



## **Recommended Practices for Colours and Layers September 24, 2001**

### Contacts:

Jochen Boy  
PROSTEP AG  
Taunusstraße 11, D-80807  
München  
+49 89 35020 210  
[jochen.boy@prostep.com](mailto:jochen.boy@prostep.com)

Phil Rosche  
PDES, Inc.  
5300 International Blvd.  
North Charleston, SC 29418  
843-760-3593  
[rosche@aticorp.org](mailto:rosche@aticorp.org)



This document has been developed by PDES, Inc. and ProSTEP iViP to aid in accelerating implementation of STEP standards. It is provided on an as-is basis with no warranties, guarantees, or representations of any kind, whether express or implied, including any warranty of merchantability or fitness for a particular purpose. This documentation is expressly not guaranteed to be error-free. PDES, Inc. and ProSTEP iViP shall have no liability for any damages including, but not limited to, direct, indirect, consequential, incidental, special, exemplary, punitive, and/or incidental damages arising out of, or relating to, the use of this documentation, even if PDES, Inc. and ProSTEP iViP have been advised of the possibility of such damages. This document, or excerpts from it, may be freely distributed provided that PDES, Inc. and ProSTEP iViP are acknowledged to be the original authors.

## Table of Contents

<b>1</b>	<b>Introduction .....</b>	<b>3</b>
<b>2</b>	<b>Scope .....</b>	<b>3</b>
<b>3</b>	<b>Layers .....</b>	<b>4</b>
<b>4</b>	<b>Colours .....</b>	<b>5</b>
4.1	COLOURING OF SOLIDS AND SURFACES .....	5
4.2	OVERRIDING SURFACE COLOURS .....	8
4.3	OVERRIDING EDGE COLOURS .....	9
<b>5</b>	<b>Part 21 file example .....</b>	<b>10</b>
5.1	CONFORMING TO AP214 .....	11
5.2	CONFORMING TO AP203 + EXTENSIONS .....	11
<b>6</b>	<b>Availability of implementation schemas .....</b>	<b>22</b>
6.1	AP214 .....	22
6.2	AP203 + EXTENSIONS .....	22

## List of Figures

FIGURE 1: LAYER REPRESENTATION .....	4
FIGURE 2: EXCHANGE FILE SEGMENT FOR LAYER SPECIFICATION (FROM FILE EXAMPLE) .....	4
FIGURE 3: COLOUR REPRESENTATION .....	5
FIGURE 4: EXCHANGE FILE SEGMENT FOR COLOUR SPECIFICATION (FROM FILE EXAMPLE) .....	7
FIGURE 5: REPRESENTATION OF OVERRIDING FACE COLOUR .....	8
FIGURE 6: EXCHANGE FILE SEGMENT FOR OVERRIDING FACE COLOUR (FROM FILE EXAMPLE) .....	9
FIGURE 7: REPRESENTATION OF OVERRIDING EDGE COLOUR .....	9
FIGURE 8: EXCHANGE FILE SEGMENT FOR OVERRIDING EDGE COLOUR (FROM FILE EXAMPLE) .....	10
FIGURE 9: GEOMETRY OF PART 21 FILE EXAMPLE .....	10

## 1 Introduction

This document describes the recommended practices for implementing to ability to exchange Colours and Layers via the STEP standard. The approaches in this document have been agreed by the CAx Implementors Forum and incorporated into the pre and post processors taking part in the interoperability testing.

If questions arise during your review of this document, please contact any of the persons listed on the cover page of this document

## 2 Scope

The scope is the definition of layers and the specification of colours for both solids and topologically bounded surfaces, as well as their constituent shape elements, and also geometrically bounded wireframe and surface data. The document currently covers how a STEP implementation, both pre and post processors, would support the specification of colours and layers and the assignment of those to the designated portions of the part shape.

**Note** that this scope does not currently cover the colouring of Instances in an assembly. This will be covered at a later date.

### 3 Layers

A layer is a general structure for the collection of geometric and annotation elements. Layers shall not be nested, i.e.; a layer cannot be put on another layer.

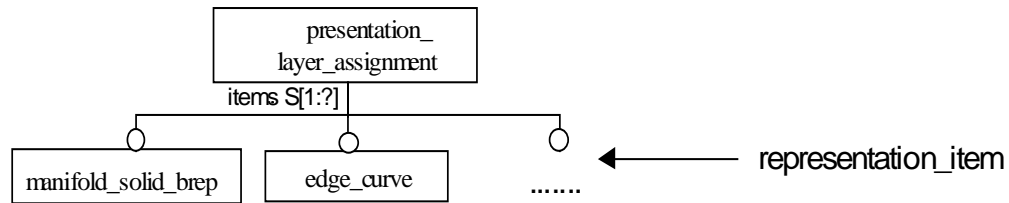


Figure 1: LAYER REPRESENTATION

The **presentation\_layer\_assignment** entity collects all items that are on the same layer in its assigned items attribute. These items are **representation\_items**. Note that this ignores the informal proposition on **layered\_item** provided in Part46. This approach was chosen by the implementor forum in order to reduce exchange file size by removing the need for **layered\_items** to be **styled\_items**, although a STEP File which applies layers through **styled\_items** is not deemed to be invalid.

In addition, a whole layer can be set to invisible, using an instance of **invisibility** referencing the **presentation\_layer\_assignment**. In general, all unstyled items are handled as being visible. In case of intended invisibility, the invisible objects shall be declared explicitly as invisible. This may happen directly for each object or indirectly via declaring the layer that contains the objects as invisible.

<b>presentation_layer_assignment</b>	Assigns an identifier to a set of <b>representation_items</b> . This set contains the pictures, or elements of pictures, that are assigned to a layer.
name	The identification of the layer represented by this entity.
description	A textual description of the layer.
assigned_items	The set of items assigned to the layer.
<b>invisibility</b>	Specifies that a collection of elements assigned to a layer by <b>presentation_layer_assignment</b> shall not be present.
invisible_items	A set of <b>styled_items</b> that are denoted as being invisible.

```

#225=PRESENTATION_LAYER_ASSIGNMENT('010','layer 010',(#206));
#206=SHELL_BASED_SURFACE_MODEL('#206',(#205));
  
```

Figure 2: EXCHANGE FILE SEGMENT FOR LAYER SPECIFICATION (FROM FILE EXAMPLE)

## 4 Colours

### 4.1 Colouring of solids, topologically and geometrically bound surfaces and wireframe

The colouring information is inherited from the solid or surface to its constituent faces and edges. The priority list for colouring elements in topological models is as follows:

1. solids (**manifold\_solid\_brep**, **brep\_with\_voids**) *OR*  
surfaces (**shell\_based\_surface\_model**)
- 1a. surfaces only (**open\_shell**, **closed\_shell**)
2. faces / edges
3. geometric surfaces / curves

Solids and surfaces may be coloured by using **fill\_area\_style\_colour**, where faces lying on a solid should be styled by overriding the solid style, as are the edges. The edges are treated as being independent of the face due to them potentially being used by two faces. This could lead to a conflict when the two faces were coloured differently. See sections 4.2 and 4.3 for handling of overriding styles.

In order to maintain similarity of style, it was decided at the 7<sup>th</sup> CAx Implementor Forum to extend this approach to include Geometrically bounded surfaces and Wireframe. Thus the priority list for colouring these elements is as follows:

1. Wireframe / Surface collector (**geometric\_set**, **geometric\_curve\_set**)
2. Geometric Representation Item (e.g. **trimmed\_curve**, **b\_spline\_surface**)

Colours should be instantiated from the top down for these hierarchies, e.g. the solid should always be coloured, then any differences in face colours applied by overriding the solid colour for the different face. Similarly for wireframe, the **geometric\_set/geometric\_curve\_set** should always be coloured the majority colour for the geometric entities, those deviating from this majority colour being overridden.

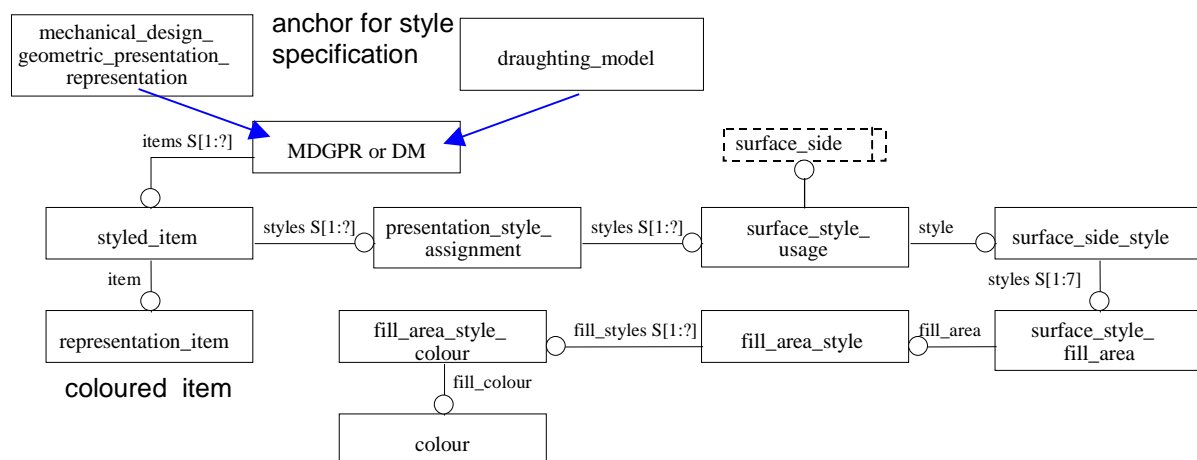


Figure 3: COLOUR REPRESENTATION FOR SURFACES/SOLIDS

As anchor for the style specification both a **mechanical\_design\_geometric\_presentation\_representation** (MDGPR) or a **draughting\_model** are applicable.

