INTERNATIONAL STANDARD ISO 10303-46:1994
TECHNICAL CORRIGENDUM 2

Published 2002-05-15

INTERNATIONAL ORGANIZATION FOR STANDARDIZATION • МЕЖДУНАРОДНАЯ ОРГАНИЗАЦИЯ ПО СТАНДАРТИЗАЦИИ • ORGANISATION INTERNATIONALE DE NORMALISATION

Industrial automation systems and integration — Product data representation and exchange —

Part 46: Integrated generic resources: Visual presentation

TECHNICAL CORRIGENDUM 2

Systèmes d'automatisation industrielle et intégration — Représentation et échange de données de produits — Partie 46: Ressources génériques intégrées: Présentation visuelle

RECTIFICATIF TECHNIQUE 2

Technical Corrigendum 2 to International Standard ISO 10303-46:1994 was prepared by Technical Committee ISO/TC 184, *Industrial automation systems and integration*, Subcommittee SC 4, *Industrial data*.

Introduction

This corrigendum applies to ISO 10303-46:1994 as corrected by ISO 10303-46:1994/Cor.1:1999. For the convenience of the user, this corrigendum also includes the content of corrigendum 1.

The purpose of the modifications to the text of ISO 10303-46:1994 is to correct errors in the EXPRESS, to clarify a definition, to correct errors in Informal propositions and Formal propositions, to correct errors identified in the ballot for ISO 10303-518, and to replace the object identifier for the document and the schemas.

ICS 25.040.40 **Ref. No. ISO 10303-46:1994/Cor.2:2002(E)**

Modifications to the text of ISO 10303-46:1994

Clause 2, p. 2
The Normative references require an additional normative reference for the correction identified in clause 7.3.21. Add the following to the list of Normative references:

ISO 3098-0:1977, *Technical product documentation — Lettering — Part 0: General requirements*

Clause 4, p. 5
The EXPRESS specification of ***camera_image_3d_with_scale*** *and* ***aspect_ratio,*** *defined below, requires additional EXPRESS external references. Remove the following:*

```
REFERENCE FROM presentation_resource_schema
   (colour,
    planar_box,
    presentation_scaled_placement);

REFERENCE FROM measure_schema
   (length_measure,
    positive_plane_angle_measure);
```

Replace with the following:

```
REFERENCE FROM presentation_resource_schema
   (colour,
    planar_box,
    planar_extent,
    presentation_scaled_placement);

REFERENCE FROM measure_schema
   (length_measure,
    positive_ratio_measure,
    positive_plane_angle_measure);
```

The EXPRESS specification for the **presentation_organization_schema** *did not include a reference to required data type. The first required data type is an entity data type, the* **annotation_occurrence** *for the Formal propositions in* **area_dependent_annotation_representation** *and* **view_dependent_annotation_representation**. *The second required data type is an entity data type, the symbol_representation for the Formal propositions in* **symbol_representation_rule**. *The third required data type is an entity data type, the* **symbol_representation_relationship** *for the Formal propositions in* **symbol_representation_rule**. *The fourth required data type is an entity data type, the* **styled_item** *for the Formal propositions in* **camera_model** *and* **light_source**. *The fifth required data type is an entity data type, the* **founded_item**. *It is required to be referenced since it is now a supertype of* **view_volume**. *Add the following to the EXPRESS specification between the 'SCHEMA presentation_organization_schema;' and the 'REFERENCE FROM presentation_resource_schema':*

```
REFERENCE FROM presentation_definition_schema
    (annotation_occurrence,
     symbol_representation,
     symbol_representation_relationship);

REFERENCE FROM presentation_appearance_schema
    (styled_item);
```

Delete the following EXPRESS specification:

```
REFERENCE FROM representation_schema
    (item_defined_transformation,
     item_in_context,
     mapped_item,
     representation,
     representation_item,
     representation_map,
     representation_relationship,
     representation_relationship_with_transformation);
```

