeature( ElementOf=@503, Id="CABLE01-WHT" );
@505:PartTransportFeature( ElementOf=@503, Id="CABLE01-BLK" );
@511:CableOccurrence( Id=IdentifierString("CABLE01"), Definition=@502,
Quantity=@512 );
  @512:NumericalValue( Unit=@8, ValueComponent=1.8 );
  @513:WireIdentification( ElementOf=@511, Id="CABLE01-WHT",
    Definition=@504, code=@70 );
  @514:WireIdentification( ElementOf=@511, Id="CABLE01-BLK",
    Definition=@505, code=@71 );
  @515:CableOccurrenceTerminalLocationGroup( ElementOf=@511, Name="end
a" );
    @521:CableOccurrenceTerminal( ElementOf=@515, AssociatedTransportFea-
ture=@513 );
    @523:CableOccurrenceTerminal( ElementOf=@515, AssociatedTransportFea-
ture=@514 );
  @516:CableOccurrenceTerminalLocationGroup( ElementOf=@511, Name="end
b" );
    @522:CableOccurrenceTerminal( ElementOf=@516, AssociatedTransportFea-
ture=@513 );
    @524:CableOccurrenceTerminal( ElementOf=@516, AssociatedTransportFea-
ture=@514 );


# Wire
@600:Part( PartTypes[i]=PartCategoryEnum(wire),
  PartTypes[i]=PartCategoryEnum(raw_material_by_length) );
@601:PartVersion;
@602:PartView;
```

```
@603:Identifier( Id=IdentifierString("83027 001100"),
  IdentificationContext=@52 )
Part_with_Name_and_PartView(@600, "Wire-C", @601, @602, @4);
@611:WireOccurrence( Id=IdentifierString("WIRE01"), Definition=@602,
  Quantity=@612 );
@612:NumericalValue( Unit=@8, ValueComponent=3.5 );
@613=WireIdentification( ElementOf=@611, DomainType="electrical" ... )
@614=WireOccurrenceTerminal( ElementOf=@611,
  AssociatedTransportFeature=@613, Name="end a" );
@615=WireOccurrenceTerminal( ElementOf=@611,
  AssociatedTransportFeature=@613, Name="end b" );


# Deutsch IMC Series cavity & contact shapes
@1000:ContactFeatureDefinitionFitRelationship( Name="Deutsch IMC Series
Size 20 fit",
  Relating=@1001, Related=@1002 );
@1001:ContactFeatureDefinition( Name="Deutsch IMC Series Size 20 cavity",
  ShapeFeatureType=cavity_profile );
@1002:ContactFeatureDefinition( Name="Deutsch IMC Series Size 20 pin",
  ShapeFeatureType=contact_profile );


# EWH-Assembly
@9000:Part;
@9001:PartVersion;
@9002:WiringHarnessAssemblyDesign;
@9003:ViewContext;
@9004:ViewContext;
part_with_WiringHarnessAssemblyDesign( @9000,"EWH Test-Case Connectivity2",
  @9001,@9002,@9003,@9004 );


@9101:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@111 );
@9102:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@211 );
@9103:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@311 );
@9104:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@411 ); # "P-
CONN01-01"
@9105:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@421 ); # "P-
CONN01-02"
@9106:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@511 );
@9107:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@611 );


# electrical connections
@9210:AssemblyShapeJoint( ElementOf=@9002,
JointType="soldered_connection" );
  @9211:AssemblyShapeJointItemRelationship( Relating=@9210, Related=@214 );
# PLUG01 / 0
  @9212:AssemblyShapeJointItemRelationship( Relating=@9210, Related=@521 );
# CABLE01-WHT / end a
@9220:AssemblyShapeJoint( ElementOf=@9002,
JointType="crimped_connection" );
  @9221:AssemblyShapeJointItemRelationship( Relating=@9220, Related=@412 );
# P-CONN01-01 / j
  @9222:AssemblyShapeJointItemRelationship( Relating=@9220, Related=@522 );
# CABLE01-WHT / end b
@9230:AssemblyShapeJoint( ElementOf=@9002,
JointType="soldered_connection" );
```

```
  @9231:AssemblyShapeJointItemRelationship( Relating=@9230, Related=@215 );
# PLUG01 / 1
  @9232:AssemblyShapeJointItemRelationship( Relating=@9230, Related=@523 );
# CABLE01-BLK / end a
  @9232:AssemblyShapeJointItemRelationship( Relating=@9230, Related=@614 );
# WIRE01 / end a
@9240:AssemblyShapeJoint( ElementOf=@9002,
JointType="crimped_connection" );
  @9241:AssemblyShapeJointItemRelationship( Relating=@9240, Related=@422 );
#  P-CONN01-02 / j
  @9242:AssemblyShapeJointItemRelationship( Relating=@9240, Related=@524 );
# CABLE01-BLK / end b
@9250:AssemblyShapeJoint( ElementOf=@9002,
JointType="crimped_connection" );
  @9251:AssemblyShapeJointItemRelationship( Relating=@9250, Related=@112 );
# LUG01 / 1
  @9252:AssemblyShapeJointItemRelationship( Relating=@9250, Related=@615 );
# WIRE01 / end b


# mechanical connections
@9260:AssemblyShapeJoint( ElementOf=@9002, JointType="snap_connection" );
  @9261:AssemblyShapeJointItemRelationship( Relating=@9260, Related=@316 );
# P-CONN01 / 1
  @9262:AssemblyShapeJointItemRelationship( Relating=@9260, Related=@317 );
# P-CONN01-01 / o
@9270:AssemblyShapeJoint( ElementOf=@9002, JointType="snap_connection" );
  @9271:AssemblyShapeJointItemRelationship( Relating=@9270, Related=@112 );
# P-CONN01 / 2
  @9272:AssemblyShapeJointItemRelationship( Relating=@9270, Related=@423 );
# P-CONN01-02 / o


sizeof(Part) = 7;
sizeof(PartVersion) = 7;
sizeof(PartView) = 6;
sizeof(WiringHarnessAssemblyDesign) = 1;
sizeof(NextAssemblyOccurrenceUsage) = 7;
sizeof(SingleOccurrence) = 5;
sizeof(WireOccurrence) = 1;
sizeof(CableOccurrence) = 1;


sizeof(PartTerminal) >= 4; # there might me interface terminals
sizeof(OccurrenceTerminal) >= 4; # there might me interface terminals
sizeof(PartContactFeature) = 3;
sizeof(OccurrenceContactFeature) = 4;
sizeof(WireColourBasedIdentificationCode) = 2;
sizeof(WireOccurrenceTerminal) = 2;
sizeof(CableOccurrenceTerminalLocationGroup) = 2;
sizeof(CableOccurrenceTerminal) = 4;
sizeof(AssemblyShapeJoint) = 7;
sizeof(AssemblyShapeJointItemRelationship) = 15;