<b>MDGPR</b>	Represents the style information for a <b>shape_representation</b> which is presented in a <b>MDGP_area</b>
name	The name by which the representation is referred to.
items	Only entities of type <b>styled_item</b> shall be used in the context of colouring.
context_of_items	The context in which the styles are assigned.
<b>draughting_model</b>	The presentation of the shape of a product for the purpose of draughting.
name	The name by which the model is referred to.
items	The <b>styled_items</b> specify the style for a <b>mapped_item</b> that is the mapping of a <b>shape_representation</b> and in this context provide the colour to be used in the model.
context_of_items	The context in which the styles are assigned.
<b>styled_item</b>	A <b>representation_item</b> with associated presentation style.
name	The name by which the item is referred to.
styles	The styles assigned to the item
item	The <b>representation_item</b> to which the styles are assigned.
<b>presentation_style_assignment</b>	A set of styles which are assigned to a <b>representation_item</b> for the purpose of presenting the item.
styles	The set of presentation styles.
<b>surface_style_usage</b>	The application of a <b>surface_side_style_select</b> to the positive side, negative side, or both sides of a surface.
side	The indication of which side of the surface to apply the style.
style	The style which shall be applied to the surface.
<b>surface_style_fill_area</b>	The surface style that presents a surface by mapping a fill area onto the surface.
fill_area	The <b>fill_area_style</b> associated with the two-dimensional parameter space of a surface that will be mapped onto the surface itself.
<b>fill_area_style</b>	A style for filling visible curve segments, annotation fill areas, or surfaces with tiles or hatching.
name	The name by which the style is referred to.
fill_styles	The set of fill area styles to use in presenting the item.
<b>fill_area_style_colour</b>	Defines a colour to be used for the <b>fill_area_style</b>
name	The name by which the colour is referred to.
fill_colour	The colour to be used for filling the area. Either a <b>draughting_pre_defined_colour</b> or a <b>colour_rgb</b> shall be used here.

<b>draughting_pre_defined_colour</b>	A <b>pre_defined_colour</b> that is identified by name.
name	Required to be one of 'red', 'green', 'blue', 'yellow', 'magenta', 'cyan', 'black' or 'white'.
<b>colour_rgb</b>	Defines a colour by specifying the intensity of red, green, and blue.
name	The name by which the colour is referred to.
red	The intensity of the red colour component. ( $0.0 \leq \text{red} \leq 1.0$ )
green	The intensity of the green colour component. ( $0.0 \leq \text{green} \leq 1.0$ )
blue	The intensity of the blue colour component. ( $0.0 \leq \text{blue} \leq 1.0$ )

```
#164=MANIFOLD_SOLID_BREP(' ',#163);
#226=DRAUGHTING_PRE_DEFINED_COLOUR('cyan');
#227=FILL_AREA_STYLE_COLOUR(' ',#226);
#228=FILL_AREA_STYLE(' ',(#227));
#229=SURFACE_STYLE_FILL_AREA(#228);
#230=SURFACE_SIDE_STYLE(' ',(#229));
#231=SURFACE_STYLE_USAGE(.BOTH.,#230);
#232=PRESENTATION_STYLE_ASSIGNMENT((#231));
#233=STYLED_ITEM(' ',(#232),#164);
#276=DRAUGHTING_MODEL(' #276 ',(#233,#241,#246,#254,#255,#263,#268,#273,
#275),#269);
```

Figure 4: EXCHANGE FILE SEGMENT FOR COLOUR SPECIFICATION (FROM FILE EXAMPLE)

#### 4.1.1 Pre-defined colours to rgb

The following rgb values shall be assumed for the pre-defined colour values :

Colour	RGB
Black	1,1,1
White	0,0,0
Red	1,0,0
Blue	0,0,1
Green	0,1,0
Yellow	1,1,0
Cyan	0,1,1
Magenta	1,0,1

## 4.2 Overriding surface colours

For colouring of a surface lying on a solid, the colour of the solid is overridden. This happens by using an instance of **over\_riding\_styled\_item**.

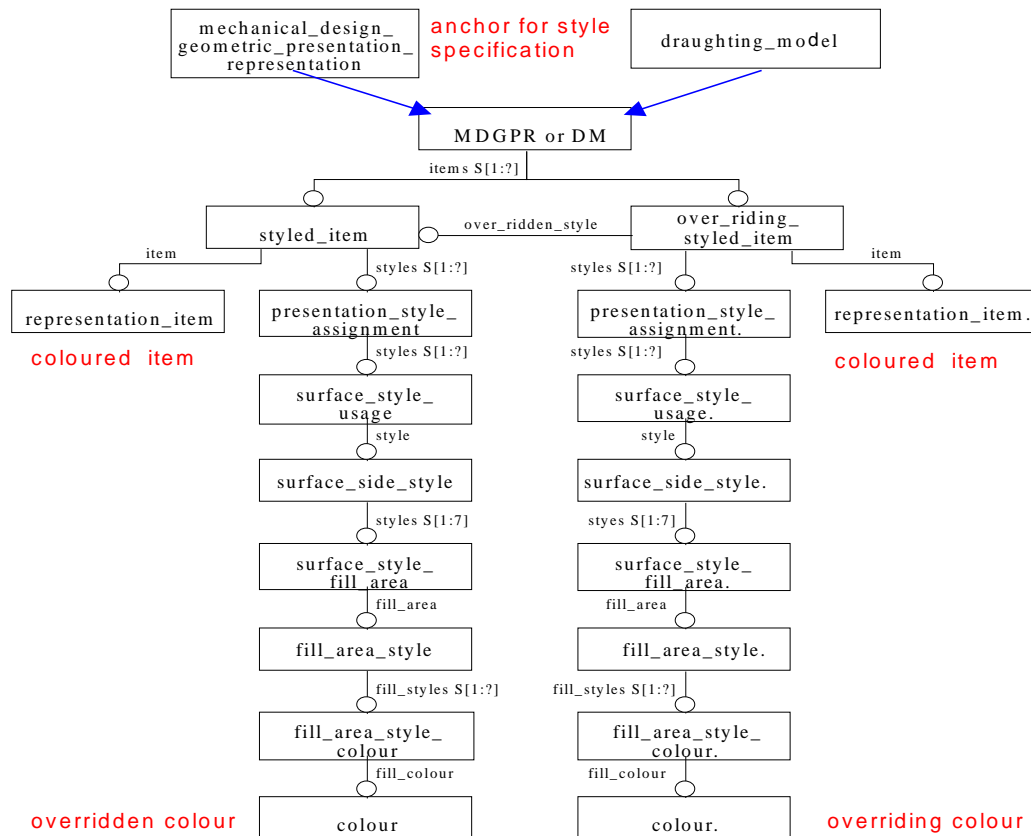


Figure 5: REPRESENTATION OF OVERRIDING FACE COLOUR

Only portions of styles redefined by **over\_riding\_styled\_item** will be overridden. Portions not redefined by the overriding style shall be handled as already defined.

<b>over_riding_styled_item</b>	A <b>styled_item</b> where the style assignment takes precedence over another assigned style. The precedence happens when the overridden <b>styled_item</b> and the <b>over_riding_styled_item</b> are both included in the same presentation.
name	The name by which the item is referred.
styles	The overriding styles assigned to the item.
item	The <b>representation_item</b> identifying the portion of the shape for which the styles are redefined.
over_ridden_style	The <b>styled_item</b> that will have it's style overridden.



```
#150=ADVANCED_FACE('',( #144), #149, .T.);
#247=DRAUGHTING_PRE_DEFINED_COLOUR('magenta');
#248=FILL_AREA_STYLE_COLOUR('',( #247);
#249=FILL_AREA_STYLE('',( #248));
#250=SURFACE_STYLE_FILL_AREA(#249);
#251=SURFACE_SIDE_STYLE('',( #250));
#252=SURFACE_STYLE_USAGE(.BOTH., #251);
#253=PRESENTATION_STYLE_ASSIGNMENT((#252));
#254=OVER RIDING_STYLED_ITEM('',( #253), #150, #233);
#276=DRAUGHTING_MODEL(' #276', (#233, #241, #246, #254, #255, #263, #268, #273,
#275), #269);
```

Figure 6: EXCHANGE FILE SEGMENT FOR OVERRIDING FACE COLOUR (FROM FILE EXAMPLE)

### 4.3 Overriding edge colours

For colouring of an edge, the colour of the solid or surface is overridden. This happens by using an instance of **over\_riding\_styled\_item**. **Note** that Edge colours are applied by a **curve\_style** rather than a **surface\_style**. This leads to a situation where the **surface\_style** of the solid or surface containing the edge could be overridden by a **curve\_style** for the edge in question.