Replace with the following EXPRESS specification:

```
REFERENCE FROM representation_schema
    (founded_item,
     item_defined_transformation,
     item_in_context,
     mapped_item,
     representation,
     representation_item,
     representation_map,
     representation_relationship,
     representation_relationship_with_transformation);
```

With the addition of the **annotation_occurrence**, **symbol_representation**, **symbol_representation_relationship** *and* **styled_item** *to the* **presentation_organization_schema**, *NOTE 1 changed. Delete NOTE 1 and replace with the following:*

NOTE 1 The schemas referenced above can be found in the following parts of ISO 10303:

Presentation_definition_schema	Clause 5 of this part of ISO 10303
Presentation_appearance_schema	Clause 6 of this part of ISO 10303
Presentation_resource_schema	Clause 7 of this part of ISO 10303
Geometry_schema	ISO 10303-42
Representation_schema	ISO 10303-43
Measure_schema	ISO 10303-41
Support_resource_schema	ISO 10303-41

Clause 4.3.45, p. 13

The Informal proposition of **layered_item** *contradicts to the intended use of* **presentation_layer_assignment**. *The type of* **representation_item**s *assigned to a layer shall not be restricted. Remove Informal proposition IP1.*

Clause 4.5.5, p. 26

The EXPRESS specification of **view_volume** *is revised to make it a subtype of* **founded_item** *in order to provide a representation context for the* **projection_point** *and* **planar_box** *attributes. Remove the EXPRESS specification and replace with the following:*

EXPRESS specification:

```
*)
ENTITY view_volume
  SUBTYPE OF (founded_item);
  projection_type            : central_or_parallel;
  projection_point           : cartesian_point;
  view_plane_distance        : length_measure;
  front_plane_distance       : length_measure;
  front_plane_clipping       : BOOLEAN;
  back_plane_distance        : length_measure;
  back_plane_clipping        : BOOLEAN;
  view_volume_sides_clipping : BOOLEAN;
  view_window                : planar_box;
END_ENTITY;
(*
```

Add the following note at the end of the entity description:

NOTE Since **view_volume** is not a subtype of **geometric_representation_item** the instances of **cartesian_point** which is the **projection_point** attribute and **planar_box** which is the **view_window** attribute are not associated in the usual way with the **geometric_representation_context** of each **representation** using a **camera_model_d3** containing this **view_volume**. The **geometric_representation_context** is associated via the **founded_item** supertype.

Clause 4.5.9, p. 31

The EXPRESS specification of **light_source** *contained logical errors in the WHERE rule. WR1 requires a role name qualified by attribute name 'ITEM' for argument 2 of built-in function USEDIN. Delete the current WR1 and replace WR1 with the following:*

```
WR1: SIZEOF(USEDIN(SELF,'PRESENTATION_APPEARANCE_SCHEMA.'+
          'STYLED_ITEM.ITEM')) = 0;
```

Clause 4.5.14, p. 35

The description of the Formal propositions does not give a correct explanation of WR2. Remove the description of WR2 and replace with the following:

WR2: The target of the mapping shall be a **planar_box**.

Clause 4.5.16, p. 35

The EXPRESS specification for ***camera_image_3d_with_scale*** *defined below are required for reference from other parts of ISO 10303. Add the following as clause 4.5.16 after clause 4.5.15*

4.5.16 camera_image_3d_with_scale

A **camera_image_3d_with_scale** is a **camera_image** that projects three-dimensional geometry and has a derived scale. The scale is the ratio between the size of the viewport and the size of the **view_window** of the **view_volume**.

