INTERNATIONAL STANDARD

ISO 10303-201

First edition
1994-12-15

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

Part 201:
Application protocol: Explicit draughting

Systèmes d'automatisation industrielle et intégration — Représentation et échange de données de produits —

Partie 201: Protocole d'application: Dessin technique explicite

Reference number
ISO 10303-201:1994(E)

Contents

Page

International Organization for Standardization
Case Postale 56 • CH-1211 Genève 20 • Switzerland

Printed in Switzerland

Annexes

Figures

Tables

Foreword

The International Organization for Standardization (ISO) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

Draft International Standards adopted by technical committees are circulated to the member bodies for voting. Publication as an International Standard requires approval by at least 75% of the member bodies casting a vote.

International Standard ISO 10303-201 was prepared by Technical Committee ISO/TC 184, *Industrial automation systems and integration*, Subcommittee SC4, *Industrial data and global manufacturing programming languages.*

ISO 10303 consists of the following parts under the general title *Industrial automation systems and integration - Product data representation and exchange:*

- Part 1, Overview and fundamental principles;
- Part 11, Description methods: The *EXPRESS* language reference manual;
- Part 21, Implementation methods: Clear text encoding of the exchange structure;
- Part 22, Implementation methods: Standard data access interface specification;
- Part 31, Conformance testing methodology and framework: General concepts;
- Part 32, Conformance testing methodology and framework: Requirements on testing laboratories and clients;
- Part 41, Integrated generic resources: Fundamentals of product description and support;
- Part 42, Integrated generic resources: Geometric and topological representation;
- Part 43, Integrated generic resources: Representation structures;
- Part 44, Integrated generic resources: Product structure configuration;
- Part 45, Integrated generic resources: Materials;
- Part 46, Integrated generic resources: Visual presentation;
- Part 47, Integrated generic resources: Shape variation tolerances;

- Part 49, Integrated generic resources: Process structure and properties;
- Part 101, Integrated application resources: Draughting;
- Part 104, Integrated application resources: Finite element analysis;
- Part 105, Integrated application resources: Kinematics;
- Part 201, Application protocol: Explicit draughting;
- Part 202, Application protocol: Associative draughting;
- Part 203, Application protocol: Configuration controlled design;
- Part 207, Application protocol: Sheet metal die planning and design;
- Part 210, Application protocol: Printed circuit assembly product design data;
- Part 213, Application protocol: Numerical control process plans for machined parts.

The structure of this International Standard is described in ISO 10303-1. The numbering of the parts of this International Standard reflects its structure:

- Part 11 specifies the description methods;
- Parts 21 and 22 specify the implementation methods;
- Parts 31 and 32 specify the conformance testing methodology and framework;
- Parts 41 to 49 specify the integrated generic resources;
- Parts 101 to 105 specify the integrated application resources;
- Parts 201 to 213 specify the application protocols.

Should further parts be published, they will follow the same numbering pattern.

Annexes A, B, C, D and E form an integral part of this part of ISO 10303. Annexes F, G, H, J and K are for information only.

Diskette

Users should note that this part of ISO 10303 comprises a diskette:
— the short names of entities given in annex B are also included on the diskette;
— the EXPRESS listings (annex A) are provided on the diskette only;
— a method to enable users to report errors in the documentation is given. Full details are provided in the file.

Introduction

ISO 10303 is an International Standard for the computer-interpretable representation and exchange of product data. The objective is to provide a neutral mechanism capable of describing product data throughout the life cycle of a product independent from any particular system. The nature of this description makes it suitable not only for neutral file exchange, but also as a basis for implementing and sharing product databases and archiving.

This International Standard is organized as a series of parts, each published separately. The parts of ISO 10303 fall into one of the following series: description methods, integrated resources, application protocols, abstract test suites, implementation methods, and conformance testing. The series are described in ISO 10303–1. This part of ISO 10303 is a member of the application protocols series.

This part of ISO 10303 specifies an application protocol (AP) for the exchange of technical CAD drawings consisting of two-dimensional geometry and two-dimensional annotation. This part satisfies the need for the exchange of such drawings within and between organizations, especially those operating in the mechanical engineering and architectural, engineering, and construction industrial sectors.

The exchange of such drawings enables the communication of product data during all stages of the lifecycle of a product for any product type. This product data, in the form of a CAD drawing, may be interpreted by people according to accepted international, national, or organization draughting standards. The use of CAD drawings for the purposes of the communication of product data is common practice in many industries. This product data may also be interpreted by appropriate CAx systems for use in applications other than draughting.

The nominal shape of a product is represented by geometry defined in two dimensions. Such representations may be the result of the application of a projection transformation on a three-dimensional shape representation; such three-dimensional shape representations, and the projection transformations that may be applied to produce a two-dimensional representation, lie outside the scope of this part of ISO 10303.

This application protocol makes use of two fundamental concepts which relate to the creation, revision, storage, and use of drawings produced using CAD systems.

A CAD drawing is a digital representation of a drawing and may include the following, in addition to lines and text presented on the face of the drawing:

- representations of the shape(s) of the product(s) depicted by the drawing;

- information used in administering drawings for purposes of configuration control, audit trails, etc.

The use of the shape representation present in the CAD drawing depends on the interpretation of a draughtsman or engineer viewing the drawing. For this reason the concept of a draughting shape model is included. A draughting shape model is a representation of the shape of a product

whose interpretation is determined through visual inspection of the drawings that depict the product.

EXAMPLE 1 – An architectural drawing may contain both isometric and perspective views of a building. The use of the two-dimensional representations depicted in these views depends on the knowledge, in the mind of a designer, that the lengths of lines may be measured or calculated in the isometric view but not in the perspective view.

This draughting shape model is also a nominal shape model; no information regarding tolerances is included in the draughting shape model.

Figure 1 contains the data planning model that provides a high-level description of the general structure of this application protocol, as well as the relationships between the basic data components.

The data planning model illustrates that a product may be described by a draughting shape model. The draughting shape model is a two-dimensional model generated by a CAD system. This CAD model is composed of geometry which represents the shape of the product and may also include annotation. The product is documented by a drawing.

The drawing is composed of drawing sheets that contain annotation and drawing views. The drawing views are two-dimensional views of the draughting shape model that may include additional annotation. Therefore, the drawing is a presentation of the draughting shape model.

Annotation, in the form of text and symbology, provides additional product data that is needed to fully define the product or interpret the drawing. The scope of this application protocol constrains the dimensionality of all data found within it to be two-dimensional only.

This application protocol defines the context, scope, and information requirements for exchange of geometrically explicit two-dimensional technical CAD drawings with explicit annotation and specifies the integrated resources necessary to satisfy these requirements.

Application protocols provide the basis for developing implementations of ISO 10303 and abstract test suites for the conformance testing of AP implementations.

Clause 1 defines the scope of the application protocol and summarizes the functionality and data covered by the AP. An application activity model that is the basis for the definition of the scope is provided in annex F. The information requirements of the application are specified in clause 4 using terminology appropriate to the application. A graphical representation of the information requirements, referred to as the application reference model, is given in annex G.

Resource constructs are interpreted to meet the information requirements. This interpretation produces the application interpreted model (AIM). This interpretation, given in 5.1, shows the correspondence between the information requirements and the AIM. The short listing of the AIM specifies the interface to the integrated resources and is given in 5.2. Note that the definitions and EXPRESS provided in the integrated resources for constructs used in the AIM may include select list items and subtypes which are not imported into the AIM. The expanded listing given in annex A contains the complete EXPRESS for the AIM without annotation. A graphical representation of the AIM is given in annex H. Additional requirements for specific

implementation methods are given in annex C.

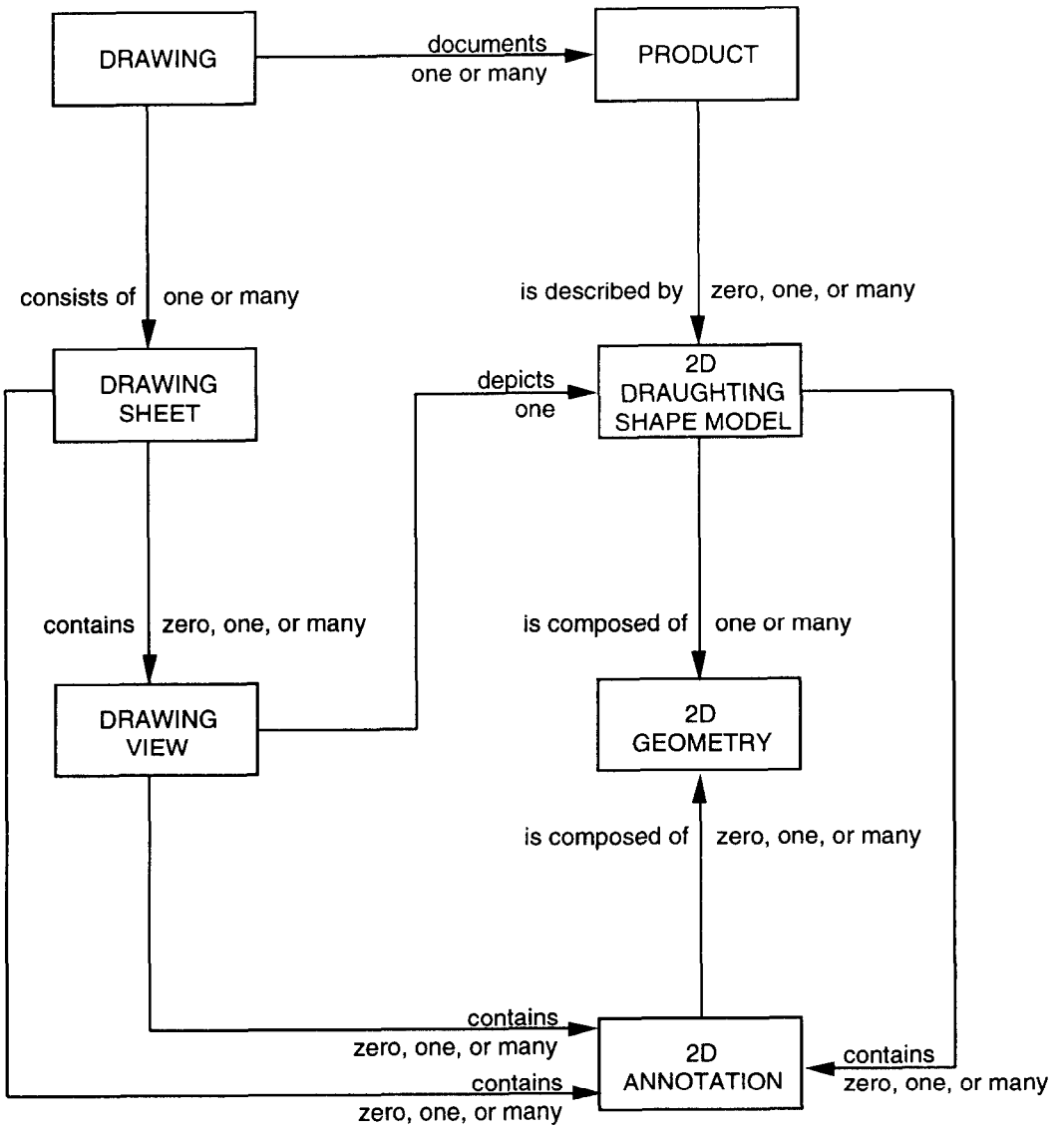

Figure 1 – Data planning model

Industrial automation systems and integration — Product data representation and exchange — Part 201 : Application protocol: Explicit draughting

1 Scope

This part of ISO 10303 specifies the use of the integrated resources necessary for the scope and information requirements for explicit draughting.

NOTE – The application activity model in annex F provides a graphical representation of the processes and information flows which are the basis for the definition of the scope of this part of ISO 10303.

This part of ISO 10303 is applicable to the inter-organization exchange of computer-interpretable drawing information and product definition data.

The following are within the scope of this part of ISO 10303:

– the representation of drawings for the purpose of exchange, especially for mechanical engineering, architectural engineering, and construction applications;

– the representation of the real size of a product depicted in a drawing to enable use by applications where true geometric equivalence is required;

EXAMPLE 2 – The representation of the shape of the product is required to support not only visual equivalence of exchanged drawings but also where true geometric equivalence is required by the receiving system. Such uses include the calculations of distances or areas and the generation of numerical control tool paths.

– the representation of a drawing that depicts any phase of the design;

– the representation of individual drawing revisions;

– the representation of the two-dimensional draughting shape model depicting the product shape and the transformations used for the generation of the drawing views;

– the presentation of non-shape product definition data depicted in a drawing by two-dimensional annotation;

– the hierarchical structure of drawings, drawing sheets, and views of the draughting shape

model;

– mechanisms for the grouping of the elements depicted on a drawing;

– the administrative data used for the purpose of drawing management;

– the administrative data identifying the product versions being documented by the drawing.

The following are outside the scope of this part of ISO 10303:

– the representation of the shape of a product using three-dimensional geometry;

– the representation of the shape of a product that is not depicted in a drawing;

– the representation of drawings that are not related to a product;

– the exchange of drawing history;

– the definition of annotation in three-dimensional coordinate systems;

– the presentation of dimensions and annotation that are associated to viewed geometry and annotation;

– a computer-interpretable bill of material structure except as conveyed by annotation on the drawing;

– strict enforcement of draughting standards;

– drawings containing non-displayable attribute data other than that required as adminstrative data;

EXAMPLE 3 – Non-displayable attribute data could be density, mass, or moment of inertia.

– the automatic generation of drawings including views, dimensions, and annotation.

– the exchange of data used exclusively for the creation of paper or or hardcopy versions of the drawing.

EXAMPLE 4 – Printer or plotter data could be pen designations, plot scale, or plot color specifications.

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this part of ISO 10303. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this part of ISO 10303 are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO 3098-1:1974, *Technical drawings - Lettering - Part 1: Currently used characters.*

ISO/IEC 8824-1:—[1)], *Information technology - Open Systems Interconnection - Abstract Syntax Notation One (ASN.1) - Part 1: Specification of Basic Notation.*

ISO 10303-1:1994, *Industrial automation systems and integration - Product data representation and exchange - Part1: Overview and fundamental principles.*

ISO 10303-11:1994, *Industrial automation systems and integration - Product data representation and exchange - Part 11: Description methods: The EXPRESS language reference manual.*

ISO 10303-21:1994, *Industrial automation systems and integration - Product data representation and exchange - Part 21: Implementation methods: Clear text encoding of the exchange structure.*

ISO 10303-31:1994, *Industrial automation systems and integration - Product data representation and exchange - Part 31: Conformance testing methodology and framework: General concepts.*

ISO 10303-41:1994, *Industrial automation systems and integration - Product data representation and exchange - Part 41: Integrated generic resources: Fundamentals of product description and support.*

ISO 10303-42:1994, *Industrial automation systems and integration - Product data representation and exchange - Part 42: Integrated generic resources: Geometric and topological representation.*

ISO 10303-43:1994, *Industrial automation systems and integration - Product data representation and exchange - Part 43: Integrated generic resources: Representation structures.*

ISO 10303-46:1994, *Industrial automation systems and integration - Product data representation and exchange - Part 46: Integrated generic resources: Visual presentation.*

ISO 10303-101:1994, *Industrial automation systems and integration - Product data representation and exchange - Part 101: Integrated application resources: Draughting.*

[1)] To be published.

3 Definitions and abbreviations

3.1 Terms defined in ISO 10303-1

This part of ISO 10303 makes use of the following terms defined in ISO 10303-1:

- application;
- application activity model (AAM);
- application interpreted model (AIM);
- application protocol (AP);
- application reference model (ARM);
- conformance testing;
- implementation method;
- integrated resource;
- model;
- product;
- product data;
- unit of functionality (UoF).

3.2 Terms defined in ISO 10303-42

This part of ISO 10303 makes use of the following terms defined in ISO 10303-42:

- curve.

3.3 Terms defined in ISO 10303-46

This part of ISO 10303 makes use of the following terms defined in ISO 10303-46:

- annotation;
- picture.

3.4 Terms defined in ISO 10303-101

This part of ISO 10303 makes use of the following terms defined in ISO 10303-101:

- callout;
- draughting;
- drawing.

3.5 Other definitions

For the purpose of this part of ISO 10303, the following definitions apply.

3.5.1 text baseline : a theoretical line on which the lowest points of the text characters, excluding descenders, rest.

3.5.2 CAD drawing : a digital representation of a drawing that includes pictures, text, or combinations of both that are presented in the drawing; information used in administering the drawing; and possibly representations of the shapes of the products depicted in the drawing.

3.5.3 cartesian coordinate system : a set of mutually perpendicular axes divided into known segments, each of whose length is a specified unit of measure. In the case of a two-dimensional coordinate system there are two axes and in the case of a three-dimensional coordinate system there are three. The point of intersection of the axes is referred to as the origin.

3.5.4 chain dimension sequence : a sequence of two or more dimensions in which the terminus of one dimension initializes the next dimension in the sequence.

3.5.5 dimension value : a representation of the numerical value for a dimension that describes its magnitude.

3.5.6 draughting shape model : a graphical representation of a product shape composed of geometric and annotation elements that are all defined in the same coordinate system. It is created and used specifically for generating a drawing of a product.

3.5.7 externally defined : an element in which an explicit description of its use and physical representation are given within another source. This source shall provide all the information necessary to unambiguously recreate the element. This information shall be referenceable by identification of the source and the element within that source.

3.5.8 independently instantiable : an entity for which instances can be created that are not referenced by instances of other entity data types.

3.5.9 parallel dimension sequence : a collection of two or more dimensions where all the dimensions are of the same type, their dimension lines are parallel to one another and/or they

share a common origin or datum.

3.5.10 predefined : an element which has an explicit description of its use and physical representation given within an annex of this part of ISO 10303. The explicit description shall provide all the information necessary to unambiguously recreate the element.

3.5.11 scale : the ratio of the linear size of a representation of a product in a drawing to the linear size of the product itself.

3.5.12 subfigure : a combination of geometric and annotation elements that are all defined in the same coordinate system.

NOTE – This combination of elements may be duplicated multiple times into the drawing or draughting shape model.

3.5.13 symbol : a combination of annotation elements which, as a collection, represents a distinct concept.

NOTE – This combination of elements may be defined externally to the drawing or draughting shape model and may be duplicated multiple times into the drawing or draughting shape model.

3.6 Abbreviations

For the purpose of this part of ISO 10303, the following abbreviations apply.

AIC	application interpreted construct.
ANSI	American National Standards Institute.
CAD	computer aided design.
CAx	computer aided technologies.

4 Information requirements

This clause specifies the information required for explicit draughting.

The information requirements are specified as a set of units of functionality, application objects, and application assertions. These assertions pertain to individual application objects and to relationships between application objects. The information requirements are defined using the terminology of the subject area of this application protocol.

NOTES

1 – A graphical representation of the information requirements is given in annex G.

2 – The information requirements correspond to those of the activities identified as being in the scope of this application protocol in annex F.

3 – The mapping table specified in 5.1 shows how the information requirements are met using the integrated resources of this International Standard. The use of the integrated resources introduces additional requirements that are common to application protocols.

The following are high-level information requirements:

– a draughting shape model shall be presented in at least one view;

– each exchange shall contain at least one drawing;

– all draughting shape models depicting the shape of a product shall be defined using one-to-one scale geometry;

– the exchange of two-dimensional draughting shape models and the information that defines the views of those models shall be supported. Two-dimensional geometry created as a result of applying the view definition will not be exchanged;

– annotation may be applied to the draughting shape model, a drawing view, or a drawing sheet without being part of a drawing view;

– all elements in a CAD model or a drawing shall be assigned to one or more layers.

4.1 Units of functionality

This subclause specifies the units of functionality for the explicit draughting application protocol. This part of ISO 10303 specifies the following units of functionality:

– 2D_draughting_shape_model;

– 2D_elements_of_appearance;

– 2D_model_viewing;

– drawing_structure_and_administration;

– elements_of_annotation;

– elements_of_draughting_annotation;

– grouping;

– product_relation.

The units of functionality and a description of the functions that each UoF supports are given below. The application objects included in the UoF's are defined in 4.2.

4.1.1 2D_draughting_shape_model

The 2D_draughting_shape_model UoF contains the application objects which represent the two-dimensional shape of a product. This representation is composed of geometric elements and may include annotation elements. This representation may be structured through the use of sub-models.

The following application objects are used by 2D_draughting_shape_model UoF:

- 2D_cartesian_coordinate_space;
- 2D_draughting_shape_model;
- 2D_geometric_element;
- Cartesian_coordinate_space;
- Model_placed_annotation;
- Sub_model;
- Sub_model_definition.

4.1.2 2D_elements_of_appearance

The 2D_elements_of_appearance UoF contains information about the presentation characteristics of two-dimensional geometric and annotation elements. It contains the information required for visual display of the drawing elements.

The following application objects are used by 2D_elements_of_appearance UoF:

- Appearance;
- Colour;
- Curve_appearance;
- Externally_defined_hatching;
- Externally_defined_line_font;
- Externally_defined_text_font;

- Externally_defined_tile;
- Externally_defined_tiling;
- Fill_area_appearance;
- Hatching_pattern;
- Line_font;
- Predefined_colour;
- Predefined_line_font;
- Predefined_text_font;
- Solid_fill_area;
- Text_appearance;
- Text_font;
- Tile;
- User_defined_colour;
- User_defined_hatching;
- User_defined_line_font;
- User_defined_tile;
- User_defined_tiling;
- Visibility.

4.1.3 2D_model_viewing

The 2D_model_viewing UoF contains the information necessary to depict a two-dimensional draughting shape model in a view on a sheet of the drawing. It establishes a relationship between the draughting shape model and the drawing view. It provides the assignment of appearance characteristics to all geometric elements and annotation that are displayed in a view of the draughting shape model.

The following application objects are used by 2D_model_viewing UoF:

- 2D_drawing_view_definition;
- View_displayed_geometry;
- View_displayed_model_annotation.

4.1.4 drawing_structure_and_administration

The drawing_structure_and_administration UoF contains information about the hierarchical organization of drawings, drawing sheets, and drawing views, together with the administrative information necessary to manage drawings and drawing sheets. Drawing sheets and drawing views are defined in their specific coordinate space. Annotation may be assigned to each drawing sheet and drawing view. The administrative information supports the exchange of drawings between environments in which configuration management of drawings is used.

The following application objects are used by drawing_structure_and_administration UoF:

- 2D_cartesian_coordinate_space;
- Approval;
- Cartesian_coordinate_space;
- Drawing;
- Drawing_sheet;
- Drawing_view;
- Organization;
- Sheet_placed_annotation;
- View_placed_annotation.

4.1.5 elements_of_annotation

The elements_of_annotation UoF contains the application objects used to compose all annotation, along with the ability to create nested annotation. Annotation may be used either in a drawing sheet, a drawing view, or a draughting shape model to present drawing elements without a specific meaning in the context of a drawing.

EXAMPLE 5 - Figure 2 shows a linear dimension and identifies within it a text object as an example of an annotation element.

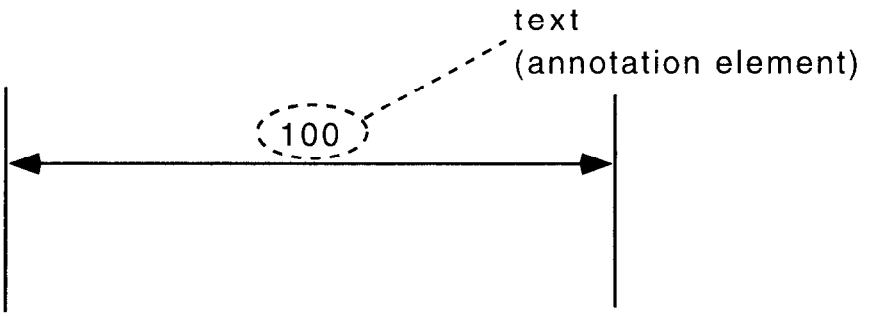

Figure 2 – Example of elements_of_annotation UoF

The following application objects are used by elements_of_annotation UoF:

- 2D_cartesian_coordinate_space;
- Annotation_curve;
- Annotation_element;
- Annotation_subfigure;
- Annotation_subfigure_definition;
- Annotation_subfigure_definition_element;
- Annotation_symbol;
- Cartesian_coordinate_space;
- Externally_defined_symbol;
- Fill_area;
- Fill_area_boundary;
- Point_marker_symbol;
- Predefined_symbol;
- Text;
- Text_string;
- User_defined_symbol;

– User_defined_symbol_definition.

4.1.6 elements_of_draughting_annotation

The elements_of_draughting_annotation UoF contains the application objects used for the creation of annotation, presented on a drawing, that has a specific meaning or purpose within the scope of draughting.

EXAMPLE 6 – Figure 3 shows a linear dimension as an example of draughting annotation. This is built from a number of components – projection lines, dimension line, dimension value – and has a known meaning: the dimensioned part or feature has a length of 100 units.

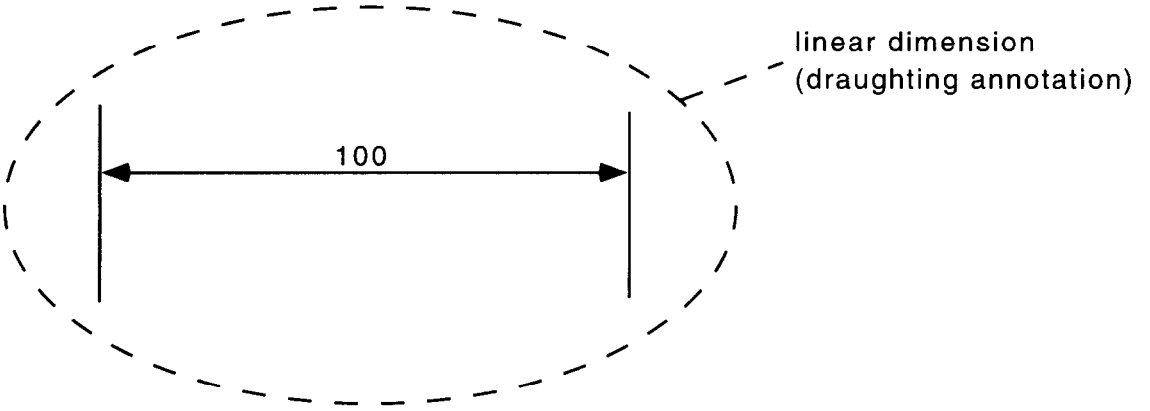

Figure 3 – Example of elements_of_draughting_annotation UoF

The following application objects are used by elements_of_draughting_annotation UoF:

– Angular_dimension;

– Chained_dimension_pair;

– Curve_dimension;

– Datum_feature_callout;

– Datum_target_callout;

– Diameter_dimension;

– Dimension;

– Dimension_callout;

– Dimension_line;

- Dimension_sequence_pair;
- Dimension_symbol;
- Directed_curve;
- Draughting_annotation;
- Draughting_callout;
- Drawing_sheet_layout;
- Geometrical_tolerance;
- Geometrical_tolerance_symbol;
- Leader;
- Leader_directed_dimension;
- Linear_dimension;
- Ordinate_dimension;
- Parallel_dimension_pair;
- Projection_line;
- Radius_dimension;
- Structured_dimension_callout;
- Terminator_symbol;
- Unstructured_dimension_callout.

4.1.7 grouping

The grouping UoF contains information about the organizational structures defined in a draughting shape model or a drawing. It includes the user capabilities to collect geometric or drawing elements for a specific discipline or viewpoint. Different mechanisms can be applied to group these elements. The primary use of grouping mechanisms is to increase efficiency during shape model design or modification by manipulation of more than one element at a time. Grouping mechanisms are not physical characteristics of the drawing, but their structure is preserved to aid in the future modification of the shape model or drawing.

The following application objects are used by grouping UoF:

- Group;
- Group_annotation_element;
- Group_element;
- Group_geometric_element;
- Layer;
- Sub_group.

4.1.8 product_relation

The product_relation UoF contains the administrative information necessary to relate a product to a drawing. The administrative data consists of the information to uniquely identify the product which is documented by the drawing.

The following application objects are used by product_relation UoF:

- Organization;
- Product_version.

4.2 Application objects

This subclause specifies the application objects for the explicit draughting application protocol. Each application object is an atomic element that embodies a unique application concept and contains attributes specifying the data elements of the object. The application objects and their definitions are given below.

4.2.1 2D_cartesian_coordinate_space

A 2D_cartesian_coordinate_space is a type of Cartesian_coordinate_space (see 4.2.14) that is defined by two mutually perpendicular axes.

4.2.2 2D_draughting_shape_model

A 2D_draughting_shape_model is a representation of a product or a portion of a product. This representation consists of points and curves defined in a two-dimensional coordinate space, and

may also contain annotation placed in the same coordinate space, or Sub_model objects. Each 2D_draughting_shape_model contains at least one 2D_geometric_element (see 4.2.4) or Sub_model (see 4.2.71). Each 2D_draughting_shape_model may be a Sub_model_definition (see 4.2.72). The data associated with a 2D_draughting_shape_model are the following:

- Model_id.

The Model_id specifies the identification of a particular draughting shape model.

4.2.3 2D_drawing_view_definition

A 2D_drawing_view_definition is the specification of a plane, within the coordinate system of a draughting shape model, to which the elements of the draughting shape model are projected. Additionally, clipping boundaries are specified to limit the content of the projection, and a scale is specified to define the size of the projected elements in the view. The data associated with a 2D_drawing_view_definition are the following:

- Clipping;
- Scale;
- Translation.

NOTE – A 2D_drawing_view_definition may be placed multiple times on a drawing, but each is a separate drawing view. The placement within a particular sheet makes each drawing view unique to that drawing sheet.

4.2.3.1 Clipping

The Clipping specifies the mathematical information necessary to define a two-dimensional area that encloses all viewable geometric and annotation elements of a draughting shape model. Only those elements, or portions of any elements, that fall within this boundary will be displayed.

4.2.3.2 Scale

The Scale specifies the ratio between the size of the elements as defined in the draughting shape model and the size of the elements as presented in the drawing view.

4.2.3.3 Translation

The Translation specifies the mathematical values that define the non-rotational displacement between elements located in the coordinate system of the draughting shape model and their location in the coordinate system of the drawing view.

4.2.4 2D_geometric_element

A 2D_geometric_element is a specified type of individual geometry that is used in the representation of the shape of a product. A 2D_geometric_element is defined in a two-dimensional coordinate space. The types of 2D_geometric_element that may be specified are: cartesian point, point on curve, straight line, composite curve, b-spline curve, offset curve, polyline, circle, ellipse, and trimmed conic.

4.2.5 Angular_dimension

An Angular_dimension is a type of Dimension (see 4.2.22) that is the graphical presentation of a value of angular distance measure where the distance is between two elements converging on a common point.

NOTE – Figure 4 illustrates three angular dimensions depicting angular distance. The figure also illustrates a radius dimension.

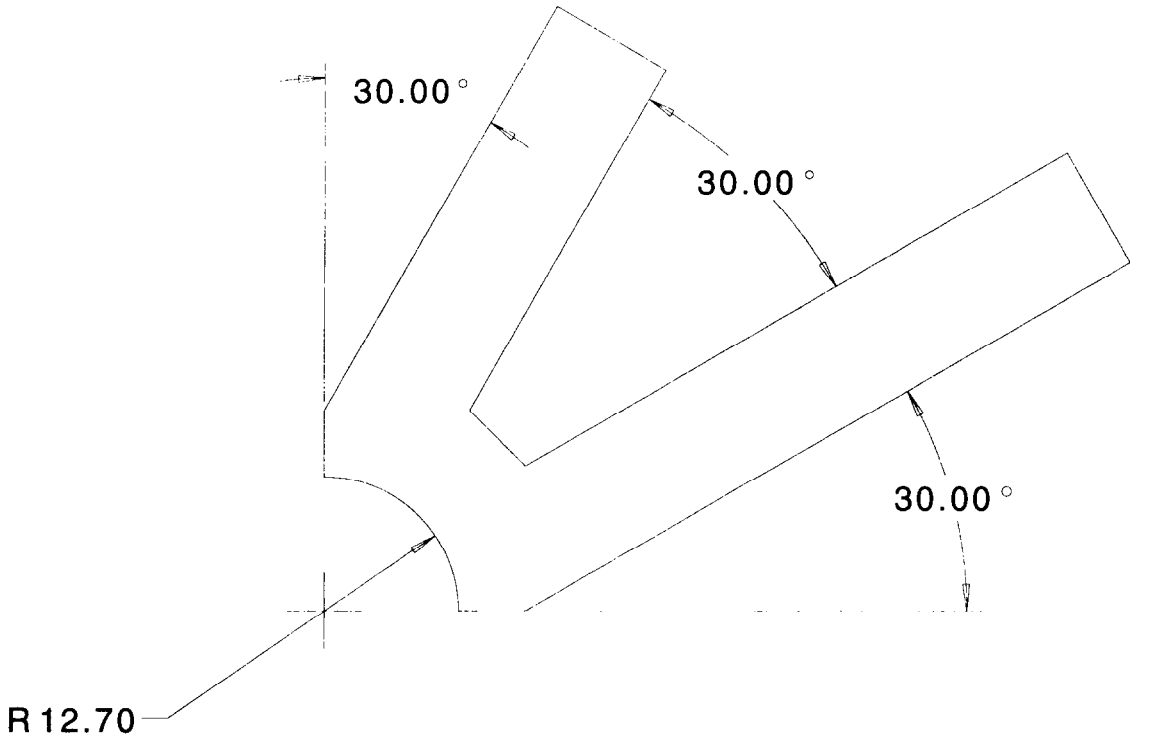

Figure 4 – Angular and radius dimensions

4.2.6 Annotation_curve

An Annotation_curve is a type of Annotation_element (see 4.2.7) that is a two-dimensional trimmed curve used only to annotate a drawing or a draughting shape model and which is defined in the coordinate system in which it is used. Each Annotation_curve may be a Directed_curve (see 4.2.27).

4.2.7 Annotation_element

An Annotation_element is a type of Draughting_annotation (see 4.2.28) that is the lowest level discrete element that can either serve as annotation itself, or be used as a constituent of other annotation. Each Annotation_element is either an Annotation_curve (see 4.2.6), an Annotation_subfigure (see 4.2.8), an Annotation_symbol (see 4.2.11), a Fill_area (see 4.2.40), or a Text (see 4.2.74).

4.2.8 Annotation_subfigure

An Annotation_subfigure is a type of Annotation_element (see 4.2.7) that is the presentation of an Annotation_subfigure_definition (see 4.2.9) located within the coordinate system of a drawing sheet, a drawing view, a draughting shape model, or another subfigure. The data associated with an Annotation_subfigure are the following:

- Location;
- Rotation;
- Scale.

4.2.8.1 Location

The Location specifies the position of the origin of the coordinate system in which the subfigure is defined relative to the origin of the coordinate system into which the subfigure is being placed.

4.2.8.2 Rotation

The Rotation specifies the angle, measured counter-clockwise, between the horizontal axis of the coordinate system in which the subfigure is defined and the horizontal axis of the coordinate system into which the subfigure is being placed.

4.2.8.3 Scale

The Scale specifies the ratio between the size of the subfigure as defined and the size of the subfigure as presented. The scale in the x-coordinate need not equal the scale in the y-coordinate.

4.2.9 Annotation_subfigure_definition

An Annotation_subfigure_definition is a collection of defined annotation elements, along with their placements, in a definitional coordinate space. The data associated with an Annotation_subfigure_definition are the following:

- Name.

The Name specifies the identification of a particular subfigure definition. The Name need not be specified for a particular Annotation_subfigure_definition.

4.2.10 Annotation_subfigure_definition_element

An Annotation_subfigure_definition_element is annotation that is used as a constituent of a subfigure definition.

4.2.11 Annotation_symbol

An Annotation_symbol is a type of Annotation_element (see 4.2.7) that is a combination of text and annotation curves which, as a collection, represent a recognized concept. Each Annotation_symbol is either an Externally_defined_symbol (see 4.2.36), a Predefined_symbol (see 4.2.62), or a User_defined_symbol (see 4.2.83).

The data associated with an Annotation_symbol are the following:

- Location;
- Rotation;
- Scale.

4.2.11.1 Location

The Location specifies the position of the origin of the coordinate system in which the symbol is defined, relative to the origin of the coordinate system into which the symbol is being placed.

4.2.11.2 Rotation

The Rotation specifies the angle, measured counter-clockwise, between the horizontal axis of the coordinate system in which the symbol is defined and the horizontal axis of the coordinate system into which the symbol is being placed.

4.2.11.3 Scale

The Scale specifies the ratio between the size of the symbol as defined and the size of the symbol as presented. The scale in the x-coordinate need not equal the scale in the y-coordinate.

4.2.12 Appearance

An Appearance is a collection of visual characteristics that govern the presentation of geometric elements or annotation elements. Each Appearance may be one of the following: a Fill_area_appearance (see 4.2.41), a Curve_appearance (see 4.2.17), or a Text_appearance (see 4.2.75).

4.2.13 Approval

An Approval is information that indicates the drawing, drawing sheet, or both have been reviewed for data content and for correctness of the presentation of that data and has been found to be acceptable. The data associated with an Approval are the following:

- Date;
- Description.

4.2.13.1 Date

The Date specifies the date on which the approval was assigned.

4.2.13.2 Description

The Description specifies the organization-specific release status or the authorized modifications for the revision of the drawing, drawing sheet, or both.

EXAMPLE 7 – A Description could be "manufacturing release", "see engineering change notice 423 for a description of the changes", or "zone C-8, dimension 8.00 was 8.25"

4.2.14 Cartesian_coordinate_space

A Cartesian_coordinate_space is a two-dimensional or three-dimensional space defined by a cartesian coordinate system. Each Cartesian_coordinate_space is a 2D_cartesian_coordinate_space (see 4.2.1). Each Cartesian_coordinate_space is the coordinate space of either an Annotation_subfigure_definition (see 4.2.9), a Drawing_sheet (see 4.2.31), a Drawing_view (see 4.2.33), a User_defined_symbol_definition (see 4.2.84), or multiple 2D_draughting_shape_model (see 4.2.2) objects. The data associated with a Cartesian_coordinate_space are the following:

- Length_measure_unit;
- Plane_angle_measure_unit.

4.2.14.1 Length_measure_unit

The Length_measure_unit specifies the increments used to define linear distances or sizes within a cartesian coordinate space.

4.2.14.2 Plane_angle_measure_unit

The Plane_angle_measure_unit specifies the increments used to define angular distances within a cartesian coordinate space.

4.2.15 Chained_dimension_pair

A Chained_dimension_pair is a type of Dimension_sequence_pair (see 4.2.25) that is the relationship between two dimensions in which the terminus of one dimension initializes the next dimension in the sequence.

NOTE – Figure 5 illustrates four horizontal linear dimensions in a chain dimension sequence. Each set of two dimensions sharing a projection line is a chained dimension pair. Additionally, the figure illustrates a single vertical linear dimension.

4.2.16 Colour

A Colour defines a basic apperance property of an element with respect to the light reflected by it. Each Colour is either a Predefined_colour (see 4.2.60) or a User_defined_colour (see 4.2.80).

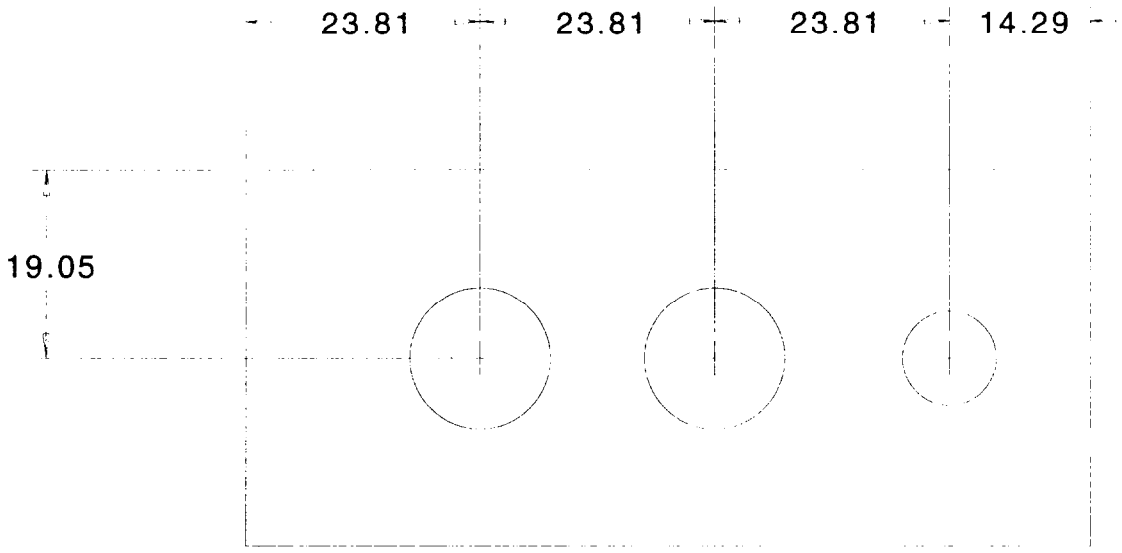

Figure 5 – Chain dimension sequence

4.2.17 Curve_appearance

A Curve_appearance is a type of Appearance (see 4.2.12) that governs the visual presentation of geometric curves and annotation curves. The data associated with a Curve_appearance are the following:

- Draughting_role;
- Width.

4.2.17.1 Draughting_role

The Draughting_role specifies the purpose within draughting for a particular curve appearance. The Draughting_role need not be specified for a particular Curve_appearance.

EXAMPLE 8 – A Draughting_role could be a centreline or section line.

4.2.17.2 Width

The Width specifies the thickness of the curve measured perpendicular to the direction of the curve. The Width of a curve is not affected by scaling.

4.2.18 Curve_dimension

A Curve_dimension is a type of Dimension (see 4.2.22) that is the graphical presentation of a value of the distance between two elements, measured along a curved path, or the length of a curved element.

NOTE – Figure 6 illustrates two curve dimensions. The dimension with a value of 36.91 depicts the distance between two elements along a path defined by a third element of geometry or annotation. The dimension with a value of 76.64 depicts the length of a curve, measured along the entire path of the curve.

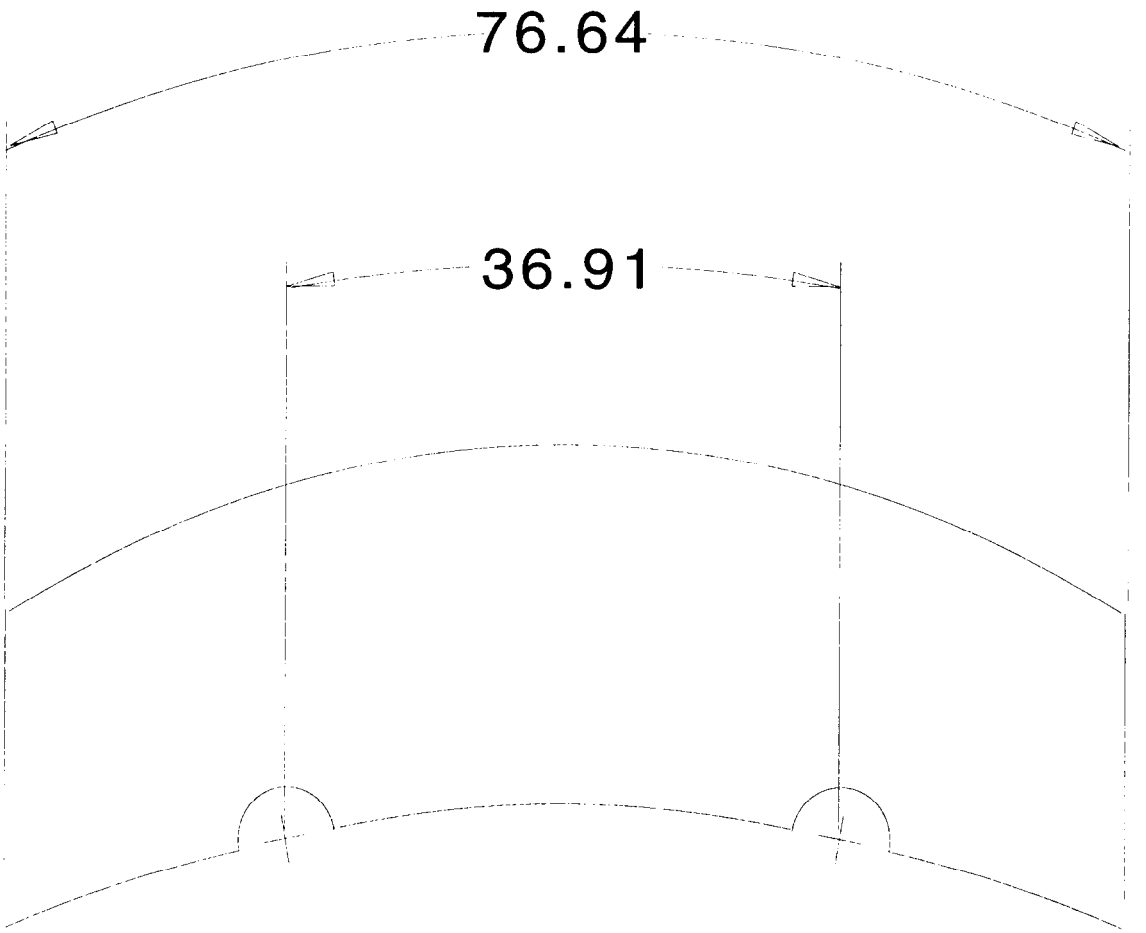

Figure 6 – Curve dimension

4.2.19 Datum_feature_callout

A Datum_feature_callout is a type of Draughting_callout (see 4.2.29) used to denote a point, line, or plane as a datum and which specifies the designation to be used as identification of that datum.