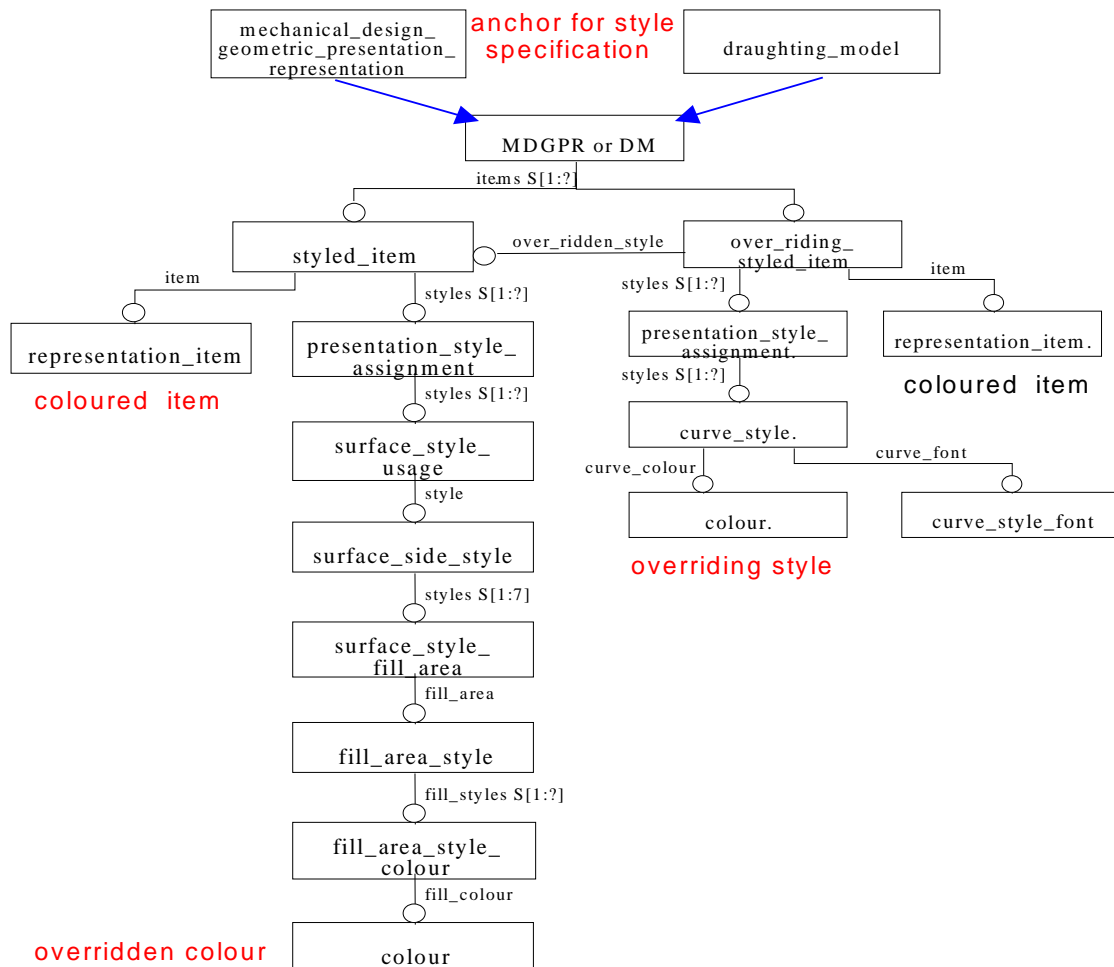


Figure 7: REPRESENTATION OF OVERRIDING EDGE COLOUR

Only portions of styles redefined by **over\_riding\_styled\_item** will be overridden. Portions not redefined by the overriding style shall be handled as already defined.

<b>curve_style</b>	Specifies the visual appearance of a curve.
name	The name by which the style is referred to.
curve_font	The <b>curve_style_font</b> which is used to present a curve.
curve_width	The width of the visible part of the presented curve in <b>presentation_area</b> units.
curve_colour	The colour of the visible part of the curve.

```
#87=EDGE_CURVE(' ',#28,#24,#66,.T.);
#242=DRAUGHTING_PRE_DEFINED_COLOUR('yellow');
#243=DRAUGHTING_PRE_DEFINED_CURVE_FONT('continuous');
#244=CURVE_STYLE(' ',#243,POSITIVE_LENGTH_MEASURE(1.0),#242);
#245=PRESENTATION_STYLE_ASSIGNMENT((#244));
#246=OVER RIDING_STYLED_ITEM(' ',(#245),#87,#233);
#276=DRAUGHTING_MODEL(' #276 ',(#233,#241,#246,#254,#255,#263,#268,#273,
#275),#269);
```

Figure 8: EXCHANGE FILE SEGMENT FOR OVERRIDING EDGE COLOUR. (FROM FILE EXAMPLE)

## 5 Part 21 file examples

The instantiation example given in this section defines a solid cube with overriding face and edge colours, a topologically bounded surface and a wireframe curve:

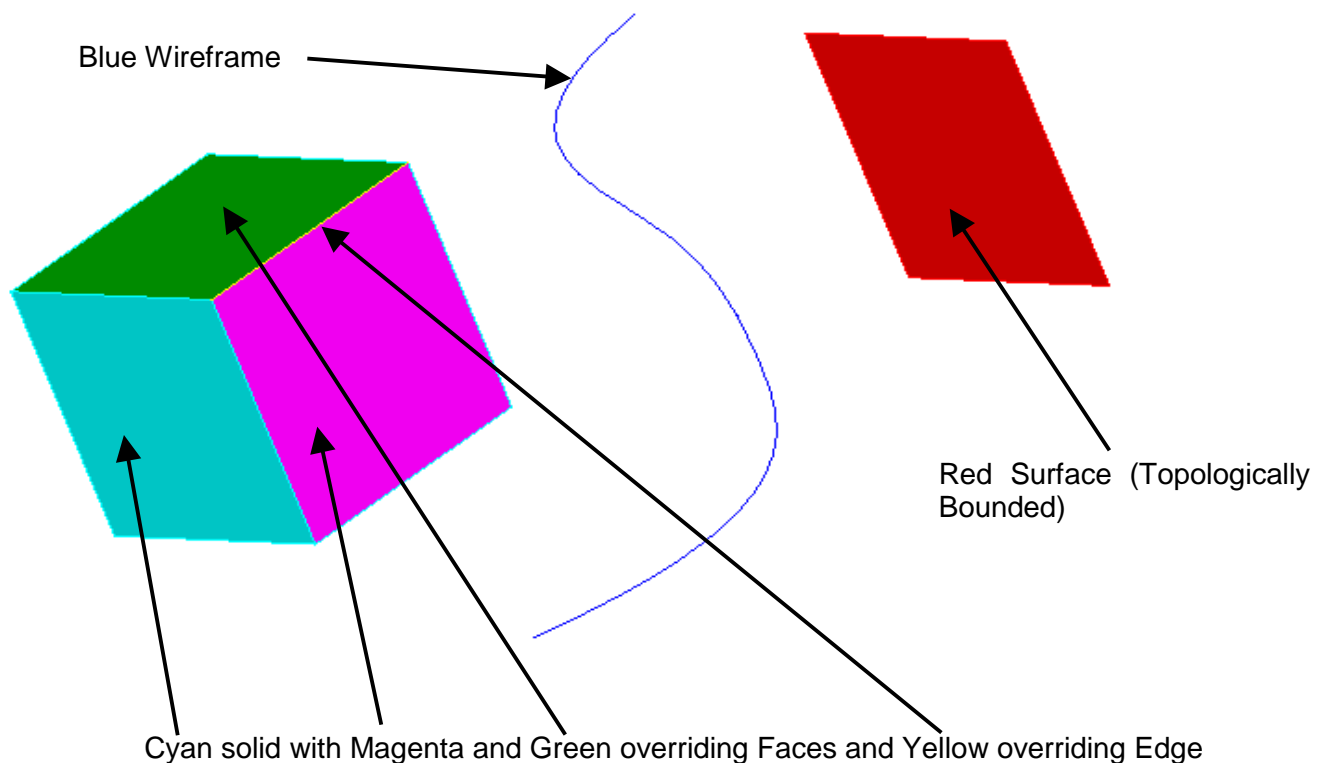


Figure 9: GEOMETRY OF PART 21 FILE EXAMPLE

**5.1 Conforming to AP214**

```

ISO-10303-21;
HEADER;
FILE_DESCRIPTION(('','2;1');
FILE_NAME('col214.stp',
'2001-10-11T11:08:10+01:00',
(' '),
(' '),
'CAx-IF Example File',
'CAD System',
'CAx-IF ');
FILE_SCHEMA(('AUTOMOTIVE_DESIGN { 1 2 10303 214 0 1 1 1 } '));
ENDSEC;
DATA;
#1=DIMENSIONAL_EXPONENTS(1.0,0.0,0.0,0.0,0.0,0.0,0.0);
#2=DIMENSIONAL_EXPONENTS(0.0,0.0,0.0,0.0,0.0,0.0,0.0);
#3=(NAMED_UNIT(*)SI_UNIT($,.STERADIAN.)SOLID_ANGLE_UNIT());
#4=(LENGTH_UNIT()NAMED_UNIT(*)SI_UNIT(.MILLI.,.METRE.));
#5=(NAMED_UNIT(*)PLANE_ANGLE_UNIT()SI_UNIT($,.RADIAN.));
#6=APPLICATION_CONTEXT(
'Core Data for Automotive Mechanical Design Process');
#7=APPLICATION_PROTOCOL_DEFINITION('draft international standard',
'automotive_design',1999,#6);
#8=PRODUCT_CONTEXT(' ',#6,'mechanical');
#9=PRODUCT_DEFINITION_CONTEXT('part definition',#6,'design');
#10=PRODUCT('product id','product name',$,(#8));
#11=PRODUCT_RELATED_PRODUCT_CATEGORY('part',$,(#10));
#12=PRODUCT_DEFINITION_FORMATION('version 0',$,#10);
#13=PRODUCT_DEFINITION(' ','unknown',#12,#9);
#14=PRODUCT_DEFINITION_SHAPE(' ',#13);
#15=CARTESIAN_POINT('#15',(0.0,0.0,0.0));
#16=VERTEX_POINT(' ',#15);
#17=CARTESIAN_POINT('#17',(0.0,50.,0.0));
#18=VERTEX_POINT(' ',#17);
#19=CARTESIAN_POINT('#19',(0.0,0.0,50.));
#20=VERTEX_POINT(' ',#19);
#21=CARTESIAN_POINT('#21',(0.0,50.,50.));
#22=VERTEX_POINT(' ',#21);
#23=CARTESIAN_POINT('#23',(50.,50.,50.));
#24=VERTEX_POINT(' ',#23);
#25=CARTESIAN_POINT('#25',(50.,50.,0.0));
#26=VERTEX_POINT(' ',#25);
#27=CARTESIAN_POINT('#27',(50.,0.0,50.));
#28=VERTEX_POINT(' ',#27);
#29=CARTESIAN_POINT('#29',(50.,0.0,0.0));
#30=VERTEX_POINT(' ',#29);
#31=CARTESIAN_POINT('#31',(0.0,0.0,0.0));
#32=DIRECTION('#32',(0.0,1.0,0.0));
#33=VECTOR('#33',#32,50.);
#34=LINE(' ',#31,#33);
#35=CARTESIAN_POINT('#35',(0.0,0.0,0.0));
#36=DIRECTION('#36',(0.0,0.0,1.0));
#37=VECTOR('#37',#36,50.);
#38=LINE(' ',#35,#37);
#39=CARTESIAN_POINT('#39',(0.0,50.,50.));
#40=DIRECTION('#40',(0.0,-1.,0.0));
#41=VECTOR('#41',#40,50.);
#42=LINE(' ',#39,#41);
#43=CARTESIAN_POINT('#43',(0.0,50.,50.));
#44=DIRECTION('#44',(0.0,0.0,-1.));

```

```

#45=VECTOR(' '#45', #44, 50.);
#46=LINE(' ', #43, #45);
#47=CARTESIAN_POINT(' '#47', (0.0, 50., 50.));
#48=DIRECTION(' '#48', (1.0, 0.0, 0.0));
#49=VECTOR(' '#49', #48, 50.);
#50=LINE(' ', #47, #49);
#51=CARTESIAN_POINT(' '#51', (50., 50., 0.0));
#52=DIRECTION(' '#52', (0.0, 0.0, 1.0));
#53=VECTOR(' '#53', #52, 50.);
#54=LINE(' ', #51, #53);
#55=CARTESIAN_POINT(' '#55', (50., 50., 0.0));
#56=DIRECTION(' '#56', (-1., 0.0, 0.0));
#57=VECTOR(' '#57', #56, 50.);
#58=LINE(' ', #55, #57);
#59=CARTESIAN_POINT(' '#59', (50., 0.0, 50.));
#60=DIRECTION(' '#60', (-1., 0.0, 0.0));
#61=VECTOR(' '#61', #60, 50.);
#62=LINE(' ', #59, #61);
#63=CARTESIAN_POINT(' '#63', (50., 0.0, 50.));
#64=DIRECTION(' '#64', (0.0, 1.0, 0.0));
#65=VECTOR(' '#65', #64, 50.);
#66=LINE(' ', #63, #65);
#67=CARTESIAN_POINT(' '#67', (0.0, 0.0, 0.0));
#68=DIRECTION(' '#68', (1.0, 0.0, 0.0));
#69=VECTOR(' '#69', #68, 50.);
#70=LINE(' ', #67, #69);
#71=CARTESIAN_POINT(' '#71', (50., 50., 0.0));
#72=DIRECTION(' '#72', (0.0, -1., 0.0));
#73=VECTOR(' '#73', #72, 50.);
#74=LINE(' ', #71, #73);
#75=CARTESIAN_POINT(' '#75', (50., 0.0, 50.));
#76=DIRECTION(' '#76', (0.0, 0.0, -1.));
#77=VECTOR(' '#77', #76, 50.);
#78=LINE(' ', #75, #77);
#79=EDGE_CURVE(' ', #16, #18, #34, .T.);
#80=EDGE_CURVE(' ', #16, #20, #38, .T.);
#81=EDGE_CURVE(' ', #22, #20, #42, .T.);
#82=EDGE_CURVE(' ', #22, #18, #46, .T.);
#83=EDGE_CURVE(' ', #22, #24, #50, .T.);
#84=EDGE_CURVE(' ', #26, #24, #54, .T.);
#85=EDGE_CURVE(' ', #26, #18, #58, .T.);
#86=EDGE_CURVE(' ', #28, #20, #62, .T.);
#87=EDGE_CURVE(' ', #28, #24, #66, .T.);
#88=EDGE_CURVE(' ', #16, #30, #70, .T.);
#89=EDGE_CURVE(' ', #26, #30, #74, .T.);
#90=EDGE_CURVE(' ', #28, #30, #78, .T.);
#91=ORIENTED_EDGE(' '#91', *, *, #79, .F.);
#92=ORIENTED_EDGE(' '#92', *, *, #80, .T.);
#93=ORIENTED_EDGE(' '#93', *, *, #81, .F.);
#94=ORIENTED_EDGE(' '#94', *, *, #82, .T.);
#95=EDGE_LOOP(' '#95', (#91, #92, #93, #94));
#96=FACE_OUTER_BOUND(' '#96', #95, .T.);
#97=CARTESIAN_POINT(' '#97', (0.0, 25., 25.));
#98=DIRECTION(' '#98', (-1., 0.0, 0.0));
#99=DIRECTION(' '#99', (0.0, -1., 0.0));
#100=AXIS2_PLACEMENT_3D(' '#100', #97, #98, #99);
#101=PLANE(' ', #100);
#102=ADVANCED_FACE(' ', (#96), #101, .T.);
#103=ORIENTED_EDGE(' '#103', *, *, #82, .F.);
#104=ORIENTED_EDGE(' '#104', *, *, #83, .T.);
#105=ORIENTED_EDGE(' '#105', *, *, #84, .F.);

```

```

#106=ORIENTED_EDGE(' #106',*,*,#85,.T.);
#107=EDGE_LOOP(' #107',(#103,#104,#105,#106));
#108=FACE_OUTER_BOUND(' #108',#107,.T.);
#109=CARTESIAN_POINT(' #109',(25.,50.,25.));
#110=DIRECTION(' #110',(0.0,1.0,0.0));
#111=DIRECTION(' #111',(0.0,0.0,1.0));
#112=AXIS2_PLACEMENT_3D(' #112',#109,#110,#111);
#113=PLANE(' ',#112);
#114=ADVANCED_FACE(' ',(#108),#113,.T.);
#115=ORIENTED_EDGE(' #115',*,*,#83,.F.);
#116=ORIENTED_EDGE(' #116',*,*,#81,.T.);
#117=ORIENTED_EDGE(' #117',*,*,#86,.F.);
#118=ORIENTED_EDGE(' #118',*,*,#87,.T.);
#119=EDGE_LOOP(' #119',(#115,#116,#117,#118));
#120=FACE_OUTER_BOUND(' #120',#119,.T.);
#121=CARTESIAN_POINT(' #121',(25.,25.,50.));
#122=DIRECTION(' #122',(0.0,0.0,1.0));
#123=DIRECTION(' #123',(-1.,0.0,0.0));
#124=AXIS2_PLACEMENT_3D(' #124',#121,#122,#123);
#125=PLANE(' ',#124);
#126=ADVANCED_FACE(' ',(#120),#125,.T.);
#127=ORIENTED_EDGE(' #127',*,*,#88,.F.);
#128=ORIENTED_EDGE(' #128',*,*,#79,.T.);
#129=ORIENTED_EDGE(' #129',*,*,#85,.F.);
#130=ORIENTED_EDGE(' #130',*,*,#89,.T.);
#131=EDGE_LOOP(' #131',(#127,#128,#129,#130));
#132=FACE_OUTER_BOUND(' #132',#131,.T.);
#133=CARTESIAN_POINT(' #133',(25.,25.,0.0));
#134=DIRECTION(' #134',(0.0,0.0,-1.));
#135=DIRECTION(' #135',(-1.,0.0,0.0));
#136=AXIS2_PLACEMENT_3D(' #136',#133,#134,#135);
#137=PLANE(' ',#136);
#138=ADVANCED_FACE(' ',(#132),#137,.T.);
#139=ORIENTED_EDGE(' #139',*,*,#89,.F.);
#140=ORIENTED_EDGE(' #140',*,*,#84,.T.);
#141=ORIENTED_EDGE(' #141',*,*,#87,.F.);
#142=ORIENTED_EDGE(' #142',*,*,#90,.T.);
#143=EDGE_LOOP(' #143',(#139,#140,#141,#142));
#144=FACE_OUTER_BOUND(' #144',#143,.T.);
#145=CARTESIAN_POINT(' #145',(50.,25.,25.));
#146=DIRECTION(' #146',(1.0,0.0,0.0));
#147=DIRECTION(' #147',(0.0,1.0,0.0));
#148=AXIS2_PLACEMENT_3D(' #148',#145,#146,#147);
#149=PLANE(' ',#148);
#150=ADVANCED_FACE(' ',(#144),#149,.T.);
#151=ORIENTED_EDGE(' #151',*,*,#80,.F.);
#152=ORIENTED_EDGE(' #152',*,*,#88,.T.);
#153=ORIENTED_EDGE(' #153',*,*,#90,.F.);
#154=ORIENTED_EDGE(' #154',*,*,#86,.T.);
#155=EDGE_LOOP(' #155',(#151,#152,#153,#154));
#156=FACE_OUTER_BOUND(' #156',#155,.T.);
#157=CARTESIAN_POINT(' #157',(25.,0.0,25.));
#158=DIRECTION(' #158',(0.0,-1.,0.0));
#159=DIRECTION(' #159',(0.0,0.0,-1.));
#160=AXIS2_PLACEMENT_3D(' #160',#157,#158,#159);
#161=PLANE(' ',#160);
#162=ADVANCED_FACE(' ',(#156),#161,.T.);
#163=CLOSED_SHELL(' #163',(#102,#114,#126,#138,#150,#162));
#164=MANIFOLD_SOLID_BREP(' ',#163);
#165=CARTESIAN_POINT(' #165',(100.,100.,0.0));
#166=VERTEX_POINT(' ',#165);