);
```

4.7 EWH-Connectivity3

This test case consists of a "simple" coaxial cable with two coaxial connectors at the ends. The design is an extract from a bigger commercial product. The used parts are:

- coaxial connector **TC-400-SM-X** from Times Microwave Systems
  https://www.timesmicrowave.com/Products/Connectors/TC-400-SM-X/

- coaxial cable **PFLX400-500** from Rockwell Collins
  https://www.collinsaerospace.com/what-we-do/Business-Aviation/Flight-Deck/Avion-ics-Integration/Avionics-Integration-Products/Cables-And-Connectors/50-And-75-Ohm-Coaxial-And-Triaxial-Cables

FROM REF DES	TERM HRDWR	SHLD/TWST	FEP	WIRE NUMBER	TO REF DES	TERM HARDWR	TEP	WIRE PART NBR	LGTH
180A13P3-1	06-672-17	088-1CX1	ST	F48A-4	212DB06P1P3B-1	06-672-17	ST	PFLX400-500	279.79
180A13P3-C/S	FERRULE	088-1CX1	SC	088-1CX1	212DB06P1P3B-C/S	FERRULE	SC	PFLX400-500	279.79
(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)

Figure 7: EWH-Connectivity3 - Wire-list

Figure 7 list the inner core and the shield. The mapping of the columns to AP242 is:

- 1, 6: the from and to reference designators are composed by the *SingleOccurrence-Id* and the *Part/OccurrenceTerminal-Id*.
- 2, 7: the terminal hardware to which the connection is established. Below these *Parts* are identified as TC-400-SM-X_contact_pin and TC-400-SM-X_ferrule to have a consistent naming.
- 3: *CableOccurrence-Id*
- 4, 8: The From and To End Preparation. AP242ed2 does not provide explicit details for wire/cable end preparation, but support the kind of *AssemblyShapeJoints* to be used
  - ST – Strip and Tin, specified by *JointType soldered_connection*
  - SC – Strip and Crimp, specified by *JointType crimped_connection*
- 5: the *WireIdentification-Id* within the cable
- 9: the cable *Part-Id*
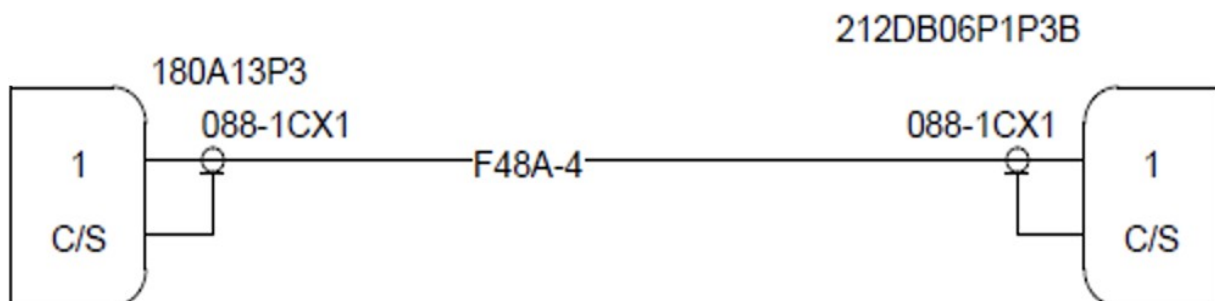- 10: the length of the *CableOccurrence* in inch

Figure 8: EWH-Connectivity3 - Wiring diagram

4.7.1 Coaxial Connector Model

Depending on the use case, a coaxial connector might be modeled in full details, but often a very simplified representation is sufficient.

- Coaxial and other connectors are typically bought as a kit that includes all the needed parts. While building an EWH they are assembled together while connecting them to the corresponding wires or cables. So strictly speaking a coaxial connector that shows up in a BOM is a collection of things, so a **CollectionDefinition** (@151) is a proper *PartView* for such a set.

- The pieces that make up a connector can also be represented as a virtual (sub) **AssemblyDefinition** *(@102)*, ignoring the wires/cables with which the joint terminals have to be connected before doing the real assembly. To link the *PartTerminals* of this assembly to the *OccurrenceTerminals* of the constituents **PartConnectivityDefinition** is used.

  - As there is no clear information provided on how the outer metal pieces "C/S" of Body, Ferrule and Shell are connected together with the shield of the coax cable somehow it between, a PartConnectivityDefinition is the right way to state that they are/will be connected.

  - To link the join_terminal of the connector_contact to the PartTerminal "1" of the connector assembly a PartConnectivityDefinition is not really the right choice. It would be better to use the attribute PartTerminal.PartDefinition to link to the corresponding OccurrenceTerminal that it reflect. An upcomming version of the standard should support his.