EXPRESS specification:

```
*)
ENTITY camera_image_3d_with_scale
  SUBTYPE OF (camera_image);
DERIVE
  scale: positive_ratio_measure := ((SELF\mapped_item.mapping_target\
         planar_extent.size_in_x) / (SELF\mapped_item.mapping_source.
         mapping_origin\camera_model_d3.perspective_of_volume.view_window.
         size_in_x));
WHERE
  WR1: ('PRESENTATION_ORGANIZATION_SCHEMA.CAMERA_MODEL_D3'
       IN TYPEOF (SELF\mapped_item.mapping_source.mapping_origin));
  WR2: aspect_ratio(SELF\mapped_item.mapping_target) =
       aspect_ratio(SELF\mapped_item.mapping_source.mapping_origin\
       camera_model_d3.perspective_of_volume.view_window);
  WR3: SELF\mapped_item.mapping_source.mapping_origin\camera_model_d3.
       perspective_of_volume.front_plane_clipping
       AND
       SELF\mapped_item.mapping_source.mapping_origin\camera_model_d3.
       perspective_of_volume.view_volume_sides_clipping;
  WR4: (SELF\mapped_item.mapping_target\planar_extent.size_in_x > 0)
       AND
       (SELF\mapped_item.mapping_target\planar_extent.size_in_y > 0);
  WR5: (SELF\mapped_item.mapping_source.mapping_origin\camera_model_d3.
       perspective_of_volume.view_window.size_in_x > 0)
       AND
       (SELF\mapped_item.mapping_source.mapping_origin\camera_model_d3.
       perspective_of_volume.view_window.size_in_y > 0);
  WR6: ('GEOMETRY_SCHEMA.' +
       'AXIS2_PLACEMENT_2D' IN TYPEOF (SELF\mapped_item.
       mapping_target\planar_box.placement))
       AND NOT ('GEOMETRY_SCHEMA.' +
       'AXIS2_PLACEMENT_3D' IN TYPEOF (SELF\mapped_item.
       mapping_target\planar_box.placement));
END_ENTITY;
(*
```

Attribute definitions:

scale: the **positive_ratio_measure** derived from the rectangular size of the viewport and the rectangular size of the **view_volume** of the **camera_model**.

Formal propositions:

WR1: The source of the projection shall be a **camera_model_d3**.

WR2: The aspect ratio of the viewport shall equal the aspect ratio of the **view_window** of the **view_volume**.

WR3: The geometry of the projected representation shall be clipped against the plane represented by the **front_plane_distance** and the planes which are the sides of the volume defined by the **view_volume**.

WR4: The rectangular size of the viewport shall be specified by positive values.

WR5: The rectangular size of the **view_window** shall be specified by positive values.

WR6: The drawing space of a **camera_image_3d_with_scale** shall be specified in a 2D coordinate system.

Informal propositions:

IP1: The horizontal and vertical components of the viewport shall be parallel to the corresponding components of the **view_window** of the **view_volume**.

Clause 4.9.1, p. 39
The EXPRESS specification for the FUNCTION **acyclic_presentation_representation_relationship** *contained logical errors in the function body. The assignment to variable 'x' requires a 'SET' and not a 'BAG'. Remove the EXPRESS specification and replace with the following:*

EXPRESS specification:

```
*)
FUNCTION acyclic_presentation_representation_relationship
  ( relation : presentation_representation_relationship;
    children : SET OF presentation_representation ) : BOOLEAN;

  LOCAL
    x : SET OF presentation_representation_relationship;
    local_children : SET OF presentation_representation;
  END_LOCAL;

  REPEAT i:=1 TO HIINDEX(children);
    IF relation\representation_relationship.rep_1 :=: children[i] THEN
       RETURN(FALSE);
    END_IF;
  END_REPEAT;

  x := bag_to_set (USEDIN ( relation\representation_relationship.rep_1,
                  'REPRESENTATION_SCHEMA.'+
                  'REPRESENTATION_RELATIONSHIP.REP_2'));
  local_children := children + relation\representation_relationship.rep_1;
```

```
  IF SIZEOF (x) > 0 THEN
    REPEAT i:=1 TO HIINDEX (x);
       IF NOT acyclic_presentation_representation_relationship
              (x[i] , local_children) THEN
         RETURN (FALSE);
       END_IF;
    END_REPEAT;
  END_IF;

  RETURN (TRUE);

END_FUNCTION;
(*
```

Clause 4.9.2, p.39

The EXPRESS specification for ***aspect_ratio*** *defined below are required for reference from other parts of ISO 10303. This entity was incorrectly defined in ISO 10303-517. Add the following as clause 4.9.2 after clause 4.9.1 and before the END_SCHEMA EXPRESS specification:*

4.9.2 aspect ratio

The **aspect_ratio** function checks that both the attributes, size_in_x and size_in_y, have positivie values and returns a **positive_ratio_measure** that is the ratio of length to height for a given **planar_box**. In other cases, an indeterminate value is returned.