```

```

#167=CARTESIAN_POINT(' #167', (100.,100.,50.));
#168=VERTEX_POINT(' ', #167);
#169=CARTESIAN_POINT(' #169', (150.,100.,0.0));
#170=VERTEX_POINT(' ', #169);
#171=CARTESIAN_POINT(' #171', (150.,100.,50.));
#172=VERTEX_POINT(' ', #171);
#173=CARTESIAN_POINT(' #173', (100.,100.,0.0));
#174=DIRECTION(' #174', (0.0,0.0,1.0));
#175=VECTOR(' #175', #174,50.);
#176=LINE(' ', #173,#175);
#177=CARTESIAN_POINT(' #177', (100.,100.,0.0));
#178=DIRECTION(' #178', (1.0,0.0,0.0));
#179=VECTOR(' #179', #178,50.);
#180=LINE(' ', #177,#179);
#181=CARTESIAN_POINT(' #181', (150.,100.,50.));
#182=DIRECTION(' #182', (0.0,0.0,-1.));
#183=VECTOR(' #183', #182,50.);
#184=LINE(' ', #181,#183);
#185=CARTESIAN_POINT(' #185', (150.,100.,50.));
#186=DIRECTION(' #186', (-1.,0.0,0.0));
#187=VECTOR(' #187', #186,50.);
#188=LINE(' ', #185,#187);
#189=EDGE_CURVE(' ', #166,#168,#176,.T.);
#190=EDGE_CURVE(' ', #166,#170,#180,.T.);
#191=EDGE_CURVE(' ', #172,#170,#184,.T.);
#192=EDGE_CURVE(' ', #172,#168,#188,.T.);
#193=ORIENTED_EDGE(' #193', *,*,#189,.F.);
#194=ORIENTED_EDGE(' #194', *,*,#190,.T.);
#195=ORIENTED_EDGE(' #195', *,*,#191,.F.);
#196=ORIENTED_EDGE(' #196', *,*,#192,.T.);
#197=EDGE_LOOP(' #197', (#193,#194,#195,#196));
#198=FACE_OUTER_BOUND(' #198', #197,.T.);
#199=CARTESIAN_POINT(' #199', (125.,100.,25.));
#200=DIRECTION(' #200', (0.0,-1.,0.0));
#201=DIRECTION(' #201', (0.0,0.0,-1.));
#202=AXIS2_PLACEMENT_3D(' #202', #199,#200,#201);
#203=PLANE(' ', #202);
#204=ADVANCED_FACE(' ', (#198),#203,.T.);
#205=OPEN_SHELL(' #205', (#204));
#206=SHELL_BASED_SURFACE_MODEL(' #206', (#205));
#207=CARTESIAN_POINT(' #207', (-53.06339474,186.41142347,0.0));
#208=CARTESIAN_POINT(' #208', (-48.87329012,158.63330064,0.0));
#209=CARTESIAN_POINT(' #209', (-22.95334534,132.62814755,0.0));
#210=CARTESIAN_POINT(' #210', (38.47821014,109.44948121,0.0));
#211=CARTESIAN_POINT(' #211', (87.02630611,74.50572274,0.0));
#212=CARTESIAN_POINT(' #212', (140.28037162,31.30872812,0.0));
#213=CARTESIAN_POINT(' #213', (144.74613919,-5.48598948,0.0));
#214=CARTESIAN_POINT(' #214', (132.77349428,-28.90239822,0.0));
#215=B_SPLINE_CURVE_WITH_KNOTS(' ', 3, (#207,#208,#209,#210,#211,#212,#213,
#214), .UNSPECIFIED., .F., .F., (4,1,1,1,1,4), (0.0,65.47002974,113.71553761,
191.84772524,248.5790028,306.27452289), .UNSPECIFIED.);
#216=GEOMETRIC_CURVE_SET(' #216', (#215));
#217=UNCERTAINTY_MEASURE_WITH_UNIT(LENGTH_MEASURE(0.1E-12), #4,
'distance_accuracy_value', 'EDGE CURVE AND VERTEX POINT ACCURACY');
#218=(GEOMETRIC_REPRESENTATION_CONTEXT(3)
GLOBAL_UNCERTAINTY_ASSIGNED_CONTEXT((#217))GLOBAL_UNIT_ASSIGNED_CONTEXT(
(#4,#5,#3))REPRESENTATION_CONTEXT('MASTER', '3D'));
#219=CARTESIAN_POINT(' #219', (0.0,0.0,0.0));
#220=DIRECTION(' #220', (1.0,0.0,0.0));
#221=DIRECTION(' #221', (0.0,0.0,1.0));
#222=AXIS2_PLACEMENT_3D(' #222', #219,#221,#220);

```

```

#223=SHAPE_REPRESENTATION(' #223', (#164,#206,#216,#222),#218);
#224=PRESENTATION_LAYER_ASSIGNMENT('020','layer 020',(#164));
#225=PRESENTATION_LAYER_ASSIGNMENT('010','layer 010',(#206));
#226=DRAUGHTING_PRE_DEFINED_COLOUR('cyan');
#227=FILL_AREA_STYLE_COLOUR('',#226);
#228=FILL_AREA_STYLE('',(#227));
#229=SURFACE_STYLE_FILL_AREA(#228);
#230=SURFACE_SIDE_STYLE('',(#229));
#231=SURFACE_STYLE_USAGE(.BOTH.,#230);
#232=PRESENTATION_STYLE_ASSIGNMENT((#231));
#233=STYLED_ITEM('',(#232),#164);
#234=DRAUGHTING_PRE_DEFINED_COLOUR('green');
#235=FILL_AREA_STYLE_COLOUR('',#234);
#236=FILL_AREA_STYLE('',(#235));
#237=SURFACE_STYLE_FILL_AREA(#236);
#238=SURFACE_SIDE_STYLE('',(#237));
#239=SURFACE_STYLE_USAGE(.BOTH.,#238);
#240=PRESENTATION_STYLE_ASSIGNMENT((#239));
#241=OVER_RIDING_STYLED_ITEM('',(#240),#126,#233);
#242=DRAUGHTING_PRE_DEFINED_COLOUR('yellow');
#243=DRAUGHTING_PRE_DEFINED_CURVE_FONT('continuous');
#244=CURVE_STYLE('',#243,POSITIVE_LENGTH_MEASURE(1.0),#242);
#245=PRESENTATION_STYLE_ASSIGNMENT((#244));
#246=OVER_RIDING_STYLED_ITEM('',(#245),#87,#233);
#247=DRAUGHTING_PRE_DEFINED_COLOUR('magenta');
#248=FILL_AREA_STYLE_COLOUR('',#247);
#249=FILL_AREA_STYLE('',(#248));
#250=SURFACE_STYLE_FILL_AREA(#249);
#251=SURFACE_SIDE_STYLE('',(#250));
#252=SURFACE_STYLE_USAGE(.BOTH.,#251);
#253=PRESENTATION_STYLE_ASSIGNMENT((#252));
#254=OVER_RIDING_STYLED_ITEM('',(#253),#150,#233);
#255=OVER_RIDING_STYLED_ITEM('',(#245),#87,#233);
#256=DRAUGHTING_PRE_DEFINED_COLOUR('red');
#257=FILL_AREA_STYLE_COLOUR('',#256);
#258=FILL_AREA_STYLE('',(#257));
#259=SURFACE_STYLE_FILL_AREA(#258);
#260=SURFACE_SIDE_STYLE('',(#259));
#261=SURFACE_STYLE_USAGE(.BOTH.,#260);
#262=PRESENTATION_STYLE_ASSIGNMENT((#261));
#263=STYLED_ITEM('',(#262),#206);
#264=DRAUGHTING_PRE_DEFINED_COLOUR('blue');
#265=DRAUGHTING_PRE_DEFINED_CURVE_FONT('continuous');
#266=CURVE_STYLE('',#265,POSITIVE_LENGTH_MEASURE(1.0),#264);
#267=PRESENTATION_STYLE_ASSIGNMENT((#266));
#268=STYLED_ITEM('',(#267),#216);
#269=(GEOMETRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CONTEXT((
#4,#5,#3))REPRESENTATION_CONTEXT('Presentation Context','3D'));
#270=CARTESIAN_POINT('#270',(0.0,0.0,0.0));
#271=DIRECTION('#271',(1.0,0.0,0.0));
#272=DIRECTION('#272',(0.0,0.0,1.0));
#273=AXIS2_PLACEMENT_3D('#273',#270,#272,#271);
#274=REPRESENTATION_MAP(#222,#223);
#275=MAPPED_ITEM('#275',#274,#273);
#276=DRAUGHTING_MODEL('#276',(#233,#241,#246,#254,#255,#263,#268,#273,
#275),#269);
#277=SHAPE_DEFINITION_REPRESENTATION(#14,#223);
ENDSEC;
END-ISO-10303-21;