- Both connector_housing (@200) and connector_contact (@300) come with a joint- and an interface-terminal. A **PartConnectivityDefinition** is used to state that these two *PartTerminal* are somehow connected without telling any further detail.

The example chosen for this test case is the TC-400-SM-X BNC connector (@100). It is of a male type and suitable to be used for an LMR-400 coax cable (@600). The connector consists of five piece parts named and categorized as:

- body / connector_housing: @200,

- contact pin / connector_contact: @300,

- insulator / <no predefined part category available> : @400,

- shell / backshell: @500,

- ferrule / cable_ferrule: @600.

Because the connector is delivered as a kit there are no individual part numbers supplied for the piece parts; so we have to make up part numbers for this test case.

The electrical *AssemblyShapeJoints* of the coaxial connector to an end of a coaxial cable is realized by the *JointType* soldered_connection for the coaxial core and by the *JointType* *crimped_connection* for the crimped_connection.

NOTICE OF PROPRIETARY RIGHTS THIS DOCUMENT CONTAINS CONFIDENTIAL TECHNICAL DATA, INCLUDING TRADE SECRETS, PROPRIETARY TO TIMES MICROWAVE SYSTEMS. DISCLOSURE OF THIS DATA IS EXPRESSLY CONDITIONED UPON YOUR ASSENT THAT ITS USE IS LIMITED TO USE WITHIN YOUR COMPANY ONLY. ANY OTHER USE IS STRICTLY PROHIBITED WITHOUT THE PRIOR WRITTEN CONSENT OF TIMES MICROWAVE SYSTEMS.

SYM	REVISION DESCRIPTION	DFTM	DATE	APPD	DATE
A	RELEASED FOR PRODUCTION	n.n.n	4/14/14	J.D.B.	4/16/14

REMARKS:
1. CENTER PIN IS SOLDERED.
2. CRIMP FERRULE HEX. CRIMP SIZE .429"

RECOMMENDED CABLE STRIPPING DIM'S

TC-400-SM-X

MATERIALS AND PLATING		UNIT: MICRO-INCHES
BODY	BRASS C3604	ALBALOY PLATEING
CONTACT PIN	BRASS C3604	GOLD 50µ" / NICKEL/COPPER
INSULATOR	TEFLON MIL-P-19468	
SHELL	BRASS C3604	ALBALOY PLATEING
FERRULE	BRASS C3604	ALBALOY PLATEING

ELECTRICAL CHARACTERISTICS	
Impedance	50 Ω
Voltage rating	335V(rms)
Frequency range	0~6GHz
Dielectric withstanding voltage	500V
Contact resistance	Center contact≤3mΩ / Outer contact≤2mΩ
Insulation resistance	≥5000MΩ
Insertion loss	N/A
RF- leakage	N/A
VSWR	1.22 max (0~6G)
(3rd Intermodulation)	N/A

(MECHANICAL CHARACTERISTICS)	
Force to engage and disengage	N/A
Center contact retention force	N/A
Recommended coupling torque	7~10 in-lbs
Coupling nut retention force	60 lbs MIN.
Durability	≥ 500 cycles

ENVIRONMENTAL CHARACTERISTICS	
Temperature range	−55°C~ +125°C
Relative Humidity	MIL-STD-202,Method 106
Vibration	MIL-STD-202,Method 204,Cond D
Shock	MIL-STD-202,Method 213,Cond I
Corrosion	MIL-STD-202,Method 101,Cond B

MATL: SEE ABOVE

USED ON:

UNLESS OTHERWISE SPECIFIED
ALL DIMENSIONS ARE IN mm
1) EDGES .015 MAX
2) REMOVE ALL BURRS
3) SURFACE ROUGHNESS Ra1.6
4) UNITS mm
5) DIMENSIONS   TOLERANCES
DECIMALS   ANGLES
.XX ±0.3
.XXX ±0.2
ANGLES ±5°

DFTM	N. N. N
DATE	4/14/14
CHKD.	J. D. B.
DATE	4/16/14
APPD.	J. D. B.
DATE	4/16/14

TIMES MICROWAVE SYSTEMS

TC-400-SM-X
SMA Male
for LMR-400

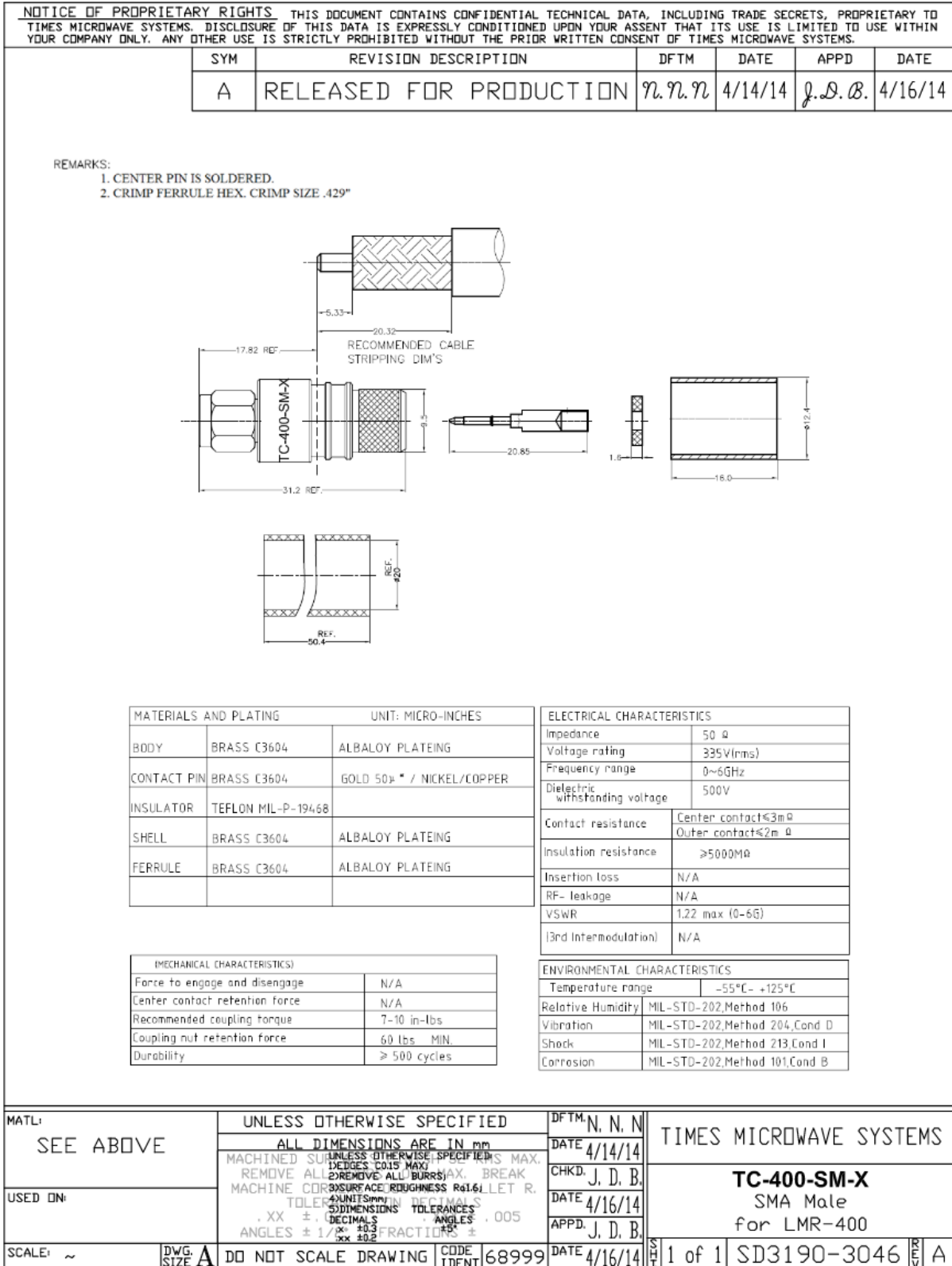SCALE: ~   DWG. SIZE A   DO NOT SCALE DRAWING   CODE IDENT 68999   SHT 1 of 1   SD3190-3046   REV A

Figure 9: Datasheet of Coax Connector TC-400-SM-X

Figure 11: Connector contact inserted into connector shell
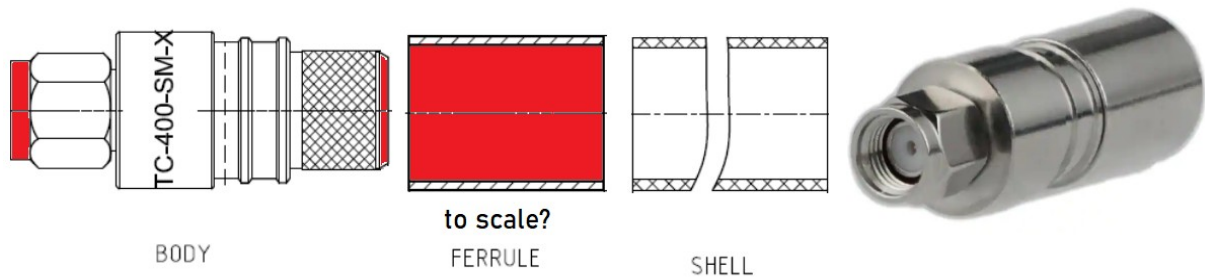
CONTACT PIN   INSULATOR   **Coax-Cable**

BODY   to scale? FERRULE   SHELL

Figure 10: Electrical contacts in red, mechanical contacts in green

General references:

```
TC-400-SM-X
https://www.timesmicrowave.com/Products/Connectors/TC-400-SM-X
=> with 3D STEP files as single piece part
https://eu.mouser.com/ProductDetail/Amphenol-Times-Microwave-Systems/TC-
400-SM-X?qs=OTrKUuiFdkYP4uoBuXBVRA==
https://eu.mouser.com/datasheet/2/18/3190-3046A-1826531.pdf
https://www.digikey.com/en/products/detail/amphenol-times-microwave-sys-
tems/TC-400-SM-X/9644101


PFLX400-500RF coaxial cable
https://www.collinsaerospace.com/-/media/project/collinsaerospace/
collinsaerospace-website/product-assets/marketing/0-9/50-75-ohm-coaxial-ca-
bles/1_50-74-coaxial-cables/30_pflx400-500_rf_coaxial_cable_datasheet.pdf?
rev=a569a39747e94bbdb1b5ee7995cacd5d&hash=10CD09CE594BB2441C31797F75708B1C
https://www.pasternack.com/50-ohm-low-loss-flexible-lmr400-pe-jacket-dou-
ble-shielded-black-lmr-400-p.aspx
```

Formal test-case specification:

```
Test EWH-Connectivity3 (

@4:ViewContext;
@5:ViewContext;
@8:Unit( Name=ClassString("inch"), Quantity=ClassString("length") );

@50:Organization( name="Times Microwaves System")
@52:Organization( name="Rockwell Collins")

@70:WireColourBasedIdentificationCode( Id="core" );
@71:WireColourBasedIdentificationCode( Id="shield" );

# Connector TC-400-SM-X, MANDATORY
@100:Part( Name="RF Connectors",
```

```
      Description="Coaxial Connectors SMA-Male (plug) crimp connector; no braid
trim"
  PartTypes[i]=PartCategoryEnum(shielded_connector),
  PartTypes[i]=PartCategoryEnum(connector_kit),
  PartTypes[i]=PartCategoryEnum(connector),
  PartTypes[i]=PartCategoryEnum(discrete) );
@101:PartVersion;

# Mandatory assembly view needed for harness
@102:AssemblyDefinition;
@103:Identifier( Id=IdentifierString("TC-400-SM-X"),
IdentificationContext=@50 )
Part_with_ID_and_PartView(@100, @103, @101, @102, @4);

# interface aspects of assembly
@104:PartTerminal( ElementOf=@102, Id="1", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal" );
@105:PartTerminal( ElementOf=@102, Id="C/S", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal" );

# occurrence level
@111:SingleOccurrence( Id=IdentifierString("connector 1"),
Definition=@102 ); # connector 1
@112:OccurrenceTerminal( ElementOf=@111, Definition=@104 );
@113:OccurrenceTerminal( ElementOf=@111, Definition=@105 );
@121:SingleOccurrence( Id=IdentifierString("connector 2"),
Definition=@102 ); # connector 2
@122:OccurrenceTerminal( ElementOf=@111, Definition=@104 );
@123:OccurrenceTerminal( ElementOf=@111, Definition=@105 );

# internal sub-assembly aspects of assembly
@162:NextAssemblyOccurrenceUsage( Relating=@102, Related=@211 ); # body /
connector_housing
@163:NextAssemblyOccurrenceUsage( Relating=@102, Related=@311 ); # pin /
connector_contact
@164:NextAssemblyOccurrenceUsage( Relating=@102, Related=@411 ); # insula-
tor
@165:NextAssemblyOccurrenceUsage( Relating=@102, Related=@511 ); # shell /
backshell
@166:NextAssemblyOccurrenceUsage( Relating=@102, Related=@611 ); #
ferrule / cable_ferrule

# possible future alternaitve for #174 below
#   @104:PartTerminal( ElementOf=@102, Id="1", DomainType="electrical",
#       InterfaceOrJoinTerminal="join_terminal", PartDefinition=@312 );
# Terminal 1 is connected with contact-pin
@174:PartConnectivityDefinition( ConnectedTerminals=(@104, @312) );
# terminal C/S is connected with connector_housing, backshell and
cable_ferrule
@175:PartConnectivityDefinition( ConnectedTerminals=(@105, @212, @512,
@612) );