EXPRESS specification:

```
*)
FUNCTION aspect_ratio (p : planar_box) : positive_ratio_measure;
(* if the dimensions of the planar_box are greater than zero,
      compute the aspect ratio and return the resulting value. *)
   IF (p.size_in_x > 0.) AND (p.size_in_y > 0.) THEN
      RETURN (p.size_in_x / p.size_in_y);
   ELSE
      RETURN (?);
   END_IF;
END_FUNCTION;
(*
```

Argument definitions:

p: The input **planar_box** to be checked.

Clause 5, p. 40

The EXPRESS specification for the **presentation_definition_schema** *did not include a reference to a required data type. The required reference is a function, the* **bag_to_set** *for the EXPRESS specifications changed in* **acyclic_presentation_representation_relationship**, **acyclic_symbol_representation_relationship** *and* **field_in_table**. *Delete the following EXPRESS specification:*

```
REFERENCE FROM support_resource_schema
    (label,
     text);
```

Replace with the following EXPRESS specification:

```
REFERENCE FROM support_resource_schema
    (label,
     text,
     bag_to_set);
```

Clause 5.4.13, p.53
The EXPRESS specification for ***table_record_representation*** *was incorrect. The local rules of table_record_representation are incorrect since the variable map_item is of type REPRESENTATION,, but it is used as argument to the function using_representations, which accepts only variables of type FOUNDED_ITEM_SELECT. Remove the EXPRESS specification and replace with the following:*

EXPRESS specification:

```
*)
ENTITY table_record_representation
  SUBTYPE OF (symbol_representation);
WHERE
  WR1: (SIZEOF(USEDIN(SELF, 'REPRESENTATION_SCHEMA.'+
                            'REPRESENTATION_RELATIONSHIP.REP_2')) > 0)
                       OR
       (SIZEOF(QUERY( map_item <* USEDIN(SELF, 'REPRESENTATION_SCHEMA.'+
                                               'REPRESENTATION_MAP.'+
                                               'MAPPED_REPRESENTATION') |
         SIZEOF(QUERY( mi <* USEDIN(map_item, 'REPRESENTATION_SCHEMA.'+
                                              'MAPPED_ITEM.'+
                                              'MAPPING_SOURCE') |
                                    'PRESENTATION_DEFINITION_SCHEMA.'+
                                              'TABLE_REPRESENTATION' IN
            TYPEOF (using_representations (mi)) )) > 0))
                  > 0);
END_ENTITY;
(*
```

Clause 5.4.14, p.54
The EXPRESS specification for ***table_record_field_representation*** *was incorrect. The local rules of table_record_field_representation are incorrect since the variable map_item is of type REPRESENTATION,, but it is used as argument to the function using_representations, which accepts only variables of type FOUNDED_ITEM_SELECT. Remove the EXPRESS specification and replace with the following:*

EXPRESS specification:

```
*)
ENTITY table_record_field_representation
  SUBTYPE OF (symbol_representation);
WHERE
  WR1: (SIZEOF(USEDIN(SELF, 'REPRESENTATION_SCHEMA.'+
                            'REPRESENTATION_RELATIONSHIP.REP_2')) > 0)
                       OR
       (SIZEOF(QUERY( map_item <* USEDIN(SELF, 'REPRESENTATION_SCHEMA.'+
                                               'REPRESENTATION_MAP.'+
                                               'MAPPED_REPRESENTATION') |
         SIZEOF(QUERY( mi <* USEDIN(map_item, 'REPRESENTATION_SCHEMA.'+
                                              'MAPPED_ITEM.'+
```

```
                                                    'MAPPING_SOURCE') |
                                                'PRESENTATION_DEFINITION_SCHEMA.'+
                                                 'TABLE_RECORD_REPRESENTATION' IN
                    TYPEOF (using_representations (mi)) )) > 0))
                           > 0);
END_ENTITY;
(*
```