```

**5.2 Conforming to AP203 + extensions**

```

ISO-10303-21;
HEADER;
FILE_DESCRIPTION(('','2;1');
FILE_NAME('col203.stp',
'2001-10-11T11:08:53+01:00',
(' '),
(' '),
'CAx-IF Example File',
'CAD System',
'CAx-IF ');
FILE_SCHEMA(('ccdclg'));
ENDSEC;
DATA;
#1=DIMENSIONAL_EXPONENTS(1.0,0.0,0.0,0.0,0.0,0.0,0.0);
#2=DIMENSIONAL_EXPONENTS(0.0,0.0,0.0,0.0,0.0,0.0,0.0);
#3=(NAMED_UNIT(*)SI_UNIT($,.STERADIAN.)SOLID_ANGLE_UNIT());
#4=(LENGTH_UNIT()NAMED_UNIT(*)SI_UNIT(.MILLI.,.METRE.));
#5=(NAMED_UNIT(*)PLANE_ANGLE_UNIT()SI_UNIT($,.RADIAN.));
#6=APPLICATION_CONTEXT(
'configuration controlled 3D designs of mechanical parts and assemblies'
);
#7=APPLICATION_PROTOCOL_DEFINITION('international standard',
'config_control_design',1994,#6);
#8=MECHANICAL_CONTEXT(' ',#6,'mechanical');
#9=DESIGN_CONTEXT(' ',#6,'design');
#10=PRODUCT('MASTER','MASTER','Description for MASTER',(#8));
#11=PRODUCT_RELATED_PRODUCT_CATEGORY('detail','MASTER',(#10));
#12=PRODUCT_CATEGORY('part','MASTER');
#13=PRODUCT_CATEGORY_RELATIONSHIP(' ','MASTER',#12,#11);
#14=PRODUCT_DEFINITION_FORMATION_WITH_SPECIFIED_SOURCE('version 0',
'Description for MASTER',#10,.NOT_KNOWN.);
#15=PRODUCT_DEFINITION('design','unknown',#14,#9);
#16=PERSON('Default_pid','Doe','John',$,$,$);
#17=ORGANIZATION('Default_oid','Default_org','Default Description');
#18=PERSON_AND_ORGANIZATION(#16,#17);
#19=COORDINATED_UNIVERSAL_TIME_OFFSET(0,0,.AHEAD.);
#20=LOCAL_TIME(10,8,0.0,#19);
#21=CALENDAR_DATE(2001,11,10);
#22=DATE_AND_TIME(#21,#20);
#23=PERSON_AND_ORGANIZATION_ROLE('design_owner');
#24=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#18,#23,(#10));
#25=PERSON_AND_ORGANIZATION_ROLE('design_supplier');
#26=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#18,#25,(#14));
#27=PERSON_AND_ORGANIZATION_ROLE('creator');
#28=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#18,#27,(#14));
#29=PERSON_AND_ORGANIZATION_ROLE('creator');
#30=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#18,#29,(#15));
#31=SECURITY_CLASSIFICATION_LEVEL('unclassified');
#32=SECURITY_CLASSIFICATION(' ','unknown',#31);
#33=CC_DESIGN_SECURITY_CLASSIFICATION(#32,(#14));
#34=PERSON_AND_ORGANIZATION_ROLE('classification_officer');
#35=CC_DESIGN_PERSON_AND_ORGANIZATION_ASSIGNMENT(#18,#34,(#32));
#36=APPROVAL_STATUS('not_yet_approved');
#37=APPROVAL(#36,'unknown');
#38=APPROVAL_ROLE('approver');
#39=APPROVAL_PERSON_ORGANIZATION(#18,#37,#38);
#40=APPROVAL_DATE_TIME(#22,#37);
#41=CC_DESIGN_APPROVAL(#37,(#32));
#42=DATE_TIME_ROLE('sign_off_date');
#43=CC_DESIGN_DATE_AND_TIME_ASSIGNMENT(#22,#42,(#39));

```



```

#44=DATE_TIME_ROLE('classification_date');
#45=CC_DESIGN_DATE_AND_TIME_ASSIGNMENT(#22,#44,(#32));
#46=APPROVAL_STATUS('not_yet_approved');
#47=APPROVAL(#46,'unknown');
#48=APPROVAL_ROLE('approver');
#49=APPROVAL_PERSON_ORGANIZATION(#18,#47,#48);
#50=APPROVAL_DATE_TIME(#22,#47);
#51=CC_DESIGN_APPROVAL(#47,(#14));
#52=DATE_TIME_ROLE('sign_off_date');
#53=CC_DESIGN_DATE_AND_TIME_ASSIGNMENT(#22,#52,(#49));
#54=APPROVAL_STATUS('not_yet_approved');
#55=APPROVAL(#54,'unknown');
#56=APPROVAL_ROLE('approver');
#57=APPROVAL_PERSON_ORGANIZATION(#18,#55,#56);
#58=APPROVAL_DATE_TIME(#22,#55);
#59=CC_DESIGN_APPROVAL(#55,(#15));
#60=DATE_TIME_ROLE('sign_off_date');
#61=CC_DESIGN_DATE_AND_TIME_ASSIGNMENT(#22,#60,(#57));
#62=DATE_TIME_ROLE('creation_date');
#63=CC_DESIGN_DATE_AND_TIME_ASSIGNMENT(#22,#62,(#15));
#64=CARTESIAN_POINT('#64',(0.0,0.0,0.0));
#65=VERTEX_POINT('',#64);
#66=CARTESIAN_POINT('#66',(0.0,50.,0.0));
#67=VERTEX_POINT('',#66);
#68=CARTESIAN_POINT('#68',(0.0,0.0,50.));
#69=VERTEX_POINT('',#68);
#70=CARTESIAN_POINT('#70',(0.0,50.,50.));
#71=VERTEX_POINT('',#70);
#72=CARTESIAN_POINT('#72',(50.,50.,50.));
#73=VERTEX_POINT('',#72);
#74=CARTESIAN_POINT('#74',(50.,50.,0.0));
#75=VERTEX_POINT('',#74);
#76=CARTESIAN_POINT('#76',(50.,0.0,50.));
#77=VERTEX_POINT('',#76);
#78=CARTESIAN_POINT('#78',(50.,0.0,0.0));
#79=VERTEX_POINT('',#78);
#80=CARTESIAN_POINT('#80',(0.0,0.0,0.0));
#81=DIRECTION('#81',(0.0,1.0,0.0));
#82=VECTOR('#82',#81,50.);
#83=LINE('',#80,#82);
#84=CARTESIAN_POINT('#84',(0.0,0.0,0.0));
#85=DIRECTION('#85',(0.0,0.0,1.0));
#86=VECTOR('#86',#85,50.);
#87=LINE('',#84,#86);
#88=CARTESIAN_POINT('#88',(0.0,50.,50.));
#89=DIRECTION('#89',(0.0,-1.,0.0));
#90=VECTOR('#90',#89,50.);
#91=LINE('',#88,#90);
#92=CARTESIAN_POINT('#92',(0.0,50.,50.));
#93=DIRECTION('#93',(0.0,0.0,-1.));
#94=VECTOR('#94',#93,50.);
#95=LINE('',#92,#94);
#96=CARTESIAN_POINT('#96',(0.0,50.,50.));
#97=DIRECTION('#97',(1.0,0.0,0.0));
#98=VECTOR('#98',#97,50.);
#99=LINE('',#96,#98);
#100=CARTESIAN_POINT('#100',(50.,50.,0.0));
#101=DIRECTION('#101',(0.0,0.0,1.0));
#102=VECTOR('#102',#101,50.);
#103=LINE('',#100,#102);
#104=CARTESIAN_POINT('#104',(50.,50.,0.0));