4.2.20 Datum_target_callout

A Datum_target_callout is a type of Draughting_callout (see 4.2.29) used to denote points, lines, and surfaces of contact which are used in establishing a reference datum on a product. The callout contains an alphanumeric designation and, where applicable, a specification of the diametrical size of the target area.

4.2.21 Diameter_dimension

A Diameter_dimension is a type of Dimension (see 4.2.22) that is the graphical presentation of a value of the diametrical size of a circular element.

NOTE – Projection lines may be used to clarify the extent of a dimension of diametrical size as illustrated in Figure 7.

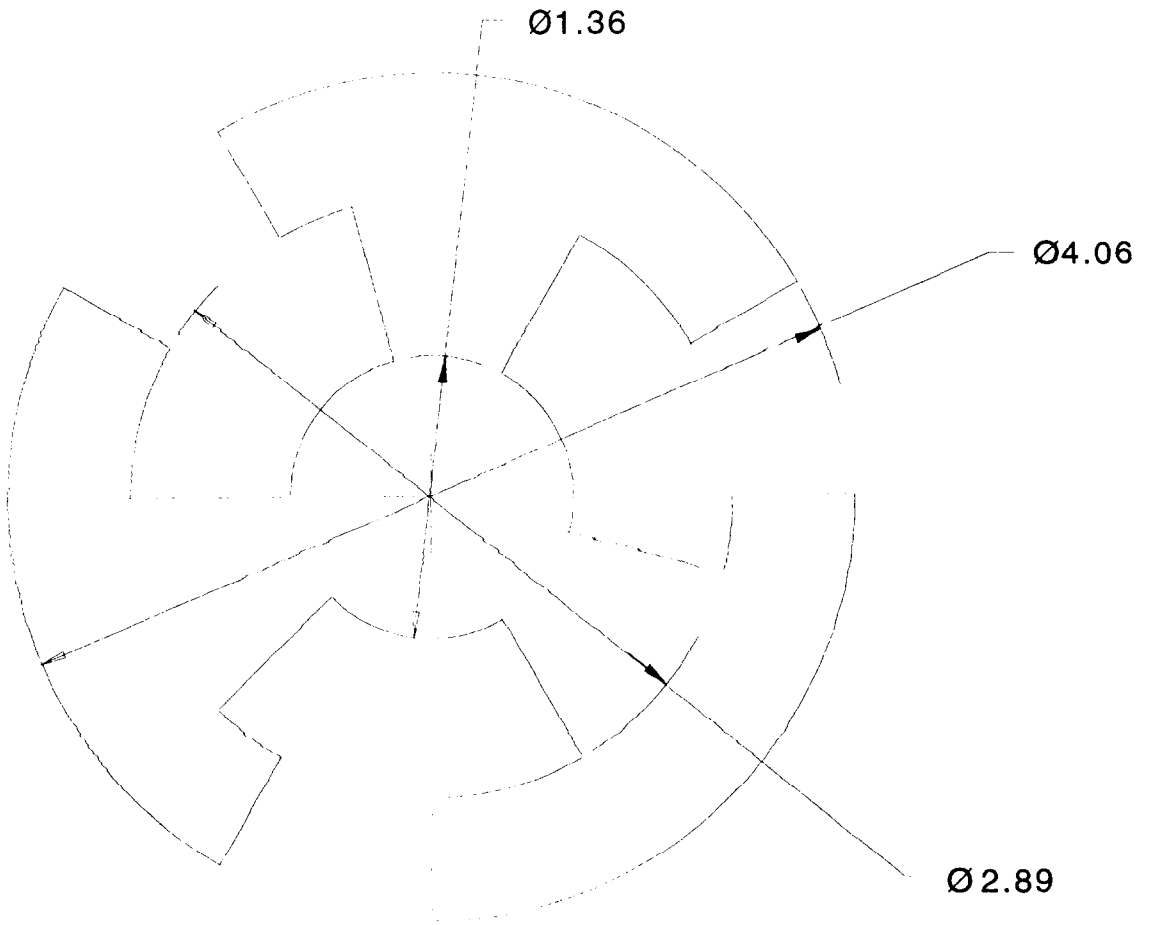

Figure 7 – Diameter dimension

4.2.22 Dimension

A Dimension is a type of Draughting_annotation (see 4.2.28) that is the graphical presentation of the dimension value, associated information, and the necessary symbology to accurately depict its area of application. Each Dimension is either an Angular_dimension (see 4.2.5), a Curve_dimension (see 4.2.18), a Diameter_dimension (see 4.2.21), a Leader_directed_dimension (see 4.2.52), a Linear_dimension (see 4.2.54), an Ordinate_dimension (see 4.2.56), or a Radius_dimension (see 4.2.66).

4.2.23 Dimension_callout

A Dimension_callout is the text and symbols in the presentation of a dimension that represent the dimension value, dimension units, tolerance information and any related notes. Each Dimension_callout is either a Structured_dimension_callout (see 4.2.69) or an Unstructured_dimension_callout (see 4.2.79). Each Dimension_callout is either primary or secondary for one dimension.

4.2.24 Dimension_line

A Dimension_line is a type of Directed_curve (see 4.2.27) used in the graphical presentation of a dimension value along with other symbology, if necessary, to show the extent of the application of the value.

EXAMPLE 9 – Terminator symbols may be associated with a Dimension_line.

4.2.25 Dimension_sequence_pair

A Dimension_sequence_pair is the relationship between two adjacent dimensions that share a projection line. Each Dimension_sequence_pair is either a Chained_dimension_pair (see 4.2.15) or a Parallel_dimension_pair (see 4.2.58).

4.2.26 Dimension_symbol

A Dimension_symbol is a type of Predefined_symbol (see 4.2.62) that is used in conjunction with a dimension value to convey the context of the dimension value.

EXAMPLE 10 – A diameter symbol may be used in conjunction with a dimension value to denote a diameter dimension.

NOTE – See 5.2.2.1.36 for a description of the dimension symbols supported by this part of ISO 10303.

4.2.27 Directed_curve

A Directed_curve is a type of Annotation_curve (see 4.2.6) that is used to guide annotation to a specific feature or area of a draughting shape model, drawing view, or drawing sheet. Each Directed_curve is either a Dimension_line (see 4.2.24), a Leader (see 4.2.51), or a Projection_line (see 4.2.65).

4.2.28 Draughting_annotation

A Draughting_annotation is text and symbology applied to either a drawing sheet, drawing view, another piece of annotation, or a draughting shape model, for the purpose of communicating product data and drawing interpretation information. Each Draughting_annotation is either an Annotation_element (see 4.2.7), a Dimension (see 4.2.22), or a Draughting_callout (see 4.2.29). Each Draughting_annotation may be one of the following: a Model_placed_annotation (see 4.2.55), a Sheet_placed_annotation (see 4.2.67), or a View_placed_annotation (see 4.2.89).

4.2.29 Draughting_callout

A Draughting_callout is a type of Draughting_annotation (see 4.2.28) that is a combination of text, annotation curves, and symbology which conveys information about a specific feature or area. Each Draughting_callout may be one of the following: a Datum_feature_callout (see 4.2.19), a Datum_target_callout (see 4.2.20), or a Geometrical_tolerance (see 4.2.43).

NOTE - Figure 8 illustrates the dimensioning and annotation of specific features of a product. The top view illustrates the specification of datums E and F and the specification of the diameter dimension with an associated geometrical tolerance. The centre view illustrates another datum specification and a draughting callout, in this case, a geometrical tolerance related by a projection line. The bottom view illustrates three datum target points and their associated datum target symbols.

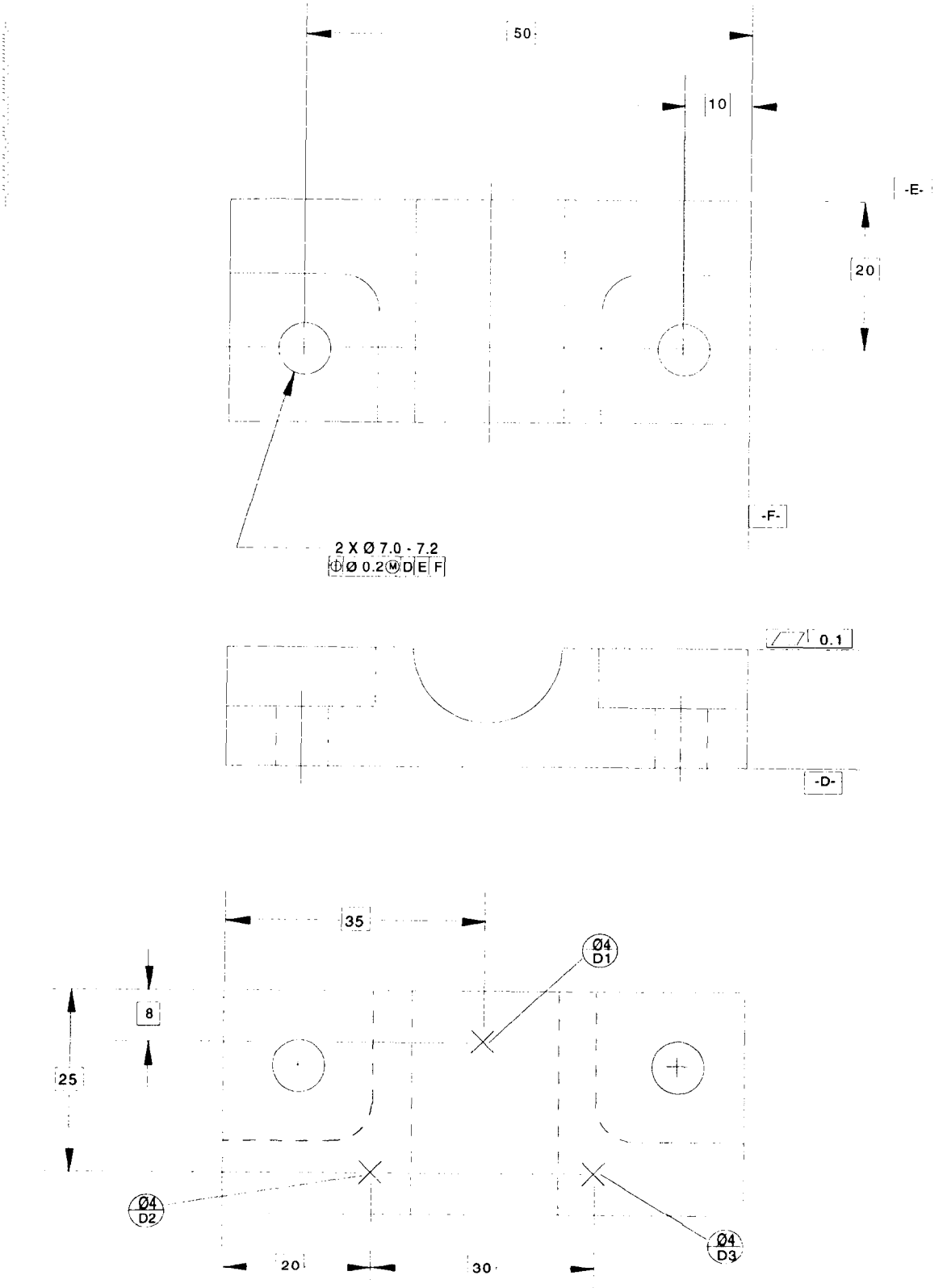

Figure 8 – Draughting callouts

4.2.30 Drawing

A Drawing is the presentation of product data in a human-interpretable form wherein the physical and functional requirements for that product are presented pictorially and textually. The data associated with a Drawing are the following:

- Contract_reference;
- Drawing_number;
- Drawing_revision_id;
- Drawing_specification;
- Dawing_type;
- Security_classification;
- Title.

4.2.30.1 Contract_reference

The Contract_reference specifies the identification of a legally enforceable agreement, between two or more parties, that defines activities related to the drawing. The Contract_reference need not be specified for a particular Drawing. There may be more than one Contract_reference for a Drawing.

4.2.30.2 Drawing_number

The Drawing_number specifies the identification of a particular drawing by an organization.

4.2.30.3 Drawing_revision_id

The Drawing_revision_id specifies the identification of a particular version of the drawing.

4.2.30.4 Drawing_specification

The Drawing_specification specifies the identification of a standard to which the drawing conforms. This standard specifies the presentation forms used in the drawing. The Drawing_specification need not be specified for a particular Drawing. There may be more than one Drawing_specification for a Drawing.

EXAMPLE 11 – A Drawing_specification may be ISO 129, Technical drawings - Dimensioning.

4.2.30.5 Drawing_type

The Drawing_type specifies the category of the drawing and may indicate the information content. The Drawing_type need not be specified for a particular Drawing.

EXAMPLE 12 – A Drawing_type may be an assembly drawing, casting drawing, elevation drawing, or a land title drawing.

4.2.30.6 Security_classification

The Security_classification specifies the organization and external designation of the level of protection from unauthorized access to the drawing. The Security_classification need not be specified for a particular Drawing. There may be more than one Security_classification for a Drawing.

4.2.30.7 Title

The Title specifies a description of the content or subject matter of the drawing. The Title may or may not be specified for a particular Drawing.

4.2.31 Drawing_sheet

A Drawing_sheet is a logical division of a drawing into a two-dimensional area for the presentation of product data. These divisions correspond to sheet paper sizes for plotting. A Drawing_sheet contains at least one Drawing_view (see 4.2.33) or one Draughting_annotation (see 4.2.28). The data associated with a Drawing_sheet are the following:

- Security_classification;
- Sheet_number;
- Sheet_revision_id;
- Size;
- Title.

4.2.31.1 Security_classification

The Security_classification specifies the organization and external designation of the level of protection from unauthorized access to the drawing sheet. The Security_classification need not be specified for a particular Drawing_sheet. There may be more than one Security_classification for a Drawing_sheet.

4.2.31.2 Sheet_number

The Sheet_number specifies the page number for a particular drawing sheet and its location in relation to other sheets of the drawing.

4.2.31.3 Sheet_revision_id

The Sheet_revision_id specifies the identification of a particular version of the drawing sheet.

4.2.31.4 Size

The Size specifies the physical size of the presentation area of the drawing sheet. This physical size also corresponds to the physical size of a sheet of paper on which the drawing sheet may be placed. The physical size is defined by the length and width of the sheet in specified units of measure.

NOTE – The standard sheet size designators for a specified draughting standard, such as ANSI Y14.2M-1979 or ISO STD HDBK 12, can be derived from these values.

4.2.31.5 Title

The Title specifies the contents or subject matter of the drawing sheet. The Title need not be specified for a particular Drawing_sheet.

4.2.32 Drawing_sheet_layout

A Drawing_sheet_layout is a type of User_defined_symbol_definition (see 4.2.84) that specifies the layout, or format, of a sheet of a drawing.

4.2.33 Drawing_view

A Drawing_view is a presentation on a drawing sheet of 2 2d projection of a draughting shape model, together with any annotation that refers to, or is relevant to, the view. The data

associated with a Drawing_view are the following:

- Location;
- Name;
- Rotation.

NOTE – Note that the 2d projection is specified by a 2D_drawing_view_definition (see 4.2.3).

4.2.33.1 Location

The Location specifies the position of the origin of the coordinate system of the drawing view relative to the origin of the coordinate system of the drawing sheet where it is placed.

4.2.33.2 Name

The Name specifies the identification of a particular drawing view. The Name need not be specified for a particular Drawing_view.

4.2.33.3 Rotation

The Rotation specifies the angle, measured counter-clockwise, between the horizontal axis of the coordinate system of the drawing view and the horizontal axis of the coordinate system of the drawing sheet where it is placed.

4.2.34 Externally_defined_hatching

An Externally_defined_hatching is a type of Fill_area_appearance (see 4.2.41) that has an explicit physical description defining the hatching and is found in a known source. This known source is agreed to by all parties involved in the exchange of the drawings on which the hatching pattern appears. An Externally_defined_hatching consists of one or more uniformly spaced geometric patterns. For each pattern, an Externally_defined_hatching shall include the specification of the curve appearance, the angle of the curves in the pattern relative to the horizontal axis of the coordinate system into which it is placed, and the displacement between adjacent curves in the pattern. The data associated with an Externally_defined_hatching are the following:

- Hatching_name;
- Hatching_reference.

4.2.34.1 Hatching_name

The Hatching_name specifies the identification of a particular hatching pattern within the known source.

4.2.34.2 Hatching_reference

The Hatching_reference specifies the known source that contains a set of patterns from which the hatching pattern is selected.

4.2.35 Externally_defined_line_font

An Externally_defined_line_font is a type of Line_font (see 4.2.53) that has an explicit physical description defining the line font and is found in a known source. This known source is agreed to by all parties involved in the exchange of the drawings on which the line font appears. An Externally_defined_line_font shall include a set of values which represent the length of the visible and invisible segments of the line font. The set of values is sufficient to define all elements which constitute a single portion of the curve. This portion is then repeated over the length of the curve. The data associated with an Externally_defined_line_font are the following:

- Font_name;
- Font_reference.

4.2.35.1 Font_name

The Font_name specifies the identification of a particular line font within the known source.

4.2.35.2 Font_reference

The Font_reference specifies the known source that contains a set of line fonts from which the line font is be selected.

4.2.36 Externally_defined_symbol

An Externally_defined_symbol is a type of Annotation_symbol (see 4.2.11) that has an explicit physical description defining the symbol and is found in a known source. This known source is agreed to by all parties involved in the exchange of the drawings on which the symbol appears. An Externally_defined_symbol shall include the specification of all the constituent components of the symbol, their size, and relative location, together with the origin or reference point of the symbol. The data associated with an Externally_defined_symbol are the following:

- Symbol_name;
- Symbol_reference.

4.2.36.1 Symbol_name

The Symbol_name specifies the identification of a particular symbol within the known source.

4.2.36.2 Symbol_reference

The Symbol_reference specifies the known source that contains a set of symbols from which the symbol is selected.

4.2.37 Externally_defined_text_font

An Externally_defined_text_font is a type of Text_font (see 4.2.76) that has an explicit physical description defining the text font and is found in a known source. This known source is agreed to by all parties involved in the exchange of the drawings on which the text font appears. An Externally_defined_text_font shall include the specification of the physical form of the characters of the font, together with the origin or reference point of each character. The data associated with an Externally_defined_text_font are the following:

- Font_name;
- Font_reference.

4.2.37.1 Font_name

The Font_name specifies the identification of a particular text font in the known source.

4.2.37.2 Font_reference

The Font_reference specifies the known source that contains a set of text fonts from which the text font is selected.

4.2.38 Externally_defined_tile

An Externally_defined_tile is a type of Tile (see 4.2.78) that has an explicit physical description defining the tile and is found in a known source. This known source is agreed to by all parties

involved in the exchange of the drawings on which the tile appears. An Externally_defined_tile shall include the specification of all the constituent components of the tile. The data associated with an Externally_defined_tile are the following:

- Tile_name;
- Tile_reference.

4.2.38.1 Tile_name

The Tile_name specifies the identification of a particular tile in the known source.

4.2.38.2 Tile_reference

The Tile_reference specifies the known source that contains a set of tiles from which the tile is selected.

4.2.39 Externally_defined_tiling

An Externally_defined_tiling is a type of Fill_area_appearance (see 4.2.41) that has an explicit physical description defining the tiling pattern and is found in a known source. This known source is agreed to by all parties involved in the exchange of the drawings on which the tiling appears. An Externally_defined_tiling specification shall include the repeat vectors used to define the relative positioning of tiles, the angle of the horizontal axis of the tile relative to the horizontal axis of the coordinate system into which it is placed, and the scale of the tile as presented to the tile as defined. The data associated with an Externally_defined_tiling are the following:

- Tiling_name;
- Tiling_reference.

4.2.39.1 Tiling_name

The Tiling_name specifies the identification of a particular tiling pattern in the known source.

4.2.39.2 Tiling_reference

The Tiling_reference specifies the known source that contains a set of patterns from which the tiling pattern is selected.

4.2.40 Fill_area

A Fill_area is a type of Annotation_element (see 4.2.7) that is a bounded area containing coloring, hatching, or tiling that indicate its extent and content. A Fill_area communicates some aspect of a physical part characteristic or distinguishes it from its surroundings. The data associated with a Fill_area are the following:

- Reference_point.

The Reference_point specifies a point within the fill area used in the placement and initiation of the fill area appearance. The Reference_point establishes a point through which a line of a hatching pattern passes or at which the origin of a tile is located. The Reference_point also establishes the point at which the first visible segment of a line font used as the curve appearance for a hatching pattern starts. The Reference_point need not be specified for a particular Fill_area.

4.2.41 Fill_area_appearance

A Fill_area_appearance is a type of Appearance (see 4.2.12) that governs the visual presentation of a fill area. Each Fill_area_appearance is either an Externally_defined_hatching (see 4.2.34), an Externally_defined_tiling (see 4.2.39), a Solid_fill_area (see 4.2.68), a User_defined_hatching (see 4.2.81), or a User_defined_tiling (see 4.2.86). The data associated with a Fill_area_appearance are the following:

- Draughting_role.

The Draughting_role specifies the purpose within draughting for a particular fill area appearance. The Draughting_role need not be specified for a particular Fill_area_appearance.

4.2.42 Fill_area_boundary

A Fill_area_boundary is an annotation curve in the same coordinate space that defines the limits of a fill area. The curve is closed and not self-intersecting.

4.2.43 Geometrical_tolerance

A Geometrical_tolerance is a type of Draughting_callout (see 4.2.29) that is the human-interpretable presentation of tolerance information. This presentation is a combination of geometric characteristic symbols, tolerance values, and datum reference designations, where applicable, to express the permissible variation from the theoretically exact size, profile, orientation, or location of a feature or datum target.

NOTE - Computer interpretable geometric tolerances are not within the scope of this application protocol.

4.2.44 Geometrical_tolerance_symbol

A Geometrical_tolerance_symbol is a type of Predefined_symbol (see 4.2.62) that is used to establish a tolerance zone within which the specified conditions of the tolerance apply.

NOTE - See 5.2.2.1.37 for a description of the geometrical tolerance symbols supported by this part of ISO 10303.

4.2.45 Group

A Group is a collection of elements and other previously defined groups into related sets. The data associated with a Group are the following:

- Name.

The Name specifies the identification of a particular group. The Name need not be specified for a particular Group.

4.2.46 Group_annotation_element

A Group_annotation_element is a type of Group_element (see 4.2.47) that is annotation.

4.2.47 Group_element

A Group_element is annotation, a geometric element, or another group that is a member of a group. Each Group_element is either a Group_annotation_element (see 4.2.46), a Group_geometric_element (see 4.2.48), or a Sub_group (see 4.2.70).

4.2.48 Group_geometric_element

A Group_geometric_element is a type of Group_element (see 4.2.47) that is a geometric element.

4.2.49 Hatching_pattern

A Hatching_pattern is a single, uniformly spaced geometric pattern of lines. The basis of the hatching pattern is an infinite straight line that is repeated across the fill area and clipped to its boundaries. A curve appearance is applied to an annotation curve based on the clipped line. The data associated with a Hatching_pattern are the following:

- Angle;

– Displacement.

4.2.49.1 Angle

The Angle specifies the angular rotation of the curves of the hatching pattern, measured counter-clockwise from the x-axis of the coordinate system into which it is placed.

4.2.49.2 Displacement

The Displacement specifies a vector that positions the adjacent lines of the hatch pattern from the current line.

4.2.50 Layer

A layer is a collection of elements that is used to organize a drawing and control the visibility of the elements. The association of specific application semantics to the assignment of elements to a layer is outside the scope of this Part of ISO 10303.

EXAMPLE 13 – the use of a convention for naming of layers to represent different aspects of a CAD model is an example of the association of specific application semantics.

– Layer_id;

– Name.

4.2.50.1 Layer_id

The Layer_id specifies the identification of a particular layer.

4.2.50.2 Name

The Name specifies a description of the content or the usage of a particular layer. The Name need not be specified for a particular Layer.

EXAMPLE 14 – The Name could be dimensions, plumbing, or furniture.

4.2.51 Leader

A Leader is a type of Directed_curve (see 4.2.27) that directs a dimension, a note, or a symbol to the intended place or point on a feature appearing on the drawing or in the draughting shape

model.

4.2.52 Leader_directed_dimension

A Leader_directed_dimension is a type of Dimension (see 4.2.22) that is the graphical presentation of a dimension value and is guided to the feature being dimensioned with a leader.

NOTE – Figure 9 illustrates a leader directed dimension used to depict information about the holes in this product. Linear dimensions are used to show the placement of the holes.

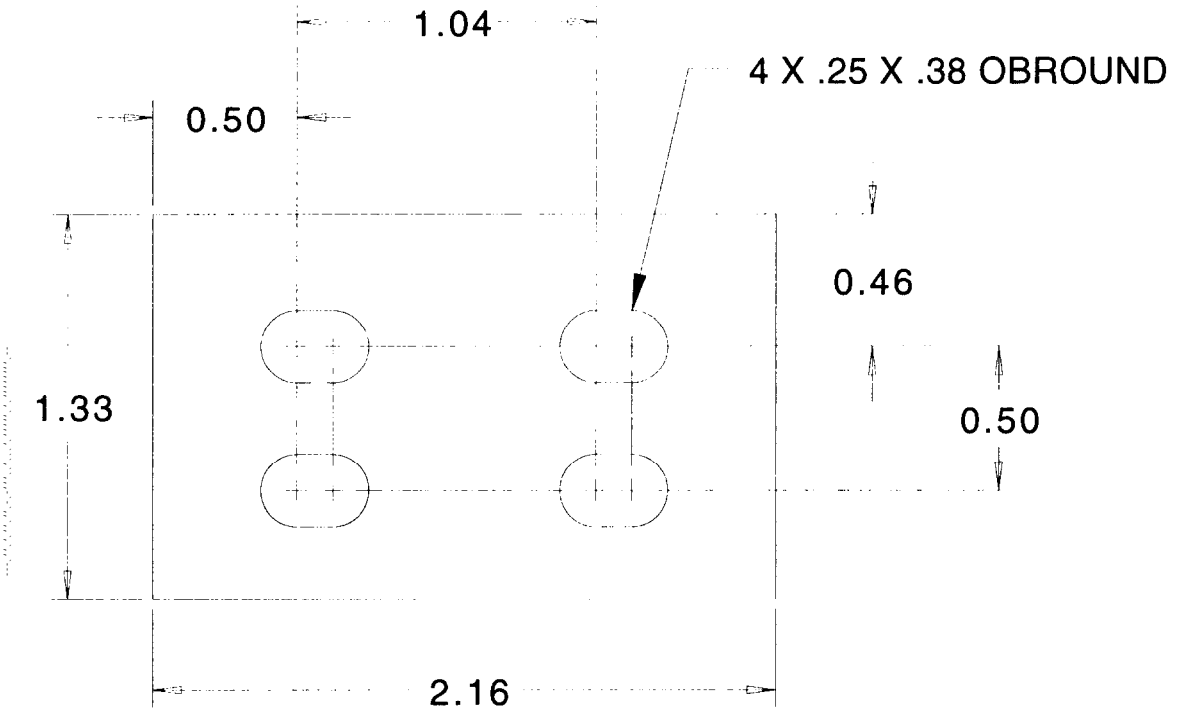

Figure 9 – Leader directed dimension

4.2.53 Line_font

A Line_font is a defined pattern of visible and invisible segments applied to a curve in a repetitive manner. Each Line_font is either an Externally_defined_line_font (see 4.2.35), a Predefined_line_font (see 4.2.61), or a User_defined_line_font (see 4.2.82).

4.2.54 Linear_dimension

A Linear_dimension is a type of Dimension (see 4.2.22) that is the graphical presentation of a value of linear distance measured between two points along a straight path.

NOTE – Figure 5, Figure 8, Figure 9, and Figure 11 show several linear dimensions used singularly or in dimension sequences.

4.2.55 Model_placed_annotation

A Model_placed_annotation is a type of Draughting_annotation (see 4.2.28) that is located in the coordinate system of the draughting shape model and is subject to view transformations for display.

4.2.56 Ordinate_dimension

An Ordinate_dimension is a type of Dimension (see 4.2.22) that is the graphical presentation of a value of linear distance measure where the linear distance is parallel to an axis of the coordinate system of the item being dimensioned. The origin or datum of the linear distance dimension is a point, line, or plane surface corresponding to, or coincident with, an axis in the plane of the dimension and perpendicular to the direction of measurement. Only the terminus of the dimension extent is indicated by a projection line parallel to the datum, the dimension value, and associated information.

NOTE – Figure 10 illustrates several ordinate dimensions used to dimension a simple mechanical product.

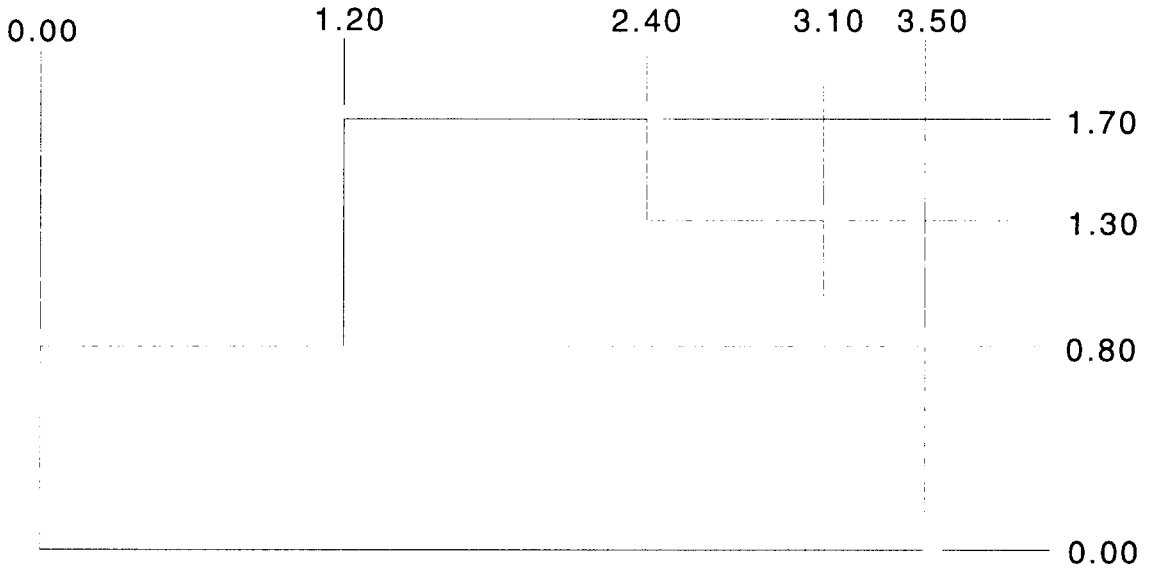

Figure 10 – Ordinate dimension

4.2.57 Organization

An Organization is a number of persons or groups that designs, produces and supplies products and services. The data associated with an Organization are the following:

- Address;
- Organization_name.

4.2.57.1 Address

The Address specifies the postal delivery or geographical location of the Organization. The Address need not be specified for a particular Organization. There may be more than one address for an Organization.

4.2.57.2 Organization_name

The Organization_name specifies the identification of a particular organization.

4.2.58 Parallel_dimension_pair

A Parallel_dimension_pair is a type of Dimension_sequence_pair (see 4.2.25) that is the relationship between two dimensions of the same type, wherein their dimension lines are parallel and share a common baseline or datum.

NOTE – Figure 11 illustrates a horizontal and a vertical parallel dimension sequence. The horizontal sequence is composed of four horizontal linear dimensions; each set of two dimensions sharing a projection line is a parallel dimension pair. The vertical sequence is composed of three vertical linear dimensions; each set of two dimensions sharing a projection line is a parallel dimension pair.

4.2.59 Point_marker_symbol

A Point_marker_symbol is a type of Predefined_symbol (see 4.2.62) that is used to visually present the location of a point in a draughting shape model, drawing sheet, drawing view, or another symbol.

NOTE – See 5.2.2.1.38 for a description of the point marker symbols supported by this part of ISO 10303.

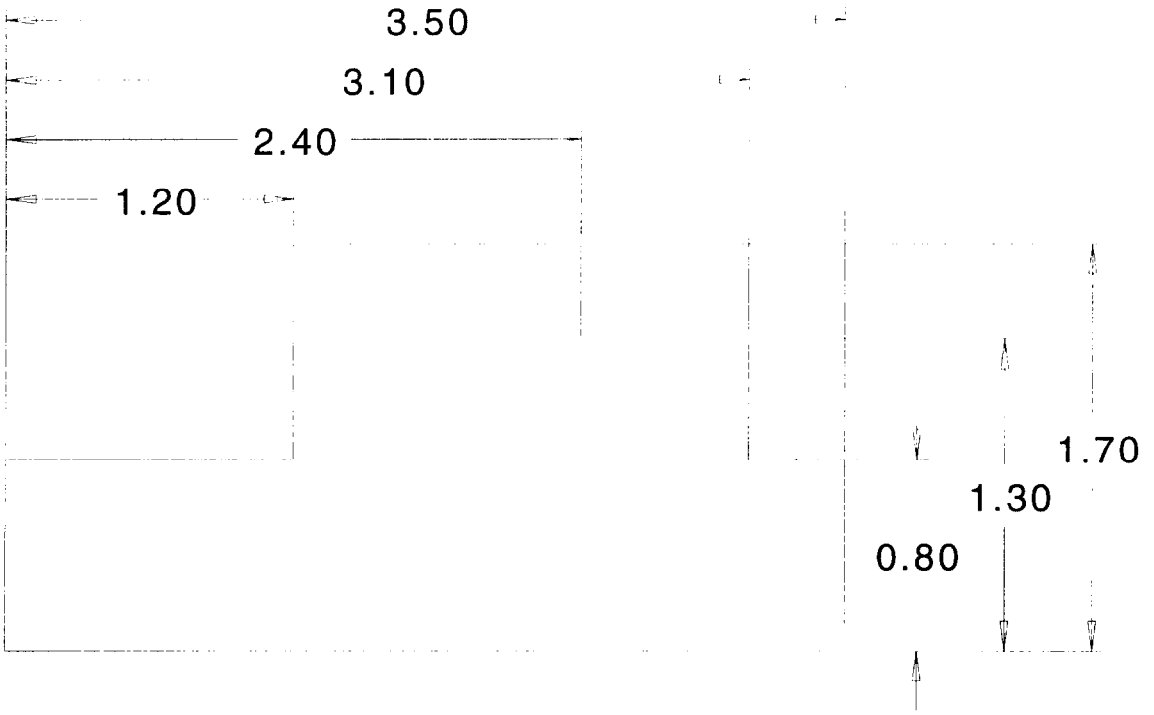

Figure 11 – Parallel dimension sequence

4.2.60 Predefined_colour

A Predefined_colour is a type of Colour (see 4.2.16) that has a specific visual appearance as defined in this part of ISO 10303.

NOTE – See 5.2.2.1.22 for a description of the colours supported by this part of ISO 10303.

4.2.61 Predefined_line_font

A Predefined_line_font is a type of Line_font (see 4.2.53) that has a specific visual appearance as defined in this part of ISO 10303.

NOTE – See 5.2.2.1.23 for a description of the line fonts supported by this part of ISO 10303.

4.2.62 Predefined_symbol

A Predefined_symbol is a type of Annotation_symbol (see 4.2.11) that has a specific visual appearance as defined in this part of ISO 10303. Each Predefined_symbol is either a Dimension_symbol (see 4.2.26), a Geometrical_tolerance_symbol (see 4.2.44), a Point_marker_symbol (see 4.2.59), or a Terminator_symbol (see 4.2.73).

4.2.63 Predefined_text_font

A Predefined_text_font is a type of Text_font (see 4.2.76) that has a specific physical appearance as defined in this part of ISO 10303.

NOTE – See 5.2.2.1.24 for a description of the text fonts supported by this part of ISO 10303.

4.2.64 Product_version

A Product_version is a variant of a product that is, or is intended to be, produced or employed in a production or construction process. The data associated with a Product_version are the following:

- Product_id;
- Revision_id.

4.2.64.1 Product_id

The Product_id specifies the identification of a product for a given organization.

4.2.64.2 Revision_id

The Revision_id specifies the identification of a variation of a product for a given organization.

4.2.65 Projection_line

A Projection_line is a type of Directed_curve (see 4.2.27) that represents the extension of a point, line, surface, or theoretical point of intersection to a location outside the part outline.

4.2.66 Radius_dimension

A Radius_dimension is a type of Dimension (see 4.2.22) that is the graphical presentation of the value of the radial distance from the centre of a circular element to a point on the element.

NOTE – Figure 4 illustrates a radius dimension. The figure also illustrates three angular dimensions depicting angular distance.

4.2.67 Sheet_placed_annotation

A Sheet_placed_annotation is a type of Draughting_annotation (see 4.2.28) that is located in the coordinate system of the drawing sheet.

4.2.68 Solid_fill_area

A Solid_fill_area is a type of Fill_area_appearance (see 4.2.41) defined by a single color which uniformly fills the fill area to which the appearance is applied.

4.2.69 Structured_dimension_callout

A Structured_dimension_callout is a type of Dimension_callout (see 4.2.23) wherein each component is identified as having the semantics of prefix information, suffix information, a dimension symbol, a dimension unit, a dimension value, or tolerance information.

NOTE – Figure 12 illustrates the components of a structured dimension callout.

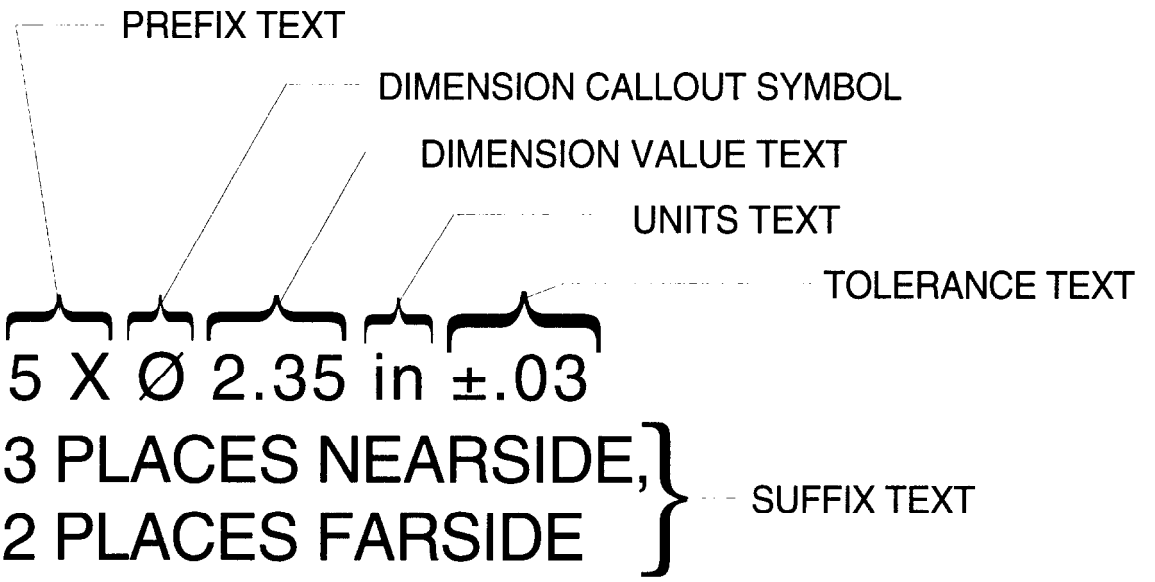

Figure 12 – Structured dimension callout

4.2.70 Sub_group

A Sub_group is a type of Group_element (see 4.2.47) that is a collection of elements previously defined as a group.

4.2.71 Sub_model

A Sub_model is the presentation of a Sub_model_definition located within the coordinate system of a draughting shape model. The data associated with a Sub_model are the following:

- Transformation.

The Transformation specifies the mathematical values that define the relationship between elements located in the coordinate system of the sub-model definition and their location in the coordinate system of the draughting shape model.

4.2.72 Sub_model_definition

A Sub_model_definition is a type of 2D_draughting_shape_model (see 4.2.2) that represents a constituent part of a product. The product is represented by a Draughting_shape_model of which one or more instances, or copies, of the Sub_model_definition are a part.

4.2.73 Terminator_symbol

A Terminator_symbol is a type of Predefined_symbol (see 4.2.62) applied to an annotation curve and used to identify the endpoint or point of application of any annotation directed by that curve.

NOTE – See 5.2.2.1.39 for a description of the terminator symbols supported by this part of ISO 10303.

4.2.74 Text

A Text is a type of Annotation_element (see 4.2.7) that is a collection of characters which convey some human-interpretable information. The data associated with a Text are the following:

- Alignment;
- Blanking_box;
- Displayed_box;
- Mirror_angle;
- Surrounding_box.

NOTE – See Figure 13 for an illustration of the blanking box of text, the location and rotation of a text string, the alignment of text, and the surrounding box of text.