Clause 5.6.2, p. 72

The EXPRESS specification for the FUNCTION **acyclic_symbol_representation_relationship** *contained logical errors in the function body. The assignment to variable 'x' requires a 'SET' and not a 'BAG'. Remove the EXPRESS specification and replace with the following:*

EXPRESS specification:

```
*)
FUNCTION acyclic_symbol_representation_relationship
  (relation : symbol_representation_relationship;
   children : SET OF symbol_representation ) : BOOLEAN;
  LOCAL
    x : SET OF symbol_representation_relationship;
    local_children : SET OF symbol_representation;
  END_LOCAL;

  REPEAT i:=1 TO HIINDEX(children);
    IF relation\representation_relationship.rep_1 :=: children[i] THEN
      RETURN(FALSE);
    END_IF;
  END_REPEAT;

  x := bag_to_set (USEDIN ( relation\representation_relationship.rep_1,
                'REPRESENTATION_SCHEMA.'+
                'REPRESENTATION_RELATIONSHIP.'+ 'REP_2'));
  local_children := children + relation\representation_relationship.rep_1;

  IF SIZEOF (x) > 0 THEN
    REPEAT i:=1 TO HIINDEX (x);
      IF NOT acyclic_symbol_representation_relationship(x[i] ,
                                                  local_children) THEN
        RETURN (FALSE);
      END_IF;
    END_REPEAT;
  END_IF;

  RETURN (TRUE);

END_FUNCTION;
(*
```

Clause 5.6.3, p. 73

The EXPRESS specification for the FUNCTION **field_in_table** *contained spelling and logical errors. The expression in the first QUERY requires a string 'PRESENTATION_DEFINITION_SCHEMA. TABLE_RECORD_REPRESENTATION' and not a string 'PRESENTATION_DEFINITIONS_ SCHEMA.TABLE_RECORD_REPRESENTATION'. The declaration of variable 'symbol_rep_rel_set' requires a 'SET' and not a 'SET[1:?]'. The declaration of variable mapped_item_set' requires a 'SET' and not a 'SET[1:?]'. The declaration of variable 'table_record_rep_set' requires a 'SET' and not a 'SET[1:?]'.The assignment to variable 'symbol_rep_rel_set' requires a 'SET' and not a 'BAG'. The built-in function USEDIN in the second QUERY requires a role name qualified by an attribute name as argument 2. Remove the EXPRESS specification and replace with the following:*

EXPRESS specification:

```
*)
FUNCTION field_in_table (field : table_record_field_representation;
                         table : annotation_table_occurrence): BOOLEAN;
  LOCAL
    table_rep : table_representation;
    symbol_rep_rel_set : SET OF symbol_representation_relationship;
    mapped_item_set : SET OF mapped_item;
    table_record_rep_set : SET OF table_record_representation := [];
  END_LOCAL;

  table_rep := table\styled_item.item\mapped_item.mapping_source.
    mapped_representation;
  mapped_item_set := QUERY(item <* table_rep.items |
                      ('REPRESENTATION_SCHEMA.MAPPED_ITEM' IN
                       TYPEOF(item))
                                AND
                      ('PRESENTATION_DEFINITION_SCHEMA.'+
                       'TABLE_RECORD_REPRESENTATION' IN
                        TYPEOF(item\mapped_item.mapping_source.
                                    mapped_representation ))
                     );
REPEAT i := 1 TO HIINDEX(mapped_item_set);
    table_record_rep_set := table_record_rep_set +
           mapped_item_set[i].mapping_source.mapped_representation;
  END_REPEAT;

  symbol_rep_rel_set := bag_to_set (USEDIN(table_rep,
                                'REPRESENTATION_SCHEMA.'+
                                'REPRESENTATION_RELATIONSHIP.REP_1'));

  REPEAT i := 1 TO HIINDEX(symbol_rep_rel_set);
     table_record_rep_set := table_record_rep_set +
              symbol_rep_rel_set[i]\representation_relationship.rep_2;
  END_REPEAT;
```

```
  IF SIZEOF(QUERY( table_record_rep <* table_record_rep_set |
        (SIZEOF(QUERY( rep_rel <* USEDIN(table_record_rep,
                              'REPRESENTATION_SCHEMA.'+
                              'REPRESENTATION_RELATIONSHIP.REP_1') |
                        ('PRESENTATION_DEFINITION_SCHEMA.' +
                        'SYMBOL_REPRESENTATION_RELATIONSHIP' IN
                        TYPEOF(rep_rel))
                        AND
                        (rep_rel.rep_2 :=: field)
                        )) > 0)
                        OR
                (SIZEOF(QUERY(item <* table_record_rep.items |
                          ('REPRESENTATION_SCHEMA.MAPPED_ITEM' IN
                           TYPEOF(item))
                                  AND
                          (field :=: item\mapped_item.mapping_source.
                                     mapped_representation )
                           )) > 0)
             )) = 0 THEN
    RETURN(FALSE);
  END_IF;

  RETURN(TRUE);

END_FUNCTION;
(*
```