# Optional collection view for bill of material, stock etc.
@151:CollectionDefinition
```

```
Part_with_ID_and_PartView(@100, @103, @101, @151, @4); // not fully correct
yet
@152:NextAssemblyOccurrenceUsage( Relating=@151, Related=@211 );
@153:NextAssemblyOccurrenceUsage( Relating=@151, Related=@311 );
@154:NextAssemblyOccurrenceUsage( Relating=@151, Related=@411 );
@155:NextAssemblyOccurrenceUsage( Relating=@151, Related=@511 );
@156:NextAssemblyOccurrenceUsage( Relating=@151, Related=@611 );


# Connector housing, OPTIONAL
@200:Part(
  PartTypes[i]=PartCategoryEnum(connector_housing),
  PartTypes[i]=PartCategoryEnum(discrete) );
@201:PartVersion;
@202:PartView;
@203:Identifier( Id=IdentifierString("TC-400-SM-X_body"), Identification-
Context=@50 )
Part_with_ID_and_PartView(@200, @203, @201, @202, @4);
@204:PartTerminal( ElementOf=@202, Id="C/S", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal" );
@205:PartTerminal( ElementOf=@202, Id="it", DomainType="electrical",
  InterfaceOrJoinTerminal="interface_terminal" );
@206:PartConnectivityDefinition( ConnectedTerminals=(@204, @205) );
#
@211:SingleOccurrence( Id=IdentifierString("housing"), Definition=@202 );
@212:OccurrenceTerminal( ElementOf=@211, Definition=@204 );


# Connector contact-pint, OPTIONAL
@300:Part(
  PartTypes[i]=PartCategoryEnum(connector_contact),
  PartTypes[i]=PartCategoryEnum(discrete) );
@301:PartVersion;
@302:PartView;
@303:Identifier( Id=IdentifierString("TC-400-SM-X_contact_pin"), Identifi-
cationContext=@50 )
Part_with_ID_and_PartView(@300, @303, @301, @302, @4);
@304:PartTerminal( ElementOf=@302, Id="", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal" );
@305:PartTerminal( ElementOf=@302, Id="", DomainType="electrical",
  InterfaceOrJoinTerminal="interface_terminal" );
@306:PartConnectivityDefinition( ConnectedTerminals=(@304, @305) );
#
@311:SingleOccurrence( Id=IdentifierString("???"), Definition=@302 );
@312:OccurrenceTerminal( ElementOf=@311, Definition=@304 );


# Connector insulator, OPTIONAL
@400:Part(
  PartTypes[i]=PartCategoryEnum(discrete) );
@401:PartVersion;
@402:PartView;
@403:Identifier( Id=IdentifierString("TC-400-SM-X_insulator"), Identifica-
tionContext=@50 )
Part_with_ID_and_PartView(@400, @403, @401, @402, @4);
@411:SingleOccurrence( Id=IdentifierString("???"), Definition=@402 );
```

```
# Connector shell, OPTIONAL
@500:Part(
  PartTypes[i]=PartCategoryEnum(backshell),
  PartTypes[i]=PartCategoryEnum(discrete) );
@501:PartVersion;
@502:PartView;
@503:Identifier( Id=IdentifierString("TC-400-SM-X_shell"), Identification-
Context=@50 )
Part_with_ID_and_PartView(@500, @503, @501, @502, @4);
@504:PartTerminal( ElementOf=@502, Id="", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal" );
@511:SingleOccurrence( Id=IdentifierString("???"), Definition=@502 );
@512:OccurrenceTerminal( ElementOf=@511, Definition=@504 );


# Connector ferrule, OPTIONAL
@600:Part(
  PartTypes[i]=PartCategoryEnum(cable_ferrule),
  PartTypes[i]=PartCategoryEnum(discrete) );
@601:PartVersion;
@602:PartView;
@603:Identifier( Id=IdentifierString("TC-400-SM-X_ferrule"), Identifica-
tionContext=@50 )
Part_with_ID_and_PartView(@600, @603, @601, @602, @4);
@604:PartTerminal( ElementOf=@602, Id="", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal" );
@611:SingleOccurrence( Id=IdentifierString("???"), Definition=@602 );
@612:OccurrenceTerminal( ElementOf=@611, Definition=@604 );


# Cable, MANDATORY
@700:Part(
  PartTypes[i]=PartCategoryEnum(cable),
  PartTypes[i]=PartCategoryEnum(raw_material_by_length) );
@701:PartVersion;
@702:PartView;
@703:Identifier( Id=IdentifierString("PFLX400-500"),
IdentificationContext=@52 )
Part_with_ID_and_PartView(@700, @703, @701, @702, @4);
@704:PartTransportFeature( ElementOf=@503, Id="core" );
@705:PartTransportFeature( ElementOf=@503, Id="shield" );
@711:CableOccurrence( Id=IdentifierString("088-1CX1"), Definition=@702,
Quantity=@712 );
  @712:NumericalValue( Unit=@8, ValueComponent=279.79 );
  @713:WireIdentification( ElementOf=@711, Id="F48A-4", Definition=@704,
code=@70 );
  @714:WireIdentification( ElementOf=@711, Id="088-1CX1", Definition=@705,
code=@71 );
  @715:CableOccurrenceTerminalLocationGroup( ElementOf=@711, Name="end
a" );
    @721:CableOccurrenceTerminal( ElementOf=@715, AssociatedTransportFea-
ture=@713); # cable1/core / end a
    @723:CableOccurrenceTerminal( ElementOf=@715, AssociatedTransportFea-
ture=@714); # cable1/shield / end a
  @716:CableOccurrenceTerminalLocationGroup( ElementOf=@711, Name="end
b" );
```

```
    @722:CableOccurrenceTerminal( ElementOf=@716, AssociatedTransportFea-
ture=@713); # cable1/core / end b
    @724:CableOccurrenceTerminal( ElementOf=@716, AssociatedTransportFea-
ture=@714); # cable1/shield / end a