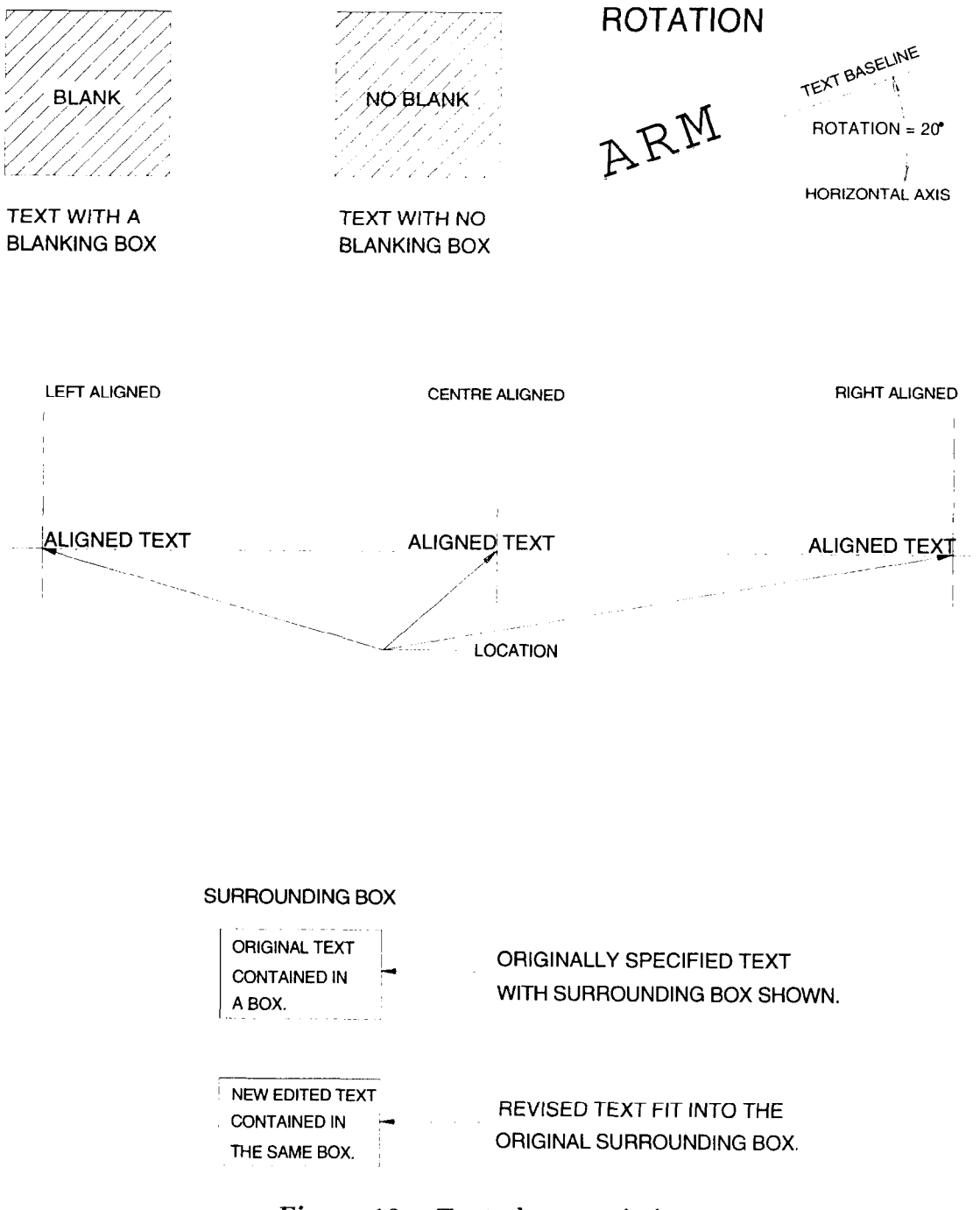

Figure 13 – Text characteristics

4.2.74.1 Alignment

The Alignment specifies a reference point on the text baseline, either left, right, or centre, that is used to align each line of the text.

4.2.74.2 Blanking_box

The Blanking_box specifies an area that the text occupies and is used to suppress the visual presentation of all other elements which are within this area. The Blanking_box need not be specified for a particular Text.

4.2.74.3 Displayed_box

The Displayed_box specifies a rectangular box, composed of annotation curves, that encloses text where one side of the box is parallel to the text baseline. The Displayed_box need not be specified for a particular Text.

4.2.74.4 Mirror_angle

The Mirror_angle specifies the angle, measured in a counter-clockwise direction from the text baseline, to an axis about which the text is mirrored. The mirror axis and text baseline intersect at the Location of the Text. The Mirror_angle need not be specified for a particular Text.

NOTE – See Figure 14 for an illustration of the mirror angle characteristic of text.

4.2.74.5 Surrounding_box

The Surrounding_box specifies the physical space that the text occupies and is defined by width, the distance of the left-most point of the left-most character to the right-most point of the right-most character measured parallel to the text baseline, and height, the distance of the lowest point of the lowest reaching character to the highest point of the highest reaching character measured perpendicular to the text baseline. The Surrounding_box need not be specified for a particular Text.

NOTE – Figure 13 is informative only; the Surrounding_box information may be used for this purpose, but is not required.

4.2.75 Text_appearance

A Text_appearance is a type of Appearance (see 4.2.12) that governs the visual presentation of text. The data associated with a Text_appearance are the following:

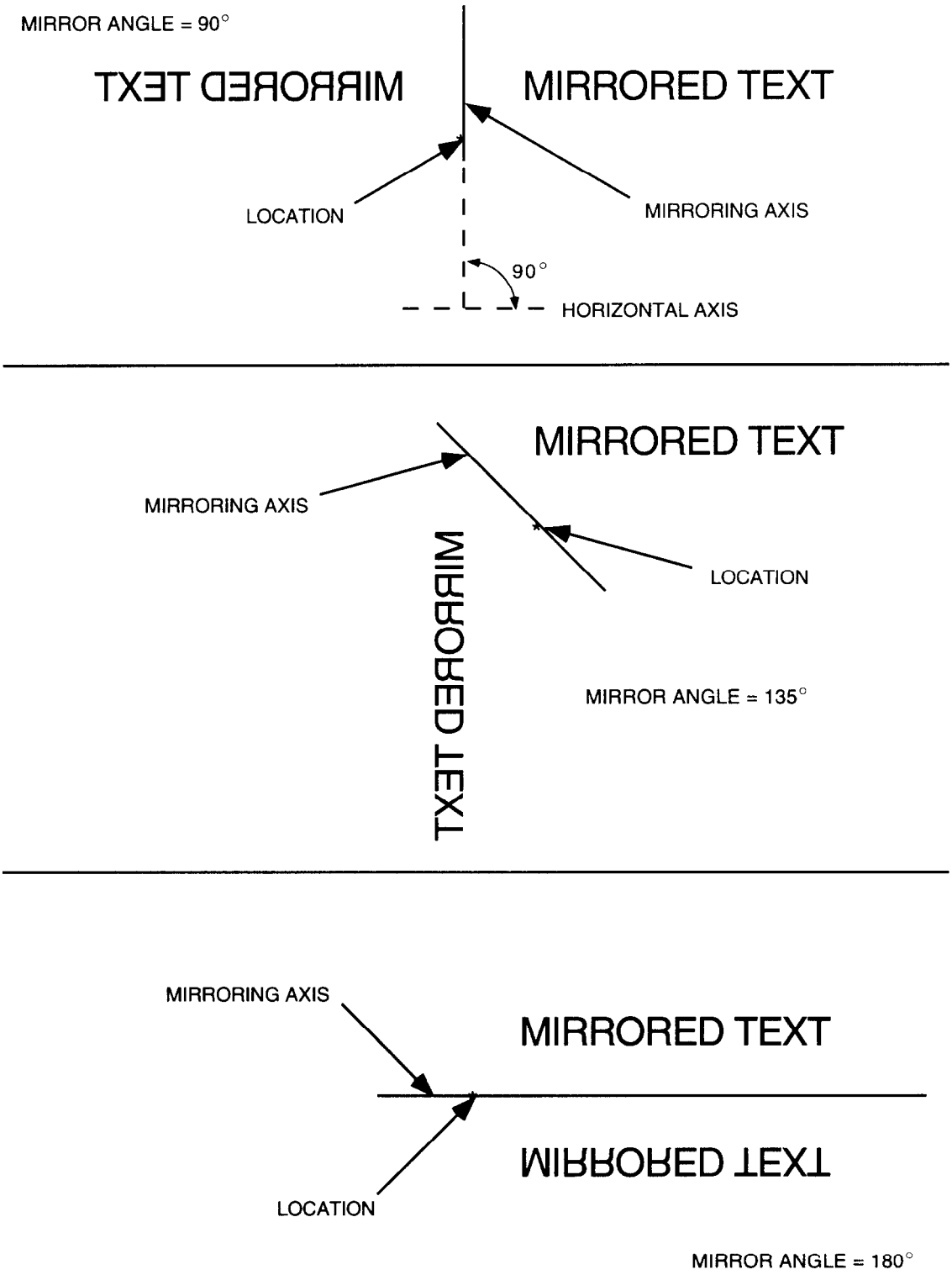

Figure 14 – Mirror angle

– Character_aspect_ratio;

– Character_rotation_angle;

– Character_scale;

– Character_slant_angle.

NOTE – See Figure 15 for an illustration of character_aspect_ratio, character_rotation_angle and character_slant_angle.

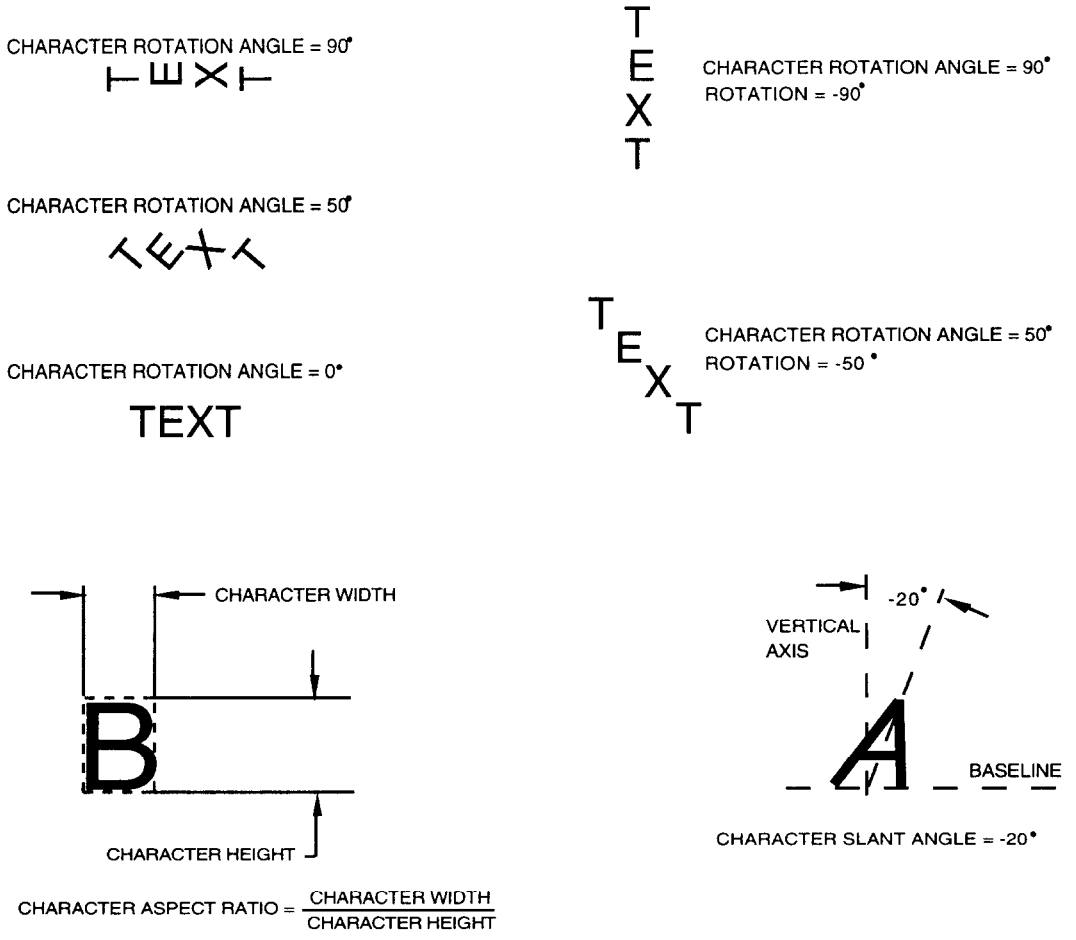

Figure 15 – Text appearance and rotation of text

4.2.75.1 Character_aspect_ratio

The Character_aspect_ratio specifies the ratio of the width of the character to the height of the character.

4.2.75.2 Character_rotation_angle

The Character_rotation_angle specifies the angular counter-clockwise rotation of each character within the text string in which it appears. The point of rotation is the left-most point of each character at its baseline.

4.2.75.3 Character_scale

The Character_scale specifies the ratio of the size of the text character as defined to the size of the text character as presented.

4.2.75.4 Character_slant_angle

The Character_slant_angle specifies the angular distance between vertical aspects of the individual character and an axis perpendicular to the baseline of the character, measured clockwise.

4.2.76 Text_font

A Text_font is the description of each individual character of a character set, including its form and spatial characteristics. Each Text_font is either an Externally_defined_text_font (see 4.2.37) or a Predefined_text_font (see 4.2.63).

NOTE – See Figure 13 for an illustration of the location and rotation of a text string.

4.2.77 Text_string

A Text_string is the smallest unit of text and is a collection of one or more characters that convey some human-interpretable information. The data associated with a Text_string are the following:

- Character_alignment;
- Character_string;
- Location;
- Overline_underline;
- Rotation;
- Surrounding_box.

4.2.77.1 Character_alignment

The Character_alignment specifies the relative position of successive characters in a string of text. The Character_alignment is either left, right, up, or down.

4.2.77.2 Character_string

The Character_string specifies the list of characters that compose the string of text.

4.2.77.3 Location

The Location specifies the position of the text string in the placement coordinate system.

4.2.77.4 Overline_underline

The Overline_underline specifies that there is a line placed either over or under the text string. The Overline_underline need not be specified for a particular Text_string.

4.2.77.5 Rotation

The Rotation specifies the angle, measured counter-clockwise, between the baseline of the text and the horizontal axis of the coordinate system into which it is being placed.

4.2.77.6 Surrounding_box

The Surrounding_box specifies the physical space that the text string occupies and is defined by width, the distance of the left-most point of the left-most character to the right-most point of the right-most character measured parallel to the text baseline, and height, the distance of the lowest point of the lowest reaching character to the highest point of the highest reaching character measured perpendicular to the text baseline.

4.2.78 Tile

A Tile is a graphical symbol defined within a containment border and used as the content of a tiling pattern. The containment border defines the edges of the tile. All elements contained within the border are duplicated for each tile. Tiles are placed within a fill area adjacently and do not overlap. The containment border may be blanked. Each Tile is either an Externally_defined_tile (see 4.2.38) or a User_defined_tile (see 4.2.85).

4.2.79 Unstructured_dimension_callout

An Unstructured_dimension_callout is a type of Dimension_callout (see 4.2.23) wherein a single draughting callout is used and the components of the dimension callout are not semantically identified.

4.2.80 User_defined_colour

A User_defined_colour is a type of Colour (see 4.2.16) that is defined by an explicit listing of the proportions of blue, green, and red. The data associated with a User_defined_colour are the following:

- Blue_proportion;
- Green_proportion;
- Name;
- Red_proportion.

4.2.80.1 Blue_proportion

The Blue_proportion specifies the level of intensity of the colour blue to be displayed.

4.2.80.2 Green_proportion

The Green_proportion specifies the level of intensity of the colour green to be displayed.

4.2.80.3 Name

The Name specifies the title for the composition of red, green and blue values of a particular User_defined_colour.

4.2.80.4 Red_proportion

The Red_proportion specifies the level of intensity of the colour red to be displayed.

4.2.81 User_defined_hatching

A User_defined_hatching is a type of Fill_area_appearance (see 4.2.41) that is defined by an explicit listing of hatch patterns.

4.2.82 User_defined_line_font

A User_defined_line_font is a type of Line_font (see 4.2.53) that is defined by an explicit listing of the visible and invisible segments that make up the pattern of the font. The data associated with a User_defined_line_font are the following:

- Name;
- Pattern.

4.2.82.1 Name

The Name specifies the identification of a particular line font. The Name need not be specified for a particular User_defined_line_font.

4.2.82.2 Pattern

The Pattern specifies a list of length values of visible and invisible segments. The number of values in the list is even and greater than or equal to two.

4.2.83 User_defined_symbol

A User_defined_symbol is a type of Annotation_symbol (see 4.2.11) that is defined by an explicit listing of annotation elements that make up the symbol along with their positions within the coordinate system in which the symbol is defined.

4.2.84 User_defined_symbol_definition

A User_defined_symbol_definition is a collection of defined annotation elements, along with their placements, in a definitional coordinate space that represents a distinct concept. Each User_defined_symbol_definition may be a Drawing_sheet_layout (see 4.2.32). The data associated with a User_defined_symbol_definition are the following:

- Symbol_definition_id.

4.2.84.1 Symbol_definition_id

The Symbol_definition_id specifies the identification of a particular symbol.

4.2.85 User_defined_tile

A User_defined_tile is a type of Tile (see 4.2.78) that is defined by an explicit listing of the components that make up the tile.

4.2.86 User_defined_tiling

A User_defined_tiling is a type of Fill_area_appearance (see 4.2.41) that is defined by an explicit listing of tiles. The data associated with a User_defined_tiling are the following:

- Angle;
- Repeat_vector_1;
- Repeat_vector_2;
- Scale.

4.2.86.1 Angle

The Angle specifies the rotation of the tile, measured counter-clockwise, relative to the x-axis of the coordinate system in which the boundary of the fill area is defined.

4.2.86.2 Repeat_vector_1

The Repeat_vector_1 specifies the direction and the distance in that direction at which to place the tile relative to the placement of a previous tile.

4.2.86.3 Repeat_vector_2

The Repeat_vector_2 specifies the secondary direction and the distance in that direction at which to place the tile relative to the placement of a previous tile.

4.2.86.4 Scale

The Scale specifies the ratio between the size of the tile as defined and the size of the tile as presented.

4.2.87 View_displayed_geometry

A View_displayed_geometry is the identification and assignment of appearance characteristics to a specific geometric element of the draughting shape model when the element is presented in a drawing view.

NOTE – When the specific element of the draughting shape model is a point which is to be displayed in a drawing view, it has an appearance assigned through the use of a point marker symbol located at the position of the geometric point.

4.2.88 View_displayed_model_annotation

A View_displayed_model_annotation is the identification and assignment of appearance characteristics to a specific annotation element of the draughting shape model when the annotation element is presented in a drawing view.

4.2.89 View_placed_annotation

A View_placed_annotation is a type of Draughting_annotation (see 4.2.28) that is located in the coordinate system of the drawing view.

4.2.90 Visibility

A Visibility is an indication of whether or not an individual element or collection of elements are displayed in the visual presentation of the drawing. Visibility takes precedence over all other appearance characteristics assigned to the element.

EXAMPLE 15 – A construction line is an element not meant for display on a drawing, therefore visibility would indicate this.

4.3 Application assertions

This subclause specifies the application assertions for the explicit draughting application protocol. Application assertions specify the relationships between application objects, the cardinality of the relationships, and the rules required for the integrity and validity of the application objects and UoFs. The application assertions and their definitions are given below.

4.3.1 2D_draughting_shape_model to 2D_geometric_element

Each 2D_draughting_shape_model consists of zero, one, or many 2D_geometric_element objects. Each 2D_geometric_element belongs to exactly one 2D_draughting_shape_model.

4.3.2 2D_draughting_shape_model to Cartesian_coordinate_space

Each 2D_draughting_shape_model has as a coordinate space exactly one Cartesian_coordinate_space. Each Cartesian_coordinate_space is the coordinate space of zero, one, or many 2D_draughting_shape_model objects.

4.3.3 2D_draughting_shape_model to Model_placed_annotation

Each 2D_draughting_shape_model contains zero, one, or many Model_placed_annotation objects. Each Model_placed_annotation is contained in exactly one 2D_draughting_shape_model.

4.3.4 2D_draughting_shape_model to Sub_model

Each 2D_draughting_shape_model consists of zero, one, or many Sub_model objects. Each Sub_model belongs to exactly one 2D_draughting_shape_model.

4.3.5 2D_drawing_view_definition to View_displayed_geometry

Each 2D_drawing_view_definition displays zero, one, or many View_displayed_geometry objects. Each View_displayed_geometry is displayed in exactly one 2D_drawing_view_definition.

4.3.6 2D_drawing_view_definition to View_displayed_model_annotation

Each 2D_drawing_view_definition displays zero, one, or many View_displayed_model_annotation objects. Each View_displayed_model_annotation is displayed in exactly one 2D_drawing_view_definition.

4.3.7 2D_geometric_element to Layer

Each 2D_geometric_element is contained in one or more Layer objects. Each Layer contains zero, one, or many 2D_geometric_element objects.

4.3.8 Angular_dimension to Dimension_line

Each Angular_dimension has the extent of exactly one Dimension_line. Each Dimension_line may show the extent of exactly one Angular_dimension.

4.3.9 Angular_dimension to Projection_line

Each Angular_dimension has as a component zero, one, or two Projection_line objects. Each Projection_line is a component of zero, one, or many Angular_dimension objects.

4.3.10 Annotation_curve to Curve_appearance

Each Annotation_curve has exactly one Curve_appearance. Each Curve_appearance applies to zero, one, or many Annotation_curve objects.

4.3.11 Annotation_subfigure to Annotation_subfigure_definition

Each Annotation_subfigure is defined by exactly one Annotation_subfigure_definition. Each Annotation_subfigure_definition defines one or more Annotation_subfigure objects.

4.3.12 Annotation_subfigure_definition to 2D_cartesian_coordinate_space

Each Annotation_subfigure_definition has as a coordinate space exactly one 2D_cartesian_coordinate_space. Each 2D_cartesian_coordinate_space is the coordinate space of zero or one Annotation_subfigure_definition.

4.3.13 Annotation_subfigure_definition to Annotation_subfigure_definition_element

Each Annotation_subfigure_definition is composed of one or more Annotation_subfigure_definition_element objects. Each Annotation_subfigure_definition_element is a component of exactly one Annotation_subfigure_definition.

4.3.14 Annotation_subfigure_definition_element to Draughting_annotation

Each Annotation_subfigure_definition_element is exactly one Draughting_annotation. Each Draughting_annotation is used as zero or one Annotation_subfigure_definition_element.

4.3.15 Annotation_subfigure_definition_element to Layer

Each Annotation_subfigure_definition_element is contained in one or more Layer objects. Each Layer contains zero, one, or many Annotation_subfigure_definition_element objects.

4.3.16 Annotation_subfigure_definition_element to Visibility

Each Annotation_subfigure_definition_element has display governed by exactly one Visibility. Each Visibility governs the display of zero or one Annotation_subfigure_definition_element.

4.3.17 Annotation_symbol to Colour

Each Annotation_symbol may have as overriding colour exactly one Colour. Each Colour provides an overriding colour for zero, one, or many Annotation_symbol objects.

4.3.18 Approval to Organization

Each Approval is provided by one or more Organization objects. Each Organization provides zero, one, or many Approval objects.

4.3.19 Curve_appearance to Colour

Each Curve_appearance is partially defined by exactly one Colour. Each Colour partially defines zero, one, or many Curve_appearance objects.

4.3.20 Curve_appearance to Line_font

Each Curve_appearance is partially defined by exactly one Line_font. Each Line_font partially defines one or more Curve_appearance objects.

4.3.21 Curve_dimension to Dimension_line

Each Curve_dimension has the extent of exactly one Dimension_line. Each Dimension_line may show the extent of exactly one Curve_dimension.

4.3.22 Curve_dimension to Projection_line

Each Curve_dimension has as a component zero, one, or two Projection_line objects. Each Projection_line is a component of zero, one, or many Curve_dimension objects.

4.3.23 Diameter_dimension to Dimension_line

Each Diameter_dimension has the extent of exactly one Dimension_line. Each Dimension_line may show the extent of exactly one Diameter_dimension.

4.3.24 Diameter_dimension to Projection_line

Each Diameter_dimension has as a component zero, one, or two Projection_line objects. Each Projection_line is a component of zero, one, or many Diameter_dimension objects.

4.3.25 Dimension to Dimension_callout

Each Dimension has as a primary callout exactly one Dimension_callout.

Each Dimension has as a secondary callout zero or one Dimension_callout.

Each Dimension_callout is the primary callout or secondary callout for exactly one Dimension.

NOTE – The primary callout is the dimension value, tolerance information, and any associated notes specified in the primary unit of measure for that drawing view, drawing sheet, or drawing. The secondary callout is the same information specified in a second unit of measure. The primary callout could be in millimeters and the secondary in inches.

4.3.26 Dimension to Dimension_sequence_pair

Each Dimension is the predecessor for zero, one, or many Dimension_sequence_pair objects. Each Dimension_sequence_pair has as a predecessor exactly one Dimension.

Each Dimension is the successor for zero, one, or many Dimension_sequence_pair objects. Each Dimension_sequence_pair has as a successor exactly one Dimension.

4.3.27 Dimension_line to Annotation_symbol

Each Dimension_line has extent indicated by zero, one, or two Annotation_symbol objects. Each Annotation_symbol may indicate the extent of exactly one Dimension_line. Each Annotation_symbol used to show the extent of a Dimension_line may only be a User_defined_symbol, an Externally_defined_symbol, or a Predefined_symbol of type Terminator_symbol.

4.3.28 Draughting_callout to Annotation_curve

Each Draughting_callout is composed of zero, one, or many Annotation_curve objects. Each Annotation_curve is a component of zero or one Draughting_callout.

4.3.29 Draughting_callout to Annotation_Symbol

Each Draughting_callout is composed of zero, one, or many Annotation_symbol objects. Each Annotation_symbol is a component of zero or one Draughting_callout.

4.3.30 Draughting_callout to Directed_curve

Each Draughting_callout is directed by zero, one, or many Directed_curve objects. Each Directed_curve shall direct exactly one Draughting_callout.

4.3.31 Draughting_callout to Text

Each Draughting_callout is composed of zero, one, or many Text. Each Text is a component of zero or one Draughting_callout.

4.3.32 Drawing to Approval

Each Drawing is governed by zero, one, or many Approval objects. Each Approval governs zero or one Drawing.

4.3.33 Drawing to Drawing_sheet

Each Drawing consists of one or more Drawing_sheet objects. Each Drawing_sheet belongs to exactly one Drawing.

4.3.34 Drawing to Organization

Each Drawing is the responsibility of one or more Organization objects. Each Organization has responsibility for zero, one, or many Drawing objects.

4.3.35 Drawing to Product_version

Each Drawing documents one or more Product_version objects. Each Product_version is documented by zero, one, or many Drawing objects. A drawing may document more than one product, but the drawing will only document one version of each product.

EXAMPLE 16 – Multi-detail drawings document more than one product version.

4.3.36 Drawing_sheet to 2D_cartesian_coordinate_space

Each Drawing_sheet has as a coordinate space exactly one 2D_cartesian_coordinate_space. Each 2D_cartesian_coordinate_space is the coordinate space of zero or one Drawing_sheet.

4.3.37 Drawing_sheet to Approval

Each Drawing_sheet is governed by zero, one, or many Approval objects. Each Approval governs zero, one, or many Drawing_sheet objects.

4.3.38 Drawing_sheet to Drawing_view

Each Drawing_sheet contains zero, one, or many Drawing_view objects. Each Drawing_view is contained in exactly one Drawing_sheet.

NOTE – A Drawing_sheet shall contain only annotation if views of the 2D_draughting_shape_model are not defined or if the Product_version is not defined by a 2D_draughting_shape_model.

4.3.39 Drawing_sheet to Organization

Each Drawing_sheet is the responsibility of zero, one, or many Organization objects. Each Organization has responsibility for zero, one, or many Drawing_sheet objects.

4.3.40 Drawing_sheet to Sheet_placed_annotation

Each Drawing_sheet contains zero, one, or many Sheet_placed_annotation objects. Each Sheet_placed_annotation is contained in exactly one Drawing_sheet.

NOTE – A Drawing_sheet shall contain only views of the 2D_draughting_shape_model which defines the Product_version and no additional annotation.

4.3.41 Drawing_view to 2D_cartesian_coordinate_space

Each Drawing_view has as a coordinate space exactly one 2D_cartesian_coordinate_space. Each 2D_cartesian_coordinate_space is the coordinate space of zero or one Drawing_view.

4.3.42 Drawing_view to 2D_drawing_view_definition

Each Drawing_view is defined by exactly one 2D_drawing_view_definition. Each 2D_drawing_view_definition defines one or more Drawing_view objects.

4.3.43 Drawing_view to View_placed_annotation

Each Drawing_view contains zero, one, or many View_placed_annotation objects. Each View_placed_annotation is contained in exactly one Drawing_view.

4.3.44 Fill_area to Fill_area_appearance

Each Fill_area has exactly one Fill_area_appearance. Each Fill_area_appearance applies to zero, one, or many Fill_area objects.

4.3.45 Fill_area to Fill_area_boundary

Each Fill_area is contained within one or more Fill_area_boundary objects. Each Fill_area_boundary bounds one or more Fill_area objects.

4.3.46 Fill_area_boundary to Annotation_curve

Each Fill_area_boundary is defined by exactly one Annotation_curve. Each Annotation_curve defines zero or one Fill_area_boundary.

4.3.47 Group to Group_element

Each Group contains one or more Group_element objects. Each Group_element is a member of one or more Group objects.

4.3.48 Group_annotation_element to Draughting_annotation

Each Group_annotation_element is exactly one Draughting_annotation. Each Draughting_annotation is used as zero or one Group_annotation_element.

4.3.49 Group_geometric_element to 2D_geometric_element

Each Group_geometric_element is exactly one 2D_geometric_element. Each 2D_geometric_element is used as zero or one Group_geometric_element.

4.3.50 Hatching_pattern to Curve_appearance

Each Hatching_pattern has exactly one Curve_appearance. Each Curve_appearance applies to zero, one, or many Hatching_pattern objects.

4.3.51 Layer to Visibility

Each Layer has display governed by exactly one Visibility. Each Visibility governs the display of zero or one Layer.

4.3.52 Leader to Annotation_symbol

Each Leader is terminated by zero or one Annotation_symbol. Each Annotation_symbol may terminate exactly one Leader. Each Annotation_symbol used to terminate a Leader may only be a User_defined_symbol, an Externally_defined_symbol, or a Predefined_symbol of type Terminator_symbol.

4.3.53 Leader_directed_dimension to Leader

Each Leader_directed_dimension has as a component exactly one Leader. Each Leader is a component of zero or one Leader_directed_dimension.

4.3.54 Linear_dimension to Dimension_line

Each Linear_dimension has the extent of exactly one Dimension_line. Each Dimension_line may show the extent of exactly one Linear_dimension.

4.3.55 Linear_dimension to Projection_line

Each Linear_dimension has as a component zero, one, or two Projection_line objects. Each Projection_line is a component of zero, one, or many Linear_dimension objects.

4.3.56 Model_placed_annotation to Layer

Each Model_placed_annotation is contained in one or more Layer objects. Each Layer contains zero, one, or many Model_placed_annotation objects.

4.3.57 Model_placed_annotation to Visibility

Each Model_placed_annotation has display governed by exactly one Visibility. Each Visibility governs the display of zero or one Model_placed_annotation.

4.3.58 Ordinate_dimension to Projection_line

Each Ordinate_dimension has as a component exactly one Projection_line. Each Projection_line is a component of zero, one, or many Ordinate_dimension objects.

4.3.59 Product_version to 2D_draughting_shape_model

Each Product_version is geometrically defined by zero, one, or many 2D_draughting_shape_model objects. Each 2D_draughting_shape_model geometrically defines zero, one, or many Product_version objects.

EXAMPLE 17 – A draughting shape model is not necessary in the exchange of textual drawings for a product.

4.3.60 Product_version to Organization

Each Product_version is the responsibility of one or more Organization objects. Each Organization has responsibility for zero, one, or many Product_version objects.

4.3.61 Radius_dimension to Dimension_line

Each Radius_dimension has the extent of exactly one Dimension_line. Each Dimension_line may show the extent of exactly one Radius_dimension.

4.3.62 Radius_dimension to Projection_line

Each Radius_dimension may have as a component exactly one Projection_line. Each Projection_line is a component of zero, one, or many Radius_dimension objects.

4.3.63 Sheet_placed_annotation to Layer

Each Sheet_placed_annotation is contained in one or more Layer objects. Each Layer contains zero, one, or many Sheet_placed_annotation objects.

4.3.64 Sheet_placed_annotation to Visibility

Each Sheet_placed_annotation has display governed by exactly one Visibility. Each Visibility governs the display of zero or one Sheet_placed_annotation.

4.3.65 Solid_fill_area to Colour

Each Solid_fill_area is defined by exactly one Colour. Each Colour defines zero, one, or many Solid_fill_area objects.

4.3.66 Structured_dimension_callout to Annotation_symbol

Each Structured_dimension_callout has as a symbol zero or one Annotation_symbol. Each Annotation_symbol shall be a symbol for exactly one Structured_dimension_callout.

4.3.67 Structured_dimension_callout to Text_string

Each Structured_dimension_callout has as a dimension value one or more Text_string objects. Each Text_string may be the dimension value for exactly one Structured_dimension_callout.

Each Structured_dimension_callout has as a tolerance value zero, one, or many Text_string objects. Each Text_string may be the tolerance value for exactly one Structured_dimension_callout.

Each Structured_dimension_callout has as unit text zero, one, or many Text_string objects. Each Text_string may be the unit text for exactly one Structured_dimension_callout.

4.3.68 Structured_dimension_callout to Draughting_callout

Each Structured_dimension_callout has as prefix zero or one Draughting_callout. Each Draughting_callout shall be the prefix for exactly one Structured_dimension_callout.

Each Structured_dimension_callout has as suffix zero or one Draughting_callout. Each Draughting_callout shall be the suffix for exactly one Structured_dimension_callout.

4.3.69 Sub_group to Group

Each Sub_group is exactly one Group. Each Group is used as zero, one, or many Sub_group objects.

4.3.70 Sub_model to Sub_model_definition

Each Sub_model is defined by exactly one Sub_model_definition. Each Sub_model_definition defines one or more Sub_model objects.

4.3.71 Text to Text_appearance

Each Text has exactly one default Text_appearance. Each Text_appearance is the default appearance for zero, one, or many Text objects.

4.3.72 Text to Text_string

Each Text is composed of one or more Text_string objects. Each Text_string is a component of exactly one Text.

4.3.73 Text_appearance to Colour

Each Text_appearance is partially defined by exactly one Colour. Each Colour partially defines zero, one, or many Text_appearance objects.

4.3.74 Text_appearance to Text_font

Each Text_appearance is partially defined by exactly one Text_font. Each Text_font partially defines one or more Text_appearance objects.

4.3.75 Text_string to Text_appearance

Each Text_string may have as an overriding appearance exactly one Text_appearance. Each Text_appearance applies to zero, one, or many Text_string objects.

4.3.76 Tile to Colour

Each Tile may have as overriding colour exactly one Colour. Each Colour provides an overriding colour for zero, one, or many Tile objects.

4.3.77 Unstructured_dimension_callout to Draughting_callout

Each Unstructured_dimension_callout is exactly one Draughting_callout. Each Draughting_callout is used as zero or one Unstructured_dimension_callout.

4.3.78 User_defined_hatching to Hatching_pattern

Each User_defined_hatching is defined by one or more Hatching_pattern objects. Each Hatching_pattern defines one or more User_defined_hatching objects.

4.3.79 User_defined_symbol to User_defined_symbol_definition

Each User_defined_symbol is defined by exactly one User_defined_symbol_definition. Each User_defined_symbol_definition defines zero, one, or many User_defined_symbol objects.

4.3.80 User_defined_symbol_definition to 2D_cartesian_coordinate_space

Each User_defined_symbol_definition has as a coordinate space exactly one 2D_cartesian_coordinate_space. Each 2D_cartesian_coordinate_space is the coordinate space of zero or one User_defined_symbol_definition.

4.3.81 User_defined_symbol_definition to Annotation_element

Each User_defined_symbol_definition is composed of one or more Annotation_element objects. Each Annotation_element is a component of zero or one User_defined_symbol_definition.

4.3.82 User_defined_tile to User_defined_symbol_definition

Each User_defined_tile is defined by exactly one User_defined_symbol_definition. Each User_defined_symbol_definition defines zero, one, or many User_defined_tile objects.

4.3.83 User_defined_tiling to Tile

Each User_defined_tiling is defined by exactly one Tile. Each Tile defines one or more User_defined_tiling objects.

4.3.84 View_displayed_geometry to 2D_geometric_element

Each View_displayed_geometry displays exactly one 2D_geometric_element. Each 2D_geometric_element is displayed as zero, one, or many View_displayed_geometry objects.

4.3.85 View_displayed_geometry to Curve_appearance

Each View_displayed_geometry is governed by exactly one Curve_appearance. Each Curve_appearance governs the display of zero, one, or many View_displayed_geometry objects.

4.3.86 View_displayed_geometry to Visibility

Each View_displayed_geometry has display governed by exactly one Visibility. Each Visibility governs the display of zero or one View_displayed_geometry.

4.3.87 View_displayed_model_annotation to Appearance

NOTE – The appearance of an element for a specific visual presentation is governed by the last set of appearance characteristics assigned to it. Appearance characteristics assigned after the element has been defined with a set of appearance characteristics will override those which are part of the definition. Each new set of characteristics which are assigned will override those previously assigned. However, when an element is instanced or viewed through a viewing pipeline, the appearance characteristics assigned for that instance or view only apply to that instance or view and will not alter the original definition of that element.

Each View_displayed_model_annotation has as overriding appearance zero, one or many Apperance objects. Each Appearance applies to zero, one, or many View_displayed_model_annotation objects.

4.3.88 View_displayed_model_annotation to Model_placed_annotation

Each View_displayed_model_annotation displays exactly one Model_placed_annotation. Each Model_placed_annotation is displayed as zero, one, or many View_displayed_model_annotation objects.

4.3.89 View_placed_annotation to Layer

Each View_placed_annotation is contained in one or more Layer objects. Each Layer contains zero, one, or many View_placed_annotation objects.

4.3.90 View_placed_annotation to Visibility

Each View_placed_annotation has display governed by exactly one Visibility. Each Visibility governs the display of zero or one View_placed_annotation.

5 Application interpreted model

5.1 Mapping table

This clause contains the mapping table that shows how each UoF and application object of this part of ISO 10303 (see clause 4) maps to one or several resource constructs (see 5.2).

The mapping table is organized in five columns. The contents of these five columns are:

- Column 1) Application element: Name of an application element as it appears in the application object definition in 4.2. Application object names are written in uppercase. Attribute names and assertions are listed after the application object to which they belong and are written in lower case.

- Column 2) AIM element: Name of an AIM element as it appears in the AIM (see 5.2). AIM entities are written in lower case. Attribute names of AIM entities are referred to as <entity name> . <attribute name>. The mapping of an application element may result in several related AIM elements. Each of these AIM elements requires a line of its own in the table.

- Column 3) Source: For those AIM elements that are interpreted from the integrated resources, this is the number of the corresponding part of ISO 10303. For those AIM elements that are created for the purpose of this part of ISO 10303, this is the number of the part being written.

- Column 4) Rules: One or more numbers may be given that refer to rules that apply to the current AIM element. For rules that are derived from relationships between application objects, the same rule is referred to by the mapping entries of all the involved AIM elements. Global rules are given for each application object or AIM element to which they apply. Immediately following the table, the rules are listed by number and by expanded name.

- Column 5) Reference path: To describe fully the mapping of an ARM element it may

be necessary to specify a reference path through several related AIM elements. A single AIM element is documented on a single row within the reference path column with a symbol that defines its relationship to the AIM element on the succeeding row in the column. The reference path column, therefore, documents the role of an AIM element relative to the AIM element in the row suceeding it. Two or more such related AIM elements define the interpretation of the integrated resources that satisfies the requirement specified by the ARM element if a reference path is provided.

For each AIM element that has been created for use within this part of ISO 10303, a reference path up to its supertype from an integrated resource is specified. In the case of a bi-directional reference from an AIM element with two attributes, each of which spawns a reference path, each reference path is enclosed by a set of parentheses. The AIM element that is the root of both reference paths is documented either between the sets of parentheses or preceeding each set of parentheses.

For the expression of reference paths the following notational conventions apply:

a) [] : multiple AIM elements or sections of the reference path are required to satisfy an information requirement;

b) () : multiple AIM elements or sections of the reference path are identified as alternatives within the mapping to satisfy an information requirement;

c) : enclosed section constrains the reference path to satisfy an information requirement;

d) $->$: attribute references the entity or select type given in the following row;

e) $<-$: entity or select type is referenced by the attribute in the following row;

f) $[i]$: attribute is an aggregation of which a single member is given in the following row;

g) $[n]$: attribute is an aggregation of which member n is given in the following row;

h) $=>$: entity is a supertype of the entity given in the following row;

i) $<=$: entity is a subtype of the entity given in the following row;

j) $=$: the select or enumeration type is constrained to a choice or value.

