



INTERNATIONAL STANDARD ISO 10303-42:2000
TECHNICAL CORRIGENDUM 1

Published 2001-04-01

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

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

Part 42:

Integrated generic resource: Geometric and topological representation

TECHNICAL CORRIGENDUM 1

*Systèmes d'automatisation industrielle et intégration — Représentation et échange de données de produits —
Partie 42: Ressource générique intégrée: Représentation géométrique et topologique*

RECTIFICATIF TECHNIQUE 1

Technical Corrigendum 1 to International Standard ISO 10303-42:2000 was prepared by Technical Committee ISO/TC 184, *Industrial automation systems and integration*, Subcommittee SC 4, *Industrial data*.

Introduction

This Technical Corrigendum corrects ISO 10303-42:2000, Industrial automation systems and integration — Product data representation and exchange — Part 42: Integrated generic resource: Geometric and topological representation.

The purpose of the modifications to the text of ISO 10303-42:1994 is to correct errors in the EXPRESS definitions likely to cause compilation problems, to replace the URL in the annex for the computer-interpretable EXPRESS, and to replace the object identifier for the document and the applicable schema.

Modifications to the text of ISO 10303-42:2000**Clause 4.4.64, p. 98**

The EXPRESS specification of **surface_of_revolution** contains an error in the DERIVE statement. The usage here of **dummy_gri** is not supported by all compilers. Remove the EXPRESS specification and replace with the following:

EXPRESS specification:

```

*)
ENTITY surface_of_revolution
  SUBTYPE OF (swept_surface);
  axis_position      : axis1_placement;
DERIVE
  axis_line : line := representation_item('')||
                        geometric_representation_item()|| curve()||
                        line(axis_position.location, representation_item('')||
                        geometric_representation_item()||
                        vector(axis_position.z, 1.0)),
END_ENTITY;
( *

```

Clause 4.6.10, p. 151

The EXPRESS specification of function **first_proj_axis** does not provide a valid result when called with **z_axis** in the direction [-1,0,0] and **arg** defaulted. Replace the second sentence of the description with:

With **arg** defaulted the result is the projection of [1,0,0] onto this plane; except that, if **z_axis** = [1,0,0], or, **z_axis** = [-1,0,0], [0,1,0] is the default for **arg**.

Remove the EXPRESS specification and replace with the following:

EXPRESS specification:

```

*)
FUNCTION first_proj_axis(z_axis, arg : direction) : direction;
  LOCAL
    x_axis : direction;
    v      : direction;
    z      : direction;
    x_vec  : vector;
  END_LOCAL;

  IF (NOT EXISTS(z_axis)) THEN
    RETURN (?);
  ELSE

```

```

z := normalise(z_axis);
IF NOT EXISTS(arg) THEN
  IF ((z.direction_ratios <> [1.0,0.0,0.0]) AND
      (z.direction_ratios <> [-1.0,0.0,0.0])) THEN
    v := dummy_gri || direction([1.0,0.0,0.0]);
  ELSE
    v := dummy_gri || direction([0.0,1.0,0.0]);
  END_IF;
ELSE
  IF (arg.dim <> 3) THEN
    RETURN (?);
  END_IF;
  IF ((cross_product(arg,z).magnitude) = 0.0) THEN
    RETURN (?);
  ELSE
    v := normalise(arg);
  END_IF;
END_IF;
x_vec := scalar_times_vector(dot_product(v, z), z);
x_axis := vector_difference(v, x_vec).orientation;
x_axis := normalise(x_axis);
END_IF;
RETURN(x_axis);
END_FUNCTION;
(*

```

Clause 4.6.29, p. 171

The EXPRESS specification of the function **list_to_array** omits the array dimensions for the return variable. Remove the EXPRESS specification and replace with the following:

EXPRESS specification:

```

*)
FUNCTION list_to_array(lis : LIST [0:?] OF GENERIC : T;
                      low,u : INTEGER) : ARRAY [low:u] OF GENERIC : T;
  LOCAL
    n : INTEGER;
    res : ARRAY [low:u] OF GENERIC : T;
  END_LOCAL;

  n := SIZEOF(lis);
  IF (n <> (u-low+1)) THEN
    RETURN(?);
  ELSE
    res := [lis[1] : n];
    REPEAT i := 2 TO n;
      res[low+i-1] := lis[i];
    END_REPEAT;
    RETURN(res);
  END_IF;
END_FUNCTION;

```

```

    END_IF;
  END_FUNCTION;
  (*

```

Clause 4.6.30, p. 172

The EXPRESS specification of the function **make_array_of_array** omits the array dimensions for the return variable. Remove the EXPRESS specification and replace with the following:

EXPRESS specification:

```

  *)
  FUNCTION make_array_of_array(lis : LIST[1:?] OF LIST [1:?] OF GENERIC : T;
                                low1, u1, low2, u2 : INTEGER):
    ARRAY [low1:u1] OF ARRAY [low2:u2] OF GENERIC : T;

  LOCAL
    res : ARRAY[low1:u1] OF ARRAY [low2:u2] OF GENERIC : T;
  END_LOCAL;

  (* Check input dimensions for consistency *)
  IF (u1-low1+1) <> SIZEOF(lis) THEN
    RETURN (?);
  END_IF;
  IF (u2 - low2 + 1 ) <> SIZEOF(lis[1]) THEN
    RETURN (?) ;
  END_IF;
  (* Initialise res with values from lis[1] *)
  res := [list_to_array(lis[1], low2, u2) : (u1-low1 + 1)];
  REPEAT i := 2 TO HIINDEX(lis);
    IF (u2-low2+1) <> SIZEOF(lis[i]) THEN
      RETURN (?);
    END_IF;
    res[low1+i-1] := list_to_array(lis[i], low2, u2);
  END_REPEAT;

  RETURN (res);
END_FUNCTION;
  (*

```

Clause 4.6.31, p. 173

The EXPRESS specification of the function **make_array_of_array_of_array** omits the array dimensions for the return variable. Remove the EXPRESS specification and replace with the following:

EXPRESS specification:

```

  *)
  FUNCTION make_array_of_array_of_array(lis : LIST[1:?] OF
    LIST [1:?] OF LIST [1:?] OF GENERIC : T;

```

```

        low1, u1, low2, u2, low3, u3 : INTEGER);
    ARRAY[low1:u1] OF ARRAY[low2:u2] OF ARRAY[low3:u3] OF GENERIC : T;
LOCAL
    res : ARRAY[low1:u1] OF ARRAY [low2:u2] OF
        ARRAY[low3:u3] OF GENERIC : T;
END_LOCAL;

(* Check input dimensions for consistency *)
IF (u1-low1+1) <> SIZEOF(lis) THEN
    RETURN (?);
END_IF;
IF (u2-low2+1) <> SIZEOF(lis[1]) THEN
    RETURN (?);
END_IF;
(* Initialise res with values from lis[1] *)
res := [make_array_of_array(lis[1], low2, u2, low3, u3) : (u1-low1 + 1)];
REPEAT i := 2 TO HIINDEX(lis);
    IF (u2-low2+1) <> SIZEOF(lis[i]) THEN
        RETURN (?);
    END_IF;
    res[low1+i-1] := make_array_of_array(lis[i], low2, u2, low3, u3);
END_REPEAT;
RETURN (res);
END_FUNCTION;
(*

```

Clause 6 p. 231

The EXPRESS specification of **geometric_model_schema** requires an additional external reference to support the corrected definition of the **box_domain** and **rectangle_domain** entities. Remove the definition and the EXPRESS specification and replace with the following:

The following EXPRESS declaration begins the **geometric_model_schema** and identifies the necessary external references.

EXPRESS specification:

```

*)
SCHEMA geometric_model_schema;
    REFERENCE FROM geometry_schema;
    REFERENCE FROM topology_schema;
    REFERENCE FROM measure_schema(length_measure,
        positive_length_measure,
        plane_angle_measure,
        plane_angle_unit,
        positive_plane_angle_measure);
    REFERENCE FROM representation_schema(founded_item);
(*

```

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

geometry_schema	Clause 4 of this part of ISO 10303
topology_schema	Clause 5 of this part of ISO 10303
measure_schema	ISO 10303-41
representation_schema	ISO 10303-43

NOTE 2 - See annex D, figures D.16 - D.18, for a graphical presentation of this schema.

Clause 6.4.34 p. 265

The EXPRESS specification of **revolved_face_solid** contains an error in the DERIVE statement. The usage here of **dummy_gri** is not supported by all compilers. Remove the EXPRESS specification and replace with the following:

EXPRESS specification:

```

*)
ENTITY revolved_face_solid SUBTYPE OF (swept_face_solid);
  axis : axis1_placement;
  angle : plane_angle_measure;
DERIVE
  axis_line : line := representation_item('')||
                        geometric_representation_item()|| curve()||
                        line(axis.location, representation_item('')||
                        geometric_representation_item()||
                        vector(axis.z, 1.0));
END_ENTITY;
( *

```

Clause 6.4.38 p. 269

The EXPRESS specification of **revolved_area_solid** contains an error in the DERIVE statement. The usage here of **dummy_gri** is not supported by all compilers. Remove the EXPRESS specification and replace with the following:

EXPRESS specification:

```

*)
ENTITY revolved_area_solid
  SUBTYPE OF (swept_area_solid);
  axis : axis1_placement;
  angle : plane_angle_measure;
DERIVE
  axis_line : line := representation_item('')||
                        geometric_representation_item()|| curve()||
                        line(axis.location, representation_item('')||

```

```

        geometric_representation_item() ||
        vector(axis.z, 1.0));
END_ENTITY;
(*

```

Clause 6.4.23, p. 258

The EXPRESS specification of the **box_domain** entity fails to ensure that the **cartesian_point** used to define the **corner** attribute is geometrically founded. This is corrected by re-defining **box_domain** as a subtype of **founded_item**. Remove the EXPRESS specification and replace with the following:

EXPRESS specification:

```

*)
ENTITY box_domain
  SUBTYPE OF (founded_item);
  corner : cartesian_point;
  xlength : positive_length_measure;
  ylength : positive_length_measure;
  zlength : positive_length_measure;
WHERE
  WR1: SIZEOF(QUERY(item <* USEDIN(SELF, '' )|
    NOT ('GEOMETRIC_MODEL_SCHEMA.BOXED_HALF_SPACE'
      IN TYPEOF(item)))) = 0;
END_ENTITY;
(*

```

Clause 6.4.31, p. 263

The EXPRESS specification of the **rectangle_domain** entity fails to ensure that the **cartesian_point** used to define the **corner** attribute is geometrically founded. This is corrected by re-defining **box_domain** as a subtype of **founded_item**. Remove the EXPRESS specification and replace with the following:

EXPRESS specification:

```

*)
ENTITY rectangle_domain
  SUBTYPE OF (founded_item);
  corner: cartesian_point;
  xlength: positive_length_measure;
  ylength: positive_length_measure;
WHERE
  WR1: SIZEOF(QUERY(item <* USEDIN(SELF, '' )| NOT
    ('GEOMETRIC_MODEL_SCHEMA.RECTANGLED_HALF_SPACE' IN TYPEOF(item)))) = 0;
END_ENTITY;
(*

```

Clause 6.5.5, p. 283

The EXPRESS specificatuion of the function **build_transformed_set** creates incomplete instances of **curve_replica**, **point_replica** and **surface_replica**. This is corrected in the replacement definition below. Remove the existing EXPRESS specification and replace with the following:

EXPRESS specification:

```

*)
FUNCTION build_transformed_set(tr: cartesian_transformation_operator;
                               gset : geometric_set) : SET [0:?] OF geometric_set_select;
LOCAL
  s          : SET [1:?] OF geometric_set_select := gset.elements;
  trset      : SET [0:?] OF geometric_set_select := [];
END_LOCAL;
REPEAT j := 1 TO SIZEOF(s);
  IF ('GEOMETRY_SCHEMA.CURVE' IN TYPEOF(s[j])) THEN
    trset := trset + dummy_gri || curve() || curve_replica(s[j],tr); ELSE
    IF ('GEOMETRY_SCHEMA.POINT' IN TYPEOF(s[j])) THEN
      trset := trset + dummy_gri || point() || point_replica(s[j],tr);
    ELSE
      IF ('GEOMETRY_SCHEMA.SURFACE' IN TYPEOF(s[j])) THEN
        trset := trset + dummy_gri || surface() || surface_replica(s[j],
          tr || cartesian_transformation_operator_3d (?));
      END_IF;
    END_IF;
  END_IF;
END_REPEAT;
RETURN(trset);
END_FUNCTION;
( *

```

Annex A, p. 290

The list of short names of entities contains two spelling mistakes. Replace the entries for **PYRAMID_VOLUME** and **QUASI_UNIFORM_VOLUME** in the existing table with the corrected versions:

Entity names	Short names
PYRAMID_VOLUME	PYRVLM
QUASI_UNIFORM_VOLUME	QSUNVL

Annex B, p. 202

With the changes identified in this Technical Corrigendum the document identifiers and the schema information object identifiers have changed. Remove the object identifier for the part and replace with the following:

To provide for unambiguous identification of an information object in an open system, the object identifier

{ iso standard 10303 part(42) version(6) }

is assigned to this part of ISO 10303. The meaning of this value is defined in ISO/IEC 8824-1, and is described in ISO 10303-1.

*Remove the object identifier for the **geometry_schema** and replace with the following:*

To provide for unambiguous identification of the geometry_schema in an open information system, the object identifier

{ iso standard 10303 part(42) version(6) object(1) geometry-schema(1) }

is assigned to the geometry_schema (see clause 4). The meaning of this value is defined in ISO/IEC 8824-1, and is described in ISO 10303-1.

*Remove the object identifier for the **topology_schema** and replace with the following:*

To provide for unambiguous identification of the topology_schema in an open information system, the object identifier

{ iso standard 10303 part(42) version(6) object(1) topology-schema(2) }

is assigned to the topology_schema (see clause 5). The meaning of this value is defined in ISO/IEC 8824-1, and is described in ISO 10303-1.

*Remove the object identifier for the **geometric_model_schema** and replace with the following:*

To provide for unambiguous identification of the geometric_model_schema in an open information system, the object identifier

{ iso standard 10303 part(42) version(6) object(1) geometric-model-schema(3) }

is assigned to the geometric_model_schema (see clause 6). The meaning of this value is defined in ISO/IEC 8824-1, and is described in ISO 10303-1.

Annex C, p. 203

With the changes identified in this Technical Corrigendum, the URL containing the EXPRESS is incorrect. Replace the contents of the annex with the following:

This annex provides a listing of the EXPRESS entity names and corresponding short names as specified in this part of ISO 10303. It also provides a listing of the complete EXPRESS schema specified in this part of ISO 10303 without comments or other explanatory text. This annex is available in computer-interpretable form and can be found at the following URLs:

Short names: <http://www.mel.nist.gov/div826/subject/apde/snr/>
EXPRESS: <http://www.mel.nist.gov/step/parts/part042e2/is/tc1/>