# EWH-Assembly
@9000:Part;
@9001:PartVersion;
@9002:WiringHarnessAssemblyDesign;
@9003:ViewContext;
@9004:ViewContext;
part_with_WiringHarnessAssemblyDesign( @9000,
  "EWH Test-Case Connectivity3-A",@9001,@9002,@9003,@9004 );
@9101:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@111 );  # Sin-
gleOccurrence "connector 1"
@9102:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@121 );  # Sin-
gleOccurrence "connector 2"
@9103:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@711 );  # Ca-
bleOccurrence "088-1CX1"
@9210:AssemblyShapeJoint( ElementOf=@9002,
JointType="soldered_connection" );
  @9211:AssemblyShapeJointItemRelationship( Relating=@9210, Related=@112 );
# connector 1/ 1
  @9212:AssemblyShapeJointItemRelationship( Relating=@9210, Related=@721 );
# 088-1CX1/core / end a
@9220:AssemblyShapeJoint( ElementOf=@9002,
JointType="soldered_connection" );
  @9221:AssemblyShapeJointItemRelationship( Relating=@9220, Related=@122 );
# connector 2/ 1
  @9222:AssemblyShapeJointItemRelationship( Relating=@9220, Related=@722 );
# 088-1CX1/core / end b
@9230:AssemblyShapeJoint( ElementOf=@9002,
JointType="crimped_connection" );
  @9232:AssemblyShapeJointItemRelationship( Relating=@9230, Related=@113 );
# connector 1/ C/S
  @9232:AssemblyShapeJointItemRelationship( Relating=@9230, Related=@723 );
# 088-1CX1/shield / end a
@9240:AssemblyShapeJoint( ElementOf=@9002,
JointType="crimped_connection" );
  @9241:AssemblyShapeJointItemRelationship( Relating=@9240, Related=@123 );
# connector 2/ C/S
  @9242:AssemblyShapeJointItemRelationship( Relating=@9240, Related=@724 );
# 088-1CX1/shield / end b


);
```

4.8 EWH-Connectivity4

This test case is similar to the previous connectivity test cases. On the left side we have a connector of type PT06A-10-6S with 6 non-removable contact, while on the right side we have a connector with 6 cavities into which either connector-contacts or if not used sealing plugs are to be inserted. Between the two connectors there is a single wire and a shielded cable with two cores. On the left side the shield of the cable is connected to connector terminal "C" by a shield sleeve of type M83519/2-8. A special "Banding and Shrink Boot Adapter" of type 440DS031NF1002-3 is screwed onto the back-shell of the adapter to guide the wire, cable and shield sleeve. On the right side the shield of the cable is directly connected to the electrified back-shell of the connector (indicated as "B/S").

Both connectors are delivered as a connector set that have to be assembled while manufacturing the harness. The PT06A-10-6S connector is delivered as three piece parts, the connector_housing, the connector_insert with integrated soldering pins, and the backshell. The MS27484T8F35SB is delivered in likely 8 parts,
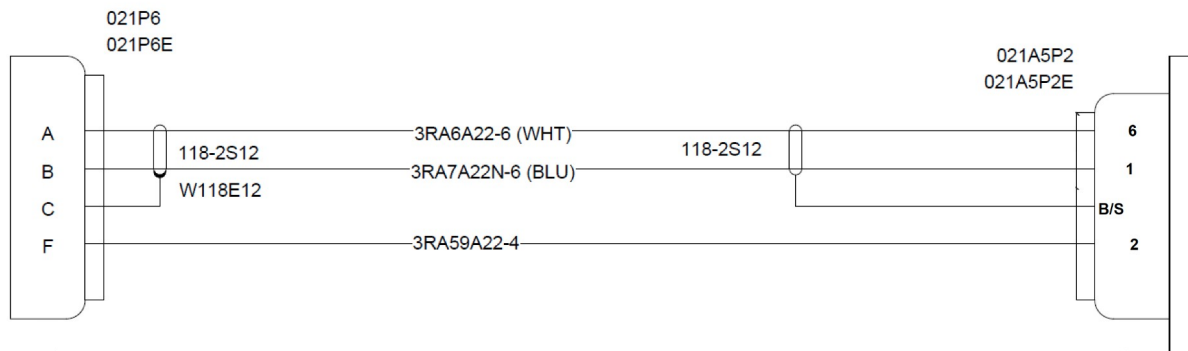

Figure 12: EWH-Connectivity4 - Wiring diagram

Name	Part Number	Nomenclature	Vendor	Test Case
W118E12	M83519/2-8	Shield Sleeve		@100
021P6	PT06A-10-6S	connector	Amphenol	@200
021P6E	440DS031NF1002-3	backshell	Glenair	@300
021A5P2	MS27484T8F35SB	RFI grounding plug	Conesys	@400
021A5P2E	PART OF 021A5P2	Backshell	- dito -	@500
118-2S12	04049A22A02J24	Two Conductor Shielded Cable		@600
3RA59A22-4	04034-22-9	Wire		@700
???	680-116-22	Dummy Contact Sealing Plug	Glenair	@800

Figure 13: EWH-Connectitvity4 - Part List

Figure 14: M83519/2-8: Solder Sleeves & Shield Tubing S-SLEEVE SHLD TRMNTR 22 AWG

Figure 15: Connector: PT06A-10-6S with 6 solder contacts of size 20

Breakdown of MS27484T8F35SB

- MS: Mil. Prefix

- 27484: RFI grounding plug

- T: With accessory thread

- 8: Shell Size = 8

- F: Finish = Aluminum shell, electroless nickel finish

- 35: Insert Arrangement

- S: S=Socket

- B: Polarization / Keying = B

Sednal Technologies Twistrip Cable Die Cross Reference Sheet   Updated: 2014-03-31

Manufacturer Cable Part Number	Twistrip Cable Die Part Number	Number of Conductors	Conductor Gauge	Cable Jacket Major OD	Cable Jacket Minor OD	Cable IPR (Pitch)	Cable Jacket Thickness	Shield Thickness	Conductor OD
04049A20A02J24	TDA-2-5836/5R-4	2	20	0.115	0.072	0.99	0.008	0.004	0.050
04049A22A02J24	TDA-2-5435/5R-5	2	22	0.107	0.070	0.98	0.008	0.003	0.040

Figure 16: Two Conductor Shielded Cable 04049A22A02J24

Formal test-case specification:

Test EWH-Connectivity4 (

@4:ViewContext;
@5:ViewContext;
@8:Unit(Name=ClassString("metre"), Quantity=ClassString("length"));

@50:Organization(name="TE Connectivity / Raychem")
@51:Organization(name="Amphenol")
@52:Organization(name="Glenair")

@70:WireColourBasedIdentificationCode(Id="white");
@71:WireColourBasedIdentificationCode(Id="blue");

Shield Sleeve
@100:Part(Name="Shield Sleeve",
  Description="Solder Sleeves & Shield Tubing S-SLEEVE SHLD TRMNTR 22 AWG"
  PartTypes[i]=PartCategoryEnum(shield_connector),
  PartTypes[i]=PartCategoryEnum(discrete));
@101:PartVersion;
@102:PartView;
@103:Identifier(Id=IdentifierString("M83519/2-8"), IdentificationContext=@50)
Part_with_ID_and_PartView(@100, @103, @101, @102, @4);
@104:PartTerminal(ElementOf=@102, Id="Shield-Terminal", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal");
@105:PartTerminal(ElementOf=@102, Id="Wire-Terminal", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal");
@111:SingleOccurrence(Id=IdentifierString("W118E12"), Definition=@102);
    @112:OccurrenceTerminal(ElementOf=@111, Definition=@104);
    @113:OccurrenceTerminal(ElementOf=@111, Definition=@105);

connector
@200:Part(name="Miniature Cylindrical Connectors"
    Description="Bayonet Coupling with Solder Contact Termination"
    PartTypes[i]=PartCategoryEnum(connector),
    PartTypes[i]=PartCategoryEnum(discrete));
@201:PartVersion;
@202:PartView;
@203:Identifier(Id=IdentifierString("PT06A-10-6S"), IdentificationContext=@51)
Part_with_ID_and_PartView(@200, @203, @201, @202, @4);
@204:PartTerminal(ElementOf=@202, Id="A", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal");
@205:PartTerminal(ElementOf=@202, Id="B", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal");
@206:PartTerminal(ElementOf=@202, Id="C", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal");
@207:PartTerminal(ElementOf=@202, Id="D", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal");

@208:PartTerminal(ElementOf=@202, Id="E", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal");
@209:PartTerminal(ElementOf=@202, Id="F", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal");
@210:PartFeature(ElementOf=@202, Id="Thread", DomainType="mechanical");
@211:SingleOccurrence(Id=IdentifierString("021P6"), Definition=@202);
  @214:OccurrenceTerminal(ElementOf=@211, Definition=@204);
  @215:OccurrenceTerminal(ElementOf=@211, Definition=@205);
  @216:OccurrenceTerminal(ElementOf=@211, Definition=@206);
  @217:OccurrenceTerminal(ElementOf=@211, Definition=@207);
  @218:OccurrenceTerminal(ElementOf=@211, Definition=@208);
  @219:OccurrenceTerminal(ElementOf=@211, Definition=@209);
  @220:OccurrenceTerminal(ElementOf=@211, Definition=@210);

backshell
@300:Part(
    PartTypes[i]=PartCategoryEnum(backshell),
    PartTypes[i]=PartCategoryEnum(discrete));
@301:PartVersion;
@302:PartView;
@303:Identifier(Id=IdentifierString("440DS031NF1002-3"), IdentificationContext=@52)
Part_with_ID_and_PartView(@300, @303, @301, @302, @4);
@307:PartContactFeature(ElementOf=@302, Id="Thread");
@308:PartFeature(ElementOf=@302, Id="Segment-Opening");
@311:SingleOccurrence(Id=IdentifierString("021P6E"), Definition=@302);
    @317:OccurrenceContactFeature(ElementOf=@311, Definition=@307);
    @318:OccurrenceShapeFeature(ElementOf=@311, Definition=@308);

connector
@400:Part(PartTypes[i]=PartCategoryEnum(connector),
PartTypes[i]=PartCategoryEnum(discrete));
@401:PartVersion;
@402:PartView;
@403:Identifier(Id=IdentifierString("MS27484T8F35SB"), IdentificationContext=@51)
Part_with_ID_and_PartView(@400, @403, @401, @402, @4);
@404:PartTerminal(ElementOf=@402, Id="1", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal");
@405:PartTerminal(ElementOf=@402, Id="2", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal");
@406:PartTerminal(ElementOf=@402, Id="3", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal");
@407:PartTerminal(ElementOf=@402, Id="4", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal");
@408:PartTerminal(ElementOf=@402, Id="5", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal");
@409:PartTerminal(ElementOf=@402, Id="6", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal");
@410:PartFeature(ElementOf=@402, Id="Tread"); # contact feature