Table 1 - Mapping table for 2D_draughting_shape_model UoF

Application element	AIM element	Source	Rules	Reference path
2D_CARTESIAN_COORDINATE_SPACE	[geometric_representation_context] [global_unit_assigned_context]	42 41	4, 13, 23	
2D_DRAUGHTING_SHAPE_MODEL	draughting_model	201		draughting_model <= representation
model_id	representation.name	43		draughting_model <= representation representation.name
2D_draughting_shape_model to 2D_geometric_element	PATH		38	draughting_model <= representation representation.items[i] ->> representation_item => mapped_item mapped_item.mapping_source ->> representation_map representation_map.mapped_representation ->> representation {=> shape_representation => geometrically_bounded_2d_shape_representation} representation.items[i] ->> representation_item => geometric_representation_item => geometric_set {=> geometric_curve_set} geometric_set.elements[i] ->> geometric_set_select
2D_draughting_shape_model to cartesian_coordinate_space	PATH			draughting_model <= representation representation.context_of_items ->> representation_context => [geometric_representation_context] [global_unit_assigned_context]

Table 1 - Mapping table for 2D_draughting_shape_model UoF (continued)

Application element	AIM element	Source	Rules	Reference path
2D_draughting_shape_model to model_placed_annotation	PATH			draughting_model <= representation representation.items[i] -> representation_item => styled_item => annotation_occurrence => draughting_annotation_occurrence
2D_draughting_shape_model to sub_model	PATH			draughting_model <= representation representation.items[i] -> representation_item => mapped_item
2D_GEOMETRIC_ELEMENT	geometric_set_select	42		
2D_geometric_element to layer	PATH		12	geometric_set_select (geometric_set_select = curve curve <=) (geometric_set_select = point point <=) geometric_representation_item <= representation_item <- styled_item.item styled_item <= representation_item layered_item = representation_item layered_item <- presentation_layer_assignment.assigned_items[i] presentation_layer_assignment
CARTESIAN_COORDINATE_SPACE	[geometric_representation_context] [global_unit_assigned_context]	42 41	4, 13, 23	
length_measure_unit	length_unit	41	37	global_unit_assigned_context global_unit_assigned_context.units[i] -> unit unit = named_unit named_unit => length_unit

Table 1 - Mapping table for 2D_draughting_shape_model UoF (continued)

Application element	AIM element	Source	Rules	Reference path
plane_angle_measure_unit	plane_angle_unit	41	37	global_unit_assigned_context global_unit_assigned_context.units[i] -> unit unit = named_unit named_unit => plane_angle_unit
MODEL_PLACED_ANNOTATION #1: If the annotation is a curve, fill area, symbol, subfigure, or text #2: If the annotation is a dimension or a draughting callout	#1 (draughting_annotation_occurrence) #2 (draughting_elements)	201 201	26, 28	#1 (draughting_annotation_occurrence <= annotation_occurrence) #2 (draughting_elements <= draughting_callout)
model_placed_annotation to layer #1: If the annotation is a curve, fill area, symbol, subfigure, or text #2: If the annotation is a dimension or a draughting callout	PATH		5	#1 (draughting_annotation_occurrence <= annotation_occurrence <= styled_item <=) #2 (draughting_elements <= draughting_callout <= geometric_representation_item <=) representation_item layered_item = representation_item layered_item <- presentation_layer_assignment.assigned_items[i] presentation_layer_assignment

Table 1 - Mapping table for 2D_draughting_shape_model UoF (continued)

Application element	AIM element	Source	Rules	Reference path
model_placed_annotation to visibility #1a: If the annotation is a curve, fill area, symbol, subfigure, or text #1b: If the annotation is a dimension or a draughting callout #2a: If the annotation is projected into one or more views and is not to be displayed in any view #2b: If the annotation is projected into a view and is not to be displayed in that view	PATH			#1a (draughting_annotation_occurrence <= annotation_occurrence <=) #1b (draughting_elements <= draughting_callout draughting_callout.contents[i] ->) draughting_callout_element (draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence <=) (draughting_callout_element = annotation_symbol_occurrence annotation_symbol_occurrence <=) (draughting_callout_element = annotation_text_occurrence annotation_text_occurrence <=) annotation_occurrence {=> draughting_annotation_occurrence} <=) styled_item invisible_item = styled_item invisible_item <- invisibility.invisible_items[i] #2a (invisibility) #2b (invisibility => context_dependent_invisibility context_dependent_invisibility.presentation_context ->) presentation_representation => presentation_view)
SUB_MODEL	mapped_item	43	6, 38	
transformation #1: The source location of the transformation #2: The target location of the transformation	#1 [representation_map.mapping_origin] #2 [mapped_item.mapping_target]	43 43		mapped_item #1 [mapped_item.mapping_source -> representation_map representation_map.mapping_origin ->] #2 [mapped_item.mapping_target ->] [representation_item => geometric_representation_item => placement => axis2_placement_2d]

Table 1 - Mapping table for 2D_draughting_shape_model UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
sub_model to sub_model_definition	PATH			mapped_item mapped_item.mapping_source ->representation_map representation_map.mapped_representation ->representation =>draughting_model
SUB_MODEL_DEFINITION	draughting_model	201		draughting_model <=representation

Table 2 - Mapping table for 2D_elements_of_appearance UoF

Application element	AIM element	Source	Rules	Reference path
APPEARANCE	presentation_style_select	46		
COLOUR	colour	46		
CURVE_APPEARANCE	curve_style	46		
draughting_role	curve_style.name	46		
width	curve_style.curve_width	46		
curve_appearance to colour	PATH			curve_style curve_style.curve_colour -> colour
curve_appearance to line_font	PATH			curve_style curve_style.curve_font -> curve_font_or_scaled_curve_font_select curve_font_or_scaled_curve_font_select = curve_style_font_select curve_style_font_select
EXTERNALLY_ DEFINED_HATCHING	externally_defined_hatch_style	46		
hatching_name	externally_defined_item.item_id	41		externally_defined_hatch_style <= externally_defined_item externally_defined_item.item_id
hatching_reference	external_source.source_id	41		externally_defined_hatch_style <= externally_defined_item externally_defined_item.source -> external_source external_source.source_id
EXTERNALLY_ DEFINED_LINE_FONT	externally_defined_curve_font	46	3	
font_name	externally_defined_item.item_id	41		externally_defined_curve_font <= externally_defined_item externally_defined_item.item_id
font_reference	external_source.source_id	41		externally_defined_curve_font <= externally_defined_item externally_defined_item.source -> external_source external_source.source_id

Table 2 - Mapping table for 2D_elements_of_appearance UoF (continued)

Application element	AIM element	Source	Rules	Reference path
EXTERNALLY_ DEFINED_TEXT_FONT	externally_defined_text_font	46	36	
font_name	externally_defined_item.item_id	41		externally_defined_text_font <= externally_defined_item externally_defined_item.item_id
font_reference	external_source.source_id	41		externally_defined_text_font <= externally_defined_item externally_defined_item.source -> external_source external_source.source_id
EXTERNALLY_ DEFINED_TILE	annotation_symbol_occurrence	46		{annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => geometric_representation_item => defined_symbol defined_symbol.definition -> defined_symbol_select defined_symbol_select = externally_defined_symbol}
tile_name	externally_defined_item.item_id	41		annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => geometric_representation_item => defined_symbol defined_symbol.definition -> defined_symbol_select defined_symbol_select = externally_defined_symbol externally_defined_symbol <= externally_defined_item externally_defined_item.item_id

Table 2 - Mapping table for 2D_elements_of_appearance UoF (continued)

Application element	AIM element	Source	Rules	Reference path
tile_reference	external_source.source_id	41		annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => geometric_representation_item => defined_symbol defined_symbol.definition -> defined_symbol_select defined_symbol_select = externally_defined_symbol externally_defined_symbol <= externally_defined_item externally_defined_item.source -> external_source external_source.source_id
EXTERNALLY_ DEFINED_TILING	externally_defined_tile_style	46		
tiling_name	externally_defined_item.item_id	41		externally_defined_tile_style <= externally_defined_item externally_defined_item.item_id
tiling_reference	external_source.source_id	41		externally_defined_tile_style <= externally_defined_item externally_defined_item.source -> external_source external_source.source_id
FILL_AREA_ APPEARANCE	fill_style_select	46		
draughting_role	fill_area_style.name	46		fill_style_select <- fill_area_style.fill_styles[1] fill_area_style fill_area_style.name
HATCHING_PATTERN	fill_area_style_hatching	46		
angle	fill_area_style_hatching. hatch_line_angle	46		

Table 2 - Mapping table for 2D_elements_of_appearance UoF (continued)

Application element	AIM element	Source	Rules	Reference path
displacement	one_direction_repeat_factor. repeat_factor	46		fill_area_style_hatching fill_area_style_hatching.start_of_next_hatch_line -> one_direction_repeat_factor one_direction_repeat_factor.repeat_factor
hatching_pattern to curve_appearance	PATH			fill_area_style_hatching fill_area_style_hatching.hatch_line_appearance -> curve_style
LINE_FONT	curve_style_font_select	46		
PREDEFINED_COLOUR	draughting_pre_defined_colour	201	30	draughting_pre_defined_colour <= pre_defined_colour
PREDEFINED_LINE_ FONT	draughting_pre_defined_curve_font	201	3, 31	draughting_pre_defined_curve_font <= pre_defined_curve_font
PREDEFINED_TEXT_ FONT	draughting_pre_defined_text_font	201	33, 36	draughting_pre_defined_text_font <= pre_defined_text_font
SOLID_FILL_AREA	fill_area_style_colour	46		
solid_fill_area to colour	PATH			fill_area_style_colour fill_area_style_colour.fill_colour -> colour
TEXT_APPEARANCE #1: If text is not to be mirrored about an axis #2: If text is to be mirrored about an axis	#1 (text_style_with_box_characteristics) #2 ([text_style_with_mirror] [text_style_with_box_characteristics])	46 46 46		
character_aspect_ratio	[box_height] [box_width]	46 46		text_style_with_box_characteristics [text_style_with_box_characteristics.characteristics[i] -> box_characteristic_select box_characteristic_select = box_height box_height] [text_style_with_box_characteristics.characteristics[j] -> box_characteristic_select box_characteristic_select = box_width box_width]

Table 2 - Mapping table for 2D_elements_of_appearance UoF (continued)

Application element	AIM element	Source	Rules	Reference path
character_rotation_angle	box_rotate_angle	46		text_style_with_box_characteristics text_style_with_box_characteristics.characteristics[i] ->box_characteristic_select box_characteristic_select = box_rotate_angle box_rotate_angle
character_scale	[box_height] [box_width]	46 46		text_style_with_box_characteristics [text_style_with_box_characteristics.characteristics[i] ->box_characteristic_select box_characteristic_select = box_height box_height] [text_style_with_box_characteristics.characteristics[j] ->box_characteristic_select box_characteristic_select = box_width box_width]
character_slant_angle	box_slant_angle	46		text_style_with_box_characteristics text_style_with_box_characteristics.characteristics[i] ->box_characteristic_select box_characteristic_select = box_slant_angle box_slant_angle
text_appearance to colour	PATH			text_style_with_box_characteristics <=text_style text_style.character_appearance ->character_style_select character_style_select = text_style_for_defined_font text_style_for_defined_font text_style_for_defined_font.text_colour ->colour

Table 2 - Mapping table for 2D_elements_of_appearance UoF (continued)

Application element	AIM element	Source	Rules	Reference path
text_appearance to text_font	PATH			text_style_with_box_characteristics <= text_style presentation_style_select = text_style presentation_style_select <- presentation_style_assignment.styles[i] presentation_style_assignment <- styled_item.styles[i] styled_item {=> annotation_occurrence => annotation_text_occurrence} styled_item.item -> representation_item => geometric_representation_item => (composite_text composite_text.collected_text[i] -> text_or_character text_or_character = text_literal) text_literal text_literal.font -> font_select
TEXT_FONT	font_select	46		
TILE	annotation_symbol_occurrence	46		{annotation_symbol_occurrence <- fill_area_style_tile_symbol_with_style.symbol}
tile to colour	PATH			annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.styles[i] -> presentation_style_assignment presentation_style_assignment.styles[i] -> presentation_style_select presentation_style_select = symbol_style symbol_style symbol_style.style_of_symbol -> symbol_style_select symbol_style_select = symbol_colour symbol_colour symbol_colour.colour_of_symbol -> colour

Table 2 - Mapping table for 2D_elements_of_appearance UoF (continued)

Application element	AIM element	Source	Rules	Reference path
USER_DEFINED_ COLOUR	colour_rgb	46		
blue_proportion	colour_rgb.blue	46		
green_proportion	colour_rgb.green	46		
name	colour_specification.name	46		colour_rgb <= colour_specification colour_specification.name
red_proportion	colour_rgb.red	46		
USER_DEFINED_ HATCHING	fill_area_style	46		fill_area_style {fill_area_style.fill_styles[i] -> fill_style_select fill_style_select = fill_area_style_hatching}
user_defined_hatching to hatching_pattern	PATH			fill_area_style fill_area_style.fill_styles[i] -> fill_style_select fill_style_select = fill_area_style_hatching fill_area_style_hatching
USER_DEFINED_LINE_ FONT	curve_style_font	46	3	
name	curve_style_font.name	46		
pattern	curve_style_font.pattern_list	46		
USER_DEFINED_TILE	annotation_symbol_occurrence	46		{annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => mapped_item {=> annotation_symbol} mapped_item.mapping_source -> representation_map => symbol_representation_map}

Table 2 - Mapping table for 2D_elements_of_appearance UoF (continued)

Application element	AIM element	Source	Rules	Reference path
user_defined_tile to user_defined_symbol_ definition	PATH			annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => mapped_item {=> annotation_symbol} mapped_item.mapping_source -> representation_map => symbol_representation_map
USER_DEFINED_ TILING	fill_area_style_tiles	46		

Table 2 - Mapping table for 2D_elements_of_appearance UoF (continued)

Application element	AIM element	Source	Rules	Reference path
angle #1: If the tile used is an externally_defined_tile #2: If the tile used is a user_defined_tile	axis2_placement_2d.ref_direction	42		fill_area_style_tiles fill_area_style_tiles.tiles[i] ->fill_area_style_tile_shape_select fill_area_style_tile_shape_select = fill_area_style_tile_symbol_with_style fill_area_style_tile_symbol_with_style.symbol ->annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <=styled_item styled_item.item ->representation_item => #1 (geometric_representation_item =>defined_symbol {defined_symbol.definition ->defined_symbol_select defined_symbol_select = externally_defined_symbol}defined_symbol.target ->) #2 (mapped_item {=> annotation_symbol}mapped_item.mapping_target ->representation_item => geometric_representation_item =>)symbol_target symbol_target.placement ->axis2_placement axis2_placement = axis2_placement_2daxis2_placement_2d axis2_placement_2d.ref_direction
repeat_vector_1	one_direction_repeat_factor.repeat_factor	46		fill_area_style_tiles fill_area_style_tiles.tiling_pattern ->two_direction_repeat_factor <=one_direction_repeat_factor one_direction_repeat_factor.repeat_factor
repeat_vector_2	two_direction_repeat_factor.second_repeat_factor	46		fill_area_style_tiles fill_area_style_tiles.tiling_pattern ->two_direction_repeat_factor two_direction_repeat_factor.second_repeat_factor
scale	fill_area_style_tiles.tiling_scale	46		

Table 2 - Mapping table for 2D_elements_of_appearance UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
user_defined_tiling to tile	PATH		11	fill_area_style_tiles fill_area_style_tiles.tiles[i] ->fill_area_style_tile_shape_select fill_area_style_tile_shape_select = fill_area_style_tile_symbol_with_style fill_area_style_tile_symbol_with_style.symbol ->annotation_symbol_occurrence
VISIBILITY	invisibility NOTE: Mapping valid only when associated elements are not to be presented.	46		

Table 3 - Mapping table for 2D_model_viewing UoF

Application element	AIM element	Source	Rules	Reference path
2D_DRAWING_VIEW_ DEFINITION	camera_model_d2	46		
clipping	camera_model_d2.view_window	46		
scale	camera_image_2d_with_scale.scale	201		camera_model_d2 <= camera_model <= geometric_representation_item <= representation_item <- representation_map.mapping_origin representation_map {=> camera_usage} <- mapped_item.mapping_source mapped_item => camera_image => camera_image_2d_with_scale camera_image_2d_with_scale.scale
translation #1: Source of translation #2: Target of translation	#1 [camera_model_d2.view_window] #2 [mapped_item.mapping_target]	46 46		#2 [camera_model_d2 <= camera_model <= geometric_representation_item <= representation_item <- representation_map.mapping_origin representation_map {=> camera_usage} <- mapped_item.mapping_source mapped_item {=> camera_image => camera_image_2d_with_scale} mapped_item.mapping_target]

Table 3 - Mapping table for 2D_model_viewing UoF (continued)

Application element	AIM element	Source	Rules	Reference path
2D_drawing_view_ definition to view_ displayed_geometry	PATH		2	camera_model_d2 <= camera_model <= geometric_representation_item <= representation_item <- representation_map.mapping_origin representation_map {=> camera_usage} representation_map.mapped_representation -> representation {=> draughting_model} representation.items[i] -> representation_item => styled_item {styled_item.item -> representation_item => geometric_representation_item => (point) (curve)}
2D_drawing_view_ definition to view_ displayed_model_ annotation	PATH		2	camera_model_d2 <= camera_model <= geometric_representation_item <= representation_item <- representation_map.mapping_origin representation_map {=> camera_usage} representation_map.mapped_representation -> representation {=> draughting_model} representation.items[i] -> representation_item => styled_item {=> annotation_occurrence => draughting_annotation_occurrence}
VIEW_DISPLAYED_ GEOMETRY	styled_item	46		{styled_item styled_item.item -> representation_item => geometric_representation_item => (point) (curve)}

Table 3 - Mapping table for 2D_model_viewing UoF (continued)

Application element	AIM element	Source	Rules	Reference path
view_displayed_geometry to 2D_geometric_element	PATH			styled_item styled_item.item ->> representation_item => geometric_representation_item => (point geometric_set_select = point) (curve geometric_set_select = curve) geometric_set_select
view_displayed_geometry to curve_appearance #1: If the element is presented in one or more views with the same appearance characteristics #2: If the element is presented in more than one view with different appearance characteristics	PATH		19, 25	styled_item styled_item.styles[i] ->> #1 (presentation_style_assignment presentation_style_assignment.styles[i] ->> presentation_style_select presentation_style_select = curve_style curve_style) #2 ([presentation_style_assignment => presentation_style_by_context presentation_style_by_context.style_context ->> style_context_select style_context_select = representation representation => presentation_representation => presentation_view} [presentation_style_assignment presentation_style_assignment.styles[i] ->> presentation_style_select presentation_style_select = curve_style curve_style])
view_displayed_geometry to visibility	PATH			styled_item invisible_item = styled_item invisible_item <- invisibility.invisible_items[i] invisibility
VIEW_DISPLAYED_MODEL_ANNOTATION	styled_item	46		{styled_item => annotation_occurrence => draughting_annotation_occurrence}

Table 3 - Mapping table for 2D_model_viewing UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
view_display_model_ annotation to appearance #1: If the annotation is presented in one or more views with the same appearance characteristics #2: If the annotation is presented in more than one view with different appearance characteristics	PATH		19	styled_item styled_item.styles[i] ->) #1 (presentation_style_assignment presentation_style_assignment.styles[i] ->) presentation_style_select) #2 ([presentation_style_assignment => presentation_style_by_context presentation_style_by_context.style_context ->) style_context_select style_context_select = representation representation => presentation_representation => presentation_view] [presentation_style_assignment presentation_style_assignment.styles[i] ->) presentation_style_select])
view_displayed_model_ annotation to model_placed_annotation #1: If the annotation is a curve, fill area, symbol, subfigure, or text #2: If the annotation is a dimension or a draughting callout	PATH			#1 (styled_item => annotation_occurrence => draughting_annotation_occurrence) #2 (styled_item => annotation_occurrence) => draughting_annotation_occurrence) => (annotation_curve_occurrence draughting_callout_element = annotation_curve_occurrence) (annotation_symbol_occurrence draughting_callout_element = annotation_symbol_occurrence) (annotation_text_occurrence draughting_callout_element = annotation_text_occurrence) draughting_callout_element <- draughting_callout.contents[1] draughting_callout <= draughting_elements)

Table 4 - Mapping table for drawing_structure_and_administration UoF

Application element	AIM element	Source	Rules	Reference path
2D_CARTESIAN_ COORDINATE_SPACE	[geometric_representation_context] [global_unit_assigned_context]	42 41	4, 13, 23	
APPROVAL	approval	41	1	
date	calendar_date	41		approval <- approval_date_time.dated_approval approval_date_time approval_date_time.date_time ->) date_time_select date_time_select = date date => calendar_date
description	approval.level	41		
approval to organization #1: If approved only by a person #2: If approved only by an organization #3: If approved by a person within an organization	PATH			approval <- approval_person_organization.authorized_approval approval_person_organization approval_person_organization.person_organization ->) person_organization_select #1 (person_organization_select = person person) #2 (person_organization_select = organization organization) #3 (person_organization_select = person_and_organization person_and_organization)
CARTESIAN_ COORDINATE_SPACE	[geometric_representation_context] [global_unit_assigned_context]	42 41	4, 13, 23	
length_measure_unit	length_unit	41	37	global_unit_assigned_context global_unit_assigned_context.units[i] ->) unit unit = named_unit named_unit => length_unit

Table 4 - Mapping table for drawing_structure_and_administration UoF (continued)

Application element	AIM element	Source	Rules	Reference path
plane_angle_measure_unit	plane_angle_unit	41	37	global_unit_assigned_context global_unit_assigned_context.units[i] -> unit unit = named_unit named_unit => plane_angle_unit
DRAWING	draughting_drawing_revision	201	29	draughting_drawing_revision <= drawing_revision
contract_reference	contract.name	41		draughting_drawing_revision <= drawing_revision contracted_item = drawing_revision contracted_item <- draughting_contract_assignment.items[i] draughting_contract_assignment <= contract_assignment contract_assignment.assigned_contract -> contract contract.name
drawing_number	drawing_definition.drawing_number	101		draughting_drawing_revision <= drawing_revision drawing_revision.drawing_identifier -> drawing_definition drawing_definition.drawing_number
drawing_revision_id	drawing_revision.revision_identifier	101		draughting_drawing_revision <= drawing_revision drawing_revision.revision_identifier

Table 4 - Mapping table for drawing_structure_and_administration UoF (continued)

Application element	AIM element	Source	Rules	Reference path
drawing_specification	document.id	41		draughting_drawing_revision <= drawing_revision specified_item = drawing_revision specified_item <- draughting_specification_reference.specified_items[i] draughting_specification_reference <= document_reference document_reference.assigned_document ->) document document.id (document.kind ->) document_type document_type.product_data_type = 'draughting specification')
drawing_type	drawing_definition.drawing_type	101		draughting_drawing_revision <= drawing_revision drawing_revision.drawing_identifier ->) drawing_definition drawing_definition.drawing_type
security_classification	security_classification_level.name	41		draughting_drawing_revision <= drawing_revision classified_item = drawing_revision classified_item <- draughting_security_classification_assignment.assigned_items[i] draughting_security_classification_assignment <= security_classification_assignment security_classification_assignment.assigned_security_classification ->) security_classification security_classification.security_level ->) security_classification_level security_classification_level.name
title	draughting_title.contents	101		draughting_drawing_revision <= drawing_revision draughting_titled_item = drawing_revision draughting_titled_item <- draughting_title.items[i] draughting_title draughting_title.contents

Table 4 - Mapping table for drawing_structure_and_administration UoF (continued)

Application element	AIM element	Source	Rules	Reference path
drawing to approval	PATH			draughting_drawing_revision <= drawing_revision approved_item = drawing_revision approved_item <- draughting_approval_assignment.approved_items[i] draughting_approval_assignment <= approval_assignment approval_assignment.assigned_approval ->) approval
drawing to drawing_sheet	PATH		24	draughting_drawing_revision <= drawing_revision <= presentation_set <- area_in_set.in_set area_in_set {=> drawing_sheet_revision_usage} area_in_set.area ->) presentation_area => drawing_sheet_revision
drawing to organization #1: If only a person is given responsibility #2: If only an organization is given responsibility #3: If a person within an organization is given responsibility	PATH			draughting_drawing_revision <= drawing_revision draughting_organization_item = drawing_revision draughting_organization_item <- #1 (draughting_person_assignment.assigned_items[i] draughting_person_assignment <= person_assignment person_assignment.assigned_person ->) person) #2 (draughting_organization_assignment.assigned_items[i] draughting_organization_assignment <= organization_assignment organization_assignment.assigned_organization ->) organization) #3 (draughting_person_and_organization_assignment.assigned_items[i] draughting_person_and_organization_assignment <= person_and_organization_assignment person_and_organization_assignment.assigned_person_and_organization ->) person_and_organization)

Table 4 - Mapping table for drawing_structure_and_administration UoF (continued)

Application element	AIM element	Source	Rules	Reference path
drawing to product_version	presented_item_representation	46		draughting_drawing_revision <= drawing_revision <= presentation_set presentation_representation_select = presentation_set presentation_representation_select <- presented_item_representation.presentation presented_item_representation presented_item_representation.item -> presented_item => draughting_presented_item draughting_presented_item.items[i] -> draughting_presented_item_select draughting_presented_item_select = product_definition_formation product_definition_formation
DRAWING_SHEET	drawing_sheet_revision	101	24	
security_classification	security_classification_level.name	41		drawing_sheet_revision classified_item = drawing_sheet_revision classified_item <- draughting_security_classification_assignment.assigned_items[i] draughting_security_classification_assignment <= security_classification_assignment security_classification_assignment.assigned_security_classification -> security_classification security_classification.security_level -> security_classification_level security_classification_level.name
sheet_number	drawing_sheet_revision_usage. sheet_number	101		drawing_sheet_revision <= presentation_area <- area_in_set.area area_in_set => drawing_sheet_revision_usage drawing_sheet_revision_usage.sheet_number
sheet_revision_id	drawing_sheet_revision. revision_identifier	101		

Table 4 - Mapping table for drawing_structure_and_administration UoF (continued)

Application element	AIM element	Source	Rules	Reference path
size	presentation_size.size	46	18	drawing_sheet_revision <= presentation_area presentation_size_assignment_select = presentation_area presentation_size_assignment_select <- presentation_size.unit presentation_size presentation_size.size
title	draughting_title.contents	101		drawing_sheet_revision draughting_titled_item = drawing_sheet_revision draughting_titled_item <- draughting_title.items[i] draughting_title draughting_title.contents
drawing_sheet to 2D_cartesian_ coordinate_space	PATH			drawing_sheet_revision <= presentation_area <= presentation_representation <= representation representation.context_of_items ->> representation_context => [geometric_representation_context] [global_unit_assigned_context]
drawing_sheet to approval	PATH			drawing_sheet_revision approved_item = drawing_sheet_revision approved_item <- draughting_approval_assignment.approved_items[i] draughting_approval_assignment <= approval_assignment approval_assignment.assigned_approval ->> approval

Table 4 - Mapping table for drawing_structure_and_administration UoF (continued)

Application element	AIM element	Source	Rules	Reference path
drawing_sheet to drawing_view	PATH		20	drawing_sheet_revision <= presentation_area <= presentation_representation <= representation representation.items[i] ->> representation_item => mapped_item mapped_item.mapping_source ->> representation_map representation_map.mapped_representation ->> representation => presentation_representation => presentation_view
drawing_sheet to organization #1: If only a person is given responsibility #2: If only an organization is given responsibility #3: If a person within an organization is given responsibility	PATH			drawing_sheet_revision draughting_organization_item = drawing_sheet_revision draughting_organization_item < #1 (draughting_person_assignment.assigned_items[i] draughting_person_assignment <= person_assignment person_assignment.assigned_person ->> person) #2 (draughting_organization_assignment.assigned_items[i] draughting_organization_assignment <= organization_assignment organization_assignment.assigned_organization ->> organization) #3 (draughting_person_and_organization_assignment.assigned_items[i] draughting_person_and_organization_assignment <= person_and_organization_assignment person_and_organization_assignment.assigned_person_and_organization ->> person_and_organization)

Table 4 - Mapping table for drawing_structure_and_administration UoF (continued)

Application element	AIM element	Source	Rules	Reference path
drawing_sheet to sheet_placed_annotation	PATH			drawing_sheet_revision <= presentation_area <= presentation_representation <= representation representation.items[i] ->> representation_item => styled_item => annotation_occurrence => draughting_annotation_occurrence
DRAWING_VIEW	presentation_view	46	20	
location	placement.location	42		presentation_view <= presentation_representation <= representation <- representation_map.mapped_representation representation_map <- mapped_item.mapping_source mapped_item mapped_item.mapping_target ->> representation_item => geometric_representation_item => placement {=> axis2_placement_2d} placement.location
name	representation.name	43		presentation_view <= presentation_representation <= representation representation.name

Table 4 - Mapping table for drawing_structure_and_administration UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rotation	axis2_placement_2d.ref_direction	42		presentation_view <= presentation_representation <= representation <- representation_map.mapped_representation representation_map <- mapped_item.mapping_source mapped_item mapped_item.mapping_target ->) representation_item => geometric_representation_item => placement => axis2_placement_2d axis2_placement_2d.ref_direction
drawing_view to 2D_cartesian_ coordinate_space	PATH			presentation_view <= presentation_representation <= representation representation.context_of_items ->) representation_context => [geometric_representation_context] [global_unit_assigned_context]
drawing_view to 2D_drawing_view_ definition	PATH		27	presentation_view <= presentation_representation <= representation representation.items[i] ->) representation_item => mapped_item => {camera_image => camera_image_2d_with_scale} mapped_item.mapping_source ->) representation_map {=> camera_usage} representation_map.mapping_origin ->) representation_item => geometric_representation_item => camera_model => camera_model_d2

Table 4 - Mapping table for drawing_structure_and_administration UoF (continued)

Application element	AIM element	Source	Rules	Reference path
drawing_view to view_placed_annotation	PATH			presentation_view <= presentation_representation <= representation representation.items[i] -> representation_item => styled_item => annotation_occurrence => draughting_annotation_occurrence
ORGANIZATION	(person) (organization) (person_and_organization)	41 41 41		
address #1: If the Organization is only a person #2: If the Organization is only an organization #3: If the Organization is a person within an organization	address	41		#1 (person <- personal_address.people[i] personal_address <= address) #2 (organization <- organizational_address.organizations[i] organizational_address <= address) #3 (person_and_organization [person_and_organization.the_person -> person <- personal_address.people[i] personal_address <= address] [person_and_organization.the_organization -> organization <- organizational_address.organizations[i] organizational_address <= address])

Table 4 - Mapping table for drawing_structure_and_administration UoF (continued)

Application element	AIM element	Source	Rules	Reference path
organization_name #1: If the Organization is only a person #2: If the Organization is only an organization #3: If the Organization is a person within an organization	#1 (person.id) #2 (organization.name) #3 ([person.id] [organization.name])	41 41 41 41		#3 (person_and_organization [person_and_organization.the_person ->person person.id] [person_and_organization.the_organization ->organization organization.name])
SHEET_PLACED_ANNOTATION #1: If the annotation is a curve, fill area, symbol, subfigure, or text #2: If the annotation is a dimension or a draughting callout	#1 (draughting_annotation_occurrence) #2 (draughting_elements)	201 201	26, 28	#1 (draughting_annotation_occurrence <= annotation_occurrence) #2 (draughting_elements <= draughting_callout)
sheet_placed_annotation to layer #1: If the annotation is a curve, fill area, symbol, subfigure, or text #2: If the annotation is a dimension or a draughting callout	PATH		8	#1 (draughting_annotation_occurrence <= annotation_occurrence <= styled_item <=) #2 (draughting_elements <= draughting_callout <= geometric_representation_item <=) representation_item layered_item = representation_item layered_item <- presentation_layer_assignment.assigned_items[1] presentation_layer_assignment

Table 4 - Mapping table for drawing_structure_and_administration UoF (continued)

Application element	AIM element	Source	Rules	Reference path
sheet_placed_annotation to visibility #1: If the annotation is a curve, fill area, symbol, subfigure, or text #2: If the annotation is a dimension or a draughting callout	PATH			#1 (draughting_annotation_occurrence <= annotation_occurrence <=) #2 (draughting_elements <= draughting_callout draughting_callout.contents[1] ->) draughting_callout_element (draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence <= (draughting_callout_element = annotation_symbol_occurrence annotation_symbol_occurrence <=) (draughting_callout_element = annotation_text_occurrence annotation_text_occurrence <=) annotation_occurrence (=> draughting_annotation_occurrence) <=) styled_item invisible_item = styled_item invisible_item <- invisibility.invisible_items[1] invisibility
VIEW_PLACED_ANNOTATION #1: If the annotation is a curve, fill area, symbol, subfigure, or text #2: If the annotation is a dimension or a draughting callout	#1 (draughting_annotation_occurrence) #2 (draughting_elements)	201 201	26, 28	#1 (draughting_annotation_occurrence <= annotation_occurrence) #2 (draughting_elements <= draughting_callout)

Table 4 - Mapping table for drawing_structure_and_administration UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
view_placed_annotation to layer #1: If the annotation is a curve, fill area, symbol, subfigure, or text #2: If the annotation is a dimension or a draughting callout	PATH		10	#1 (draughting_annotation_occurrence <= annotation_occurrence <= styled_item <=) #2 (draughting_elements <= draughting_callout <= geometric_representation_item <=) representation_item layered_item = representation_item layered_item <- presentation_layer_assignment.assigned_items[i] presentation_layer_assignment
view_placed_annotation to visibility #1: If the annotation is a curve, fill area, symbol, subfigure, or text #2: If the annotation is a dimension or a draughting callout	PATH			#1 (draughting_annotation_occurrence <= annotation_occurrence <=) #2 (draughting_elements <= draughting_callout draughting_callout.contents[i] ->) draughting_callout_element (draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence <= (draughting_callout_element = annotation_symbol_occurrence annotation_symbol_occurrence <=) (draughting_callout_element = annotation_text_occurrence annotation_text_occurrence <=) annotation_occurrence {=> draughting_annotation_occurrence} <=) styled_item invisible_item = styled_item invisible_item <- invisibility.invisible_items[i] invisibility

Table 5 - Mapping table for elements_of_annotation UoF

Application element	AIM element	Source	Rules	Reference path
2D_CARTESIAN_ COORDINATE_SPACE	[geometric_representation_context] [global_unit_assigned_context]	42 41	4, 13, 23	
ANNOTATION_CURVE	annotation_curve_occurrence	46		
annotation_curve to curve_appearance	PATH			annotation_curve_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.styles[i] -> presentation_style_assignment presentation_style_assignment.styles[i] -> presentation_style_select presentation_style_select = curve_style curve_style
ANNOTATION_ ELEMENT	draughting_annotation_occurrence	201		draughting_annotation_occurrence <= annotation_occurrence
ANNOTATION_ SUBFIGURE	annotation_subfigure_occurrence	201		annotation_subfigure_occurrence <= annotation_symbol_occurrence
location	placement.location	42		annotation_subfigure_occurrence <= annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => mapped_item {=> annotation_symbol} mapped_item.mapping_target -> representation_item => geometric_representation_item => symbol_target symbol_target.placement -> axis2_placement axis2_placement = axis2_placement_2d axis2_placement_2d <= placement placement.location

Table 5 - Mapping table for elements_of_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rotation	axis2_placement_2d.ref_direction	42		annotation_subfigure_occurrence <= annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item ->> representation_item => mapped_item {=> annotation_symbol} mapped_item.mapping_target ->> representation_item => geometric_representation_item => symbol_target symbol_target.placement ->> axis2_placement axis2_placement = axis2_placement_2d axis2_placement_2d axis2_placement_2d.ref_direction
scale	[symbol_target.x_scale] [symbol_target.y_scale]	46 46		annotation_subfigure_occurrence <= annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item ->> representation_item => mapped_item {=> annotation_symbol} mapped_item.mapping_target ->> representation_item => geometric_representation_item => symbol_target [symbol_target.x_scale] [symbol_target.y_scale]

Table 5 - Mapping table for elements_of_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
annotation_subfigure to annotation_subfigure_ definition	PATH			annotation_subfigure_occurrence <= annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => mapped_item {=> annotation_symbol} mapped_item.mapping_source -> representation_map => symbol_representation_map
ANNOTATION_ SUBFIGURE_ DEFINITION	symbol_representation_map	46		symbol_representation_map {<= representation_map representation_map.mapped_representation -> representation => symbol_representation => draughting_subfigure_representation}
name	representation.name	43		symbol_representation_map <= representation_map representation_map.mapped_representation -> representation {=> symbol_representation => draughting_subfigure_representation} representation.name
annotation_subfigure_ definition to 2D_cartesian_coordinate_ space	PATH			symbol_representation_map <= representation_map representation_map.mapped_representation -> representation {=> symbol_representation => draughting_subfigure_representation} representation.context_of_items -> representation_context => [geometric_representation_context] [global_unit_assigned_context]

Table 5 - Mapping table for elements_of_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path

annotation_subfigure_definition to annotation_subfigure_definition_element
#1: If the element is a curve, fill area, symbol, subfigure, or text
#2: If the element is a dimension or a draughting callout
	PATH			symbol_representation_map <=
				representation_map
				representation_map.mapped_representation ->{
				representation {=>
				symbol_representation => draughting_subfigure_representation}
				representation.items[i] ->{
				representation_item =>
				#1 (styled_item =>
				annotation_occurrence =>
				draughting_annotation_occurrence)
				#2 (geometric_representation_item =>
				draughting_callout =>
				draughting_elements)}

ANNOTATION_SUBFIGURE_DEFINITION_ELEMENT #1: If the element is a curve, fill area, symbol, subfigure, or text #2: If the element is a dimension or a draughting callout	#1 (draughting_annotation_occurrence) #2 (draughting_elements)	201 201	26, 28	#1 (draughting_annotation_occurrence <= annotation_occurrence) #2 (draughting_elements <= draughting_callout)
annotation_subfigure_definition_element to draughting_annotation	IDENTICAL MAPPING			

Table 5 - Mapping table for elements_of_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
annotation_subfigure_definition_element to layer #1: If the element is a curve, fill area, symbol, subfigure, or text #2. If the element is a dimension or a draughting callout	PATH		7	#1 (draughting_annotation_occurrence <= annotation_occurrence <= styled_item <=) #2 (draughting_elements <= draughting_callout <= geometric_representation_item <=) representation_item layered_item = representation_item layered_item <- presentation_layer_assignment.assigned_items[i] presentation_layer_assignment
annotation_subfigure_definition_element to visibility #1: If the element is a curve, fill area, symbol, subfigure, or text #2: If the element is a dimension or a draughting callout	PATH			#1 (draughting_annotation_occurrence <= annotation_occurrence <=) #2 (draughting_elements <= draughting_callout draughting_callout.contents[i] ->> draughting_callout_element (draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence <= (draughting_callout_element = annotation_symbol_occurrence annotation_symbol_occurrence <=) (draughting_callout_element = annotation_text_occurrence annotation_text_occurrence <=) annotation_occurrence {=> draughting_annotation_occurrence} <=) styled_item invisible_item = styled_item invisible_item <- invisibility.invisible_items[i] invisibility
ANNOTATION_SYMBOL	annotation_symbol_occurrence	46		

Table 5 - Mapping table for elements_of_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
location	placement.location	42		annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item ->> representation_item => mapped_item {=> annotation_symbol} mapped_item.mapping_target ->> representation_item => geometric_representation_item => symbol_target symbol_target.placement ->> axis2_placement axis2_placement = axis2_placement_2d axis2_placement_2d <= placement placement.location
rotation	axis2_placement_2d.ref_direction	42		annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item ->> representation_item => mapped_item {=> annotation_symbol} mapped_item.mapping_target ->> representation_item => geometric_representation_item => symbol_target symbol_target.placement ->> axis2_placement axis2_placement = axis2_placement_2d axis2_placement_2d axis2_placement_2d.ref_direction

Table 5 - Mapping table for elements_of_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
scale	[symbol_target.x_scale] [symbol_target.y_scale]	46 46		annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item ->> representation_item => mapped_item {=> annotation_symbol} mapped_item.mapping_target ->> representation_item => geometric_representation_item => symbol_target [symbol_target.x_scale] [symbol_target.y_scale]
annotation_symbol to colour	PATH			annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.styles[i] ->> presentation_style_assignment presentation_style_assignment.styles[i] ->> presentation_style_select presentation_style_select = symbol_style symbol_style symbol_style.style_of_symbol ->> symbol_style_select symbol_style_select = symbol_colour symbol_colour symbol_colour.colour_of_symbol ->> colour
CARTESIAN_ COORDINATE_SPACE	[geometric_representation_context] [global_unit_assigned_context]	42 41	4, 13, 23	
length_measure_unit	length_unit	41	37	global_unit_assigned_context global_unit_assigned_context.units[i] ->> unit unit = named_unit named_unit => length_unit

Table 5 - Mapping table for elements_of_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
plane_angle_measure_unit	plane_angle_unit	41	37	global_unit_assigned_context global_unit_assigned_context.units[i] -> unit unit = named_unit named_unit => plane_angle_unit
EXTERNALLY_ DEFINED_SYMBOL	annotation_symbol_occurrence	46		{annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => geometric_representation_item => defined_symbol defined_symbol.definition -> defined_symbol_select defined_symbol_select = externally_defined_symbol}
symbol_name	externally_defined_item.item_id	41		annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => geometric_representation_item => defined_symbol defined_symbol.definition -> defined_symbol_select defined_symbol_select = externally_defined_symbol externally_defined_symbol <= externally_defined_item externally_defined_item.item_id

Table 5 - Mapping table for elements_of_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
symbol_reference	external_source.source_id	41		annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => geometric_representation_item => defined_symbol defined_symbol.definition -> defined_symbol_select defined_symbol_select = externally_defined_symbol externally_defined_symbol <= externally_defined_item externally_defined_item.source -> external_source external_source.source_id
FILL_AREA	annotation_fill_area_occurrence	46		
reference_point	annotation_fill_area_occurrence. fill_style_target	46		
fill_area to fill_area_ appearance	PATH			annotation_fill_area_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.styles[i] -> presentation_style_assignment presentation_style_assignment.styles[i] -> presentation_style_select presentation_style_select = fill_area_style fill_area_style fill_area_style.fill_styles[i] -> fill_style_select

Table 5 - Mapping table for elements_of_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
fill_area to fill_area_boundary	PATH			annotation_fill_area_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => geometric_representation_item => annotation_fill_area annotation_fill_area.boundaries[i] -> curve
FILL_AREA_ BOUNDARY	curve	42		
fill_area_boundary to annotation_curve	PATH			curve <= geometric_representation_item <= representation_item <- styled_item.item styled_item => annotation_occurrence {=> draughting_annotation_occurrence} => annotation_curve_occurrence
POINT_MARKER_ SYMBOL	annotation_symbol_occurrence	46	32	{annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => geometric_representation_item => defined_symbol defined_symbol.definition -> defined_symbol_select defined_symbol_select = pre_defined_symbol pre_defined_symbol => pre_defined_point_marker_symbol}

Table 5 - Mapping table for elements_of_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
PREDEFINED_SYMBOL	annotation_symbol_occurrence	46	32	(annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item ->) representation_item => geometric_representation_item => defined_symbol defined_symbol.definition ->) defined_symbol_select defined_symbol_select = pre_defined_symbol pre_defined_symbol)
TEXT	annotation_text_occurrence	46		
alignment	text_alignment	46		annotation_text_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item ->) representation_item => geometric_representation_item => (composite_text composite_text.collected_text[i] ->) text_or_character text_or_character = text_literal) text_literal text_literal.alignment ->) text_alignment

Table 5 - Mapping table for elements_of_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
blanking_box #1: If the text contains multiple text_strings #2: If the text contains only one text_string	#1 (composite_text_with_blanking_box.blanking_box) #2 (text_literal_with_blanking_box.blanking_box)	46 46		annotation_text_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item ->representation_item => geometric_representation_item => #1 (composite_text => composite_text_with_blanking_box composite_text_with_blanking_box.blanking_box) #2 (text_literal => text_literal_with_blanking_box text_literal_with_blanking_box.blanking_box)
displayed_box #1: If the text contains multiple text_strings #2: If the text contains only one text_string	#1 (composite_text_with_associated_curves.associated_curves) #2 (text_literal_with_associated_curves.associated_curves)	46 46		annotation_text_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item ->representation_item => geometric_representation_item => #1 (composite_text => composite_text_with_associated_curves composite_text_with_associated_curves.associated_curves) #2 (text_literal => text_literal_with_associated_curves text_literal_with_associated_curves.associated_curves)
mirror_angle	text_style_with_mirror.mirror_placement	46		annotation_text_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.styles[i] ->presentation_style_assignment presentation_style_assignment.styles[i] ->presentation_style_select presentation_style_select = text_style text_style => text_style_with_mirror text_style_with_mirror.mirror_placement

Table 5 - Mapping table for elements_of_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
surrounding_box	composite_text_with_extent.extent	46		annotation_text_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => geometric_representation_item => composite_text => composite_text_with_extent composite_text_with_extent.extent
text to text_appearance #1: If text is not to be mirrored about an axis #2: If text is to be mirrored about an axis	PATH			annotation_text_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.styles[i] -> presentation_style_assignment presentation_style_assignment.styles[i] -> presentation_style_select presentation_style_select = text_style #1 (text_style => text_style_with_box_characteristics) #2 (text_style => [text_style_with_mirror] [text_style_with_box_characteristics])
text to text_string	PATH			annotation_text_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => geometric_representation_item => (composite_text composite_text.collected_text[i] -> text_or_character text_or_character = text_literal) text_literal
TEXT_STRING	text_literal	46		
character_alignment	text_literal.path	46		
character_string	text_literal.literal	46		

Table 5 - Mapping table for elements_of_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
location	placement.location	42		text_literal text_literal.placement ->axis2_placement axis2_placement = axis2_placement_2d axis2_placement_2d <=placement placement.location
overline_underline	text_literal_with_delineation.delineation	46		text_literal => text_literal_with_delineation {=> draughting_text_literal_with_delineation} text_literal_with_delineation.delineation
rotation	axis2_placement_2d.ref_direction	42		text_literal text_literal.placement ->axis2_placement axis2_placement = axis2_placement_2d axis2_placement_2d axis2_placement_2d.ref_direction
surrounding_box	text_literal_with_extent.extent	46		text_literal => text_literal_with_extent text_literal_with_extent.extent

Table 5 - Mapping table for elements_of_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
text_string to text_appearance	PATH			text_literal <= geometric_representation_item <= representation_item <- styled_item.item styled_item {=> over_riding_styled_item => context_dependent_over_riding_styled_item context_dependent_over_riding_styled_item.style_context[1] -> representation_item => geometric_representation_item => composite_text} styled_item.styles[i] -> presentation_style_assignment presentation_style_assignment.styles[i] -> presentation_style_select presentation_style_select = text_style text_style => text_style_with_box_characteristics
USER_DEFINED_ SYMBOL	annotation_symbol_occurrence	46		{annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => mapped_item => {annotation_symbol} mapped_item.mapping_source -> symbol_representation_map}
user_defined_symbol to user_defined_symbol definition	PATH			annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => mapped_item => {annotation_symbol} mapped_item.mapping_source -> symbol_representation_map

Table 5 - Mapping table for elements_of_annotation UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
USER_DEFINED_ SYMBOL_DEFINITION	symbol_representation_map	46		symbol_representation_map {<= representation_map representation_map.mapped_representation -> representation => symbol_representation => draughting_symbol_representation}
symbol_definition_id	representation.name	43		symbol_representation_map <= representation_map representation_map.mapped_representation -> representation {=> symbol_representation => draughting_symbol_representation} representation.name
user_defined_symbol_ definition to 2D_cartesian_coordinate_ space	PATH			symbol_representation_map <= representation_map representation_map.mapped_representation -> representation {=> symbol_representation => draughting_symbol_representation} representation.context_of_items -> representation_context => [geometric_representation_context] [global_unit_assigned_context]
user_defined_symbol_ definition to annotation_element	PATH			symbol_representation_map <= representation_map representation_map.mapped_representation -> representation {=> symbol_representation => draughting_symbol_representation} representation.items[i] -> representation_item => styled_item => annotation_occurrence {=> draughting_annotation_occurrence} => (annotation_curve_occurrence) (annotation_fill_area_occurrence) (annotation_symbol_occurrence) (annotation_text_occurrence)