Clause 6, p. 74

The EXPRESS specification for the **presentation_appearance_schema** *did not include a reference to required data type. The first required data type is an entity data type, the group for the amended SELECT type* **style_context_select**. *The second required reference is a function, the* **bag_to_set** *for the EXPRESS specifications changed in* **acyclic_occlusion_precedence**. *Add the following EXPRESS specification before the 'REFERENCE FROM MEASURE_SCHEMA':*

```
REFERENCE FROM group_schema
    (group);
```

Delete the following EXPRESS specification:

```
REFERENCE FROM support_resource_schema
    (label);
```

Replace with the following EXPRESS specification:

```
REFERENCE FROM support_resource_schema
    (label,
     bag_to_set);
```

Clause 6.3.1, p. 80
The possibility to control the presentation style by a layer is a fundamental concept of ISO 10303-46. However the EXPRESS specification for the type **style_context_select** *did not include the necessary entities. Remove the EXPRESS specification and replace with the following:*

EXPRESS specification:

```
*)
TYPE style_context_select = SELECT
  (group,
   presentation_layer_assignment,
   representation,
   representation_item,
   presentation_set);
END_TYPE;
(*
```

Clause 6.3.43, p. 96
The restriction of invisibility to **presentation_representation** *does not satisfy the requirement to define a complete model as invisible. Include the entity representation instead of* **presentation_representation** *in the SELECT type* **invisible_item**. *Remove the EXPRESS specification and replace with the following:*

EXPRESS specification:

```
*)
TYPE invisible_item = SELECT
  (styled_item,
   presentation_layer_assignment,
   representation);
END_TYPE;
(*
```

Clause 6.6.12, p.106
The EXPRESS specification for ***draughting_pre_defined_curve_font*** *defined below are required for reference from other parts of ISO 10303. Add the following as clause 6.6.12 after clause 6.6.11.*

6.6.12 draughting_pre_defined_curve_font

A **draughting_pre_defined_curve_font** is a **pre_defined_curve_font** that is identified by name.

EXPRESS specification:

```
*)
ENTITY draughting_pre_defined_curve_font
  SUBTYPE OF (pre_defined_curve_font);
WHERE
  WR1: SELF.name IN
```

```
        ['continuous',
         'chain',
         'chain double dash',
         'dashed',
         'dotted'];
END_ENTITY;
(*
```

Formal propositions:

WR1: The name of the **draughting_pre_defined_curve_font** shall be 'continuous', 'chain', 'chain double dash', 'dashed', or 'dotted'.

Attribute value definitions:

Table 2 states the lengths of each line segment and space, in millimetres, corresponding to each of the predefined curve fonts that are specified in this part of ISO 10303. If the **pre_defined_curve_font** is used as part of the definition of a **curve_style_font_and_scaling**, then the given lengths are those when the **curve_font_scaling** attribute has the value 1.0.