```

```

#105=DIRECTION(' #105', (-1., 0.0, 0.0));
#106=VECTOR(' #106', #105, 50.);
#107=LINE(' ', #104, #106);
#108=CARTESIAN_POINT(' #108', (50., 0.0, 50.));
#109=DIRECTION(' #109', (-1., 0.0, 0.0));
#110=VECTOR(' #110', #109, 50.);
#111=LINE(' ', #108, #110);
#112=CARTESIAN_POINT(' #112', (50., 0.0, 50.));
#113=DIRECTION(' #113', (0.0, 1.0, 0.0));
#114=VECTOR(' #114', #113, 50.);
#115=LINE(' ', #112, #114);
#116=CARTESIAN_POINT(' #116', (0.0, 0.0, 0.0));
#117=DIRECTION(' #117', (1.0, 0.0, 0.0));
#118=VECTOR(' #118', #117, 50.);
#119=LINE(' ', #116, #118);
#120=CARTESIAN_POINT(' #120', (50., 50., 0.0));
#121=DIRECTION(' #121', (0.0, -1., 0.0));
#122=VECTOR(' #122', #121, 50.);
#123=LINE(' ', #120, #122);
#124=CARTESIAN_POINT(' #124', (50., 0.0, 50.));
#125=DIRECTION(' #125', (0.0, 0.0, -1.));
#126=VECTOR(' #126', #125, 50.);
#127=LINE(' ', #124, #126);
#128=EDGE_CURVE(' ', #65, #67, #83, .T.);
#129=EDGE_CURVE(' ', #65, #69, #87, .T.);
#130=EDGE_CURVE(' ', #71, #69, #91, .T.);
#131=EDGE_CURVE(' ', #71, #67, #95, .T.);
#132=EDGE_CURVE(' ', #71, #73, #99, .T.);
#133=EDGE_CURVE(' ', #75, #73, #103, .T.);
#134=EDGE_CURVE(' ', #75, #67, #107, .T.);
#135=EDGE_CURVE(' ', #77, #69, #111, .T.);
#136=EDGE_CURVE(' ', #77, #73, #115, .T.);
#137=EDGE_CURVE(' ', #65, #79, #119, .T.);
#138=EDGE_CURVE(' ', #75, #79, #123, .T.);
#139=EDGE_CURVE(' ', #77, #79, #127, .T.);
#140=ORIENTED_EDGE(' #140', *, *, #128, .F.);
#141=ORIENTED_EDGE(' #141', *, *, #129, .T.);
#142=ORIENTED_EDGE(' #142', *, *, #130, .F.);
#143=ORIENTED_EDGE(' #143', *, *, #131, .T.);
#144=EDGE_LOOP(' #144', (#140, #141, #142, #143));
#145=FACE_OUTER_BOUND(' #145', #144, .T.);
#146=CARTESIAN_POINT(' #146', (0.0, 25., 25.));
#147=DIRECTION(' #147', (-1., 0.0, 0.0));
#148=DIRECTION(' #148', (0.0, -1., 0.0));
#149=AXIS2_PLACEMENT_3D(' #149', #146, #147, #148);
#150=PLANE(' ', #149);
#151=ADVANCED_FACE(' ', (#145), #150, .T.);
#152=ORIENTED_EDGE(' #152', *, *, #131, .F.);
#153=ORIENTED_EDGE(' #153', *, *, #132, .T.);
#154=ORIENTED_EDGE(' #154', *, *, #133, .F.);
#155=ORIENTED_EDGE(' #155', *, *, #134, .T.);
#156=EDGE_LOOP(' #156', (#152, #153, #154, #155));
#157=FACE_OUTER_BOUND(' #157', #156, .T.);
#158=CARTESIAN_POINT(' #158', (25., 50., 25.));
#159=DIRECTION(' #159', (0.0, 1.0, 0.0));
#160=DIRECTION(' #160', (0.0, 0.0, 1.0));
#161=AXIS2_PLACEMENT_3D(' #161', #158, #159, #160);
#162=PLANE(' ', #161);
#163=ADVANCED_FACE(' ', (#157), #162, .T.);
#164=ORIENTED_EDGE(' #164', *, *, #132, .F.);
#165=ORIENTED_EDGE(' #165', *, *, #130, .T.);

```

```

#166=ORIENTED_EDGE(' #166',*,*,#135,.F.);
#167=ORIENTED_EDGE(' #167',*,*,#136,.T.);
#168=EDGE_LOOP(' #168',(#164,#165,#166,#167));
#169=FACE_OUTER_BOUND(' #169',#168,.T.);
#170=CARTESIAN_POINT(' #170',(25.,25.,50.));
#171=DIRECTION(' #171',(0.0,0.0,1.0));
#172=DIRECTION(' #172',(-1.,0.0,0.0));
#173=AXIS2_PLACEMENT_3D(' #173',#170,#171,#172);
#174=PLANE(' ',#173);
#175=ADVANCED_FACE(' ',(#169),#174,.T.);
#176=ORIENTED_EDGE(' #176',*,*,#137,.F.);
#177=ORIENTED_EDGE(' #177',*,*,#128,.T.);
#178=ORIENTED_EDGE(' #178',*,*,#134,.F.);
#179=ORIENTED_EDGE(' #179',*,*,#138,.T.);
#180=EDGE_LOOP(' #180',(#176,#177,#178,#179));
#181=FACE_OUTER_BOUND(' #181',#180,.T.);
#182=CARTESIAN_POINT(' #182',(25.,25.,0.0));
#183=DIRECTION(' #183',(0.0,0.0,-1.));
#184=DIRECTION(' #184',(-1.,0.0,0.0));
#185=AXIS2_PLACEMENT_3D(' #185',#182,#183,#184);
#186=PLANE(' ',#185);
#187=ADVANCED_FACE(' ',(#181),#186,.T.);
#188=ORIENTED_EDGE(' #188',*,*,#138,.F.);
#189=ORIENTED_EDGE(' #189',*,*,#133,.T.);
#190=ORIENTED_EDGE(' #190',*,*,#136,.F.);
#191=ORIENTED_EDGE(' #191',*,*,#139,.T.);
#192=EDGE_LOOP(' #192',(#188,#189,#190,#191));
#193=FACE_OUTER_BOUND(' #193',#192,.T.);
#194=CARTESIAN_POINT(' #194',(50.,25.,25.));
#195=DIRECTION(' #195',(1.0,0.0,0.0));
#196=DIRECTION(' #196',(0.0,1.0,0.0));
#197=AXIS2_PLACEMENT_3D(' #197',#194,#195,#196);
#198=PLANE(' ',#197);
#199=ADVANCED_FACE(' ',(#193),#198,.T.);
#200=ORIENTED_EDGE(' #200',*,*,#129,.F.);
#201=ORIENTED_EDGE(' #201',*,*,#137,.T.);
#202=ORIENTED_EDGE(' #202',*,*,#139,.F.);
#203=ORIENTED_EDGE(' #203',*,*,#135,.T.);
#204=EDGE_LOOP(' #204',(#200,#201,#202,#203));
#205=FACE_OUTER_BOUND(' #205',#204,.T.);
#206=CARTESIAN_POINT(' #206',(25.,0.0,25.));
#207=DIRECTION(' #207',(0.0,-1.,0.0));
#208=DIRECTION(' #208',(0.0,0.0,-1.));
#209=AXIS2_PLACEMENT_3D(' #209',#206,#207,#208);
#210=PLANE(' ',#209);
#211=ADVANCED_FACE(' ',(#205),#210,.T.);
#212=CLOSED_SHELL(' #212',(#151,#163,#175,#187,#199,#211));
#213=MANIFOLD_SOLID_BREP(' ',#212);
#214=CARTESIAN_POINT(' #214',(100.,100.,0.0));
#215=VERTEX_POINT(' ',#214);
#216=CARTESIAN_POINT(' #216',(100.,100.,50.));
#217=VERTEX_POINT(' ',#216);
#218=CARTESIAN_POINT(' #218',(150.,100.,0.0));
#219=VERTEX_POINT(' ',#218);
#220=CARTESIAN_POINT(' #220',(150.,100.,50.));
#221=VERTEX_POINT(' ',#220);
#222=CARTESIAN_POINT(' #222',(100.,100.,0.0));
#223=DIRECTION(' #223',(0.0,0.0,1.0));
#224=VECTOR(' #224',#223,50.);
#225=LINE(' ',#222,#224);
#226=CARTESIAN_POINT(' #226',(100.,100.,0.0));