Table 6 - Mapping table for elements_of_draughting_annotation UoF

Application element	AIM element	Source	Rules	Reference path
ANGULAR_DIMENSION	angular_dimension	201		angular_dimension <= dimension_curve_directed_callout
angular_dimension to dimension_line	PATH			angular_dimension <= dimension_curve_directed_callout <= draughting_callout draughting_callout.contents[i] -> draughting_callout_element draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence => dimension_curve
angular_dimension to projection_line	PATH			angular_dimension <= dimension_curve_directed_callout <= draughting_callout draughting_callout.contents[i] -> draughting_callout_element draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence => projection_curve
CHAINED_ DIMENSION_PAIR	dimension_pair	201		{dimension_pair <= draughting_callout_relationship draughting_callout_relationship.name = 'chained'}
CURVE_DIMENSION	curve_dimension	201		curve_dimension <= dimension_curve_directed_callout
curve_dimension to dimension_line	PATH			curve_dimension <= dimension_curve_directed_callout <= draughting_callout draughting_callout.contents[i] -> draughting_callout_element draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence => dimension_curve

Table 6 - Mapping table for elements_of_draughting_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
curve_dimension to projection_line	PATH			curve_dimension <= dimension_curve_directed_callout <= draughting_callout draughting_callout.contents[i] -> draughting_callout_element draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence => projection_curve
DATUM_FEATURE_ CALLOUT	datum_feature_callout	201		datum_feature_callout <= draughting_callout
DATUM_TARGET_ CALLOUT	datum_target_callout	201		datum_target_callout <= draughting_callout
DIAMETER_ DIMENSION	diameter_dimension	201		diameter_dimension <= dimension_curve_directed_callout
diameter_dimension to dimension_line	PATH			diameter_dimension <= dimension_curve_directed_callout <= draughting_callout draughting_callout.contents[i] -> draughting_callout_element draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence => dimension_curve
diameter_dimension to projection_line	PATH			diameter_dimension <= dimension_curve_directed_callout <= draughting_callout draughting_callout.contents[i] -> draughting_callout_element draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence => projection_curve

Table 6 - Mapping table for elements_of_draughting_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
DIMENSION	(angular_dimension) (curve_dimension) (diameter_dimension) (leader_directed_dimension) (linear_dimension) (ordinate_dimension) (radius_dimension)	201 201 201 201 201 201 201		(angular_dimension <= dimension_curve_directed_callout) (curve_dimension <= dimension_curve_directed_callout) (diameter_dimension <= dimension_curve_directed_callout) (leader_directed_dimension <= leader_directed_callout) (linear_dimension <= dimension_curve_directed_callout) (ordinate_dimension <= projection_directed_callout) (radius_dimension <= dimension_curve_directed_callout)
dimension to dimension_ callout (as primary callout)	dimension_callout_relationship	201		(angular_dimension <= dimension_curve_directed_callout <=) (curve_dimension <= dimension_curve_directed_callout <=) (diameter_dimension <= dimension_curve_directed_callout <=) (leader_directed_dimension <= leader_directed_callout <=) (linear_dimension <= dimension_curve_directed_callout <=) (ordinate_dimension <= projection_directed_callout <=) (radius_dimension <= dimension_curve_directed_callout <=) draughting_callout <- draughting_callout_relationship.relating_draughting_callout draughting_callout_relationship {=> dimension_callout_relationship} {draughting_callout_relationship.name = 'primary'} draughting_callout_relationship.related_draughting_callout -> draughting_callout

Table 6 - Mapping table for elements_of_draughting_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
dimension to dimension_ callout (as secondary callout)	dimension_callout_relationship	201		(angular_dimension <= dimension_curve_directed_callout <=) (curve_dimension <= dimension_curve_directed_callout <=) (diameter_dimension <= dimension_curve_directed_callout <=) (leader_directed_dimension <= leader_directed_callout <=) (linear_dimension <= dimension_curve_directed_callout <=) (ordinate_dimension <= projection_directed_callout <=) (radius_dimension <= dimension_curve_directed_callout <=) draughting_callout <- draughting_callout <- draughting_callout_relationship.relating_draughting_callout draughting_callout_relationship {=> dimension_callout_relationship} {draughting_callout_relationship.name = 'secondary'} draughting_callout_relationship.related_draughting_callout -> draughting_callout
dimension to dimension_ sequence_pair (as predecessor)	PATH			(angular_dimension <= dimension_curve_directed_callout <=) (curve_dimension <= dimension_curve_directed_callout <=) (diameter_dimension <= dimension_curve_directed_callout <=) (linear_dimension <= dimension_curve_directed_callout <=) (ordinate_dimension <= projection_directed_callout <=) (radius_dimension <= dimension_curve_directed_callout <=) draughting_callout <- draughting_callout_relationship.relating_draughting_callout draughting_callout_relationship => dimension_pair

Table 6 - Mapping table for elements_of_draughting_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
dimension to dimension_ sequence_pair (as successor)	PATH			(angular_dimension <= dimension_curve_directed_callout <=) (curve_dimension <= dimension_curve_directed_callout <=) (diameter_dimension <= dimension_curve_directed_callout <=) (linear_dimension <= dimension_curve_directed_callout <=) (ordinate_dimension <= projection_directed_callout <=) (radius_dimension <= dimension_curve_directed_callout <=) draughting_callout <- draughting_callout_relationship.related_draughting_callout draughting_callout_relationship => dimension_pair
DIMENSION_CALLOUT	draughting_callout	101		
DIMENSION_LINE	dimension_curve	101		
dimension_line to annotation_symbol	PATH		35	dimension_curve <= annotation_curve_occurrence <- terminator_symbol.annotated_curve terminator_symbol {=> dimension_curve_terminator} <= annotation_symbol_occurrence
DIMENSION_ SEQUENCE_PAIR	dimension_pair	201		dimension_pair <= draughting_callout_relationship

Table 6 - Mapping table for elements_of_draughting_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
DIMENSION_SYMBOL	annotation_symbol_occurrence	46	32	{annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => geometric_representation_item => defined_symbol defined_symbol.definition -> defined_symbol_select defined_symbol_select = pre_defined_symbol pre_defined_symbol => pre_defined_dimension_symbol}
DIRECTED_CURVE	(dimension_curve) (leader_curve) (projection_curve)	101 101 101		
DRAUGHTING_ ANNOTATION #1: If the annotation is a curve, fill area, symbol, subfigure, or text #2: If the annotation is a dimension or a draughting callout	#1 (draughting_annotation_occurrence) #2 (draughting_elements)	201 201	26, 28	#1 (draughting_annotation_occurrence <= annotation_occurrence) #2 (draughting_elements <= draughting_callout)
DRAUGHTING_ CALLOUT	draughting_callout	101		
draughting_callout to annotation_curve	PATH			draughting_callout draughting_callout.contents[i] -> draughting_callout_element draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence

Table 6 - Mapping table for elements_of_draughting_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
draughting_callout to annotation_symbol	PATH			draughting_callout draughting_callout.contents[1] ->> draughting_callout_element draughting_callout_element = annotation_symbol_occurrence annotation_symbol_occurrence
draughting_callout to directed_curve	PATH			draughting_callout draughting_callout.contents[1] ->> draughting_callout_element draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence => (dimension_curve) (leader_curve) (projection_curve)
draughting_callout to text	PATH			draughting_callout draughting_callout.contents[1] ->> draughting_callout_element draughting_callout_element = annotation_text_occurrence annotation_text_occurrence
DRAWING_SHEET_ LAYOUT	drawing_sheet_layout	201	9	drawing_sheet_layout <= draughting_symbol_representation <= symbol_representation
GEOMETRICAL_ TOLERANCE	geometrical_tolerance_callout	201		geometrical_tolerance_callout <= draughting_callout
GEOMETRICAL_ TOLERANCE_SYMBOL	annotation_symbol_occurrence	46	32	{annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item ->> representation_item => geometric_representation_item => defined_symbol defined_symbol.definition ->> defined_symbol_select defined_symbol_select = pre_defined_symbol pre_defined_symbol => pre_defined_geometrical_tolerance_symbol}

Table 6 - Mapping table for elements_of_draughting_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
LEADER	leader_curve	101		
leader to annotation_symbol	PATH		35	leader_curve <= annotation_curve_occurrence <- terminator_symbol.annotated_curve terminator_symbol [=> leader_terminator] <= annotation_symbol_occurrence
LEADER_DIRECTED_ DIMENSION	leader_directed_dimension	201		leader_directed_dimension <= leader_directed_callout
leader_directed_ dimension to leader	PATH			leader_directed_dimension <= leader_directed_callout <= draughting_callout draughting_callout.contents[i] -> draughting_callout_element draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence => leader_curve
LINEAR_DIMENSION	linear_dimension	201		linear_dimension <= dimension_curve_directed_callout
linear_dimension to dimension_line	PATH			linear_dimension <= dimension_curve_directed_callout <= draughting_callout draughting_callout.contents[i] -> draughting_callout_element draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence => dimension_curve
linear_dimension to projection_line	PATH			linear_dimension <= dimension_curve_directed_callout <= draughting_callout draughting_callout.contents[i] -> draughting_callout_element draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence => projection_curve

Table 6 - Mapping table for elements_of_draughting_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
ORDINATE_ DIMENSION	ordinate_dimension	201		ordinate_dimension <= projection_directed_callout
ordinate_dimension to projection_line	PATH			ordinate_dimension <= projection_directed_callout <= draughting_callout draughting_callout.contents[i] -> draughting_callout_element draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence => projection_curve
PARALLEL_ DIMENSION_PAIR	dimension_pair	201		{dimension_pair <= draughting_callout_relationship draughting_callout_relationship.name = 'parallel'}
PROJECTION_LINE	projection_curve	101		
RADIUS_DIMENSION	radius_dimension	201		radius_dimension <= dimension_curve_directed_callout
radius_dimension to dimension_line	PATH			radius_dimension <= dimension_curve_directed_callout <= draughting_callout draughting_callout.contents[i] -> draughting_callout_element draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence => dimension_curve
radius_dimension to projection_line	PATH			radius_dimension <= dimension_curve_directed_callout <= draughting_callout draughting_callout.contents[i] -> draughting_callout_element draughting_callout_element = annotation_curve_occurrence annotation_curve_occurrence => projection_curve
STRUCTURED_ DIMENSION_CALLOUT	structured_dimension_callout	201		structured_dimension_callout <= draughting_callout

Table 6 - Mapping table for elements_of_draughting_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
structured_dimension_ callout to annotation_ symbol	PATH			structured_dimension_callout <= draughting_callout draughting_callout.contents[i] -> draughting_callout_element draughting_callout_element = annotation_symbol_occurrence annotation_symbol_occurrence
structured_dimension_ callout to text_string (as dimension value)	PATH			structured_dimension_callout <= draughting_callout draughting_callout.contents[i] -> draughting_callout_element draughting_callout_element = annotation_text_occurrence annotation_text_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item {<= representation_item representation_item.name = 'dimension value'} styled_item.item -> representation_item => geometric_representation_item => (composite_text composite_text.collected_text[i] -> text_or_character text_or_character = text_literal) text_literal

Table 6 - Mapping table for elements_of_draughting_annotation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
structured_dimension_callout to text_string (as tolerance value)	PATH			structured_dimension_callout <= draughting_callout draughting_callout.contents[i] -> draughting_callout_element draughting_callout_element = annotation_text_occurrence annotation_text_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item {<= representation_item representation_item.name = 'tolerance value'} styled_item.item ->) representation_item => geometric_representation_item => (composite_text composite_text.collected_text[i] -> text_or_character text_or_character = text_literal) text_literal
structured_dimension_callout to text_string (as unit text)	PATH			structured_dimension_callout <= draughting_callout draughting_callout.contents[i] -> draughting_callout_element draughting_callout_element = annotation_text_occurrence annotation_text_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item {<= representation_item representation_item.name = 'unit text'} styled_item.item -> representation_item => geometric_representation_item => (composite_text composite_text.collected_text[i] -> text_or_character text_or_character = text_literal) text_literal

Table 6 - Mapping table for elements_of_draughting_annotation UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
structured_dimension_ callout to draughting_ callout (as prefix)	dimension_callout_component_ relationship	201		structured_dimension_callout <- draughting_callout_relationship.relating_draughting_callout draughting_callout_relationship {=> dimension_callout_component_relationship} {draughting_callout_relationship.name = 'prefix'} draughting_callout_relationship.related_draughting_callout -> draughting_callout
structured_dimension_ callout to draughting_ callout (as suffix)	dimension_callout_component_ relationship	201		structured_dimension_callout <- draughting_callout_relationship.relating_draughting_callout draughting_callout_relationship {=> dimension_callout_component_relationship} {draughting_callout_relationship.name = 'suffix'} draughting_callout_relationship.related_draughting_callout -> draughting_callout
TERMINATOR_ SYMBOL	annotation_symbol_occurrence	46	32, 35	{annotation_symbol_occurrence <= annotation_occurrence {=> draughting_annotation_occurrence} <= styled_item styled_item.item -> representation_item => geometric_representation_item => defined_symbol defined_symbol.definition -> defined_symbol_select defined_symbol_select = pre_defined_symbol pre_defined_symbol => pre_defined_terminator_symbol}
UNSTRUCTURED_ DIMENSION_CALLOUT	draughting_callout	101		
unstructured_dimension_ callout to draughting_callout	IDENTICAL MAPPING			

Table 7 - Mapping table for grouping UoF

Application element	AIM element	Source	Rules	Reference path
GROUP	group	41	14	
name	group.name	41		
group to group_element #1: If the group_element is a geometric or annotation element #2: If the group_element is a sub_group	PATH			group <- #1 (group_assignment.assigned_group group_assignment => draughting_group_assignment draughting_group_assignment.items[i] ->> draughting_grouped_item) #2 (group_relationship.relating_group group_relationship group_relationship.related_group ->> group)
GROUP_ANNOTATION_ELEMENT #1: If the element is a curve, fill area, symbol, subfigure, or text #2: If the element is a dimension or a draughting callout	#1 (draughting_annotation_occurrence) #2 (draughting_elements)	201 201	26, 28	#1 (draughting_annotation_occurrence <= annotation_occurrence) #2 (draughting_elements <= draughting_callout)
group_annotation_element to draughting_annotation	IDENTICAL MAPPING			
GROUP_ELEMENT #1: If the group_element is a geometric or annotation element #2: If the group_element is a sub_group	#1 (draughting_grouped_item) #2 (group)	201 41		
GROUP_GEOMETRIC_ELEMENT	geometric_set_select	42		
group_geometric_element to 2D_geometric_element	IDENTICAL MAPPING			
LAYER	presentation_layer_assignment	46	15, 16	

Table 7 - Mapping table for grouping UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
layer_id	presentation_layer_assignment.name	46		
name	presentation_layer_assignment. description	46		
layer to visibility	PATH			presentation_layer_assignment invisible_item = presentation_layer_assignment invisible_item <- invisibility.invisible_items[i] invisibility
SUB_GROUP	group	41	14	
sub_group to group	IDENTICAL MAPPING			

Table 8 - Mapping table for product_relation UoF

Application element	AIM element	Source	Rules	Reference path
ORGANIZATION	(person) (organization) (person_and_organization)	41 41 41		
address #1: If the Organization is only a person #2: If the Organization is only an organization #3: If the Organization is a person within an organization	address	41		#1 (person <- personal_address.people[i] personal_address <=) #2 (organization <- organizational_address.organizations[i] organizational_address <=) #3 (person_and_organization [person_and_organization.the_person ->] person <- personal_address.people[i] personal_address <=] [person_and_organization.the_organization ->] organization <- organizational_address.organizations[i] organizational_address <=]) address
organization_name #1: If Organization is only a person #2: If Organization is only an organization #3: If Organization is a person within an organization	#1 (person.id) #2 (organization.name) #3 ([person.id] [(organization.name)])	41 41 41 41		#3 (person_and_organization [person_and_organization.the_person ->] person person.id] [person_and_organization.the_organization ->] organization organization.name))
PRODUCT_VERSION	product_definition_formation	41	17, 21	
product_id	product.id	41		product_definition_formation product_definition_formation.of_product ->] product product.id
revision_id	product_definition_formation.id	41		

Table 8 - Mapping table for product_relation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
product_version to 2D_ draughting_shape_model	PATH		22, 34	product_definition_formation <- product_definition.formation product_definition characterized_product_definition = product_definition characterized_product_definition characterized_definition = characterized_product_definition characterized_definition <- property_definition.definition property_definition {=> product_definition_shape} <- property_definition_representation.definition property_definition_representation {=> shape_definition_representation} property_definition_representation.used_representation ->> representation {=> shape_representation => geometrically_bounded_2d_shape_representation} <- representation_map.mapped_representation representation_map <- mapped_item.mapping_source mapped_item <- representation.items[1] representation => draughting_model

Table 8 - Mapping table for product_relation UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
product_version to organization #1: If the responsibility of the product_version is given only to a person #2: If the responsibility of the product_version is given only to an organization #3: If the responsibility of the product_version is given to a person within an organization	PATH			product_definition_formation draughting_organization_item = product_definition_formation draughting_organization_item <- #1 (draughting_person_assignment.assigned_items[i] draughting_person_assignment <= person_assignment person_assignment.assigned_person -> person) #2 (draughting_organization_assignment.assigned_items[i] draughting_organization_assignment <= organization_assignment organization_assignment.assigned_organization -> organization) #3 (draughting_person_and_organization_assignment.assigned_items[i] draughting_person_and_organization_assignment <= person_and_organization_assignment person_and_organization_assignment.assigned_person_and_organization -> person_and_organization)

The following rules are referenced in the preceding table:

1) approvals_are_assigned
2) camera_usage_mapped_representation
3) curve_font_usage
4) dimensionality_is_two
5) draughting_model_annotation_layers
6) draughting_sub_model_mapped_item
7) draughting_subfigure_representation_layers
8) drawing_sheet_annotation_layers
9) drawing_sheet_layout_usage
10) drawing_view_annotation_layers
11) fill_area_style_tile_symbol_constraint
12) geometric_set_layers
13) global_length_and_angle_units
14) group_usage_constraint
15) presentation_layer_assignment_constraint
16) presentation_layer_assignment_requires_usage
17) presentation_of_product_definition_formation
18) presentation_size_constraint
19) presentation_style_by_context_constraint
20) presentation_view_presented_once
21) product_requires_version
22) product_shape_has_representation
23) representation_context_global_units
24) sheets_belong_to_one_drawing
25) styled_curve
26) subtype_mandatory_annotation_occurrence
27) subtype_mandatory_camera_image
28) subtype_mandatory_draughting_callout
29) subtype_mandatory_drawing_revision
30) subtype_mandatory_pre_defined_colour
31) subtype_mandatory_pre_defined_curve_font
32) subtype_mandatory_pre_defined_symbol
33) subtype_mandatory_pre_defined_text_font
34) subtype_mandatory_shape_representation
35) terminator_symbol_constraint
36) text_font_usage
37) units_prefix_constraint
38) use_of_geometrically_bounded_2d_wireframe_representation

5.2 AIM EXPRESS short listing

This clause specifies the EXPRESS schema that uses elements from the integrated resources and contains the types, entity specializations, rules, and functions that are specific to this part of ISO 10303. This clause also specifies modifications to the textual material for constructs that are imported from the integrated resources. The definitions and EXPRESS provided in the integrated resources for constructs used in the AIM may include select list items and subtypes which are not imported into the AIM. Requirements stated in the integrated resources which refer to such items and subtypes apply exclusively to those items which are imported into the AIM.

EXPRESS specification:

```
*)
SCHEMA explicit_draughting;

USE FROM application_context_schema   -- ISO 10303-41
  (application_context,
   application_protocol_definition);

USE FROM approval_schema   -- ISO 10303-41
  (approval,
   approval_date_time,
   approval_person_organization);

USE FROM date_time_schema   -- ISO 10303-41
  (calendar_date);

USE FROM draughting_element_schema   -- ISO 10303-101
  (dimension_curve,
   dimension_curve_directed_callout,
   dimension_curve_terminator,
   draughting_callout,
   draughting_callout_relationship,
   leader_curve,
   leader_directed_callout,
   leader_terminator,
   projection_curve,
   projection_directed_callout,
   terminator_symbol);

USE FROM drawing_definition_schema   -- ISO 10303-101
  (draughting_title,
   drawing_revision,
   drawing_sheet_revision,
   drawing_sheet_revision_usage);

USE FROM geometric_model_schema   -- ISO 10303-42
  (geometric_curve_set,
   geometric_set,
   geometric_set_select);
```

```
USE FROM geometry_schema   -- ISO 10303-42
  (axis2_placement_2d,
   b_spline_curve_with_knots,
   bezier_curve,
   cartesian_point,
   circle,
   composite_curve,
   direction,
   ellipse,
   geometric_representation_context,
   geometric_representation_item,
   hyperbola,
   line,
   offset_curve_2d,
   parabola,
   point,
   point_on_curve,
   polyline,
   quasi_uniform_curve,
   rational_b_spline_curve,
   trimmed_curve,
   uniform_curve);

USE FROM group_schema   -- ISO 10303-41
  (group,
   group_relationship);

USE FROM management_resources_schema   -- ISO 10303-41
  (approval_assignment,
   contract_assignment,
   document_reference,
   group_assignment,
   organization_assignment,
   person_and_organization_assignment,
   person_assignment,
   security_classification_assignment);

USE FROM measure_schema   -- ISO 10303-41
  (conversion_based_unit,
   global_unit_assigned_context,
   length_measure_with_unit,
   length_unit,
   measure_with_unit,
   named_unit,
   plane_angle_measure_with_unit,
   plane_angle_unit,
   positive_ratio_measure,
   si_unit);

USE FROM person_organization_schema   -- ISO 10303-41
  (organizational_address,
```

```
   personal_address);

USE FROM presentation_appearance_schema -- ISO 10303-46
  (box_height,
   box_rotate_angle,
   box_slant_angle,
   box_width,
   context_dependent_invisibility,
   curve_style,
   curve_style_font,
   externally_defined_curve_font,
   externally_defined_hatch_style,
   externally_defined_tile_style,
   fill_area_style,
   fill_area_style_colour,
   fill_area_style_hatching,
   fill_area_style_tile_symbol_with_style,
   fill_area_style_tiles,
   null_style,
   pre_defined_curve_font,
   presentation_style_by_context,
   styled_item,
   symbol_colour,
   symbol_style,
   text_style,
   text_style_for_defined_font,
   text_style_with_box_characteristics,
   text_style_with_mirror);

USE FROM presentation_definition_schema -- ISO 10303-46
  (annotation_curve_occurrence,
   annotation_fill_area,
   annotation_fill_area_occurrence,
   annotation_occurrence,
   annotation_symbol,
   annotation_symbol_occurrence,
   annotation_text_occurrence,
   composite_text,
   composite_text_with_associated_curves,
   composite_text_with_blanking_box,
   composite_text_with_extent,
   defined_symbol,
   externally_defined_symbol,
   font_select,
   pre_defined_symbol,
   symbol_representation,
   symbol_representation_map,
   text_alignment,
   text_literal,
   text_literal_with_associated_curves,
   text_literal_with_blanking_box,
   text_literal_with_delineation,
```

```
   text_literal_with_extent);

USE FROM presentation_resource_schema -- ISO 10303-46
  (colour_rgb,
   externally_defined_text_font,
   planar_box,
   planar_extent,
   pre_defined_colour,
   pre_defined_text_font);

USE FROM presentation_organization_schema   -- ISO 10303-46
  (camera_image,
   camera_model_d2,
   camera_usage,
   presentation_layer_assignment,
   presentation_layer_usage,
   presentation_size,
   presentation_view,
   presented_item,
   presented_item_representation);

USE FROM product_definition_schema   -- ISO 10303-41
  (product,
   product_definition,
   product_definition_formation);

USE FROM product_property_definition_schema -- ISO 10303-41
  (product_definition_shape);

USE FROM product_property_representation_schema   -- ISO 10303-41
  (shape_definition_representation,
   shape_representation);

USE FROM representation_schema   -- ISO 10303-43
  (mapped_item,
   representation,
   representation_context,
   representation_item);

(*
```

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

application_context_schema	ISO 10303-41
approval_schema	ISO 10303-41
date_time_schema	ISO 10303-41
draughting_element_schema	ISO 10303-101
drawing_definition_schema	ISO 10303-101
geometric_model_schema	ISO 10303-42
geometry_schema	ISO 10303-42
group_schema	ISO 10303-41
management_resources_schema	ISO 10303-41
measure_schema	ISO 10303-41
person_organization_schema	ISO 10303-41
presentation_appearance_schema	ISO 10303-46
presentation_definition_schema	ISO 10303-46
presentation_organization_schema	ISO 10303-46
presentation_resource_schema	ISO 10303-46
product_definition_schema	ISO 10303-41
product_property_representation_schema	ISO 10303-41
representation_schema	ISO 10303-43

5.2.1 explicit_draughting type definitions

5.2.1.1 approved_item

An **approved_item** type specifies those objects to which an **approval** may be assigned.

EXPRESS specification:

```
*)
TYPE approved_item = SELECT
  (drawing_revision,
   drawing_sheet_revision);
END_TYPE;
(*
```

5.2.1.2 classified_item

A **classified_item** type specifies those objects to which a security classification may be assigned.

EXPRESS specification:

```
*)
TYPE classified_item = SELECT
  (drawing_revision,
   drawing_sheet_revision);
END_TYPE;
(*
```

5.2.1.3 contracted_item

A **contracted_item** type specifies those objects to which a contract may be assigned.

EXPRESS specification:

```
*)
TYPE contracted_item = SELECT
  (drawing_revision);
END_TYPE;
(*
```

5.2.1.4 draughting_grouped_item

A **draughting_grouped_item** type specifies those objects that can be part of a **group**.

EXPRESS specification:

```
*)
```

```
TYPE draughting_grouped_item = SELECT
  (annotation_occurrence,
   geometric_set_select);
END_TYPE;
(*
```

5.2.1.5 draughting_organization_item

A **draughting_organization_item** type specifies the objects to which a person, an organization, or a person and an organization may be assigned.

EXPRESS specification:

```
*)
TYPE draughting_organization_item = SELECT
  (product_definition_formation,
   drawing_revision,
   drawing_sheet_revision);
END_TYPE;
(*
```

5.2.1.6 draughting_presented_item_select

A **draughting_presented_item_select** type specifies those objects that may be presented by a drawing.

EXPRESS specification:

```
*)
TYPE draughting_presented_item_select = SELECT
  (product_definition_formation);
END_TYPE;
(*
```

5.2.1.7 specified_item

A **specified_item** type specifies the objects to which a **draughting_specification_reference** may be assigned.

EXPRESS specification:

```
*)
TYPE specified_item = SELECT
  (drawing_revision);
END_TYPE;
(*
```

5.2.2 explicit_draughting entities

5.2.2.1 explicit_draughting entity definitions

5.2.2.1.1 angular_dimension

An **angular_dimension** is a **draughting_callout** directed by a dimension curve that presents a value of angular distance measure.

EXPRESS specification:

```
*)
ENTITY angular_dimension
  SUBTYPE of (dimension_curve_directed_callout);
END_ENTITY;
(*
```

5.2.2.1.2 annotation_subfigure_occurrence

An **annotation_subfigure_occurrence** is a grouping of annotation that is recognized as a subfigure.

EXPRESS specification:

```
*)
ENTITY annotation_subfigure_occurrence
  SUBTYPE OF (annotation_symbol_occurrence);
WHERE
  WR1: SIZEOF (QUERY (sty <* SELF\styled_item.styles |
         NOT (SIZEOF (sty.styles) = 1)
       )) = 0;
  WR2: SIZEOF (QUERY (sty <* SELF\styled_item.styles |
         NOT ('EXPLICIT_DRAUGHTING.NULL_STYLE'
               IN TYPEOF (sty.styles[1]))
       ))=0;
  WR3: ('EXPLICIT_DRAUGHTING.ANNOTATION_SYMBOL'
          IN TYPEOF (SELF\styled_item.item));
  WR4: ('EXPLICIT_DRAUGHTING.DRAUGHTING_SUBFIGURE_REPRESENTATION'
        IN TYPEOF
       (SELF\styled_item.item\mapped_item.mapping_source.
        mapped_representation));
END_ENTITY;
(*
```

Formal propositions:

WR1: The **annotation_subfigure_occurrence** shall have exactly one style.

WR2: The type of style shall be **null_style**.

WR3: The item of the **annotation_subfigure_occurrence** shall be an **annotation_symbol.**

WR4: The source of the **annotation_subfigure_occurrence** shall be a **draughting_subfigure_representation.**

5.2.2.1.3 camera_image_2d_with_scale

A **camera_image_2d_with_scale** is a **camera_image** with a derived scale. The scale is the ratio between the size of the viewport and the size of the view window of the **camera_model.**

EXPRESS specification:

```
*)
ENTITY camera_image_2d_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_d2.
           view_window.size_in_x));
WHERE
  WR1: ('EXPLICIT_DRAUGHTING.CAMERA_MODEL_D2' 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_d2.view_window);
  WR3: SELF\mapped_item.mapping_source.mapping_origin\camera_model_d2.
                     view_window_clipping;
END_ENTITY;
(*
```

Attribute definitions:

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

Formal propositions:

WR1: The source of the projection shall be a **camera_model_d2.**

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

WR3: The clipping attribute of the **camera_image** supertype shall be true.

Informal propositions:

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

5.2.2.1.4 curve_dimension

A **curve_dimension** is a **draughting_callout** directed by a dimension curve that presents a value of distance between two elements, measured along a curved path, or presents the length of a curved element.

EXPRESS specification:

```
*)
ENTITY curve_dimension
  SUBTYPE OF (dimension_curve_directed_callout);
END_ENTITY;
(*
```

5.2.2.1.5 datum_feature_callout

A **datum_feature_callout** is a **draughting_callout** that presents a datum. It shall contain an alphanumeric designation to be used as identification of the datum.

EXPRESS specification:

```
*)
ENTITY datum_feature_callout
  SUBTYPE of (draughting_callout);
END_ENTITY;
(*
```

5.2.2.1.6 datum_target_callout

A **datum_target_callout** is a **draughting_callout** that presents a datum target. It shall contain an alphanumeric designation and, where applicable, a specification of the diametrical size of the target area.

EXPRESS specification:

```
*)
ENTITY datum_target_callout
  SUBTYPE of (draughting_callout);
END_ENTITY;
(*
```

5.2.2.1.7 diameter_dimension

A **diameter_dimension** is a **draughting_callout** directed by a dimension curve that presents a value of the diametrical size of a circular element.

EXPRESS specification:

```
*)
ENTITY diameter_dimension
  SUBTYPE OF (dimension_curve_directed_callout);
END_ENTITY;
(*
```

5.2.2.1.8 dimension_callout_component_relationship

A **dimension_callout_component_relationship** is a **draughting_callout_relationship** that identifies the association between a **structured_dimension_callout** and a **draughting_callout** that participates in its definition either as prefix information or suffix information. A dimension prefix specification is information used in interpreting the dimension or its applicability and is physically located before the dimension value as the dimension is read. A dimension suffix specification is physically located after the dimension value as the dimension is read and either contains information used in interpreting the dimension or its applicability or contains additional information that is used in conjunction with the dimension.

EXPRESS specification:

```
*)
ENTITY dimension_callout_component_relationship
  SUBTYPE OF (draughting_callout_relationship);
WHERE
  WR1: SELF.name IN ['prefix', 'suffix'];
  WR2: 'EXPLICIT_DRAUGHTING.STRUCTURED_DIMENSION_CALLOUT'
          IN TYPEOF (SELF.relating_draughting_callout);
  WR3: SIZEOF (TYPEOF (SELF.related_draughting_callout) *
       ['EXPLICIT_DRAUGHTING.LEADER_DIRECTED_CALLOUT',
        'EXPLICIT_DRAUGHTING.PROJECTION_DIRECTED_CALLOUT',
        'EXPLICIT_DRAUGHTING.DIMENSION_CURVE_DIRECTED_CALLOUT',
        'EXPLICIT_DRAUGHTING.STRUCTURED_DIMENSION_CALLOUT']) = 0;
  WR4: SELF.related_draughting_callout.contents *
       SELF.relating_draughting_callout.contents =
       SELF.related_draughting_callout.contents;
END_ENTITY;
(*
```

Attribute definitions:

SELF\draughting_callout_relationship.relating_draughting_callout: the **structured_dimension_callout** with which another **draughting_callout** is associated.

SELF\draughting_callout_relationship.related_draughting_callout: the **draughting_callout** that participates in the definition of a dimension callout.

Formal propositions:

WR1: The name of the **dimension_callout_component_relationship** shall be either 'prefix' or 'suffix'.

WR2: The **relating_draughting_callout** shall be a **structured_dimension_callout**.

WR3: The **related_draughting_callout** shall not be a **leader_directed_callout**, **projection_directed_callout**, **dimension_curve_directed_callout**, or **structured_dimension_callout**.

WR4: Each element of the **related_draughting_callout** shall also be an element of the **structured_dimension_callout**.

5.2.2.1.9 dimension_callout_relationship

A **dimension_callout_relationship** is a **draughting_callout_relationship** that relates the presentation of a dimension to the presentation of the dimension callout either as the primary callout or as the secondary callout. A primary callout is a **draughting_callout** that presents the dimension value in the primary unit of measure. A secondary callout is a **draughting_callout** that presents the dimension value in another unit of measure.

EXPRESS specification:

```
*)
ENTITY dimension_callout_relationship
  SUBTYPE OF (draughting_callout_relationship);
WHERE
  WR1: SELF.name IN ['primary', 'secondary'];
  WR2: SIZEOF (TYPEOF (SELF.relating_draughting_callout) *
       ['EXPLICIT_DRAUGHTING.ANGULAR_DIMENSION',
        'EXPLICIT_DRAUGHTING.CURVE_DIMENSION',
        'EXPLICIT_DRAUGHTING.DIAMETER_DIMENSION',
        'EXPLICIT_DRAUGHTING.LEADER_DIRECTED_DIMENSION',
        'EXPLICIT_DRAUGHTING.LINEAR_DIMENSION',
        'EXPLICIT_DRAUGHTING.ORDINATE_DIMENSION',
        'EXPLICIT_DRAUGHTING.RADIUS_DIMENSION'])=1;
  WR3: SIZEOF (TYPEOF (SELF.related_draughting_callout) *
       ['EXPLICIT_DRAUGHTING.DIMENSION_CURVE_DIRECTED_CALLOUT',
        'EXPLICIT_DRAUGHTING.PROJECTION_DIRECTED_CALLOUT',
        'EXPLICIT_DRAUGHTING.LEADER_DIRECTED_CALLOUT']) = 0;
  WR4: SELF.related_draughting_callout.contents *
       SELF.relating_draughting_callout.contents =
       SELF.related_draughting_callout.contents;
END_ENTITY;
(*
```

Attribute definitions:

SELF\draughting_callout_relationship.relating_draughting_callout: the dimension with which a dimension callout is associated.

SELF\draughting_callout_relationship.related_draughting_callout: the dimension callout.

Formal propositions:

WR1: The name of the **dimension_callout_relationship** shall be either 'primary' or 'secondary'.

WR2: The dimension shall be an **angular_dimension**, **curve_dimension**, **diameter_dimension**, **leader_directed_dimension**, **linear_dimension**, **ordinate_dimension**, or **radius_dimension**.

WR3: The dimension callout shall not be a **dimension_curve_directed_callout**, **projection_curve_directed_callout**, or **leader_directed_callout**.

WR4: Each element of the dimension callout shall also be an element of the dimension.

5.2.2.1.10 dimension_pair

A **dimension_pair** is a **draughting_callout_relationship** that identifies the relationship between dimensions that participate in a chained or parallel dimension sequence. A chained dimension sequence is a sequence of two or more dimensions in which the terminus of one dimension initializes the next dimension in the sequence. A parallel dimension sequence is a collection of two or more dimensions where all the dimensions are of the same type, their dimension lines are parallel to one another, the dimensions share a common origin or datum, or all three.

EXPRESS specification:

```
*)
ENTITY dimension_pair
  SUBTYPE OF (draughting_callout_relationship);
WHERE
  WR1: SELF.name IN ['chained', 'parallel'];
  WR2: SIZEOF ( TYPEOF (SELF.relating_draughting_callout) *
       ['EXPLICIT_DRAUGHTING.ANGULAR_DIMENSION',
        'EXPLICIT_DRAUGHTING.CURVE_DIMENSION',
        'EXPLICIT_DRAUGHTING.DIAMETER_DIMENSION',
        'EXPLICIT_DRAUGHTING.LINEAR_DIMENSION',
        'EXPLICIT_DRAUGHTING.ORDINATE_DIMENSION',
        'EXPLICIT_DRAUGHTING.RADIUS_DIMENSION'])=1;
  WR3: SIZEOF (TYPEOF (SELF.related_draughting_callout) *
       ['EXPLICIT_DRAUGHTING.ANGULAR_DIMENSION',
        'EXPLICIT_DRAUGHTING.CURVE_DIMENSION',
        'EXPLICIT_DRAUGHTING.DIAMETER_DIMENSION',
        'EXPLICIT_DRAUGHTING.LINEAR_DIMENSION',
        'EXPLICIT_DRAUGHTING.ORDINATE_DIMENSION',
        'EXPLICIT_DRAUGHTING.RADIUS_DIMENSION'])=1;
END_ENTITY;
(*
```

Attribute definitions:

SELF\draughting_callout_relationship.relating_draughting_callout: the predecessor dimension.

SELF\draughting_callout_relationship.related_draughting_callout: the successor dimension.

Formal propositions:

WR1: The name of the **dimension_pair** shall be either 'chained' or 'parallel'.

WR2: The predecessor dimension in the **dimension_pair** shall be an **angular_dimension**, **curve_dimension**, **diameter_dimension**, **linear_dimension**, **ordinate_dimension**, or **radius_dimension**.

WR3: The successor dimension in the **dimension_pair** shall be an **angular_dimension**, **curve_dimension**, **diameter_dimension**, **linear_dimension**, **ordinate_dimension**, or **radius_dimension**.

5.2.2.1.11 draughting_annotation_occurrence

A **draughting_annotation_occurrence** is an **annotation_occurrence** that specifies constraints within the context of draughting.

NOTE – This entity denotes a shared scope with the Application Interpreted Model of ISO 10303-202.