```
@411:SingleOccurrence( Id=IdentifierString("021A5P2"), Definition=@402 );
  @414:OccurrenceTerminal( ElementOf=@411, Definition=@404 );
  @415:OccurrenceTerminal( ElementOf=@411, Definition=@405 );
  @416:OccurrenceTerminal( ElementOf=@411, Definition=@406 );
  @417:OccurrenceTerminal( ElementOf=@411, Definition=@407 );
  @418:OccurrenceTerminal( ElementOf=@411, Definition=@408 );
  @419:OccurrenceTerminal( ElementOf=@411, Definition=@409 );
  @420:OccurrenceTerminal( ElementOf=@411, Definition=@410 );

# electrified backshell
@500:Part(
    PartTypes[i]=PartCategoryEnum(electrified_backshell),
    PartTypes[i]=PartCategoryEnum(discrete) );
@501:PartVersion;
@502:PartView;
@503:Identifier( Id=IdentifierString("PART OF 021A5P2"), IdentificationContext=@51 )
Part_with_ID_and_PartView(@500, @503, @501, @502, @4);
@506:PartTerminal( ElementOf=@502, Id="BS", DomainType="electrical",
  InterfaceOrJoinTerminal="join_terminal" );
@507:PartContactFeature( ElementOf=@502, Id="Thread");
@508:PartFeature( ElementOf=@502, Id="Segment-Opening");
@511:SingleOccurrence( Id=IdentifierString("021A5P2E"), Definition=@502 );
    @516:OccurrenceTerminal( ElementOf=@511, Definition=@506 );
    @517:OccurrenceContactFeature( ElementOf=@511, Definition=@507 );
    @518:OccurrenceShapeFeature( ElementOf=@511, Definition=@508 );