```

```

#227=DIRECTION(' #227', (1.0,0.0,0.0));
#228=VECTOR(' #228', #227,50.);
#229=LINE('', #226, #228);
#230=CARTESIAN_POINT(' #230', (150.,100.,50.));
#231=DIRECTION(' #231', (0.0,0.0,-1.));
#232=VECTOR(' #232', #231,50.);
#233=LINE('', #230, #232);
#234=CARTESIAN_POINT(' #234', (150.,100.,50.));
#235=DIRECTION(' #235', (-1.,0.0,0.0));
#236=VECTOR(' #236', #235,50.);
#237=LINE('', #234, #236);
#238=EDGE_CURVE('', #215, #217, #225, .T.);
#239=EDGE_CURVE('', #215, #219, #229, .T.);
#240=EDGE_CURVE('', #221, #219, #233, .T.);
#241=EDGE_CURVE('', #221, #217, #237, .T.);
#242=ORIENTED_EDGE(' #242', *, *, #238, .F.);
#243=ORIENTED_EDGE(' #243', *, *, #239, .T.);
#244=ORIENTED_EDGE(' #244', *, *, #240, .F.);
#245=ORIENTED_EDGE(' #245', *, *, #241, .T.);
#246=EDGE_LOOP(' #246', (#242, #243, #244, #245));
#247=FACE_OUTER_BOUND(' #247', #246, .T.);
#248=CARTESIAN_POINT(' #248', (125.,100.,25.));
#249=DIRECTION(' #249', (0.0,-1.,0.0));
#250=DIRECTION(' #250', (0.0,0.0,-1.));
#251=AXIS2_PLACEMENT_3D(' #251', #248, #249, #250);
#252=PLANE('', #251);
#253=ADVANCED_FACE('', (#247), #252, .T.);
#254=OPEN_SHELL(' #254', (#253));
#255=SHELL_BASED_SURFACE_MODEL(' #255', (#254));
#256=CARTESIAN_POINT(' #256', (-53.06339474,186.41142347,0.0));
#257=CARTESIAN_POINT(' #257', (-48.87329012,158.63330064,0.0));
#258=CARTESIAN_POINT(' #258', (-22.95334534,132.62814755,0.0));
#259=CARTESIAN_POINT(' #259', (38.47821014,109.44948121,0.0));
#260=CARTESIAN_POINT(' #260', (87.02630611,74.50572274,0.0));
#261=CARTESIAN_POINT(' #261', (140.28037162,31.30872812,0.0));
#262=CARTESIAN_POINT(' #262', (144.74613919,-5.48598948,0.0));
#263=CARTESIAN_POINT(' #263', (132.77349428,-28.90239822,0.0));
#264=B_SPLINE_CURVE_WITH_KNOTS('', 3, (#256, #257, #258, #259, #260, #261, #262,
#263), .UNSPECIFIED., .F., .F., (4,1,1,1,1,4), (0.0,65.47002974,113.71553761,
191.84772524,248.5790028,306.27452289), .UNSPECIFIED.);
#265=GEOMETRIC_CURVE_SET(' #265', (#264));
#266=UNCERTAINTY_MEASURE_WITH_UNIT(LENGTH_MEASURE(0.1E-12), #4,
'distance_accuracy_value', 'EDGE CURVE AND VERTEX POINT ACCURACY');
#267=(GEOMETRIC_REPRESENTATION_CONTEXT(3)
GLOBAL_UNCERTAINTY_ASSIGNED_CONTEXT((#266))GLOBAL_UNIT_ASSIGNED_CONTEXT(
(#4, #5, #3))REPRESENTATION_CONTEXT('MASTER', '3D'));
#268=CARTESIAN_POINT(' #268', (0.0,0.0,0.0));
#269=DIRECTION(' #269', (1.0,0.0,0.0));
#270=DIRECTION(' #270', (0.0,0.0,1.0));
#271=AXIS2_PLACEMENT_3D(' #271', #268, #270, #269);
#272=ADVANCED_BREP_SHAPE_REPRESENTATION(' #272', (#213, #271), #267);
#273=(GEOMETRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CONTEXT(
(#4, #5, #3))REPRESENTATION_CONTEXT('MASTER', '3D'));
#274=CARTESIAN_POINT(' #274', (0.0,0.0,0.0));
#275=DIRECTION(' #275', (1.0,0.0,0.0));
#276=DIRECTION(' #276', (0.0,0.0,1.0));
#277=AXIS2_PLACEMENT_3D(' #277', #274, #276, #275);
#278=MANIFOLD_SURFACE_SHAPE_REPRESENTATION(' #278', (#255, #277), #273);
#279=SHAPE_REPRESENTATION_RELATIONSHIP(' #279', 'MASTER related shape',
#272, #278);
#280=(GEOMETRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CONTEXT(

```

```

#4,#5,#3))REPRESENTATION_CONTEXT('MASTER','3D'));
#281=CARTESIAN_POINT('#281',(0.0,0.0,0.0));
#282=DIRECTION('#282',(1.0,0.0,0.0));
#283=DIRECTION('#283',(0.0,0.0,1.0));
#284=AXIS2_PLACEMENT_3D('#284',#281,#283,#282);
#285=GEOMETRICALLY_BOUNDED_WIREFRAME_SHAPE_REPRESENTATION('#285',(#265,
#284),#280);
#286=SHAPE_REPRESENTATION_RELATIONSHIP('#286','MASTER related shape',
#272,#285);
#287=PRESENTATION_LAYER_ASSIGNMENT('020','layer 020',(#213));
#288=PRESENTATION_LAYER_ASSIGNMENT('010','layer 010',(#255));
#289=DRAUGHTING_PRE_DEFINED_COLOUR('cyan');
#290=FILL_AREA_STYLE_COLOUR('',#289);
#291=FILL_AREA_STYLE('',(#290));
#292=SURFACE_STYLE_FILL_AREA(#291);
#293=SURFACE_SIDE_STYLE('',(#292));
#294=SURFACE_STYLE_USAGE(.BOTH.,#293);
#295=PRESENTATION_STYLE_ASSIGNMENT((#294));
#296=STYLED_ITEM('',(#295),#213);
#297=DRAUGHTING_PRE_DEFINED_COLOUR('green');
#298=FILL_AREA_STYLE_COLOUR('',#297);
#299=FILL_AREA_STYLE('',(#298));
#300=SURFACE_STYLE_FILL_AREA(#299);
#301=SURFACE_SIDE_STYLE('',(#300));
#302=SURFACE_STYLE_USAGE(.BOTH.,#301);
#303=PRESENTATION_STYLE_ASSIGNMENT((#302));
#304=OVER_RIDING_STYLED_ITEM('',(#303),#175,#296);
#305=DRAUGHTING_PRE_DEFINED_COLOUR('yellow');
#306=DRAUGHTING_PRE_DEFINED_CURVE_FONT('continuous');
#307=CURVE_STYLE('',#306,POSITIVE_LENGTH_MEASURE(1.0),#305);
#308=PRESENTATION_STYLE_ASSIGNMENT((#307));
#309=OVER_RIDING_STYLED_ITEM('',(#308),#136,#296);
#310=DRAUGHTING_PRE_DEFINED_COLOUR('magenta');
#311=FILL_AREA_STYLE_COLOUR('',#310);
#312=FILL_AREA_STYLE('',(#311));
#313=SURFACE_STYLE_FILL_AREA(#312);
#314=SURFACE_SIDE_STYLE('',(#313));
#315=SURFACE_STYLE_USAGE(.BOTH.,#314);
#316=PRESENTATION_STYLE_ASSIGNMENT((#315));
#317=OVER_RIDING_STYLED_ITEM('',(#316),#199,#296);
#318=OVER_RIDING_STYLED_ITEM('',(#308),#136,#296);
#319=DRAUGHTING_PRE_DEFINED_COLOUR('red');
#320=FILL_AREA_STYLE_COLOUR('',#319);
#321=FILL_AREA_STYLE('',(#320));
#322=SURFACE_STYLE_FILL_AREA(#321);
#323=SURFACE_SIDE_STYLE('',(#322));
#324=SURFACE_STYLE_USAGE(.BOTH.,#323);
#325=PRESENTATION_STYLE_ASSIGNMENT((#324));
#326=STYLED_ITEM('',(#325),#255);
#327=DRAUGHTING_PRE_DEFINED_COLOUR('blue');
#328=DRAUGHTING_PRE_DEFINED_CURVE_FONT('continuous');
#329=CURVE_STYLE('',#328,POSITIVE_LENGTH_MEASURE(1.0),#327);
#330=PRESENTATION_STYLE_ASSIGNMENT((#329));
#331=STYLED_ITEM('',(#330),#265);
#332=(GEOMETRIC_REPRESENTATION_CONTEXT(3)GLOBAL_UNIT_ASSIGNED_CONTEXT((
#4,#5,#3))REPRESENTATION_CONTEXT('Presentation Context','3D'));
#333=CARTESIAN_POINT('#333',(0.0,0.0,0.0));
#334=DIRECTION('#334',(1.0,0.0,0.0));
#335=DIRECTION('#335',(0.0,0.0,1.0));
#336=AXIS2_PLACEMENT_3D('#336',#333,#335,#334);
#337=REPRESENTATION_MAP(#271,#272);

```

```
#338=MAPPED_ITEM( '#338', #337, #336 );  
#339=MECHANICAL_DESIGN_GEOMETRIC_PRESENTATION_REPRESENTATION( '#339',  
( #296, #304, #309, #317, #318, #326, #331, #336, #338 ), #332 );  
#340=PRODUCT_DEFINITION_SHAPE( 'MASTER', 'MASTER', #15 );  
#341=SHAPE_DEFINITION_REPRESENTATION( #340, #272 );  
ENDSEC;  
END-ISO-10303-21;
```

## 6 Availability of implementation schemas

### 6.1 AP214

The AP214 DIS schema supports the implementation of the capabilities as described. The schema can be retrieved on the Internet at:

<http://www.cax-if.org/> and <http://www.cax-if.de/> under 'Joint Testing Info'.

### 6.2 AP203 + Extensions

The long form EXPRESS schema combining AP 203 and colours and layers can be accessed via the Internet at:

<http://www.cax-if.org/> and <http://www.cax-if.de/> under 'Joint Testing Info'.