EXPRESS specification:

```
*)
ENTITY draughting_annotation_occurrence
  SUBTYPE OF (annotation_occurrence);
WHERE
  curve_has_curve_style:
     NOT ('EXPLICIT_DRAUGHTING.ANNOTATION_CURVE_OCCURRENCE'
           IN TYPEOF (SELF)) OR
     (SIZEOF (QUERY (sty <* SELF\styled_item.styles |
     NOT ((SIZEOF (sty.styles) = 1)
          AND ('EXPLICIT_DRAUGHTING.CURVE_STYLE'
               IN TYPEOF (sty.styles[1]))
     ))) = 0);
  fill_area_has_fill_style:
     NOT ('EXPLICIT_DRAUGHTING.ANNOTATION_FILL_AREA_OCCURRENCE'
          IN TYPEOF (SELF)) OR
     (SIZEOF (QUERY (sty <* SELF\styled_item.styles |
     NOT ((SIZEOF (sty.styles) = 1)
          AND ('EXPLICIT_DRAUGHTING.FILL_AREA_STYLE'
               IN TYPEOF (sty.styles[1]))
     ))) = 0);
  styled_fill_boundaries:
```

```
   NOT ('EXPLICIT_DRAUGHTING.ANNOTATION_FILL_AREA_OCCURRENCE'
        IN TYPEOF (SELF)) OR
   (SIZEOF (QUERY (bound <*
            SELF\styled_item.item\annotation_fill_area.boundaries |
   NOT (SIZEOF (QUERY (si <* USEDIN (bound, 'EXPLICIT_DRAUGHTING.' +
                                          'STYLED_ITEM.ITEM') |
   ('EXPLICIT_DRAUGHTING.ANNOTATION_CURVE_OCCURRENCE' IN TYPEOF (si)))) > 0)
   )) = 0);
symbol_has_symbol_style:
   NOT ('EXPLICIT_DRAUGHTING.ANNOTATION_SYMBOL_OCCURRENCE'
        IN TYPEOF (SELF)) OR
   (SIZEOF (QUERY (sty <* SELF\styled_item.styles |
   NOT ((SIZEOF (sty.styles) = 1)
        AND (SIZEOF (TYPEOF (sty.styles[1]) *
                ['EXPLICIT_DRAUGHTING.SYMBOL_STYLE',
                 'EXPLICIT_DRAUGHTING.NULL_STYLE']) = 1))
   )) = 0);
allowable_symbol_representations:
   NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_SYMBOL_OCCURRENCE'
        IN TYPEOF (SELF)) AND
       ('EXPLICIT_DRAUGHTING.ANNOTATION_SYMBOL'
         IN TYPEOF (SELF\styled_item.item))) OR
    (SIZEOF (['EXPLICIT_DRAUGHTING.' +
              'DRAUGHTING_SYMBOL_REPRESENTATION',
              'EXPLICIT_DRAUGHTING.' +
              'DRAUGHTING_SUBFIGURE_REPRESENTATION'] *
     TYPEOF (SELF\styled_item.item\mapped_item.mapping_source.
              mapped_representation))=1);
text_has_text_style:
   NOT ('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
        IN TYPEOF (SELF)) OR
   (SIZEOF (QUERY (sty <* SELF\styled_item.styles |
   NOT ((SIZEOF (sty.styles) = 1)
        AND ('EXPLICIT_DRAUGHTING.TEXT_STYLE'
             IN TYPEOF (sty.styles[1]))
   ))) = 0);
text_not_nested:
   NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
         IN TYPEOF (SELF)) AND
        ('EXPLICIT_DRAUGHTING.COMPOSITE_TEXT'
         IN TYPEOF (SELF\styled_item.item))) OR
   SIZEOF (QUERY (text <* SELF\styled_item.item\composite_text.
         collected_text |
   NOT (('EXPLICIT_DRAUGHTING.TEXT_LITERAL' IN TYPEOF (text))
   )))=0;
  allowable_text :
    NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
            IN TYPEOF (SELF)) AND
          (SIZEOF (TYPEOF(SELF\styled_item.item) *
             ['EXPLICIT_DRAUGHTING.COMPOSITE_TEXT',
              'EXPLICIT_DRAUGHTING.TEXT_LITERAL']) = 1));
text_alignment_literals:
```

```
    NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
          IN TYPEOF (SELF)) AND
         ('EXPLICIT_DRAUGHTING.TEXT_LITERAL'
          IN TYPEOF (SELF\styled_item.item))) OR
    (SELF\styled_item.item\text_literal.alignment IN
     ['baseline left', 'baseline centre', 'baseline right']);
text_alignment_composites:
    NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
          IN TYPEOF (SELF)) AND
         ('EXPLICIT_DRAUGHTING.COMPOSITE_TEXT'
          IN TYPEOF (SELF\styled_item.item))) OR
    SIZEOF (QUERY (literal <* QUERY (text <*
                   SELF\styled_item.item\composite_text.collected_text |
                   ('EXPLICIT_DRAUGHTING.TEXT_LITERAL' IN TYPEOF (text))) |
    NOT (literal.alignment IN
        ['baseline left', 'baseline centre', 'baseline right']
    )))=0;
text_literal_not_rotated:
    NOT ((('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
          IN TYPEOF (SELF)) AND
         ('EXPLICIT_DRAUGHTING.TEXT_LITERAL'
          IN TYPEOF (SELF\styled_item.item))) AND
    (EXISTS (SELF\styled_item.item\text_literal.
              placement\axis2_placement_2d.ref_direction)));
literals_in_composites_not_rotated:
    NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
          IN TYPEOF (SELF)) AND
         ('EXPLICIT_DRAUGHTING.COMPOSITE_TEXT'
          IN TYPEOF (SELF\styled_item.item))) OR
    SIZEOF (QUERY (literal <* QUERY (text <*
                   SELF\styled_item.item\composite_text.collected_text |
                   ('EXPLICIT_DRAUGHTING.TEXT_LITERAL' IN TYPEOF (text))) |
    EXISTS (literal.placement\axis2_placement_2d.ref_direction)
    ))=0;
single_text_alignment:
    NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
          IN TYPEOF (SELF)) AND
         ('EXPLICIT_DRAUGHTING.COMPOSITE_TEXT'
          IN TYPEOF (SELF\styled_item.item))) OR
    check_text_alignment (SELF\styled_item.item);
single_text_font:
    NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
          IN TYPEOF (SELF)) AND
         ('EXPLICIT_DRAUGHTING.COMPOSITE_TEXT'
          IN TYPEOF (SELF\styled_item.item))) OR
    check_text_font (SELF\styled_item.item);
allowable_text_literals:
    NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
          IN TYPEOF (SELF)) AND
         ('EXPLICIT_DRAUGHTING.COMPOSITE_TEXT'
          IN TYPEOF (SELF\styled_item.item))) OR
    SIZEOF (QUERY (literal <* QUERY (text <*
```

```
                SELF\styled_item.item\composite_text.collected_text |
                ('EXPLICIT_DRAUGHTING.TEXT_LITERAL' IN TYPEOF (text))) |
   NOT (SIZEOF (TYPEOF (literal) *
               ['EXPLICIT_DRAUGHTING.' +
                'TEXT_LITERAL_WITH_BLANKING_BOX',
                'EXPLICIT_DRAUGHTING.' +
                'TEXT_LITERAL_WITH_ASSOCIATED_CURVES']) = 0
   )))=0;
styled_text_literal_curves:
   NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
         IN TYPEOF (SELF)) AND
        ('EXPLICIT_DRAUGHTING.TEXT_LITERAL_WITH_ASSOCIATED_CURVES'
         IN TYPEOF (SELF\styled_item.item))) OR
   SIZEOF (QUERY (crv <*
   SELF\styled_item.item\text_literal_with_associated_curves.
         associated_curves |
   NOT (SIZEOF (QUERY (si <* USEDIN (crv, 'EXPLICIT_DRAUGHTING.' +
                                          'STYLED_ITEM.ITEM') |
        ('EXPLICIT_DRAUGHTING.ANNOTATION_CURVE_OCCURRENCE' IN TYPEOF (si))
        )) >0
 )))=0;
styled_composite_text_curves:
   NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
         IN TYPEOF (SELF)) AND
        ('EXPLICIT_DRAUGHTING.COMPOSITE_TEXT_WITH_ASSOCIATED_CURVES'
         IN TYPEOF (SELF\styled_item.item))) OR
   SIZEOF (QUERY (crv <*
   SELF\styled_item.item\composite_text_with_associated_curves.
         associated_curves |
   NOT (SIZEOF (QUERY (si <* USEDIN (crv, 'EXPLICIT_DRAUGHTING.' +
                                          'STYLED_ITEM.ITEM') |
        ('EXPLICIT_DRAUGHTING.ANNOTATION_CURVE_OCCURRENCE' IN TYPEOF (si))
        )) >0
   )))=0;
curve_style_has_width:
   SIZEOF (QUERY (cs <* QUERY (sty <* SELF\styled_item.styles |
                       ('EXPLICIT_DRAUGHTING.CURVE_STYLE' IN
                        TYPEOF (sty.styles[1]))) |
   NOT (('EXPLICIT_DRAUGHTING.LENGTH_MEASURE_WITH_UNIT'
         IN TYPEOF (cs.styles[1]\curve_style.curve_width)) AND
        ('EXPLICIT_DRAUGHTING.POSITIVE_LENGTH_MEASURE'
         IN TYPEOF (cs.styles[1]\curve_style.curve_width.value_component))
   ))) = 0;
tiling_constraints:
   SIZEOF (QUERY (fas <* QUERY (sty <* SELF\styled_item.styles |
          ('EXPLICIT_DRAUGHTING.FILL_AREA_STYLE' IN
           TYPEOF (sty.styles[1]))) |
   NOT (
   (SIZEOF (QUERY (fs <* fas.styles[1]\fill_area_style.fill_styles |
           ('EXPLICIT_DRAUGHTING.FILL_AREA_STYLE_TILES'
            IN TYPEOF (fs)))) <= 1) AND
   (SIZEOF (QUERY (fst <* QUERY (fs <*
```

```
                    fas.styles[1]\fill_area_style.fill_styles |
                   ('EXPLICIT_DRAUGHTING.FILL_AREA_STYLE_TILES'
                    IN TYPEOF (fs))) |
     NOT (SIZEOF (fst.tiles) = 1))) = 0)
     ))) = 0;
  hatching_constraints:
     SIZEOF (QUERY (fas <* QUERY (sty <* SELF\styled_item.styles |
            ('EXPLICIT_DRAUGHTING.FILL_AREA_STYLE' IN
             TYPEOF (sty.styles[1]))) |
     NOT (
     SIZEOF (QUERY (fst <* QUERY (fs <*
                    fas.styles[1]\fill_area_style.fill_styles |
                   ('EXPLICIT_DRAUGHTING.FILL_AREA_STYLE_HATCHING'
                    IN TYPEOF (fs))) |
     NOT (fst.point_of_reference_hatch_line :=: fst.pattern_start
     ))) = 0
     ))) = 0;
  text_style_constraint:
     SIZEOF (QUERY (ts <* QUERY (sty <* SELF\styled_item.styles |
                                ('EXPLICIT_DRAUGHTING.TEXT_STYLE'
                                 IN TYPEOF (sty.styles[1]))) |
     NOT ('EXPLICIT_DRAUGHTING.TEXT_STYLE_WITH_BOX_CHARACTERISTICS'
          IN TYPEOF (ts.styles[1])
     ))) = 0;
  text_style_characteristics:
     SIZEOF (QUERY (ts <* QUERY (sty <* SELF\styled_item.styles |
                                ('EXPLICIT_DRAUGHTING.' +
                                 'TEXT_STYLE_WITH_BOX_CHARACTERISTICS'
                                 IN TYPEOF (sty.styles[1]))) |
     NOT (
     SIZEOF
     (ts.styles[1]\text_style_with_box_characteristics.characteristics)
     = 4
     ))) = 0;
END_ENTITY;
(*
```

Formal propositions:

curve_has_curve_style: Each **draughting_annotation_occurrence** that is an **annotation_curve_occurrence** shall have exactly one style that is a **curve_style**.

fill_area_has_fill_style: Each **draughting_annotation_occurrence** that is an **annotation_fill_area_occurrence** shall have exactly one style that is a **fill_area_style**.

styled_fill_boundaries: : If the **draughting_annotation_occurrence** is an **annotation_fill_area_occurrence**, each boundary of the **annotation_fill_area** shall be the item for an **annotation_curve_occurrence**.

symbol_has_symbol_style: Each **draughting_annotation_occurrence** that is an **annotation_symbol_occurrence** shall have exactly one style that is either a **symbol_style** or **null_**

style.

allowable_symbol_representations: If the **draughting_annotation_occurrence** is an **annotation_symbol_occurrence** whose item is an **annotation_symbol**, the **symbol_representation** shall be a **draughting_symbol_representation** or **draughting_subfigure_representation**.

text_has_text_style: Each **draughting_annotation_occurrence** that is an **annotation_text_occurrence** shall have exactly one style that is a **text_style**.

text_not_nested: If the **draughting_annotation_occurrence** is an **annotation_text_occurrence**, each **composite_text** shall only collect **text_literal**.

text_alignment_literals: If the **draughting_annotation_occurrence** is an **annotation_text_occurrence** whose item is a **text_literal**, the **text_literal** shall have the value of 'baseline left', 'baseline centre', or 'baseline right' for the alignment.

text_alignment_composites: If the **draughting_annotation_occurrence** is an **annotation_text_occurrence** whose item is a **composite_text**, each **text_literal** included in the **composite_text** shall have the value of 'baseline left', 'baseline centre', or 'baseline right' for the alignment.

text_literal_not_rotated: If the **draughting_annotation_occurrence** is an **annotation_text_occurrence** whose item is a **text_literal**, the **text_literal** shall not be rotated.

literals_in_composites_not_rotated: If the **draughting_annotation_occurrence** is an **annotation_text_occurrence** whose item is a **composite_text**, each **text_literal** included in the **composite_text** shall not be rotated.

single_text_alignment: If the **draughting_annotation_occurrence** is an **annotation_text_occurrence**, each **text_literal** shall have the same alignment.

single_text_font: If the **draughting_annotation_occurrence** is an **annotation_text_occurrence**, each **text_literal** shall have the same font.

allowable_text_literals: If the item of the **draughting_annotation_occurrence** is an **annotation_text_occurrence**, each **composite_text** shall not contain **text_literal_with_blanking_box** nor **text_literal_with_associated_curves**.

styled_text_literal_curves: If the **draughting_annotation_occurrence** is an **annotation_text_occurrence** whose item is a **text_literal_with_associated_curves**, each **curve** associated with the text shall be the item for an **annotation_curve_occurrence.**

styled_composite_text_curves: If the **draughting_annotation_occurrence** is an **annotation_text_occurrence** whose item is a **composite_text_with_associated_curves**, each **curve** associated with the text shall be the item for an **annotation_curve_occurrence.**

curve_style_has_width: If the style of the **draughting_annotation_occurrence** is a **curve_style**, this style shall specify the units of measure associated with its **curve_width**.

tiling_constraints: If the style of the **draughting_annotation_occurrence** is a **fill_area_style**, this style shall contain, in the set of **fill_styles**, no more than one **fill_area_style_tiles** and that **fill_area_style_tiles** shall contain exactly one tile.

hatching_constraints: If the style of the **draughting_annotation_occurrence** is a **fill_area_style**, for each **fill_area_style_hatching** in the set of **fill_styles**, the **point_of_reference_hatch_line** shall be the same as the **pattern_start**.

text_style_constraint: If the style of the **draughting_annotation_occurrence** is a **text_style**, this style shall be a **text_style_with_box_characteristics**.

text_style_characteristics: If the style of the **draughting_annotation_occurrence** is a **text_style_with_box_characteristics**, this style shall have exactly four elements in the set of characteristics.

5.2.2.1.12 draughting_approval_assignment

A **draughting_approval_assignment** is the assignment of an **approval** to a **drawing_revision** or **drawing_sheet_revision**.

EXPRESS specification:

```
*)
ENTITY draughting_approval_assignment
  SUBTYPE OF (approval_assignment);
  approved_items : SET [1:?] OF approved_item;
WHERE
  WR1: SIZEOF (QUERY (item <* approved_items |
       'EXPLICIT_DRAUGHTING.DRAWING_REVISION' IN TYPEOF (item)))<=1;
END_ENTITY;
(*
```

Attribute definitions:

approved_items: a set of **approved_items** which identify the **drawing_revision** or **drawing_sheet_revision** to which an **approval** is assigned.

Formal propositions:

WR1: The set of **approved_items** shall contain at most one **drawing_revision**.

5.2.2.1.13 draughting_contract_assignment

A **draughting_contract_assignment** is the assignment of a contract to a **drawing_revision**, for the purpose of identifying the **drawing_revision** as relevant to the given contract.

EXPRESS specification:

```
*)
ENTITY draughting_contract_assignment
  SUBTYPE OF (contract_assignment);
  items : SET [1:?] OF contracted_item;
END_ENTITY;
(*
```

Attribute definitions:

items: the **drawing_revisions** that are related to the given contract.

5.2.2.1.14 draughting_drawing_revision

A **draughting_drawing_revision** is a **drawing_revision** that specifies constraints within the context of draughting.

NOTE – This entity denotes a shared scope with the Application Interpreted Model of ISO 10303-202.

EXPRESS specification:

```
*)
ENTITY draughting_drawing_revision
  SUBTYPE OF (drawing_revision);
WHERE
  drawing_requires_sheets:
    (SIZEOF (USEDIN (SELF,'EXPLICIT_DRAUGHTING.AREA_IN_SET.IN_SET')) >= 1)
    AND
    (SIZEOF (QUERY (ais <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                          'AREA_IN_SET.IN_SET') |
      NOT ('EXPLICIT_DRAUGHTING.DRAWING_SHEET_REVISION' IN TYPEOF (ais.area))
    )) = 0);
  drawing_approval_has_date:
    SIZEOF (QUERY (app_ass <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                          'DRAUGHTING_APPROVAL_ASSIGNMENT.APPROVED_ITEMS') |
       NOT (SIZEOF (USEDIN (app_ass.assigned_approval,
          'EXPLICIT_DRAUGHTING.APPROVAL_DATE_TIME.DATED_APPROVAL')) = 1)
    )) = 0;
  sheet_approval_has_date:
    SIZEOF (QUERY (ais <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                        'AREA_IN_SET.IN_SET') |
      NOT (SIZEOF (QUERY (app_ass <* USEDIN (ais.area,
          'EXPLICIT_DRAUGHTING.' +
```

```
        'DRAUGHTING_APPROVAL_ASSIGNMENT.APPROVED_ITEMS') |
      NOT (SIZEOF (USEDIN (app_ass.assigned_approval,
          'EXPLICIT_DRAUGHTING.APPROVAL_DATE_TIME.DATED_APPROVAL')) = 1)
    )) = 0)
  )) = 0;
drawing_approval_has_authorization:
  SIZEOF (QUERY (app_ass <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                      'DRAUGHTING_APPROVAL_ASSIGNMENT.APPROVED_ITEMS') |
     NOT (SIZEOF (USEDIN (app_ass.assigned_approval,
         'EXPLICIT_DRAUGHTING.' +
         'APPROVAL_PERSON_ORGANIZATION.AUTHORIZED_APPROVAL')) >= 1)
  )) = 0;
sheet_approval_has_authorization:
  SIZEOF (QUERY (ais <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                    'AREA_IN_SET.IN_SET') |
    NOT (SIZEOF (QUERY (app_ass <* USEDIN (ais.area,
        'EXPLICIT_DRAUGHTING.' +
        'DRAUGHTING_APPROVAL_ASSIGNMENT.APPROVED_ITEMS') |
      NOT (SIZEOF (USEDIN (app_ass.assigned_approval,
          'EXPLICIT_DRAUGHTING.' +
          'APPROVAL_PERSON_ORGANIZATION.AUTHORIZED_APPROVAL')) >= 1)
    )) = 0)
  )) = 0;
drawing_title:
  SIZEOF (USEDIN (SELF,'EXPLICIT_DRAUGHTING.DRAUGHTING_TITLE.ITEMS')) <= 1;

sheet_title:
  SIZEOF (QUERY (ais <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                    'AREA_IN_SET.IN_SET') |
    NOT (SIZEOF (USEDIN (ais.area, 'EXPLICIT_DRAUGHTING.' +
                                  'DRAUGHTING_TITLE.ITEMS')) <= 1)
  )) = 0;
allowable_items_of_sheet:
  SIZEOF (QUERY (ais <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                    'AREA_IN_SET.IN_SET') |
    NOT (SIZEOF (QUERY (item <* ais.area\representation.items |
      NOT (SIZEOF (['EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE',
                    'EXPLICIT_DRAUGHTING.MAPPED_ITEM',
                    'EXPLICIT_DRAUGHTING.AXIS2_PLACEMENT',
                    'EXPLICIT_DRAUGHTING.PLANAR_BOX'] *
               TYPEOF(item)) = 1)
    )) = 0)
  )) = 0;
sheet_contains_view_or_annotation:
  SIZEOF (QUERY (ais <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                    'AREA_IN_SET.IN_SET') |
    NOT (SIZEOF (QUERY (item <* ais.area\representation.items |
        (SIZEOF (['EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE',
                  'EXPLICIT_DRAUGHTING.MAPPED_ITEM'] *
                 TYPEOF(item)) = 1)
    )) > 0)
  )) = 0;
```

```
planar_box_in_sheet:
  SIZEOF (QUERY (ais <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                     'AREA_IN_SET.IN_SET') |
    NOT (SIZEOF (QUERY (p_b <* QUERY (item <* ais.area\representation.items |
                ('EXPLICIT_DRAUGHTING.PLANAR_BOX' IN TYPEOF(item))) |
      NOT (SIZEOF (USEDIN (p_b, 'EXPLICIT_DRAUGHTING.' +
                                'PRESENTATION_SIZE.SIZE')) = 1)
    )) = 0)
  )) = 0;
sheets_contain_views:
  SIZEOF (QUERY (ais <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                     'AREA_IN_SET.IN_SET') |
    NOT (SIZEOF (QUERY (mi <* QUERY (item <* ais.area\representation.items |
                ('EXPLICIT_DRAUGHTING.MAPPED_ITEM' IN TYPEOF(item))) |
      NOT ('EXPLICIT_DRAUGHTING.PRESENTATION_VIEW' IN
              TYPEOF(mi.mapping_source.mapped_representation))
    )) = 0)
  )) = 0;
sheets_contain_placements:
  SIZEOF (QUERY (ais <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                     'AREA_IN_SET.IN_SET') |
    NOT (SIZEOF (QUERY (a2p <* QUERY (item <* ais.area\representation.items |
                ('EXPLICIT_DRAUGHTING.AXIS2_PLACEMENT' IN TYPEOF(item))) |
      NOT (SIZEOF (USEDIN (a2p, 'EXPLICIT_DRAUGHTING.' +
                                'MAPPED_ITEM.MAPPING_TARGET')) > 0)
    )) = 0)
  )) = 0;
sheets_have_unique_context:
  SIZEOF (QUERY (ais <* USEDIN (SELF,
          'EXPLICIT_DRAUGHTING.AREA_IN_SET.IN_SET') |
    NOT (SIZEOF (ais.area\representation.context_of_items.
                    representations_in_context) = 1)
  )) = 0;
views_presented_once:
  SIZEOF (QUERY (ais <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                     'AREA_IN_SET.IN_SET') |
    NOT (SIZEOF (QUERY (mi <* QUERY (item <* ais.area\representation.items |
                ('EXPLICIT_DRAUGHTING.MAPPED_ITEM' IN TYPEOF(item))) |
      NOT (SIZEOF (USEDIN (mi.mapping_source.mapped_representation,
                          'EXPLICIT_DRAUGHTING.REPRESENTATION_MAP.' +
                          'MAPPED_REPRESENTATION')) = 1)
    )) = 0)
  )) = 0;
allowable_items_of_view:
  SIZEOF (QUERY (ais <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                     'AREA_IN_SET.IN_SET') |
    NOT (SIZEOF (QUERY (mi <* QUERY (item <* ais.area\representation.items |
                ('EXPLICIT_DRAUGHTING.MAPPED_ITEM' IN TYPEOF(item))) |
      NOT (SIZEOF (QUERY (pv_item <* mi.mapping_source.
                           mapped_representation\representation.items |
        NOT (SIZEOF (['EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE',
                      'EXPLICIT_DRAUGHTING.CAMERA_IMAGE',
```

```
                    'EXPLICIT_DRAUGHTING.AXIS2_PLACEMENT'] *
                TYPEOF(pv_item)) = 1)
        )) = 0)
      )) = 0)
    )) = 0;
  views_contain_one_projection:
    SIZEOF (QUERY (ais <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                        'AREA_IN_SET.IN_SET') |
      NOT (SIZEOF (QUERY (mi <* QUERY (item <* ais.area\representation.items |
                  ('EXPLICIT_DRAUGHTING.MAPPED_ITEM' IN TYPEOF(item))) |
        NOT (SIZEOF (QUERY (pv_item <* mi.mapping_source.
                            mapped_representation\representation.items |
            ('EXPLICIT_DRAUGHTING.CAMERA_IMAGE' IN TYPEOF(pv_item))
        )) = 1)
      )) = 0)
    )) = 0;
  views_contain_one_placement:
    SIZEOF (QUERY (ais <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                        'AREA_IN_SET.IN_SET') |
      NOT (SIZEOF (QUERY (mi <* QUERY (item <* ais.area\representation.items |
                  ('EXPLICIT_DRAUGHTING.MAPPED_ITEM' IN TYPEOF(item))) |
        NOT (SIZEOF (QUERY (a2p <* QUERY (pv_item <* mi.mapping_source.
                            mapped_representation\representation.items |
            ('EXPLICIT_DRAUGHTING.AXIS2_PLACEMENT' IN TYPEOF(pv_item))) |
          ('EXPLICIT_DRAUGHTING.MAPPING_SOURCE.MAPPING_ORIGIN' IN
                  ROLESOF (a2p))
        )) = 1)
      )) = 0)
    )) = 0;
  views_have_unique_context:
    SIZEOF (QUERY (ais <* USEDIN (SELF,
           'EXPLICIT_DRAUGHTING.AREA_IN_SET.IN_SET') |
      NOT (SIZEOF (QUERY (mi <* QUERY (item <* ais.area\representation.items |
            ('EXPLICIT_DRAUGHTING.MAPPED_ITEM'
                 IN TYPEOF(item))) |
        NOT (SIZEOF (mi.mapping_source.
                    mapped_representation\representation.context_of_items.
                    representations_in_context) = 1)
      )) = 0)
    )) = 0;
END_ENTITY;
(*
```

Formal propositions:

drawing_requires_sheets: The **draughting_drawing_revision** shall be related to at least one **presentation_area**, each of which is a **drawing_sheet_revision.**

drawing_approval_has_date: Each approval related to the **draughting_drawing_revision** shall have exactly one date.

sheet_approval_has_date: Each approval related to each sheet of the **draughting_drawing_revision** shall have exactly one date.

drawing_approval_has_authorization: Each approval related to the **draughting_drawing_revision** shall be authorized by one or more organizations.

sheet_approval_has_authorization: Each approval related to each sheet of the **draughting_drawing_revision** shall be authorized by one or more organizations.

drawing_title: The **draughting_drawing_revision** shall be one of the items of no more than one **draughting_title.**

sheet_title: Each sheet of the **draughting_drawing_revision** shall be one of the items of no more than one **draughting_title.**

allowable_items_of_sheet: The items of each sheet of the **draughting_drawing_revision** shall be an **annotation_occurrence**, a **mapped_item**, an **axis2_placement**, or a **planar_box**.

sheet_contains_view_or_annotation: The items of each sheet of the **draughting_drawing_revision** shall include at least one **annotation_occurrence** or **mapped_item**.

planar_box_in_sheet: Each sheet of the **draughting_drawing_revision** shall contain exactly one **planar_box** that is the **size** of a **presentation_size**.

sheets_contain_views: If there is a **mapped_item** in a sheet of the **draughting_drawing_revision**, the source of the **mapped_item** shall be a **presentation_view.**

sheets_contain_placements: Every **axis2_placement** in a sheet of the **draughting_drawing_revision** shall be the **mapping_target** for some **mapped_item.**

sheets_have_unique_context: The **representation_context** of a sheet of the **draughting_drawing_revision** shall not be the context of any other **representation**.

views_presented_once: Each view in a sheet of the **draughting_drawing_revision** shall be the **mapped_representation** for exactly one **mapped_item.**

allowable_items_of_view: The items of each view of each sheet of the **draughting_drawing_revision** shall be an **annotation_occurrence**, a **camera_image**, or an **axis2_placement.**

views_contain_one_projection: Each view of each sheet of the **draughting_drawing_revision** shall contain exactly one **camera_image.**

views_contain_one_placement: Each view of each sheet of the **draughting_drawing_revision** shall contain exactly one **axis2_placement** which is the **mapping_origin** for a **mapped_item.**

views_have_unique_context: The **representation_context** of each view of each sheet of the **draughting_drawing_revision** shall not be the context of any other **representation**.

5.2.2.1.15 draughting_elements

A **draughting_elements** is a **draughting_callout** that specifies constraints on the grouping of annotation within the context of draughting.

NOTE – This entity denotes a shared scope with the Application Interpreted Model of ISO ISO 10303-202.

EXPRESS specification:

```
*)
ENTITY draughting_elements
  SUBTYPE OF (draughting_callout);
WHERE
  WR1: SIZEOF (QUERY (l_c <*
               QUERY (con <* SELF.contents |
                     ('EXPLICIT_DRAUGHTING.LEADER_CURVE' IN TYPEOF(con))) |
       NOT (SIZEOF (QUERY (ldc <* USEDIN (l_c, 'EXPLICIT_DRAUGHTING.' +
                                              'DRAUGHTING_CALLOUT.CONTENTS') |
       ('EXPLICIT_DRAUGHTING.LEADER_DIRECTED_CALLOUT'
       IN TYPEOF (ldc)))) <= 1)
       ))=0;
  WR2: NOT ('EXPLICIT_DRAUGHTING.DIMENSION_CURVE_DIRECTED_CALLOUT'
       IN TYPEOF (SELF)) OR
       (SIZEOF (QUERY (con <* SELF.contents |
               ('EXPLICIT_DRAUGHTING.PROJECTION_CURVE' IN
                TYPEOF (con))
       )) <= 2);
  WR3: SIZEOF (QUERY (rc <* USEDIN (SELF,
              'EXPLICIT_DRAUGHTING.DIMENSION_CALLOUT_' +
              'RELATIONSHIP.RELATING_DRAUGHTING_CALLOUT') |
               rc.name = 'primary')) <= 1;
  WR4: SIZEOF (QUERY (rc <* USEDIN (SELF,
              'EXPLICIT_DRAUGHTING.DIMENSION_CALLOUT_' +
              'RELATIONSHIP.RELATING_DRAUGHTING_CALLOUT') |
               rc.name = 'secondary')) <= 1;
  WR5: SIZEOF (QUERY (sec <* QUERY (rc <* USEDIN (SELF,
               'EXPLICIT_DRAUGHTING.DIMENSION_CALLOUT_' +
               'RELATIONSHIP.RELATING_DRAUGHTING_CALLOUT') |
           rc.name = 'secondary') |
         NOT (SIZEOF (QUERY (prim <* USEDIN (SELF,
               'EXPLICIT_DRAUGHTING.DIMENSION_CALLOUT_' +
               'RELATIONSHIP.RELATING_DRAUGHTING_CALLOUT') |
            prim.name = 'primary')) = 1)
       )) = 0;
END_ENTITY;
(*
```

Formal propositions:

WR1: Each **leader_curve** that is in the contents of a **draughting_callout** shall be used by at most one **leader_directed_callout**.

WR2: Each **dimension_curve_directed_callout** shall contain at most two **projection_curves** in the set of contents.

WR3: Each **draughting_callout** shall be the specified dimension of at most one **dimension_callout_relationship** with a name of 'primary'.

WR4: Each **draughting_callout** shall be the specified dimension of at most one **dimension_callout_relationship** with a name of 'secondary'.

WR5: Each **draughting_callout** shall not participate in a secondary **dimension_callout_relationship** unless it also participates in a primary **dimension_callout_relationship**.

5.2.2.1.16 draughting_group_assignment

A **draughting_group_assignment** is an assignment of one or more **points**, **curves** or **annotation_occurrences** to a **group**.

EXPRESS specification:

```
*)
ENTITY draughting_group_assignment
  SUBTYPE OF (group_assignment);
  items : SET [1:?] OF draughting_grouped_item;
END_ENTITY;
(*
```

Attribute definitions:

items: the **draughting_grouped_items** that are assigned to a **group**.

5.2.2.1.17 draughting_model

A **draughting_model** is a representation of the shape of a product for the purpose of draughting. A **draughting_model** may contain **mapped_items** that are the mappings of **shape_representations** or of other **draughting_models**. A **mapped_item** that is the mapping of a **shape_representation** specifies an explicit representation of the shape of a product that is to be presented in the drawing. A **mapped_item** that is the mapping of another **draughting_model** specifies the use of a sub-model.

The **draughting_model** may contain **styled_items**. These **styled_items** specify the style for a **mapped_item** that is the mapping of a **shape_representation**, and provide the **curve_style** to be used in presenting the **shape_representation** in the drawing.

The **draughting_model** may contain **annotation_occurrences**. These **annotation_occurrences** specify annotation that is placed in the same coordinate system as the **shape_representation**, and is projected into a view when the **draughting_model** is presented in a drawing.

EXPRESS specification:

```
*)
ENTITY draughting_model
  SUBTYPE OF (representation);
UNIQUE
  UR1: SELF\representation.name;
WHERE
  WR1: SIZEOF (QUERY (it <* SELF.items |
         NOT (SIZEOF (['EXPLICIT_DRAUGHTING.MAPPED_ITEM',
                       'EXPLICIT_DRAUGHTING.STYLED_ITEM',
                       'EXPLICIT_DRAUGHTING.AXIS2_PLACEMENT'] *
              TYPEOF (it)) = 1
       ))) = 0;
  WR2: SIZEOF (          QUERY (mi <* QUERY (it <* SELF.items |
         ('EXPLICIT_DRAUGHTING.MAPPED_ITEM' IN TYPEOF(it))) |
         NOT (SIZEOF (['EXPLICIT_DRAUGHTING.SHAPE_REPRESENTATION',
                       'EXPLICIT_DRAUGHTING.DRAUGHTING_MODEL'] *
              TYPEOF (mi\mapped_item.mapping_source.mapped_representation)) = 1
       ))) = 0;
  WR3: SIZEOF (QUERY (smi <* QUERY (si <* QUERY (it <* SELF.items |
         ('EXPLICIT_DRAUGHTING.STYLED_ITEM' IN TYPEOF(it))) |
         ('EXPLICIT_DRAUGHTING.MAPPED_ITEM' IN TYPEOF(si\styled_item.item))) |
         NOT (('EXPLICIT_DRAUGHTING.SHAPE_REPRESENTATION' IN
TYPEOF(smi\styled_item.item\mapped_item.mapping_source.mapped_representation))
             AND
              SIZEOF (QUERY (sty <* smi\styled_item.styles |
                NOT (SIZEOF (QUERY (psa <* sty.styles |
                       NOT ('EXPLICIT_DRAUGHTING.CURVE_STYLE' IN
                              TYPEOF(psa)))) = 1
             ))) = 1)
       )) = 0;
END_ENTITY;
(*
```

Formal propositions:

UR1: The name of a **draughting_model** shall be unique.

WR1: Each item of a **draughting_model** shall be a **mapped_item**, **styled_item**, or **axis2_placement**.

WR2: If there is a **mapped_item** in the **draughting_model**, the source of the **mapped_item** shall be a **shape_representation** or another **draughting_model**.

WR3: If a **mapped_item** is styled, the source of the **mapped_item** shall be a **shape_**

representation and the style applied to the **mapped_item** shall be a **curve_style**.

Associated global rules:

The following global rules are associated with this entity and restrict its use or its relationship with other entities:

draughting_model_annotation_layers: The **draughting_model_annotation_layers** rule ensures that every **annotation_occurrence** that is an item in a **draughting_model** is assigned to at least one layer.

5.2.2.1.18 draughting_organization_assignment

A **draughting_organization_assignment** relates an organization to a **product_definition_formation**, a **drawing_revision** or a **drawing_sheet_revision**.

EXPRESS specification:

```
*)
ENTITY draughting_organization_assignment
  SUBTYPE OF (organization_assignment);
  assigned_items : SET [1:?] OF draughting_organization_item;
END_ENTITY;
(*
```

Attribute definitions:

assigned_items: a set of **draughting_organization_items** that identify the **product_definition_formation**, **drawing_revision** or **drawing_sheet_revision** to which the organization is related.

5.2.2.1.19 draughting_person_and_organization_assignment

A **draughting_person_and_organization_assignment** relates a **person_and_organization** to a **product_definition_formation**, a **drawing_revision** or a **drawing_sheet_revision**.

EXPRESS specification:

```
*)
ENTITY draughting_person_and_organization_assignment
  SUBTYPE OF (person_and_organization_assignment);
  assigned_items : SET [1:?] OF draughting_organization_item;
END_ENTITY;
(*
```

Attribute definitions:

assigned_items: a set of **draughting_organization_items** that identify the **product_definition_formation**, **drawing_revision** or **drawing_sheet_revision** to which the **person_and_-**

organization is related.

5.2.2.1.20 draughting_person_assignment

A **draughting_person_assignment** relates a person to a **product_definition_formation**, a **drawing_revision** or a **drawing_sheet_revision**.

EXPRESS specification:

```
*)
ENTITY draughting_person_assignment
  SUBTYPE OF (person_assignment);
  assigned_items : SET [1:?] OF draughting_organization_item;
END_ENTITY;
(*
```

Attribute definitions:

assigned_items: a set of **draughting_organization_items** that identify the **product_definition_formation**, **drawing_revision** or **drawing_sheet_revision** to which the person is related.

5.2.2.1.21 draughting_presented_item

A **draughting_presented_item** defines the presentation by a drawing of a **product_definition_formation.**

NOTE – This entity denotes a shared scope with the Application Interpreted Model of ISO 10303-202.

EXPRESS specification:

```
*)
ENTITY draughting_presented_item
  SUBTYPE OF (presented_item);
  items :        SET [1:?] OF draughting_presented_item_select;
WHERE
  presented_item_presentation:
      SIZEOF (QUERY (pir <* USEDIN (SELF,
          'EXPLICIT_DRAUGHTING.' +
          'PRESENTED_ITEM_REPRESENTATION.ITEM') |
          NOT ('EXPLICIT_DRAUGHTING.DRAWING_REVISION'
            IN TYPEOF (pir.presentation))
       )) = 0;
END_ENTITY;
(*
```

Attribute definitions:

items: a set of **draughting_presented_item_select** that identifies the particular **product_definition_formations** that are presented.

Formal propositions:

presented_item_presentation: Each **presented_item_representation** shall have as its presentation a **drawing_revision.**

5.2.2.1.22 draughting_pre_defined_colour

A **draughting_pre_defined_colour** is a **pre_defined_colour** that is identified by name.

EXPRESS specification:

```
*)
ENTITY draughting_pre_defined_colour
  SUBTYPE OF (pre_defined_colour);
WHERE
  WR1: SELF.name IN
       ['red',
        'green',
        'blue',
        'yellow',
        'magenta',
        'cyan',
        'black',
        'white'];
END_ENTITY;
(*
```

Formal propositions:

WR1: The name of the **draughting_pre_defined_colour** shall be 'red', 'green', 'blue', 'yellow', 'magenta', 'cyan', 'black', or 'white'.

Attribute value definitions:

Table 10 states the RGB values corresponding to each of the predefined colours that shall be supported by implementations of this part of ISO 10303.

5.2.2.1.23 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
```

Table 10 – RGB colours for pre-defined colours

Colour name	Red	Green	Blue
black	0.0	0.0	0.0
red	1.0	0.0	0.0
green	0.0	1.0	0.0
blue	0.0	0.0	1.0
yellow	1.0	1.0	0.0
magenta	1.0	0.0	1.0
cyan	0.0	1.0	1.0
white	1.0	1.0	1.0

```
  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 11 gives the lengths of each line segment and space, in millimetres. 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.

NOTES

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

2 - Illustrations of line font lists are given in Figure 16.

5.2.2.1.24 draughting_pre_defined_text_font

A **draughting_pre_defined_text_font** is a **pre_defined_text_font** that is identified by name. The definition of the appearance of each **draughting_pre_defined_text_font** is given in ISO 3098-1.

Table 11 – Line segment and space lengths for pre-defined 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

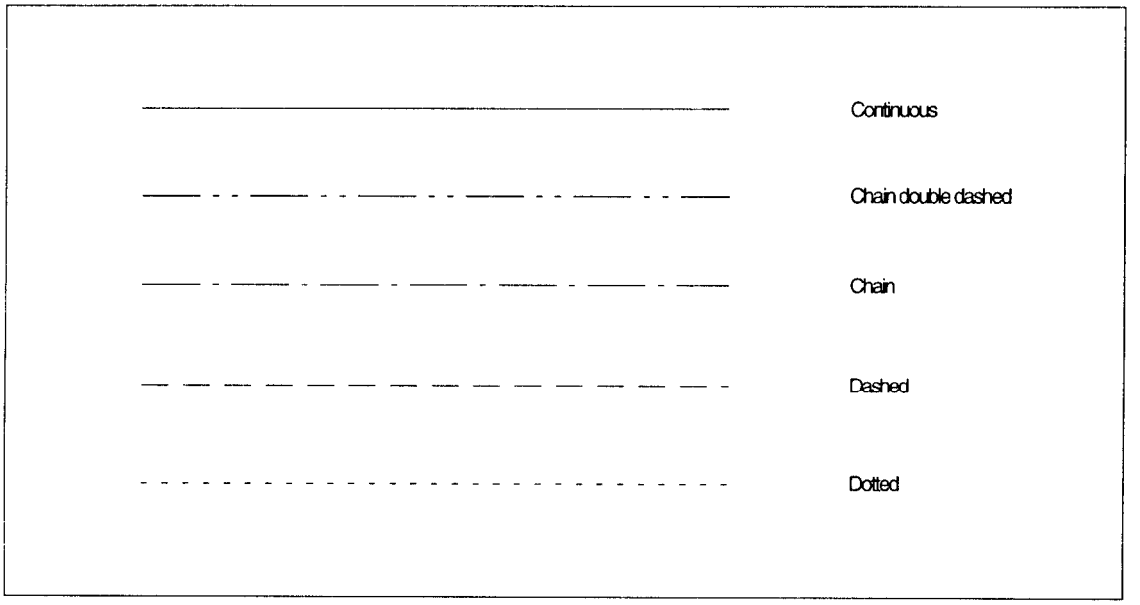

Figure 16 – Illustration of predefined curve font lists

EXPRESS specification:

```
*)
ENTITY draughting_pre_defined_text_font
  SUBTYPE of (pre_defined_text_font);
WHERE
  WR1: SELF.name IN
       ['ISO 3098-1 font A', 'ISO 3098-1 font B'];
END_ENTITY;
(*
```

Formal propositions:

WR1: The name of the **draughting_pre_defined_text_font** shall be 'ISO 3098-1 font A' or 'ISO 3098-1 font B'.

Attribute value definitions:

The **draughting_pre_defined_text_font**s are:

ISO 3098-1 font A: The text font denoted as Lettering A in clause 3 of ISO 3098-1.

ISO 3098-1 font B: The text font denoted as Lettering B in clause 3 of ISO 3098-1.

5.2.2.1.25 draughting_security_classification_assignment

A **draughting_security_classification_assignment** is the assignment of a security classification to a drawing or a drawing sheet.

EXPRESS specification:

```
*)
ENTITY draughting_security_classification_assignment
  SUBTYPE OF (security_classification_assignment);
  assigned_items : SET [1:?] OF classified_item;
END_ENTITY;
(*
```

Attribute definitions:

assigned_items: the **drawing_revisions** and **drawing_sheet_revisions** to which a security classification is assigned.

5.2.2.1.26 draughting_specification_reference

A **draughting_specification_reference** is the designation of an external draughting specification for a **drawing_revision**.

EXPRESS specification:

```
*)
ENTITY draughting_specification_reference
  SUBTYPE OF (document_reference);
  specified_items : SET [1:?] OF specified_item;
WHERE
  WR1: (SELF.assigned_document.kind.product_data_type =
          'draughting specification');
END_ENTITY;
(*
```

Attribute definitions:

specified_items: the **drawing_revisions** to which a draughting specification is applied.

Formal propositions:

WR1: Each document assigned shall be a draughting specification.

5.2.2.1.27 draughting_subfigure_representation

A **draughting_subfigure_representation** is a **symbol_representation** that collects **annotation_occurrences** as the definition of a subfigure.

EXPRESS specification:

```
*)
ENTITY draughting_subfigure_representation
  SUBTYPE OF (symbol_representation);
WHERE
  WR1: SIZEOF (QUERY (item <* SELF\representation.items |
       NOT (
       SIZEOF (['EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE',
                'EXPLICIT_DRAUGHTING.DRAUGHTING_CALLOUT',
                'EXPLICIT_DRAUGHTING.AXIS2_PLACEMENT'] *
                TYPEOF (item)) = 1
       ))) = 0;
  WR2: SIZEOF (QUERY (item <* SELF\representation.items |
       (SIZEOF (['EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE',
                 'EXPLICIT_DRAUGHTING.DRAUGHTING_CALLOUT'] *
                 TYPEOF (item)) = 1)
       )) >= 1;
  WR3: SIZEOF (QUERY (srm <* QUERY (rm <*
                           USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                         'REPRESENTATION_MAP.' +
                                         'MAPPED_REPRESENTATION') |
                           (('EXPLICIT_DRAUGHTING.SYMBOL_REPRESENTATION_MAP')
                             IN TYPEOF (rm))) |
     NOT (SIZEOF (QUERY (a_s <* QUERY (mi <* srm.map_usage |
                                 (('EXPLICIT_DRAUGHTING.ANNOTATION_SYMBOL')
                                   IN TYPEOF (mi))) |
          NOT (SIZEOF (QUERY (aso <* USEDIN (a_s, 'EXPLICIT_DRAUGHTING.' +
                                                  'STYLED_ITEM.ITEM') |
                      NOT (('EXPLICIT_DRAUGHTING.' +
                            'ANNOTATION_SUBFIGURE_OCCURRENCE')
                           IN TYPEOF (aso))
               )) = 0)
          )) = 0)
     )) > 0;
  WR4: NOT (acyclic_mapped_item_usage (SELF));
  WR5: SIZEOF (SELF.context_of_items.representations_in_context) = 1;
END_ENTITY;
(*
```

Formal propositions:

WR1: The items in a **draughting_subfigure_representation** shall be **annotation_occurrence**, **draughting_callout**, or **axis2_placement**.

WR2: At least one of the items in a **draughting_subfigure_representation** shall be an **annotation_occurrence** or a **draughting_callout**.

WR3: A **draughting_subfigure_representation** shall be the definition of at least one **annotation_subfigure_occurrence**.

WR4: A **draughting_subfigure_representation** shall not be the source of any **mapped_item** that participates in its definition.

WR5: The **representation_context** of a **draughting_subfigure_representation** shall not be the context of any other **representation**.

Associated global rules:

The following global rules are associated with this entity and restrict its use or its relationship with other entities:

draughting_subfigure_representation_layers: The **draughting_subfigure_representation_layers** rule ensures that every **annotation_occurrence** that is an item in an **draughting_subfigure_representation** is assigned to at least one layer.

5.2.2.1.28 draughting_symbol_representation

A **draughting_symbol_representation** is a **symbol_representation** that collects **annotation_curve_occurrences**, **annotation_symbol_occurrences**, **annotation_text_occurrences**, and **annotation_fill_area_occurrences** as the definition of a symbol.

EXPRESS specification:

```
*)
ENTITY draughting_symbol_representation
  SUBTYPE of (symbol_representation);
UNIQUE
  UR1: SELF\representation.name;
WHERE
  WR1: SIZEOF (QUERY (item <* SELF\representation.items |
       NOT (SIZEOF (['EXPLICIT_DRAUGHTING.' +
                     'ANNOTATION_CURVE_OCCURRENCE',
                     'EXPLICIT_DRAUGHTING.' +
                     'ANNOTATION_SYMBOL_OCCURRENCE',
                     'EXPLICIT_DRAUGHTING.' +
                     'ANNOTATION_FILL_AREA_OCCURRENCE',
                     'EXPLICIT_DRAUGHTING.' +
                     'ANNOTATION_TEXT_OCCURRENCE',
                     'EXPLICIT_DRAUGHTING.' +
                     'AXIS2_PLACEMENT'] *
            TYPEOF (item)) = 1
       ))) = 0;
  WR2: SIZEOF (QUERY (item <* SELF\representation.items |
       (SIZEOF (['EXPLICIT_DRAUGHTING.' +
                 'ANNOTATION_CURVE_OCCURRENCE',
                 'EXPLICIT_DRAUGHTING.' +
```

```
                'ANNOTATION_SYMBOL_OCCURRENCE',
                'EXPLICIT_DRAUGHTING.' +
                'ANNOTATION_FILL_AREA_OCCURRENCE',
                'EXPLICIT_DRAUGHTING.' +
                'ANNOTATION_TEXT_OCCURRENCE'] *
            TYPEOF (item)) = 1
      ))) >= 1;
  WR3: SIZEOF (QUERY (item <* SELF\representation.items |
      ('EXPLICIT_DRAUGHTING.ANNOTATION_SUBFIGURE_OCCURRENCE'
       IN TYPEOF (item))
      )) = 0;
  WR4: SIZEOF (QUERY (srm <* QUERY (rm <*
                          USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                        'REPRESENTATION_MAP.' +
                                        'MAPPED_REPRESENTATION') |
                          (('EXPLICIT_DRAUGHTING.SYMBOL_REPRESENTATION_MAP')
                            IN TYPEOF (rm))) |
     NOT (SIZEOF (QUERY (a_s <* QUERY (mi <* srm.map_usage |
                              ((('EXPLICIT_DRAUGHTING.ANNOTATION_SYMBOL')
                                IN TYPEOF (mi))) |
          NOT (SIZEOF (QUERY (aso <* USEDIN (a_s, 'EXPLICIT_DRAUGHTING.' +
                                                  'STYLED_ITEM.ITEM') |
                       NOT (('EXPLICIT_DRAUGHTING.' +
                             'ANNOTATION_SYMBOL_OCCURRENCE')
                            IN TYPEOF (aso))
               )) = 0)
          )) = 0)
     )) > 0;
  WR5: NOT (acyclic_mapped_item_usage (SELF));
  WR6: SIZEOF (SELF.context_of_items.representations_in_context) = 1;
END_ENTITY;
(*
```

Formal propositions:

UR1: The name of a **draughting_symbol_representation** shall be unique.