NOTE 1 - The **curve_style_font_and_scaling** entity is defined in the **presentation_appearance_schema** in ISO 10303-46.

NOTE 2 - Illustrations of curve fonts are given in Figure 1.

Table 2 – Line segment and space lengths for predefined curve fonts

Curve pattern name	Segment (mm)	Space (mm)	Segment (mm)	Space (mm)	Segment (mm)	Space (mm)	Number of segments
continuous							0
dashed	4.0	1.5					2
Chain	7.0	1.0	1.0	1.0			4
Chain double dash	7.0	1.0	1.0	1.0	1.0	1.0	6
dotted	1.0	1.0					2

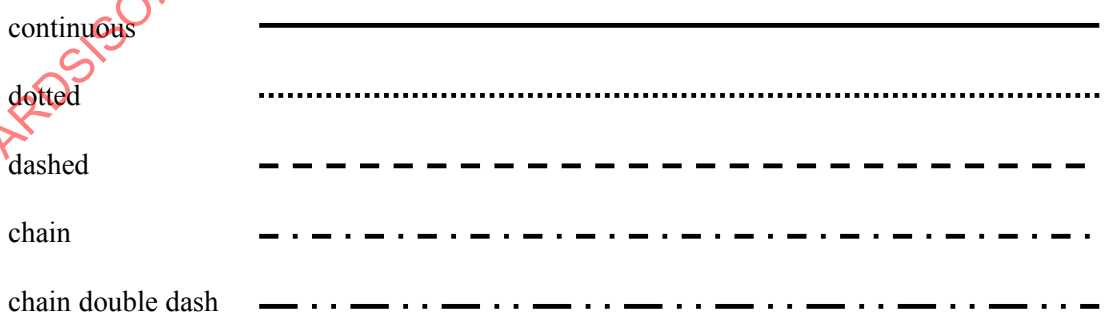

Figure 1 – Illustration of predefined curve fonts

Clause 6.9.10, p.124
*The EXPRESS specification for **text_style_with_mirror** does not specify in the definition or in the EXPRESS specification that the axis_2_placement has to be founded in the appropriate context. Add the following paragraph after Attribute definitions: and before clause 6.9.11.*

Informal propositions:

IP1: Text_style_with_mirror.mirror_placement shall have the axis2_placement founded in the same context as the text that is being mirrored.

Clause 6.13.1, p. 130

The EXPRESS specification for the FUNCTION **acyclic_occlusion_precedence** *contained logical errors in the function body. The assignment to variable 'x' requires a 'SET' and not a 'BAG'. Remove the EXPRESS specification and replace with the following:*

EXPRESS specification:

```
*)
FUNCTION acyclic_occlusion_precedence
   ( relation : occlusion_precedence;
     set_of_lower : SET OF hiding_or_blanking_select ) : BOOLEAN;
   LOCAL
      x : SET OF occlusion_precedence;
      local_set_of_lower : SET OF hiding_or_blanking_select;
   END_LOCAL;
   REPEAT i:=1 TO HIINDEX(set_of_lower);
      IF relation.higher_precedence :=: set_of_lower[i] THEN
         RETURN(FALSE);
      END_IF;
   END_REPEAT;
   x := bag_to_set (USEDIN ( relation.higher_precedence,
                   'PRESENTATION_APPEARANCE_SCHEMA.'+
           'OCCLUSION_PRECEDENCE.LOWER_PRECEDENCE'));
   local_set_of_lower := set_of_lower + relation.higher_precedence;
   IF SIZEOF (x) > 0 THEN
      REPEAT i:=1 TO HIINDEX (x);
         If NOT acyclic_occlusion_precedence(x[i] ,
                                             local_set_of_lower) THEN
            RETURN (FALSE);
         END_IF;
      END_REPEAT;
   END_IF;
   RETURN (TRUE);
END_FUNCTION;
(*
```

Clause 7.3.13, p.139

The EXPRESS specification of **colour_associated** *contained logical errors in the attribute declaration. Attribute 'name' requires a type 'label' and not 'colour'. Remove the EXPRESS specification and replace with the following:*