# Cable
@600:Part( PartTypes[i]=PartCategoryEnum(cable),
PartTypes[i]=PartCategoryEnum(raw_material_by_length) );
@601:PartVersion;
@602:PartView;
@603:Identifier( Id=IdentifierString("04049A22A02J24"), IdentificationContext=@52 )
Part_with_ID_and_PartView(@600, @603, @601, @602, @4);
@604:PartTransportFeature( ElementOf=@603, Id="white" );
@605:PartTransportFeature( ElementOf=@603, Id="blue" );
@606:PartTransportFeature( ElementOf=@603, Id="shield" );
@611:CableOccurrence( Id=IdentifierString("118-2S12"), Definition=@602, Quantity=@612 );
    @612:NumericalValue( Unit=@8, ValueComponent=1.8 );
    @613:WireIdentification( ElementOf=@611, Id="3RA6A22-6 (WHT)", Definition=@604,
code=@70 );
    @614:WireIdentification( ElementOf=@611, Id="3RA7A22N-6 (BLU)", Definition=@605,
code=@71 );
    @615:WireIdentification( ElementOf=@611, Id="118-2S12", Definition=@606,
code=@71 );
    @615:CableOccurrenceTerminalLocationGroup( ElementOf=@611, Name="end a" );
        @621:CableOccurrenceTerminal( ElementOf=@615,
AssociatedTransportFeature=@613);
        @622:CableOccurrenceTerminal( ElementOf=@615,
AssociatedTransportFeature=@614);
```

```
      @623:CableOccurrenceTerminal( ElementOf=@615,
AssociatedTransportFeature=@615);
   @616:CableOccurrenceTerminalLocationGroup( ElementOf=@611, Name="end b" );
      @624:CableOccurrenceTerminal( ElementOf=@616,
AssociatedTransportFeature=@613);
      @625:CableOccurrenceTerminal( ElementOf=@616,
AssociatedTransportFeature=@614);
      @626:CableOccurrenceTerminal( ElementOf=@616,
AssociatedTransportFeature=@615);


# Wire
@700:Part( PartTypes[i]=PartCategoryEnum(wire),
  PartTypes[i]=PartCategoryEnum(raw_material_by_length) );
@701:PartVersion;
@702:PartView;
@703:Identifier( Id=IdentifierString("04034-22-9"), IdentificationContext=@52 )
Part_with_ID_and_PartView(@700, @703, @701, @702, @4);
@711:WireOccurrence( Id=IdentifierString("3RA59A22-4"), Definition=@602, Quantity=@712
);
   @712:NumericalValue( Unit=@8, ValueComponent=3.5 );
   @713:WireIdentification( ElementOf=@611, DomainType="electrical" ... )
   @714:WireOccurrenceTerminal( ElementOf=@711, AssociatedTransportFeature=@713,
   Name="end a" );
   @715:WireOccurrenceTerminal( ElementOf=@711, AssociatedTransportFeature=@713,
   Name="end b" );


# EWH-Assembly
@9000:Part;
@9001:PartVersion;
@9002:WiringHarnessAssemblyDesign;
@9003:ViewContext;
@9004:ViewContext;
part_with_WiringHarnessAssemblyDesign( @9000,
  "EWH Test-Case Connectivity1",@9001,@9002,@9003,@9004 );


@9101:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@111 ); # left/
shield_connector
@9102:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@211 ); # left connec-
tor
@9103:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@311 ); # left backshell
@9104:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@411 ); # right connec-
tor
@9105:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@511 ); # right electri-
fied_backshell
@9106:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@611 ); # cable
@9107:NextAssemblyOccurrenceUsage( Relating=@9002, Related=@711 ); # wire


# connections
@9210:AssemblyShapeJoint( ElementOf=@9002 );
```

@9211:AssemblyShapeJointItemRelationship(Relating=@9210, Related=@621); # left white/end a

@9212:AssemblyShapeJointItemRelationship(Relating=@9210, Related=@214); # left connector/A

@9220:AssemblyShapeJoint(ElementOf=@9002);

@9221:AssemblyShapeJointItemRelationship(Relating=@9220, Related=@622); # left blue/end a

@9222:AssemblyShapeJointItemRelationship(Relating=@9220, Related=@215); # left connector/B

@9230:AssemblyShapeJoint(ElementOf=@9002);

@9231:AssemblyShapeJointItemRelationship(Relating=@9230, Related=@623); # left shield/end a

@9232:AssemblyShapeJointItemRelationship(Relating=@9230, Related=@112); # shield_connector

@9240:AssemblyShapeJoint(ElementOf=@9002);

@9241:AssemblyShapeJointItemRelationship(Relating=@9240, Related=@113); # shield_connector

@9242:AssemblyShapeJointItemRelationship(Relating=@9240, Related=@216); # left connector/C

@9250:AssemblyShapeJoint(ElementOf=@9002);

@9251:AssemblyShapeJointItemRelationship(Relating=@9250, Related=@714); # wire/ end a

@9252:AssemblyShapeJointItemRelationship(Relating=@9250, Related=@219); # left connector/F

@9260:AssemblyShapeJoint(ElementOf=@9002);

@9251:AssemblyShapeJointItemRelationship(Relating=@9260, Related=@317); # left backshell/Thread

@9252:AssemblyShapeJointItemRelationship(Relating=@9260, Related=@220); # left connector/Thread

@9310:AssemblyShapeJoint(ElementOf=@9002);

@9311:AssemblyShapeJointItemRelationship(Relating=@9310, Related=@624); # right white/end b

@9312:AssemblyShapeJointItemRelationship(Relating=@9310, Related=@419); # right connector/6

@9320:AssemblyShapeJoint(ElementOf=@9002);

@9321:AssemblyShapeJointItemRelationship(Relating=@9320, Related=@625); # right blue/end b

@9322:AssemblyShapeJointItemRelationship(Relating=@9320, Related=@414); # right connector/1

@9330:AssemblyShapeJoint(ElementOf=@9002);

@9331:AssemblyShapeJointItemRelationship(Relating=@9330, Related=@626); # right shield/end b

@9332:AssemblyShapeJointItemRelationship(Relating=@9330, Related=@516); # right electrified_backshell

@9340:AssemblyShapeJoint(ElementOf=@9002);

@9341:AssemblyShapeJointItemRelationship(Relating=@9340, Related=@715); # wire/ end b

@9342:AssemblyShapeJointItemRelationship(Relating=@9340, Related=@415); # right connector/2

@9350:AssemblyShapeJoint(ElementOf=@9002);

@9351:AssemblyShapeJointItemRelationship(Relating=@9350, Related=@517); # right electrified_backshell/Thread

@9352:AssemblyShapeJointItemRelationship(Relating=@9350, Related=@420); # right connector/Thread

);