WR1: The items in a **draughting_symbol_representation** shall be **annotation_curve_occurrence**, **annotation_text_occurrence**, **annotation_symbol_occurrence**, **annotation_fill_area_occurrence**, or **axis2_placement**.

WR2: At least one of the items in a **draughting_symbol_representation** shall be **annotation_curve_occurrence**, **annotation_text_occurrence**, **annotation_symbol_occurrence**, or **annotation_fill_area_occurrence**.

WR3: The items in a **draughting_symbol_representation** shall not be **annotation_subfigure_occurrence**.

WR4: A **draughting_symbol_representation** shall be the definition of at least one **annotation_symbol_occurrence**.

WR5: No **draughting_symbol_representation** shall be the source of any **mapped_item** that participates in its definition.

WR6: The **representation_context** of a **draughting_symbol_representation** shall not be the context of any other **representation**.

5.2.2.1.29 draughting_text_literal_with_delineation

A **draughting_text_literal_with_delineation** is a **text_literal** that specifies the delineation of the text as overline or underline.

EXPRESS specification:

```
*)
ENTITY draughting_text_literal_with_delineation
  SUBTYPE OF (text_literal_with_delineation);
WHERE
  WR1: SELF\text_literal_with_delineation.delineation IN
       ['underline', 'overline'];
END_ENTITY;
(*
```

Formal propositions:

WR1: Every **draughting_text_literal_with_delineation** shall have either an underline or an overline delineation.

5.2.2.1.30 drawing_sheet_layout

A **drawing_sheet_layout** is a symbol that presents the layout or format for a drawing sheet.

EXPRESS specification:

```
*)
ENTITY drawing_sheet_layout
  SUBTYPE OF (draughting_symbol_representation);
END_ENTITY;
(*
```

5.2.2.1.31 geometrically_bounded_2d_wireframe_representation

A **geometrically_bounded_2d_wireframe_representation** is a **shape_representation** that represents the shape of a product by two-dimensional wireframe geometry without topology. These representations are formed by the use of two-dimensional points and curves only. All unbounded curves shall be explicitly trimmed unless they are closed. The geometric entities that are used to support the definition of another geometric entity shall not exist themselves in the elements set of a **geometric_set**.

EXAMPLE 18 – A circular arc is to be used to define the corner radius on a part that is being represented using a **geometrically_bounded_2d_wireframe_representation**. The representation of that arc is a **circle** that is referenced by a **trimmed_curve** as its **basis_curve**. The **geometric_set** that contains the shape of the part has the **trimmed_curve** in its set of elements, but does not contain the **circle** that is used as the **basis_curve**.

NOTE – This entity denotes a shared scope with the Application Interpreted Model of ISO 10303-202.

EXPRESS specification:

```
*)
ENTITY geometrically_bounded_2d_wireframe_representation
  SUBTYPE OF (shape_representation);
WHERE
  WR1: SELF.context_of_items\geometric_representation_context.
       coordinate_space_dimension = 2;
  WR2: SIZEOF (QUERY (item <* SELF\representation.items |
       NOT (
       SIZEOF (TYPEOF (item) *
              ['EXPLICIT_DRAUGHTING.GEOMETRIC_CURVE_SET',
               'EXPLICIT_DRAUGHTING.AXIS2_PLACEMENT_2D',
               'EXPLICIT_DRAUGHTING.MAPPED_ITEM']) = 1
       ))) = 0;
  WR3: SIZEOF (QUERY (gcs <* QUERY (item <* SELF\representation.items |
              ('EXPLICIT_DRAUGHTING.GEOMETRIC_CURVE_SET'
                IN TYPEOF (item))) |
       NOT (
       SIZEOF (QUERY (elem <* gcs\geometric_set.elements |
       NOT (
       SIZEOF (TYPEOF (elem) *
              ['EXPLICIT_DRAUGHTING.LINE',
               'EXPLICIT_DRAUGHTING.HYPERBOLA',
               'EXPLICIT_DRAUGHTING.PARABOLA']) = 0
       ))) = 0
       ))) = 0;
  WR4: SIZEOF (QUERY (mi <* QUERY (item <* SELF\representation.items |
              ('EXPLICIT_DRAUGHTING.MAPPED_ITEM'
                IN TYPEOF (item))) |
       NOT (
       'EXPLICIT_DRAUGHTING.GEOMETRICALLY_BOUNDED_2D_WIREFRAME_REPRESENTATION'
        IN TYPEOF (mi\mapped_item.mapping_source.mapped_representation)
       ))) = 0;
  WR5: SIZEOF (QUERY (gcs <* QUERY (item <* SELF\representation.items |
              ('EXPLICIT_DRAUGHTING.GEOMETRIC_CURVE_SET'
                IN TYPEOF (item))) |
       NOT (
       SIZEOF (QUERY (pl <* QUERY (elem <* gcs\geometric_set.elements |
              ('EXPLICIT_DRAUGHTING.POLYLINE'
                IN TYPEOF (elem))) |
       NOT (
       SIZEOF (pl\polyline.points) > 2
       ))) = 0
```

```
      ))) = 0;
  WR6: SIZEOF (QUERY (item <* SELF\representation.items |
      (SIZEOF (TYPEOF (item) *
              ['EXPLICIT_DRAUGHTING.GEOMETRIC_CURVE_SET',
               'EXPLICIT_DRAUGHTING.MAPPED_ITEM']) = 1
      ))) >= 1;
  WR7: SIZEOF (QUERY (gcs <* QUERY (item <* SELF.items |
             ('EXPLICIT_DRAUGHTING.GEOMETRIC_CURVE_SET'
               IN TYPEOF (item))) |
      NOT (
      SIZEOF (QUERY (pnt <* QUERY (elem <* gcs\geometric_set.elements |
             ('EXPLICIT_DRAUGHTING.POINT'
                 IN TYPEOF(elem))) |
      NOT (
      SIZEOF (TYPEOF (pnt) *
             ['EXPLICIT_DRAUGHTING.CARTESIAN_POINT',
              'EXPLICIT_DRAUGHTING.POINT_ON_CURVE']) = 1
      ))) = 0
      ))) = 0;
END_ENTITY;
(*
```

Formal propositions:

WR1: The **geometrically_bounded_2d_wireframe_representation** shall have a **coordinate_space_dimension** equal to two.

WR2: The items in a **geometrically_bounded_2d_wireframe_representation** shall be **geometric_curve_set**, **axis2_placement_2d**, or **mapped_item**.

WR3: No **line**, **parabola**, or **hyperbola** shall be an element in a **geometric_curve_set** of a **geometrically_bounded_2d_wireframe_representation**.

WR4: If there is a **mapped_item** in a **geometrically_bounded_2d_wireframe_representation**, the source of the **mapped_item** shall be a **geometrically_bounded_2d_wireframe_representation**.

WR5: Every **polyline** in the elements of a **geometric_curve_set** of a **geometrically_bounded_2d_wireframe_representation** shall contain more than two distinct points.

WR6: The items of a **geometrically_bounded_2d_wireframe_representation** shall include at least one **mapped_item** or one **geometric_curve_set**.

WR7: Every **point** in the elements of a **geometric_curve_set** of a **geometrically_bounded_2d_wireframe_representation** shall be either a **cartesian_point** or a **point_on_curve**.

Associated global rules:

The following global rules are associated with this entity and restrict its use or its relationship with other entities:

use_of_geometrically_bounded_2d_wireframe_representation: The **use_of_geometrically_bounded_2d_wireframe_representation** rule specifies that each **geometrically_bounded_2d_wireframe_representation** shall be mapped into at least one other **representation**, and that the only **representations** into which a **geometrically_bounded_2d_wireframe_representation** is mapped shall be **draughting_model** or another **geometrically_bounded_2d_wireframe_representation.**

5.2.2.1.32 geometrical_tolerance_callout

A **geometrical_tolerance_callout** is a **draughting_callout** that presents a geometric tolerance.

EXPRESS specification:

```
*)
ENTITY geometrical_tolerance_callout
  SUBTYPE OF (draughting_callout);
END_ENTITY;
(*
```

5.2.2.1.33 leader_directed_dimension

A **leader_directed_dimension** is a **draughting_callout** directed by a leader curve that presents some dimension value.

EXPRESS specification:

```
*)
ENTITY leader_directed_dimension
  SUBTYPE OF (leader_directed_callout);
WHERE
  WR1: SIZEOF (QUERY (con <* SELF.contents |
       'EXPLICIT_DRAUGHTING.LEADER_CURVE' IN TYPEOF (con)))=1;
END_ENTITY;
(*
```

Formal propositions:

WR1: The contents of a **leader_directed_dimension** shall contain exactly one **leader_curve.**

5.2.2.1.34 linear_dimension

A **linear_dimension** is a **draughting_callout** directed by a dimension curve that presents a value of distance between two elements, measured along a linear path, or presents the length of a linear element.

EXPRESS specification:

```
*)
ENTITY linear_dimension
  SUBTYPE OF (dimension_curve_directed_callout);
END_ENTITY;
(*
```

5.2.2.1.35 ordinate_dimension

An **ordinate_dimension** is a **draughting_callout** directed by a projection curve that presents a dimension value.

EXPRESS specification:

```
*)
ENTITY ordinate_dimension
  SUBTYPE OF (projection_directed_callout);
END_ENTITY;
(*
```

5.2.2.1.36 pre_defined_dimension_symbol

A **pre_defined_dimension_symbol** is a **pre_defined_symbol** that presents a dimension symbol and is identified by name.

EXPRESS specification:

```
*)
ENTITY pre_defined_dimension_symbol
  SUBTYPE OF (pre_defined_symbol);
WHERE
  WR1: SELF.name IN
       ['arc length',
        'conical taper',
        'counterbore',
        'countersink',
        'depth',
        'diameter',
        'plus minus',
        'radius',
        'slope',
        'spherical diameter',
        'spherical radius',
        'square'];
END_ENTITY;
(*
```

Formal propositions:

WR1: The name of the **pre_defined_dimension_symbol** shall be 'arc length', 'conical taper', 'counterbore', 'countersink', 'depth', 'diameter', 'plus minus', 'radius', 'slope', 'spherical diameter', 'spherical radius', or 'square'.

Attribute value definitions:

Figure 17 defines each **pre_defined_dimension_symbol** which shall be supported by all implementations of this part of ISO 10303. Each symbol definition includes the size of the symbol prior to scaling. All of the symbols defined in Figure 17 are oriented as they appear on a horizontal dimension line.

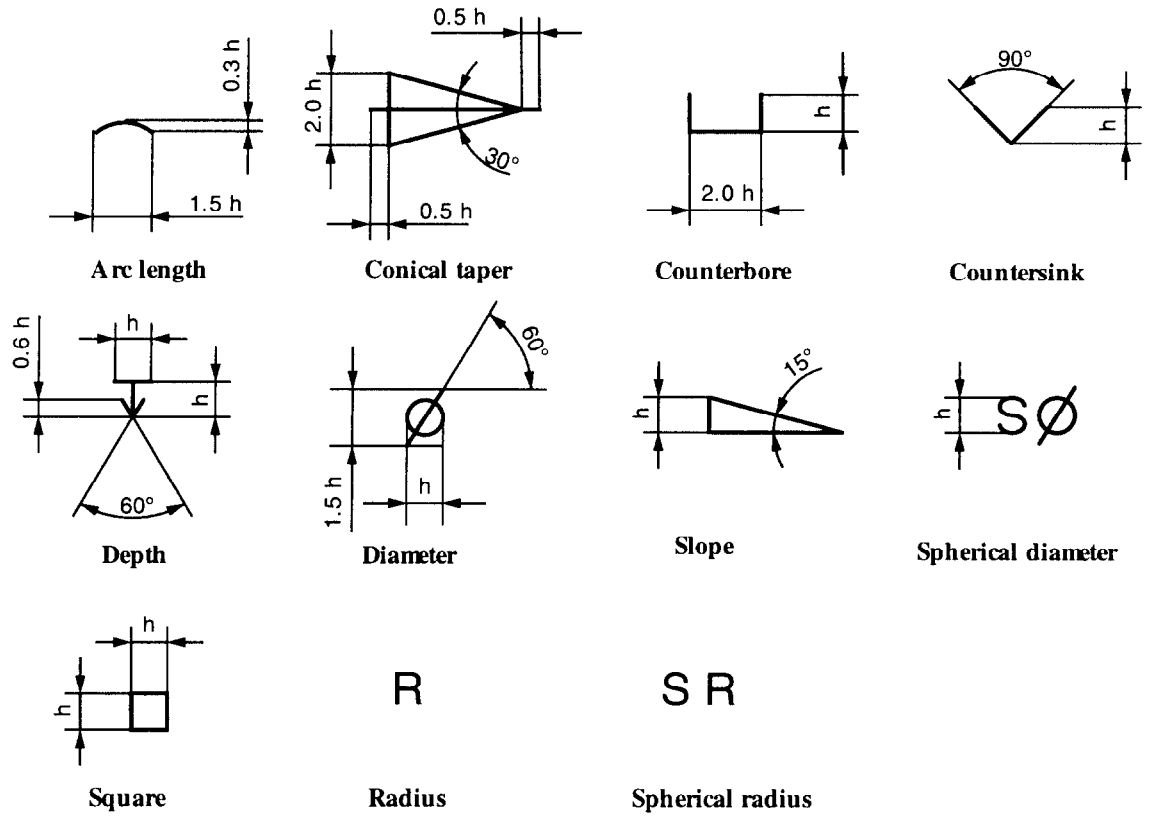

Figure 17 – Predefined dimension symbols

The **pre_defined_dimension_symbols** are:

arc length: a graphical symbol used in conjunction with a dimension value to indicate that the curve dimension is an arc length measured along the curved line or surface. An arc length symbol is depicted as one line segment that forms a part of a circle. The start point of the arc line is on the left end of the symbol and is referred to as the symbol origin.

conical taper: a graphical symbol used in conjunction with a dimension value to indicate that the value given is the ratio of the difference in the diameters of two sections taken perpendicular to the datum axis of a cone and the distance between the two sections. A

conical taper symbol is depicted as an isosceles triangle and a line. A line bisecting the angle divides the isosceles triangle into two equal right triangles. The origin of the symbol corresponds to the intersection point of the two equal sides.

counterbore: a graphical symbol used in conjunction with a dimension value to indicate that the dimension value specifies the size of the counterbored hole. A counterbored hole is a cylindrical hole of larger diameter than its basis hole. Both holes share a common centre axis, and the bottom of the counterbored hole is planar and perpendicular to the centre axis. A counterbore symbol is depicted as an open rectangle. The origin of the symbol is the geometrical centre of the rectangle.

countersink: a graphical symbol used in conjunction with a dimension value to indicate that the associated dimension value applies to a countersink which is concentric to a basis hole. A countersink is a conical taper detail located at the end of an existing or basis hole. The diameter of the taper detail is larger than the diameter of the basis hole at the surface and decreases at a constant rate as a function of distance along the centre axis of the basis hole until two diameters are equal. A countersink is depicted as an open triangle. The origin of the symbol is the intersection point of the two visible sides.

depth: a graphical symbol used in conjunction with a dimension value to indicate that the dimension value applies to the depth of the feature. A depth symbol is depicted as an arrow with a line placed perpendicular to the end opposite the arrowhead. The origin of the symbol is the tip of the arrowhead.

diameter: a graphical symbol used in conjunction with a dimension value to indicate that the dimension value applies diametrically. A diameter is depicted as an circle crossed by a line going through its centre. The origin of the symbol is the centre-point of the circle.

plus minus: a graphical symbol used in conjunction with a dimension value to indicate that the tolerance value may be applied to the dimension value. A plus minus symbol is depicted as a plus character placed above a minus character. The origin of the symbol is the centre-point of the plus.

radius: a graphical symbol used in conjunction with a dimension value to indicate that the dimension value applies radially. A radius is depicted by the letter R. The origin of the symbol is the origin of the letter.

slope: a graphical symbol used in conjunction with a dimension value to indicate that the dimension value specifies the ratio of the change in the vertical direction to the change in horizontal direction. A slope symbol is depicted as a right triangle. The origin of the symbol is the intersection of the two perpendicular sides of the triangle.

spherical diameter: a graphical symbol used in conjunction with a dimension value to indicate that the diametral dimension value applies to the surface or surfaces of a sphere. A spherical diameter is depicted as a diameter symbol as noted above and the letter S before it. The origin of the symbol is the geometrical centre of an imaginary surrounding box around the whole symbol.

spherical radius: a graphical symbol used in conjunction with a dimension value to indicate that the dimension value applies radially to all points on the dimensioned surface. A spherical radius is depicted a the letters S and R. The origin of the symbol is the geometrical centre of an imaginary surrounding box around the whole symbol.

square: a graphical symbol used in conjunction with a dimension value to indicate that the dimension value applies linearly to two orthogonal geometric elements which form two adjacent sides of a square feature. A square is depicted as a rectangle. The origin of the symbol is the geometrical centre of the rectangle.

5.2.2.1.37 pre_defined_geometrical_tolerance_symbol

A **pre_defined_geometrical_tolerance_symbol** is a **pre_defined_symbol** that presents a geometrical tolerance and is identified by name.

EXPRESS specification:

```
*)
ENTITY pre_defined_geometrical_tolerance_symbol
  SUBTYPE OF (pre_defined_symbol);
WHERE
  WR1: SELF.name IN
       ['angularity',
        'circular runout',
        'circularity',
        'concentricity',
        'cylindricity',
        'diameter',
        'flatness',
        'least material condition',
        'maximum material condition',
        'parallelism',
        'perpendicularity',
        'position',
        'profile of a line',
        'profile of a surface',
        'projected tolerance zone',
        'regardless of feature size',
        'straightness',
        'symmetry',
        'target point',
        'total runout'];
END_ENTITY;
(*
```

Formal propositions:

WR1: The name of the **pre_defined_geometrical_tolerance_symbol** shall be 'angularity', 'circular runout', 'circularity', 'concentricity', 'cylindricity', 'diameter', 'flatness', 'least material condition', 'maximum material condition', 'parallelism', 'perpendicularity', 'position', 'profile of

a line', 'profile of a surface', 'projected tolerance zone', 'regardless of feature size', 'straightness', 'symmetry', 'target point', or 'total runout'.

Attribute value definitions:

Figure 18 defines each **pre_defined_geometrical_tolerance_symbol** which shall be supported by all implementations of this part of ISO 10303. Each symbol definition includes the size of the symbol prior to scaling. All of the symbols defined in Figure 18 are oriented as they appear in a horizontal-placed tolerance frame.

The **pre_defined_geometrical_tolerance_symbols** are:

angularity: a graphical symbol used to define the condition of a surface or line which is at the specified angle, other than 90 degrees, from a datum plane or axis. An angularity symbol is depicted as two line segments that form an open triangle. The first segment is always parallel to the baseline of the text of the dimension value. The triangle opens from left to right as depicted in Figure 18. The origin of the symbol is the intersection point of the two lines.

circular runout: a graphical symbol used to define the maximum permissible surface variation at any fixed point during one complete rotation of the part about the datum axis. A circular runout symbol is depicted as a leader, terminated by an arrow. The origin of the symbol is the non-terminated end of the leader line.

circularity: a graphical symbol used to define the condition of a surface of revolution where all points of the surface intersected by any plane perpendicular to a common axis or passing through a common centre are equidistant from the axis. A circularity symbol is depicted as a circle. The origin of the symbol is the centre of the circle.

concentricity: a graphical symbol used to define the condition wherein the axes of all cross-sectional elements of a cylinder, cone, or sphere are common to a datum axis. A concentricity symbol is depicted as two concentric circles. The origin of the symbol is the common centre of the circles.

cylindricity: a graphical symbol used to define two concentric cylinders between which all elements of the specified surface must lie. A cylindricity symbol is depicted as a circle, combined with two tangential and parallel lines. The origin of the symbol is the centre of the circle.

diameter: a graphical symbol used to indicate that the associated notation applies diametrically. A diameter symbol is depicted as a circle, crossed by a line segment. The origin of the symbol is the centre of the circle.

flatness: a graphical symbol used to define the allowable perpendicular deviation of surface elements from the plane in which they reside. A flatness symbol is depicted as parallelogram. The origin of the symbol is the lower left corner of the parallelogram.

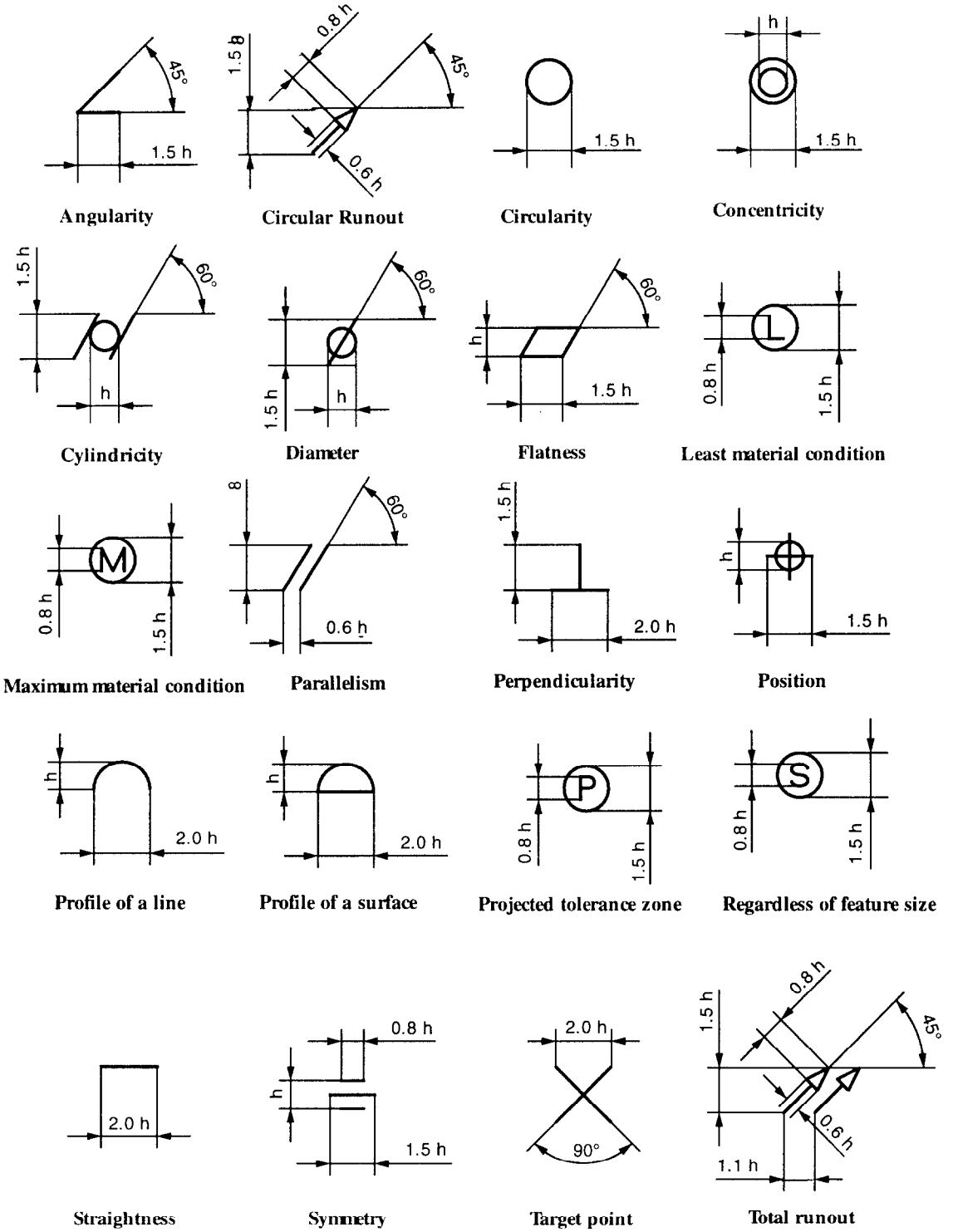

Figure 18 – Predefined geometrical tolerance symbols

least material condition: a graphical symbol used to define that the given tolerance applies to the part feature at the tolerance limit where the material content is at its minimum. A least material condition symbol is depicted as a circle with the character L positioned in its centre. The origin of the symbol is the centre of the circle.

maximum material condition: a graphical symbol used to define that the given tolerance applies to the part feature at the tolerance limit where the material content is at its maximum. A maximum material condition symbol is depicted as a circle with the character M positioned in its centre. The origin of the symbol is the centre of the circle.

parallelism: a graphical symbol used to define the condition of a surface, axis, or line which is equidistant at all points from a datum plane or axis. A parallelism symbol is depicted as two parallel line segments. The origin of the symbol is the lower point of the first line segment on the left.

perpendicularity: a graphical symbol used to define the condition of a surface, axis, or line which is at right angles to a datum plane or axis. A perpendicularity symbol is depicted as two lines, one placed perpendicular to the other. The origin of the symbol is the point of intersection of the two line segments.

position: a graphical symbol used to define either a condition wherein a part or feature has the same contour and is on opposite sides of a central plane, or a condition in which a feature is symmetrically disposed about the central plane of a datum feature. A position symbol is depicted as a circle, crossed by two perpendicular lines, intersected in the centre of the circle. The origin of the symbol is the centre of the circle.

profile of a line: a graphical symbol used to define a tolerance zone, always perpendicular to the profile at points of the profile, within which the specified line must lie. A profile of a line symbol is depicted as an arc. The origin of the symbol is the start point on the left of the arc line.

profile of a surface: a graphical symbol used to define a tolerance zone, always perpendicular to the profile at all points of the profile, within which the specified surface must lie. A profile of a surface symbol is depicted as a closed arc. The origin of the symbol is the left intersection point of the arc and the flat bottom segment.

projected tolerance zone: a graphical symbol used to define the height or depth to which a tolerance of location applies. A projected tolerance zone symbol is depicted as a circle with the character P positioned in its centre. The origin of the symbol is the centre of the circle.

regardless of feature size: a graphical symbol used to specify that the given tolerance applies to the feature regardless of its size variation. A regardless of feature size symbol is depicted as a circle with the character S positioned in its centre. The origin of the symbol is the centre of the circle.

straightness: a graphical symbol used to define a tolerance zone within which the considered element or axis must lie when the element is represented by a straight line. A

straightness symbol is depicted as a straight line segment, horizontal to the tolerance frame. The origin of the symbol is the left end point of the segment.

symmetry: a graphical symbol used to define the condition wherein all elements of the feature being toleranced must lie equidistant from the specified datum within the zone defined by the tolerance. A symmetry symbol is depicted as three parallel and horizontal line segments. The origin of the symbol is the midpoint of the middle line segment.

target point: a graphical symbol used to identify a specific point which is to be used as the datum reference point. A target point is depicted as two perpendicular line segments. The origin of the symbol is the intersection point of the two line segments.

total runout: a graphical symbol used to define the maximum permissible surface variation of all surface elements during one complete rotation of the part about the datum axis. A total runout symbol is depicted as two parallel leader lines, both terminated by an arrow. The origin of the symbol is the non-terminated end of the leader.

5.2.2.1.38 pre_defined_point_marker_symbol

A **pre_defined_point_marker_symbol** is a **pre_defined_symbol** that presents the position of a **point** and is identified by name.

EXPRESS specification:

```
*)
ENTITY pre_defined_point_marker_symbol
  SUBTYPE OF (pre_defined_symbol);
WHERE
  WR1: SELF.name IN
       ['asterisk',
        'circle',
        'dot',
        'plus',
        'square',
        'triangle',
        'x'];
END_ENTITY;
(*
```

Formal propositions:

WR1: The name of the **pre_defined_point_marker_symbol** shall be 'asterisk', 'circle', 'dot', 'plus', 'square', 'triangle', or 'x'.

Attribute value definitions:

Figure 19 defines each **pre_defined_point_marker_symbol** which shall be supported by all implementations of this part of ISO 10303. Each symbol definition includes the size of the

symbol prior to scaling. The orientation of the individual symbol is exactly as shown in Figure 19.

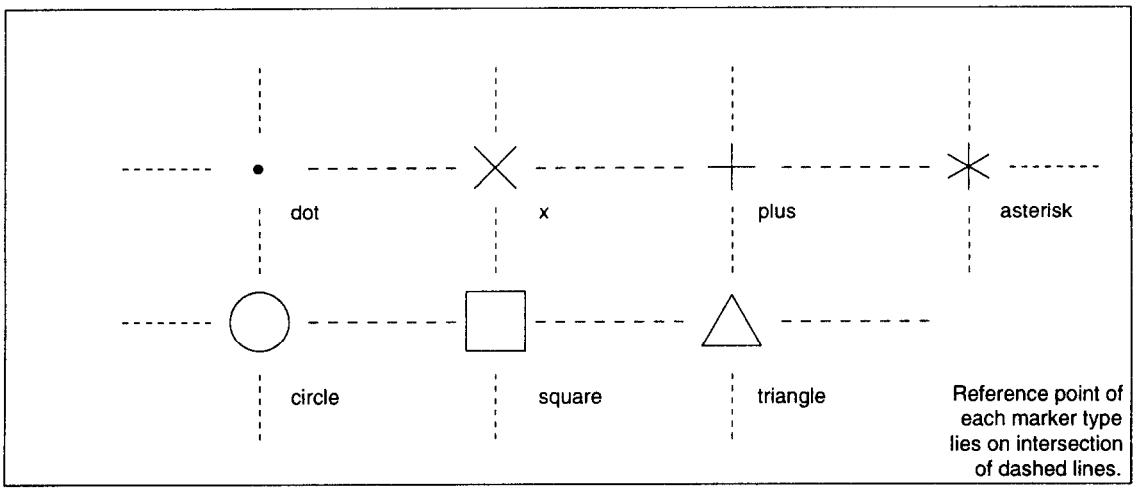

Figure 19 – Predefined point marker symbols

The **pre_defined_point_marker_symbols** are:

asterisk: a graphical symbol depicted as three line segments of equal length that intersect at their midpoints forming the origin of the symbol. One of the segments is parallel to the vertical axis of the coordinate system into which the symbol is placed. The other two segments are at angles of 60 and 120 degrees from the first segment, rotated about the origin.

circle: a graphical symbol depicted as a circle. The origin of the symbol is the geometric centre of the circle.

dot: a graphical symbol depicted as a circle with a fill-pattern applied to it. The origin of the dot symbol is the centre of the circle.

plus: a graphical symbol depicted as two perpendicular line segments. The origin of the symbol is the intersection point of the two lines.

square: a graphical symbol depicted as an even-sided rectangle. The origin of the symbol is the geometrical centre of the rectangle.

triangle: a graphical symbol depicted as three line segments that form an equilateral triangle. The origin of the triangle corresponds to the geometric centre of the triangle. One side of the triangle is parallel to the horizontal axis of the coordinate system into which the symbol is placed.

x: a graphical symbol depicted as two line segments of equal length that intersect at their midpoints forming the origin of the symbol. One line segment is at an angle of 45 degrees

to the vertical axis of the coordinate system into which the symbol is placed. The other segment is perpendicular to the first.

5.2.2.1.39 pre_defined_terminator_symbol

A **pre_defined_terminator_symbol** is a **pre_defined_symbol** that presents a terminator and is identified by name.

EXPRESS specification:

```
*)
ENTITY pre_defined_terminator_symbol
  SUBTYPE OF (pre_defined_symbol);
WHERE
  WR1: SELF.name IN
       ['blanked arrow',
        'blanked box',
        'blanked dot',
        'dimension origin',
        'filled arrow',
        'filled box',
        'filled dot',
        'integral symbol',
        'open arrow',
        'slash',
        'unfilled arrow'];
END_ENTITY;
(*
```

Formal propositions:

WR1: The name of the **pre_defined_terminator_symbol** shall be 'blanked arrow', 'blanked box', 'blanked dot', 'dimension origin', 'filled arrow', 'filled box', 'filled dot', 'integral symbol', 'open arrow', 'slash', or 'unfilled arrow'.

Attribute value definitions:

Figure 20 defines each **pre_defined_terminator_symbol** which shall be supported by all implementations of this part of ISO 10303. Each symbol definition includes the size of the symbol prior to scaling. The orientation of each symbol as positioned on an **annotation_curve** parallel to the x-axis is zero degrees as shown in Figure 20. The **annotation_curve**s shown in the figure are not part of the symbol definition.

The **pre_defined_terminator_symbols** are:

blanked arrow: a graphical symbol depicted as three line segments which form an isosceles triangle. The origin of the symbol corresponds to the intersection point of the two equal sides. The annotation curve, to which the symbol is applied, acts as a bisector to the angle created by the two equal sides. The area within the symbol is blanked.

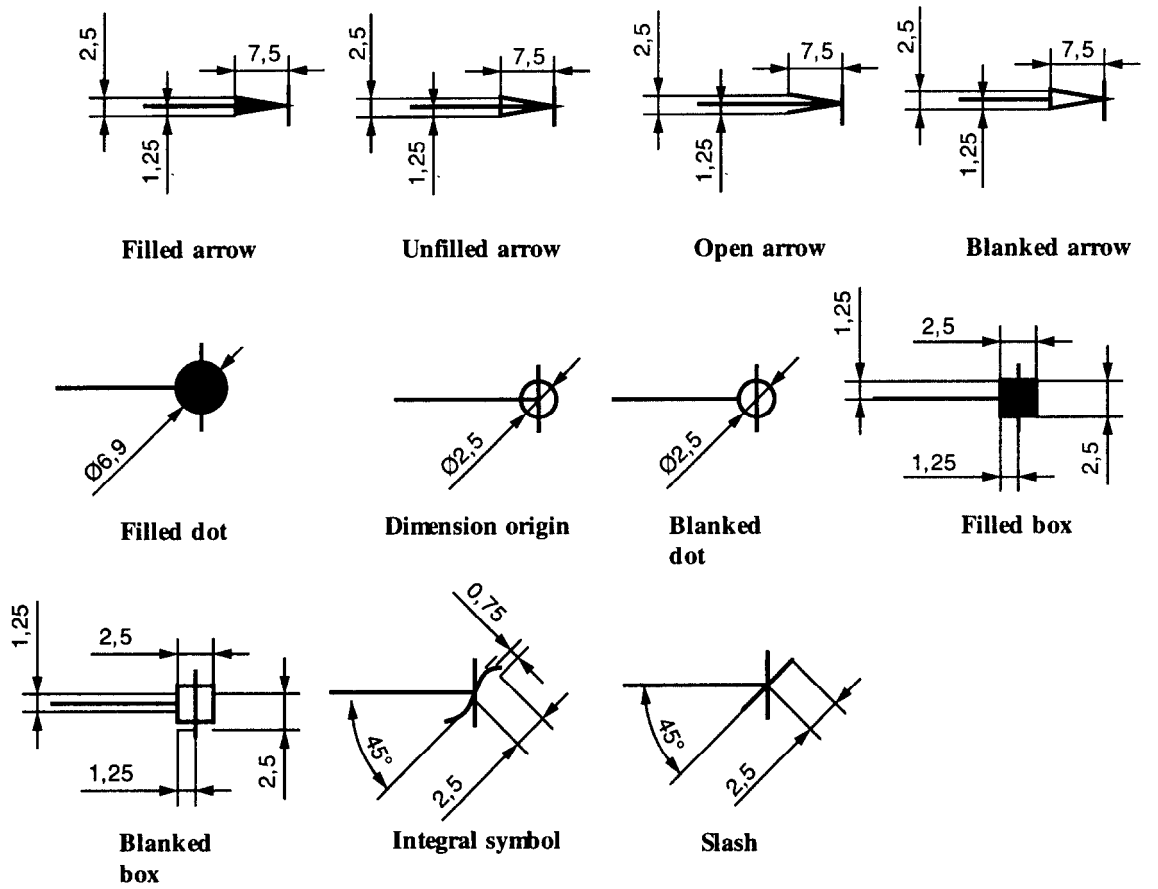

Figure 20 – Predefined terminator symbols

blanked box: a graphical symbol depicted as four line segments which form a square. Two opposite sides of the square are parallel to the annotation curve to which the symbol is applied. The origin of the symbol is the geometric centre of the square. The area within the symbol is blanked.

blanked dot: a graphical symbol depicted as a circle. The origin of the symbol is the centre of the circle. The area within the symbol is blanked.

dimension origin: a graphical symbol depicted as a circle. The origin of the symbol is the centre of the circle. Placement of the symbol is accomplished by specifying a point of location for the origin of the symbol. When the accompanying annotation curve is parallel to the X axis and in the positive X direction, the rotation is defined as zero degrees.

filled arrow: a graphical symbol depicted as three line segments which form an isosceles triangle. The origin of the symbol corresponds to the intersection point of the two equal sides. The annotation curve, to which the symbol is applied, acts as a bisector to the angle created by the two equal sides. The area within the symbol is shaded. Placement of the symbol is accomplished by specifying a point of location for the origin of the symbol. When the accompanying annotation curve is parallel to the X axis and in the positive X direction, the rotation is defined as zero degrees.

filled box: a graphical symbol depicted as four line segments which form a square. Two opposite sides of the square are parallel to the annotation curve to which the symbol is applied. The origin of the symbol is the geometric centre of the square. The area that lies within the symbol is shaded.

filled dot: a graphical symbol depicted as a circle. The origin of the symbol is the centre of the circle. The area within the symbol is shaded.

integral symbol: a graphical symbol depicted as one line segment forming two adjacent arcs. The origin of the symbol is the midpoint between the two arcs. Placement of the symbol is accomplished by specifying a point of location for the origin of the symbol. When the accompanying annotation curve is parallel to the X axis and in the positive X direction, the rotation is defined as zero degrees.

open arrow: a graphical symbol depicted as three line segments which form an isosceles triangle where the third side of the triangle is blanked. The origin of the symbol corresponds to the intersection point of the two equal sides. The annotation curve, to which the symbol is applied, acts as a bisector to the angle created by the two equal sides. Placement of the symbol is accomplished by specifying a point of location for the origin of the symbol. When the accompanying annotation curve is parallel to the X axis and in the positive X direction, the rotation is defined as zero degrees.

slash: a graphical symbol depicted as a line segment with the midpoint of the segment being the origin and lying on the annotation curve to which it is applied. Placement of the symbol is accomplished by specifying a point of location for the origin of the symbol. When the

accompanying annotation curve is parallel to the X axis and in the positive X direction, the rotation is defined as zero degrees.

unfilled arrow: a graphical symbol depicted as three line segments which form an isosceles triangle. The origin of the symbol corresponds to the intersection point of the two equal sides. The annotation curve, to which the symbol is applied, acts as a bisector to the angle created by the two equal sides. Placement of the symbol is accomplished by specifying a point of location for the origin of the symbol. When the accompanying annotation curve is parallel to the X axis and in the positive X direction, the rotation is defined as zero degrees.

5.2.2.1.40 radius_dimension

A **radius_dimension** is a **draughting_callout** directed by a dimension curve that presents a value of the radial size of a circular element.

EXPRESS specification:

```
*)
ENTITY radius_dimension
  SUBTYPE OF (dimension_curve_directed_callout);
WHERE
  WR1: SIZEOF (QUERY (con <* SELF.contents |
       'EXPLICIT_DRAUGHTING.PROJECTION_CURVE' IN TYPEOF (con)))<=1;
END_ENTITY;
(*
```

Formal propositions:

WR1: The contents of a **radius_dimension** shall contain at most one **projection_curve**.

5.2.2.1.41 structured_dimension_callout

A **structured_dimension_callout** is a **draughting_callout** that presents information associated with a dimension. Individual elements presenting different components of the dimensional information may be identified and distinguished by role names.

EXPRESS specification:

```
*)
ENTITY structured_dimension_callout
  SUBTYPE OF (draughting_callout);
WHERE
  WR1: SIZEOF (TYPEOF (SELF) *
       ['EXPLICIT_DRAUGHTING.DATUM_FEATURE_CALLOUT',
        'EXPLICIT_DRAUGHTING.DATUM_TARGET_CALLOUT',
        'EXPLICIT_DRAUGHTING.GEOMETRICAL_TOLERANCE_CALLOUT',
        'EXPLICIT_DRAUGHTING.LEADER_DIRECTED_CALLOUT',
        'EXPLICIT_DRAUGHTING.PROJECTION_DIRECTED_CALLOUT',
```

```
        'EXPLICIT_DRAUGHTING.DIMENSION_CURVE_DIRECTED_CALLOUT']) = 0;
  WR2: SIZEOF (QUERY (ato <* QUERY (con <* SELF.contents |
        ('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
              IN TYPEOF (con))) |
         NOT (ato\representation_item.name IN
             ['dimension value', 'tolerance value', 'unit text']))) = 0;
  WR3: SIZEOF (QUERY (ato <* QUERY (con <* SELF.contents |
       ('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
           IN TYPEOF (con))) |
       (ato\representation_item.name = 'dimension value')
      )) >= 1;
  WR4: SIZEOF (QUERY (dcr <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
         'DIMENSION_CALLOUT_COMPONENT_RELATIONSHIP.' +
         'RELATING_DRAUGHTING_CALLOUT') |
         dcr.name = 'prefix')) <= 1;
  WR5: SIZEOF (QUERY (dcr <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
         'DIMENSION_CALLOUT_COMPONENT_RELATIONSHIP.' +
         'RELATING_DRAUGHTING_CALLOUT') |
         dcr.name = 'suffix')) <= 1;
END_ENTITY;
(*
```

Formal propositions:

WR1: The **structured_dimension_callout** shall not be a **datum_feature_callout**, **datum_target_callout**, **geometrical_tolerance_callout**, **leader_directed_callout**, **projection_directed_callout**, or **dimension_curve_directed_callout**.

WR2: Each **annotation_text_occurrence** in the **structured_dimension_callout** shall have a name of 'dimension value', 'tolerance value', or 'unit text'.

WR3: The contents of the **structured_dimension_callout** shall contain at least one **annotation_text_occurrence** that has the name 'dimension value'.

WR4: The **structured_dimension_callout** shall participate as the **dimension_callout** in at most one **dimension_callout_component_relationship** with name of 'prefix'.

WR5: The **structured_dimension_callout** shall participate as the **dimension_callout** in at most one **dimension_callout_component_relationship** with name of 'suffix'.

5.2.2.2 explicit_draughting imported entity modifications

5.2.2.2.1 annotation_occurrence

The base definition of the entity **annotation_occurrence** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **annotation_occur-**

rence entity:

- subtype_mandatory_annotation_occurrence (see 5.2.3.45).

5.2.2.2.2 application_context

The base definition of the entity **application_context** is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **application_context** entity:

- application_context_requires_ap_definition (see 5.2.3.1).

5.2.2.2.3 application_protocol_definition

The base definition of the entity **application_protocol_definition** is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **application_protocol_definition** entity:

- application_context_requires_ap_definition (see 5.2.3.1).

5.2.2.2.4 approval

The base definition of the entity **approval** is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **approval** entity:

- approvals_are_assigned (see 5.2.3.2).

5.2.2.2.5 calendar_date

The base definition of the entity **calendar_date** is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **calendar_date** entity:

- calendar_date_instantiation (see 5.2.3.3).

5.2.2.2.6 camera_image

The base definition of the entity **camera_image** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **camera_image** entity:

- subtype_mandatory_camera_image (see 5.2.3.46).

5.2.2.2.7 camera_usage

The base definition of the entity **camera_usage** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **camera_usage** entity:

- camera_usage_mapped_representation (see 5.2.3.4).

5.2.2.2.8 colour_rgb

The base definition of the entity **colour_rgb** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **colour_rgb** entity:

- colour_rgb_instantiation (see 5.2.3.5).

5.2.2.2.9 curve_style

The base definition of the entity **curve_style** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **curve_style** entity:

- curve_style_instantiation (see 5.2.3.8).

5.2.2.2.10 curve_style_font

The base definition of the entity **curve_style_font** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rules defined in this part of ISO 10303 apply to the **curve_style_font** entity:

- curve_font_usage (see 5.2.3.6);
- curve_style_font_instantiation (see 5.2.3.7).

5.2.2.2.11 draughting_callout

The base definition of the entity **draughting_callout** is given in ISO 10303-101. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **draughting_callout** entity:

- subtype_mandatory_draughting_callout (see 5.2.3.47).

5.2.2.2.12 drawing_revision

The base definition of the entity **drawing_revision** is given in ISO 10303-101. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **drawing_revision** entity:

- subtype_mandatory_drawing_revision (see 5.2.3.48).

5.2.2.2.13 drawing_sheet_revision

The base definition of the entity **drawing_sheet_revision** is given in ISO 10303-101. The following modifications apply to this part of ISO 10303.

The following global rules defined in this part of ISO 10303 apply to the **drawing_sheet_revision** entity:

- drawing_sheet_annotation_layers (see 5.2.3.13);
- presentation_view_presented_once (see 5.2.3.33);
- sheets_belong_to_one_drawing (see 5.2.3.43).

5.2.2.2.14 drawing_sheet_revision_usage

The base definition of the entity **drawing_sheet_revision_usage** is given in ISO 10303-101. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **drawing_sheet_revision_usage** entity:

- sheets_belong_to_one_drawing (see 5.2.3.43).

5.2.2.2.15 externally_defined_curve_font

The base definition of the entity **externally_defined_curve_font** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rules defined in this part of ISO 10303 apply to the **externally_defined_curve_font** entity:

- curve_font_usage (see 5.2.3.6);
- externally_defined_curve_font_instantiation (see 5.2.3.16).

5.2.2.2.16 externally_defined_symbol

The base definition of the entity **externally_defined_symbol** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **externally_defined_symbol** entity:

- externally_defined_symbol_instantiation (see 5.2.3.17).

5.2.2.2.17 externally_defined_text_font

The base definition of the entity **externally_defined_text_font** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rules defined in this part of ISO 10303 apply to the **externally_defined_text_font** entity:

- externally_defined_text_font_instantiation (see 5.2.3.18);
- text_font_usage (see 5.2.3.56).

5.2.2.2.18 fill_area_style

The base definition of the entity **fill_area_style** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **fill_area_style** entity:

- fill_area_style_instantiation (see 5.2.3.20).

5.2.2.2.19 fill_area_style_colour

The base definition of the entity **fill_area_style_colour** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **fill_area_style_colour** entity:

- fill_area_style_colour_instantiation (see 5.2.3.19).

5.2.2.2.20 fill_area_style_tile_symbol_with_style

The base definition of the entity **fill_area_style_tile_symbol_with_style** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **fill_area_style_tile_symbol_with_style** entity:

- fill_area_style_tile_symbol_constraint (see 5.2.3.21).

5.2.2.2.21 geometric_representation_context

The base definition of the entity **geometric_representation_context** is given in ISO 10303-42. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **geometric_representation_context** entity:

- dimensionality_is_two (see 5.2.3.9).

5.2.2.2.22 geometric_set

The base definition of the entity **geometric_set** is given in ISO 10303-42. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **geometric_set** entity:

- geometric_set_layers (see 5.2.3.22).

5.2.2.2.23 global_unit_assigned_context

The base definition of the entity **global_unit_assigned_context** is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **global_unit_assigned_context** entity:

- global_length_and_angle_units (see 5.2.3.23).

5.2.2.2.24 group

The base definition of the entity **group** is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **group** entity:

- group_usage_constraint (see 5.2.3.24).

5.2.2.2.25 group_assignment

The base definition of the entity **group_assignment** is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **group_assignment** entity:

- group_usage_constraint (see 5.2.3.24).

5.2.2.2.26 mapped_item

The base definition of the entity **mapped_item** is given in ISO 10303-43. The following modifications apply to this part of ISO 10303.

The following global rules defined in this part of ISO 10303 apply to the **mapped_item** entity:

- draughting_sub_model_mapped_item (see 5.2.3.12);
- drawing_sheet_layout_usage (see 5.2.3.14);
- use_of_geometrically_bounded_2d_wireframe_representation (see 5.2.3.59).

5.2.2.2.27 measure_with_unit

The base definition of the entity **measure_with_unit** is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **measure_with_unit** entity:

- measure_with_unit_instantiation (see 5.2.3.25).

5.2.2.2.28 named_unit

The base definition of the entity **named_unit** is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **named_unit** entity:

- named_unit_instantiation (see 5.2.3.26).

5.2.2.2.29 presentation_layer_assignment

The base definition of the entity **presentation_layer_assignment** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rules defined in this part of ISO 10303 apply to the **presentation_layer_assignment** entity:

- presentation_layer_assignment_constraint (see 5.2.3.27);
- presentation_layer_assignment_requires_usage (see 5.2.3.28).

5.2.2.2.30 presentation_layer_usage

The base definition of the entity **presentation_layer_usage** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **presentation_layer_usage** entity:

- presentation_layer_assignment_requires_usage (see 5.2.3.28).

5.2.2.2.31 presentation_size

The base definition of the entity **presentation_size** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **presentation_size** entity:

- presentation_size_constraint (see 5.2.3.30).

5.2.2.2.32 presentation_style_by_context

The base definition of the entity **presentation_style_by_context** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rules defined in this part of ISO 10303 apply to the **presentation_style_by_context** entity:

- presentation_style_by_context_constraint (see 5.2.3.31);
- presentation_style_by_context_instantiation (see 5.2.3.32).

5.2.2.2.33 presentation_view

The base definition of the entity **presentation_view** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rules defined in this part of ISO 10303 apply to the **presentation_view** entity:

- drawing_view_annotation_layers (see 5.2.3.15);
- presentation_view_presented_once (see 5.2.3.33).

5.2.2.2.34 pre_defined_colour

The base definition of the entity **pre_defined_colour** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rules defined in this part of ISO 10303 apply to the **pre_defined_colour** entity:

- pre_defined_colour_instantiation (see 5.2.3.34);

– subtype_mandatory_pre_defined_colour (see 5.2.3.49).

5.2.2.2.35 pre_defined_curve_font

The base definition of the entity **pre_defined_curve_font** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rules defined in this part of ISO 10303 apply to the **pre_defined_curve_font** entity:

– curve_font_usage (see 5.2.3.6);

– pre_defined_curve_font_instantiation (see 5.2.3.35);

– subtype_mandatory_pre_defined_curve_font (see 5.2.3.50).

5.2.2.2.36 pre_defined_symbol

The base definition of the entity **pre_defined_symbol** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rules defined in this part of ISO 10303 apply to the **pre_defined_symbol** entity:

– pre_defined_symbol_instantiation (see 5.2.3.36);

– subtype_mandatory_pre_defined_symbol (see 5.2.3.51).

5.2.2.2.37 pre_defined_text_font

The base definition of the entity **pre_defined_text_font** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rules defined in this part of ISO 10303 apply to the **pre_defined_text_font** entity:

– pre_defined_text_font_instantiation (see 5.2.3.37);

– subtype_mandatory_pre_defined_text_font (see 5.2.3.52);

– text_font_usage (see 5.2.3.56).

5.2.2.2.38 product

The base definition of the entity **product** is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **product** entity:

- product_requires_version (see 5.2.3.39).

5.2.2.2.39 product_definition

The base definition of the entity **product_definition** is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **product_definition** entity:

- product_definition_instantiation (see 5.2.3.38).

5.2.2.2.40 product_definition_formation

The base definition of the entity **product_definition_formation** is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

The following global rules defined in this part of ISO 10303 apply to the **product_definition_formation** entity:

- presentation_of_product_definition_formation (see 5.2.3.29);
- product_requires_version (see 5.2.3.39).

5.2.2.2.41 product_definition_shape

The base definition of the entity **product_definition_shape** is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **product_definition_shape** entity:

- product_shape_has_representations (see 5.2.3.40).

5.2.2.2.42 representation

The base definition of the entity **representation** is given in ISO 10303-43. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **representation** entity:

- representation_instantiation (see 5.2.3.42).

5.2.2.2.43 representation_context

The base definition of the entity **representation_context** is given in ISO 10303-43. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **representation_context** entity:

- representation_context_global_units (see 5.2.3.41).

5.2.2.2.44 shape_representation

The base definition of the entity **shape_representation** is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **shape_representation** entity:

- subtype_mandatory_shape_representation (see 5.2.3.53).

5.2.2.2.45 styled_item

The base definition of the entity **styled_item** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **styled_item** entity:

- styled_curve (see 5.2.3.44).

5.2.2.2.46 symbol_colour

The base definition of the entity **symbol_colour** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **symbol_colour** entity:

- symbol_colour_instantiation (see 5.2.3.54).

5.2.2.2.47 terminator_symbol

The base definition of the entity **terminator_symbol** is given in ISO 10303-101. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **terminator_symbol** entity:

- terminator_symbol_constraint (see 5.2.3.55).

5.2.2.2.48 text_style

The base definition of the entity **text_style** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **text_style** entity:

- text_style_instantiation (see 5.2.3.58).

5.2.2.2.49 text_style_for_defined_font

The base definition of the entity **text_style_for_defined_font** is given in ISO 10303-46. The following modifications apply to this part of ISO 10303.

The following global rule defined in this part of ISO 10303 applies to the **text_style_for_defined_font** entity:

- text_style_for_defined_font_instantiation (see 5.2.3.57).

5.2.3 explicit_draughting rule definitions

5.2.3.1 application_context_requires_ap_definition

The **application_context_requires_ap_definition** rule specifies that each instance of **application_context** shall be referenced by exactly one **application_protocol_definition**. The rule also specifies that the context for all product data in the scope of this part of ISO 10303 is draughting, and that this context is associated with the schema name 'EXPLICIT_DRAUGHTING'.

EXPRESS specification:

```
*)
RULE application_context_requires_ap_definition FOR
  (application_context, application_protocol_definition);
WHERE
  WR1: SIZEOF (QUERY (ac <* application_context |
       NOT (SIZEOF (QUERY (apd <* application_protocol_definition |
       (ac :=: apd.application)
       AND
       (apd.application_interpreted_model_schema_name =
       'EXPLICIT_DRAUGHTING')
       AND
       (ac.application = 'draughting'))) = 1 ))) = 0;
END_RULE;
(*
```

Argument definitions:

application_context: the set of all instances of the **application_context** entity.

application_protocol_definition: the set of all instances of the **application_protocol_definition** entity.

Formal propositions:

WR1: For each instance of **application_context**, there shall be exactly one instance of **application_protocol_definition** which contains the instance of **application_context** as its application attribute. Each **application_context** shall specify the application to be 'draughting', and each **application_protocol_definition** shall specify the **application_interpreted_model_schema_name** to be **'EXPLICIT_DRAUGHTING'**.

5.2.3.2 approvals_are_assigned

The **approvals_are_assigned** rule ensures that all approvals are associated with product data.

EXPRESS specification:

```
*)
RULE approvals_are_assigned FOR (approval);
WHERE
  WR1: SIZEOF (QUERY (a <* approval |
       NOT (SIZEOF (USEDIN (a, 'EXPLICIT_DRAUGHTING.' +
                               'APPROVAL_ASSIGNMENT.' +
                               'ASSIGNED_APPROVAL')) >= 1
       )))=0;
END_RULE;
(*
```

Argument definitions:

approval: the set of all instances of the **approval** entity.

Formal propositions:

WR1: Each **approval** shall be the **assigned_approval** of at least one **approval_assignment.**

5.2.3.3 calendar_date_instantiation

The **calendar_date_instantiation** rule ensures that a **calendar_date** is not independently instantiated.

EXPRESS specification:

```
*)
RULE calendar_date_instantiation FOR (calendar_date);
WHERE
  WR1: SIZEOF (QUERY (each <* calendar_date |
         NOT (SIZEOF (USEDIN (each, '')) >= 1
       ))) = 0;
END_RULE;
(*
```

Argument definitions:

calendar_date: the set of all instances of the **calendar_date** entity.

Formal propositions:

WR1: Each **calendar_date** shall be used at least once.

5.2.3.4 camera_usage_mapped_representation

The **camera_usage_mapped_representation** rule ensures that a **camera_usage** is the mapping of a **draughting_model.**

EXPRESS specification:

```
*)
RULE camera_usage_mapped_representation FOR (camera_usage);
WHERE
  WR1: SIZEOF (QUERY (cu <* camera_usage |
       NOT ('EXPLICIT_DRAUGHTING.DRAUGHTING_MODEL'
            IN TYPEOF (cu.mapped_representation))))=0;
END_RULE;
(*
```

Argument definitions:

camera_usage: the set of all instances of the **camera_usage** entity.

Formal propositions:

WR1: Each **camera_usage** shall have as its **mapped_representation** a **draughting_model**.

5.2.3.5 colour_rgb_instantiation

The **colour_rgb_instantiation** rule ensures that a **colour_rgb** is not independently instantiated.

EXPRESS specification:

```
*)
RULE colour_rgb_instantiation FOR (colour_rgb);
WHERE
  WR1: SIZEOF (QUERY (each <* colour_rgb |
         NOT (SIZEOF (USEDIN (each, '')) >= 1
       ))) = 0;
END_RULE;
(*
```

Argument definitions:

colour_rgb: the set of all instances of the **colour_rgb** entity.

Formal propositions:

WR1: Each **colour_rgb** shall be used at least once.

5.2.3.6 curve_font_usage

The **curve_font_usage** rule ensures that each **curve_style_font**, **pre_defined_curve_font**, and **externally_defined_curve_font** shall define the curve font of at least one **curve_style**.

EXPRESS specification:

```
*)
RULE curve_font_usage FOR
  (curve_style_font,
   pre_defined_curve_font,
   externally_defined_curve_font);
WHERE
  WR1: SIZEOF (QUERY (csf <* curve_style_font |
       NOT (SIZEOF (USEDIN (csf, 'EXPLICIT_DRAUGHTING.' +
                                 'CURVE_STYLE.CURVE_FONT'))>=1)))=0;
  WR2: SIZEOF (QUERY (pdcf <* pre_defined_curve_font |
       NOT (SIZEOF (USEDIN (pdcf, 'EXPLICIT_DRAUGHTING.' +
                                  'CURVE_STYLE.CURVE_FONT'))>=1)))=0;
  WR3: SIZEOF (QUERY (edcf <* externally_defined_curve_font |
       NOT (SIZEOF (USEDIN (edcf, 'EXPLICIT_DRAUGHTING.' +
                                  'CURVE_STYLE.CURVE_FONT'))>=1)))=0;
END_RULE;
(*
```

Argument definitions:

curve_style_font: the set of all instances of the **curve_style_font** entity.

pre_defined_curve_font: the set of all instances of the **pre_defined_curve_font** entity.

externally_defined_curve_font: the set of all instances of the **externally_defined_curve_font** entity.

Formal propositions:

WR1: Each **curve_style_font** shall participate in the definition of at least one **curve_style**.

WR2: Each **pre_defined_curve_font** shall participate in the definition of at least one **curve_style**.

WR3: Each **externally_defined_curve_font** shall participate in the definition of at least one **curve_style**.

5.2.3.7 curve_style_font_instantiation

The **curve_style_font_instantiation** rule ensures that a **curve_style_font** is not independently instantiated.

EXPRESS specification:

```
*)
RULE curve_style_font_instantiation FOR (curve_style_font);
WHERE
  WR1: SIZEOF (QUERY (each <* curve_style_font |
         NOT (SIZEOF (USEDIN (each, ''')) >= 1
       ))) = 0;
END_RULE;
(*
```

Argument definitions:

curve_style_font: the set of all instances of the **curve_style_font** entity.

Formal propositions:

WR1: Each **curve_style_font** shall be used at least once.

5.2.3.8 curve_style_instantiation

The **curve_style_instantiation** rule ensures that a **curve_style** is not independently instantiated.

EXPRESS specification:

```
*)
RULE curve_style_instantiation FOR (curve_style);
WHERE
  WR1: SIZEOF (QUERY (each <* curve_style |
         NOT (SIZEOF (USEDIN (each, ''')) >= 1
       ))) = 0;
END_RULE;
(*
```

Argument definitions:

curve_style: the set of all instances of the **curve_style** entity.

Formal propositions:

WR1: Each **curve_style** shall be used at least once.

5.2.3.9 dimensionality_is_two

The **dimensionality_is_two** rule ensures that all representations are defined in two-dimensional coordinate systems.

EXPRESS specification:

```
*)
RULE dimensionality_is_two FOR (geometric_representation_context);
WHERE
  WR1: SIZEOF (QUERY (g <* geometric_representation_context |
               NOT ((g.coordinate_space_dimension = 2)
              )))=0;
END_RULE;
(*
```

Argument definitions:

geometric_representation_context: the set of all instances of the **geometric_representation_context** entity.

Formal propositions:

WR1: Each **geometric_representation_context** shall have a **coordinate_space_dimension** equal to two.

5.2.3.10 draughting_model_annotation_layers

The **draughting_model_annotation_layers** rule ensures that every **annotation_occurrence** that is an item in a **draughting_model** is assigned to at least one layer.

EXPRESS specification:

```
*)
RULE draughting_model_annotation_layers FOR
  (draughting_model);
WHERE
  WR1: SIZEOF
       (QUERY (dm <* draughting_model |
       NOT (SIZEOF (QUERY (ao <* QUERY (it <* dm\representation.items |
          ('EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE' IN TYPEOF (it))) |
          NOT (SIZEOF (USEDIN (ao, 'EXPLICIT_DRAUGHTING.' +
                                   'PRESENTATION_LAYER_ASSIGNMENT.' +
                                   'ASSIGNED_ITEMS')) >= 1)
           ))=0)
       ))=0;
END_RULE;
(*
```

Argument definitions:

draughting_model: the set of all instances of the **draughting_model** entity.

Formal propositions:

WR1: Every item of each **draughting_model** that is an **annotation_occurrence** shall be one of the **assigned_items** of at least one **presentation_layer_assignment**.

5.2.3.11 draughting_subfigure_representation_layers

The **draughting_subfigure_representation_layers** rule ensures that every **annotation_occurrence** that is an item in a **draughting_subfigure_representation** is assigned to at least one layer.

EXPRESS specification:

```
*)
RULE draughting_subfigure_representation_layers FOR
  (draughting_subfigure_representation);
WHERE
  WR1: SIZEOF (QUERY (dsr <* draughting_subfigure_representation |
       NOT (SIZEOF (QUERY (item <* dsr\representation.items |
           ('EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE'
            IN TYPEOF (item)) AND
           (SIZEOF (USEDIN (item, 'EXPLICIT_DRAUGHTING.' +
                                  'PRESENTATION_LAYER_ASSIGNMENT.' +
                                  'ASSIGNED_ITEMS'))=0)
           ))=0)
       ))=0;
END_RULE;
(*
```

Argument definitions:

draughting_subfigure_representation: the set of all instances of the **draughting_subfigure_representation** entity.

Formal propositions:

WR1: Every item of each **draughting_subfigure_representation** that is an **annotation_occurrence** shall be one of the **assigned_items** of at least one **presentation_layer_assignment**.

5.2.3.12 draughting_sub_model_mapped_item

The **draughting_sub_model_mapped_item** rule ensures that each **mapped_item** that is the mapping of a **shape_representation** participates in the definition of a **draughting_model** or another **shape_representation**.

EXPRESS specification:

```
*)
RULE draughting_sub_model_mapped_item FOR
  (mapped_item);
WHERE
  WR1: SIZEOF (QUERY (sr <* QUERY (mi <* mapped_item |
         ('EXPLICIT_DRAUGHTING.SHAPE_REPRESENTATION'
             IN TYPEOF (mi.mapping_source.mapped_representation))) |
        NOT (SIZEOF (USEDIN (sr, ''))  =
             SIZEOF (QUERY (mr <* USEDIN (sr, 'EXPLICIT_DRAUGHTING.' +
                                              'REPRESENTATION.ITEMS') |
                SIZEOF (['EXPLICIT_DRAUGHTING.DRAUGHTING_MODEL',
                         'EXPLICIT_DRAUGHTING.SHAPE_REPRESENTATION'] *
                   TYPEOF (mr))= 1)))
       ))=0;
END_RULE;
(*
```

Argument definitions:

mapped_item: the set of all instances of the **mapped_item** entity.

Formal propositions:

WR1: Each **mapped_item** that is the mapping of a **shape_representation** shall be one of the items of a **draughting_model** or **shape_representation**.

5.2.3.13 drawing_sheet_annotation_layers

The **drawing_sheet_annotation_layers** rule ensures that every **annotation_occurrence** that is an item in a **drawing_sheet_revision** is assigned to at least one layer.

EXPRESS specification:

```
*)
RULE drawing_sheet_annotation_layers FOR (drawing_sheet_revision);
WHERE
  WR1: SIZEOF
       (QUERY (dsr <* drawing_sheet_revision |
       NOT (SIZEOF (QUERY (ao <* QUERY (it <* dsr\representation.items |
          ('EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE' IN TYPEOF (it))) |
          NOT (SIZEOF (USEDIN (ao, 'EXPLICIT_DRAUGHTING.' +
                                   'PRESENTATION_LAYER_ASSIGNMENT.' +
                                   'ASSIGNED_ITEMS')) >= 1)
           ))=0)
       ))=0;
END_RULE;
(*
```

Argument definitions:

drawing_sheet_revision: the set of all instances of the **drawing_sheet_revision** entity.

Formal propositions:

WR1: Every item of each **drawing_sheet_revision** that is an **annotation_occurrence** shall be one of the **assigned_items** of at least one **presentation_layer_assignment**.

5.2.3.14 drawing_sheet_layout_usage

The **drawing_sheet_layout_usage** rule ensures that a **drawing_sheet_layout** is mapped as one of the items of a **drawing_sheet_revision**.

EXPRESS specification:

```
*)
RULE drawing_sheet_layout_usage FOR (mapped_item);
WHERE
  WR1: SIZEOF (QUERY (dsl <* QUERY (mi <* mapped_item |
         ('EXPLICIT_DRAUGHTING.DRAWING_SHEET_LAYOUT'
             IN TYPEOF (mi.mapping_source.mapped_representation))) |
         NOT (SIZEOF (USEDIN (dsl, ''')) =
             SIZEOF (QUERY (dsr <* USEDIN (dsl, 'EXPLICIT_DRAUGHTING.' +
                                'REPRESENTATION.ITEMS') |
             ('EXPLICIT_DRAUGHTING.DRAWING_SHEET_REVISION'
                IN TYPEOF(dsr)))))
       )) = 0;
END_RULE;
(*
```

Argument definitions:

mapped_item: the set of all instances of the **mapped_item** entity.

Formal propositions:

WR1: Each **mapped_item** that is the mapping of a **drawing_sheet_layout** shall be one of the items in a **drawing_sheet_revision**.

5.2.3.15 drawing_view_annotation_layers

The **drawing_view_annotation_layers** rule ensures that every **annotation_occurrence** that is an item in a **presentation_view** is assigned to at least one layer.

EXPRESS specification:

```
*)
RULE drawing_view_annotation_layers FOR (presentation_view);
WHERE
  WR1: SIZEOF
       (QUERY (pv <* presentation_view |
       NOT (SIZEOF (QUERY (ao <* QUERY (it <* pv\representation.items |
          ('EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE' IN TYPEOF (it))) |
          NOT (SIZEOF (USEDIN (ao, 'EXPLICIT_DRAUGHTING.' +
                                   'PRESENTATION_LAYER_ASSIGNMENT.' +
                                   'ASSIGNED_ITEMS')) >= 1)
           ))=0)
       ))=0;
END_RULE;
(*
```

Argument definitions:

presentation_view: the set of all instances of the **presentation_view** entity.

Formal propositions:

WR1: Every item of each **presentation_view** that is an **annotation_occurrence** shall be one of the **assigned_items** of at least one **presentation_layer_assignment**.

5.2.3.16 externally_defined_curve_font_instantiation

The **externally_defined_curve_font_instantiation** rule ensures that a **externally_defined_curve_font** is not independently instantiated.

EXPRESS specification:

```
*)
RULE externally_defined_curve_font_instantiation FOR
  (externally_defined_curve_font);
WHERE
  WR1: SIZEOF (QUERY (each <* externally_defined_curve_font |
         NOT (SIZEOF (USEDIN (each, ''')) >= 1
       ))) = 0;
END_RULE;
```

```
(*
```

Argument definitions:

externally_defined_curve_font: the set of all instances of the **externally_defined_curve_font** entity.

Formal propositions:

WR1: Each **externally_defined_curve_font** shall be used at least once.

5.2.3.17 externally_defined_symbol_instantiation

The **externally_defined_symbol_instantiation** rule ensures that a **externally_defined_symbol** is not independently instantiated.

EXPRESS specification:

```
*)
RULE externally_defined_symbol_instantiation FOR (externally_defined_symbol);
WHERE
  WR1: SIZEOF (QUERY (each <* externally_defined_symbol |
         NOT (SIZEOF (USEDIN (each, '')) >= 1
       ))) = 0;
END_RULE;
(*
```

Argument definitions:

externally_defined_symbol: the set of all instances of the **externally_defined_symbol** entity.

Formal propositions:

WR1: Each **externally_defined_symbol** shall be used at least once.

5.2.3.18 externally_defined_text_font_instantiation

The **externally_defined_text_font_instantiation** rule ensures that a **externally_defined_text_font** is not independently instantiated.

EXPRESS specification:

```
*)
RULE externally_defined_text_font_instantiation FOR
  (externally_defined_text_font);
WHERE
  WR1: SIZEOF (QUERY (each <* externally_defined_text_font |
         NOT (SIZEOF (USEDIN (each, '')) >= 1
```

```
      ))) = 0;
END_RULE;
(*
```

Argument definitions:

externally_defined_text_font: the set of all instances of the **externally_defined_text_font** entity.

Formal propositions:

WR1: Each **externally_defined_text_font** shall be used at least once.

5.2.3.19 fill_area_style_colour_instantiation

The **fill_area_style_colour_instantiation** rule ensures that a **fill_area_style_colour** is not independently instantiated.

EXPRESS specification:

```
*)
RULE fill_area_style_colour_instantiation FOR (fill_area_style_colour);
WHERE
  WR1: SIZEOF (QUERY (each <* fill_area_style_colour |
         NOT (SIZEOF (USEDIN (each, ''')) >= 1
       ))) = 0;
END_RULE;
(*
```

Argument definitions:

fill_area_style_colour: the set of all instances of the **fill_area_style_colour** entity.

Formal propositions:

WR1: Each **fill_area_style_colour** shall be used at least once.

5.2.3.20 fill_area_style_instantiation

The **fill_area_style_instantiation** rule ensures that a **fill_area_style** is not independently instantiated.

EXPRESS specification:

```
*)
RULE fill_area_style_instantiation FOR (fill_area_style);
WHERE
  WR1: SIZEOF (QUERY (each <* fill_area_style |
         NOT (SIZEOF (USEDIN (each, ''')) >= 1
```

```
      ))) = 0;
END_RULE;
(*
```

Argument definitions:

fill_area_style: the set of all instances of the **fill_area_style** entity.

Formal propositions:

WR1: Each **fill_area_style** shall be used at least once.

5.2.3.21 fill_area_style_tile_symbol_constraint

The **fill_area_style_tile_symbol_constraint** rule ensures that a **pre_defined_symbol** does not participate in the definition of a **fill_area_style_tile_symbol_with_style.**

EXPRESS specification:

```
*)
RULE fill_area_style_tile_symbol_constraint FOR
  (fill_area_style_tile_symbol_with_style);
WHERE
  WR1: SIZEOF (QUERY (fast <* fill_area_style_tile_symbol_with_style |
       (('EXPLICIT_DRAUGHTING.DEFINED_SYMBOL' IN
          TYPEOF(fast.symbol\styled_item.item))
        AND NOT
        ('EXPLICIT_DRAUGHTING.EXTERNALLY_DEFINED_SYMBOL' IN
          TYPEOF (fast.symbol\styled_item.
                  item\defined_symbol.definition)))
       )) = 0;
END_RULE;
(*
```

Argument definitions:

fill_area_style_tile_symbol_with_style: the set of all instances of the **fill_area_style_tile_symbol_with_style** entity.

Formal propositions:

WR1: Each **fill_area_style_tile_symbol_with_style** that has an implicit definition shall be defined by an **externally_defined_symbol.**

5.2.3.22 geometric_set_layers

The **geometric_set_layers** rule ensures that each element of a **geometric_set** is assigned to at least one layer.

EXPRESS specification:

```
*)
RULE geometric_set_layers FOR (geometric_set);
WHERE
  WR1: SIZEOF (QUERY (gs <* geometric_set |
       NOT (SIZEOF (QUERY (element <* gs.elements |
            NOT (SIZEOF (USEDIN (element, 'EXPLICIT_DRAUGHTING.' +
                                          'PRESENTATION_LAYER_ASSIGNMENT.' +
                                          'ASSIGNED_ITEMS')) >=1
                 )))=0
       )))=0;
END_RULE;
(*
```

Argument definitions:

geometric_set: the set of all instances of the **geometric_set** entity.

Formal propositions:

WR1: Every element of each **geometric_set** shall be one of the **assigned_items** of at least one **presentation_layer_assignment**.

5.2.3.23 global_length_and_angle_units

The **global_length_and_angle_units** rule ensures that units of length and plane angle are given for every **global_unit_assigned_context**.

EXPRESS specification:

```
*)
RULE global_length_and_angle_units FOR (global_unit_assigned_context);
WHERE
  WR1: SIZEOF (QUERY (guac <* global_unit_assigned_context |
       NOT (SIZEOF (guac.units)=2)))=0;
  WR2: SIZEOF (QUERY (guac <* global_unit_assigned_context |
       NOT ((SIZEOF (QUERY (x <* guac.units |
             'EXPLICIT_DRAUGHTING.LENGTH_UNIT' IN TYPEOF (x)))=1) AND
            (SIZEOF (QUERY (x <* guac.units |
             'EXPLICIT_DRAUGHTING.PLANE_ANGLE_UNIT' IN TYPEOF (x)))=1)
       )))=0;
END_RULE;
(*
```

Argument definitions:

global_unit_assigned_context: the set of all instances of the **global_unit_assigned_context** entity.

Formal propositions:

WR1: Each **global_unit_assigned_context** shall have exactly two elements in its set of units.

WR2: Each **global_unit_assigned_context** shall have exactly one **length_unit** and exactly one **plane_angle_unit** in its set of units.

5.2.3.24 group_usage_constraint

The **group_usage_constraint** rule ensures that every **group** is assigned to exactly one set of items as members of the **group.**

EXPRESS specification:

```
*)
RULE group_usage_constraint FOR (group, group_assignment);
WHERE
  WR1: SIZEOF (QUERY (g <* group |
       NOT (SIZEOF (QUERY (ga <* group_assignment |
       g :=: ga.assigned_group )) = 1 ))) = 0;
END_RULE;
(*
```

Argument definitions:

group: the set of all instances of the **group** entity.

Formal propositions:

WR1: For each instance of **group**, there shall be exactly one instance of **group_assignment** which contains the instance of **group** as its **assigned_group** attribute.

5.2.3.25 measure_with_unit_instantiation

The **measure_with_unit_instantiation** rule ensures that a **measure_with_unit** is not independently instantiated, and that every **measure_with_unit** is a **length_measure_with_unit** or a **plane_angle_measure_with_unit.**

EXPRESS specification:

```
*)
RULE measure_with_unit_instantiation FOR (measure_with_unit);
WHERE
  WR1: SIZEOF (QUERY (m <* measure_with_unit |
         NOT (SIZEOF (USEDIN (m, ''')) >= 1
       ))) = 0;
  WR2: SIZEOF (QUERY (m <* measure_with_unit |
         NOT (SIZEOF (TYPEOF (m) *
                      ['EXPLICIT_DRAUGHTING.LENGTH_MEASURE_WITH_UNIT',
                       'EXPLICIT_DRAUGHTING.PLANE_ANGLE_MEASURE_WITH_UNIT']) = 1
       ))) = 0;
END_RULE;
(*
```

Attribute definitions:

measure_with_unit: the set of all instances of the **measure_with_unit** entity.

Formal propositions:

WR1: Each **measure_with_unit** shall be used at least once.

WR2: Each **measure_with_unit** shall be a **length_measure_with_unit** or a **plane_angle_measure_with_unit**.

5.2.3.26 named_unit_instantiation

The **named_unit_instantiation** rule ensures that a **named_unit** is not independently instantiated, and that every **named_unit** is a **conversion_based_unit**, **length_unit**, **plane_angle_unit**, or **si_unit**.

EXPRESS specification:

```
*)
RULE named_unit_instantiation FOR (named_unit);
WHERE
  WR1: SIZEOF (QUERY (n <* named_unit |
         NOT (SIZEOF (USEDIN (n, '')) >= 1
       ))) = 0;
  WR2: SIZEOF (QUERY (n <* named_unit |
         NOT (SIZEOF (TYPEOF (n) *
                      ['EXPLICIT_DRAUGHTING.CONVERSION_BASED_UNIT',
                       'EXPLICIT_DRAUGHTING.LENGTH_UNIT',
                       'EXPLICIT_DRAUGHTING.PLANE_ANGLE_UNIT',
                       'EXPLICIT_DRAUGHTING.SI_UNIT']) = 2
       ))) = 0;
END_RULE;
(*
```

Attribute definitions:

named_unit: the set of all instances of the **named_unit** entity.

Formal propositions:

WR1: Each **named_unit** shall be used at least once.

WR2: Each **named_unit** shall be a **conversion_based_unit** or an **si_unit**, and a **length_unit** or a **plane_angle_unit**.

5.2.3.27 presentation_layer_assignment_constraint

The **presentation_layer_assignment_constraint** rule ensures that every element of a layer that is a **point** or **curve** is an element in a **geometric_set**, and that every element of a layer that is an **annotation_occurrence** is one of the items in an **annotation_subfigure_representation**, **draughting_model**, **drawing_sheet_revision**, or **presentation_view**.

EXPRESS specification:

```
*)
RULE presentation_layer_assignment_constraint FOR
  (presentation_layer_assignment);
WHERE
  WR1: SIZEOF (QUERY (pla <* presentation_layer_assignment |
       NOT (SIZEOF (QUERY (item <* pla.assigned_items |
       NOT (SIZEOF (TYPEOF (item) *
                   ['EXPLICIT_DRAUGHTING.POINT',
                    'EXPLICIT_DRAUGHTING.CURVE',
                    'EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE'])=1)))=0
       )))=0;
  WR2: SIZEOF (QUERY (pla <* presentation_layer_assignment |
       NOT (SIZEOF (QUERY (pnt <* QUERY (item <* pla.assigned_items |
            ('EXPLICIT_DRAUGHTING.POINT' IN TYPEOF (item))) |
          NOT (SIZEOF (USEDIN (pnt, 'EXPLICIT_DRAUGHTING.' +
                                    'GEOMETRIC_SET.ELEMENTS')) = 0)
            ))=0
       )))=0;
  WR3: SIZEOF (QUERY (pla <* presentation_layer_assignment |
       NOT (SIZEOF (QUERY (crv <* QUERY (item <* pla.assigned_items |
            ('EXPLICIT_DRAUGHTING.CURVE' IN TYPEOF (item))) |
          NOT (SIZEOF (USEDIN (crv, 'EXPLICIT_DRAUGHTING.' +
                                    'GEOMETRIC_SET.ELEMENTS')) = 0)
            ))=0
       )))=0;
  WR4: SIZEOF (QUERY (pla <* presentation_layer_assignment |
       NOT (SIZEOF (QUERY (ao <* QUERY (item <* pla.assigned_items |
              ('EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE'
                IN TYPEOF (item))) |
         NOT (SIZEOF (QUERY (ur <* using_representations(ao) |
           NOT (SIZEOF (TYPEOF(ur) *
              ['EXPLICIT_DRAUGHTING.' +
               'DRAUGHTING_SUBFIGURE_REPRESENTATION',
               'EXPLICIT_DRAUGHTING.DRAUGHTING_MODEL',
               'EXPLICIT_DRAUGHTING.DRAWING_SHEET_REVISION',
               'EXPLICIT_DRAUGHTING.PRESENTATION_VIEW']) = 1
            )))=0
         )))=0
       )))=0;
END_RULE;
(*
```

Argument definitions:

presentation_layer_assignment: the set of all instances of the **presentation_layer_assignment** entity.

Formal propositions:

WR1: Each of the **assigned_items** of every **presentation_layer_assignment** shall be a point, curve, or **annotation_occurrence**.

WR2: Every point that is one of the **assigned_items** of a **presentation_layer_assignment** shall also be a member of the set of elements of a **geometric_set**.

WR3: Every curve that is one of the **assigned_items** of a **presentation_layer_assignment** shall also be a member of the set of elements of a **geometric_set**.

WR4: Every **annotation_occurrence** that is one of the **assigned_items** of a **presentation_layer_assignment** shall also be a member of the set of items of an **annotation_subfigure_representation**, **draughting_model**, **drawing_sheet_revision**, or **presentation_view**.

5.2.3.28 presentation_layer_assignment_requires_usage

The **presentation_layer_assignment_requires_usage** rule specifies that each instance of **presentation_layer_assignment** shall be referenced by at least one **presentation_layer_usage**.

EXPRESS specification:

```
*)
RULE presentation_layer_assignment_requires_usage FOR
  (presentation_layer_assignment,
   presentation_layer_usage);
WHERE
  WR1: SIZEOF (QUERY (e <* presentation_layer_assignment |
       NOT (SIZEOF (QUERY (s <* presentation_layer_usage |
       e :=: s.assignment )) = 1 ))) = 0;
END_RULE;
(*
```

Attribute definitions:

presentation_layer_assignment: the set of all instances of the **presentation_layer_assignment** entity.

presentation_layer_usage: the set of all instances of the **presentation_layer_usage** entity.

Formal propositions:

WR1: For each instance of **presentation_layer_assignment**, there shall be exactly one instance of **presentation_layer_usage** which contains the instance of **presentation_layer_assignment** as its assignment attribute.

5.2.3.29 presentation_of_product_definition_formation

The **presentation_of_product_definition_formation** rule ensures that each **product_definition_formation** is presented in at least one **drawing_revision**, or has a defined shape, or both. This rule satisfies the requirement that a **product_definition_formation** is either depicted in a drawing, or, through a sub-model, participates in the definition of another **product_definition_formation** that is thus depicted.

EXPRESS specification:

```
*)
RULE presentation_of_product_definition_formation FOR
  (product_definition_formation);
WHERE
  WR1: SIZEOF (QUERY (pdf <* product_definition_formation |
       NOT (
       (SIZEOF (USEDIN (pdf, 'EXPLICIT_DRAUGHTING.' +
                             'DRAUGHTING_PRESENTED_ITEM.ITEMS'))>=1) OR

       (SIZEOF (QUERY (pd <* USEDIN (pdf, 'EXPLICIT_DRAUGHTING.' +
                                          'PRODUCT_DEFINITION.FORMATION') |
       NOT (
       SIZEOF (QUERY (pds <* QUERY (prop_def <*
                             USEDIN (pd, 'EXPLICIT_DRAUGHTING.' +
                                         'PROPERTY_DEFINITION.DEFINITION') |
                             'EXPLICIT_DRAUGHTING.PRODUCT_DEFINITION_SHAPE'
                             IN TYPEOF (prop_def)) |
       NOT (
       SIZEOF (QUERY (pdr <* USEDIN (pds, 'EXPLICIT_DRAUGHTING.' +
                                     'PROPERTY_DEFINITION_REPRESENTATION.' +
                                        'DEFINITION') |
              'EXPLICIT_DRAUGHTING.SHAPE_DEFINITION_REPRESENTATION'
              IN TYPEOF (pdr))) >= 1
       ))) = 0
       ))) = 0
       )
       ))) = 0;
END_RULE;
(*
```

Argument definitions:

product_definition_formation: the set of instances of the **product_definition_formation** entity.

Formal propositions:

WR1: Each **product_definition_formation** shall be referenced as the item of at least one **draughting_presented_item**, or shall be referenced by at least one **product_definition** that is referenced by at least one **product_definition_shape** that is referenced by at least one **property_definition_representation** that is referenced by at least one **property_definition_**

representation.

5.2.3.30 presentation_size_constraint

The **presentation_size_constraint** rule ensures that each **presentation_size** is used only to specify the size of a **drawing_sheet_revision.**

EXPRESS specification:

```
*)
RULE presentation_size_constraint FOR (presentation_size);
WHERE
  WR1: SIZEOF (QUERY ( ps <* presentation_size |
                NOT ('EXPLICIT_DRAUGHTING.' +
                     'DRAWING_SHEET_REVISION' IN
                     TYPEOF (ps.unit))))=0;
END_RULE;
(*
```

Argument definitions:

presentation_size: the set of all instances of the **presentation_size** entity.

Formal propositions:

WR1: Each **presentation_size** shall specify a **drawing_sheet_revision** as the unit.

5.2.3.31 presentation_style_by_context_constraint

The **presentation_style_by_context_constraint** rule specifies that only **presentation_view** provides a context for the assignment of style.

EXPRESS specification:

```
*)
RULE presentation_style_by_context_constraint FOR
  (presentation_style_by_context);
WHERE
  WR1: SIZEOF (QUERY (psbc <* presentation_style_by_context |
       NOT ('EXPLICIT_DRAUGHTING.PRESENTATION_VIEW'
            IN TYPEOF (psbc.style_context)
       )))=0;
END_RULE;
(*
```

Argument definitions:

presentation_style_by_context: the set of all instances of the **presentation_style_by_context** entity.

Formal propositions:

WR1: The **style_context** of each **presentation_style_by_context** shall be **presentation_view**.

5.2.3.32 presentation_style_by_context_instantiation

The **presentation_style_by_context_instantiation** rule ensures that a **presentation_style_by_context** is not independently instantiated.

EXPRESS specification:

```
*)
RULE presentation_style_by_context_instantiation FOR
  (presentation_style_by_context);
WHERE
  WR1: SIZEOF (QUERY (each <* presentation_style_by_context |
         NOT (SIZEOF (USEDIN (each, '')) >= 1
       ))) = 0;
END_RULE;
(*
```

Argument definitions:

presentation_style_by_context: the set of all instances of the **presentation_style_by_context** entity.

Formal propositions:

WR1: Each **presentation_style_by_context** shall be used at least once.

5.2.3.33 presentation_view_presented_once

The **presentation_view_presented_once** rule ensures that a **presentation_view** is presented in exactly one **drawing_sheet_revision.**

EXPRESS specification:

```
*)
RULE presentation_view_presented_once FOR
  (presentation_view,
   drawing_sheet_revision);
WHERE
  WR1: SIZEOF (QUERY (pv <* presentation_view |
       NOT (
       SIZEOF (QUERY (rm <* USEDIN (pv, 'EXPLICIT_DRAUGHTING.' +
                                        'REPRESENTATION_MAP.' +
                                        'MAPPED_REPRESENTATION') |
       NOT (
```

```
        SIZEOF (QUERY (mi <* rm.map_usage |
        NOT (
        SIZEOF (QUERY (dsr <* drawing_sheet_revision |
        (mi IN dsr\representation.items))) = 1
        ))) = 0
        ))) = 0
        ))) = 0;
END_RULE;
(*
```

Argument definitions:

presentation_view: the set of all instances of the **presentation_view** entity.

drawing_sheet_revision: the set of all instances of the **drawing_sheet_revision** entity.

Formal propositions:

WR1: Each **presentation_view** shall be mapped as an item in exactly one **drawing_sheet_revision.**

5.2.3.34 pre_defined_colour_instantiation

The **pre_defined_colour_instantiation** rule ensures that a **pre_defined_colour** is not independently instantiated.

EXPRESS specification:

```
*)
RULE pre_defined_colour_instantiation FOR (pre_defined_colour);
WHERE
  WR1: SIZEOF (QUERY (each <* pre_defined_colour |
         NOT (SIZEOF (USEDIN (each, ''')) >= 1
       ))) = 0;
END_RULE;
(*
```

Argument definitions:

pre_defined_colour: the set of all instances of the **pre_defined_colour** entity.

Formal propositions:

WR1: Each **pre_defined_colour** shall be used at least once.

5.2.3.35 pre_defined_curve_font_instantiation

The **pre_defined_curve_font_instantiation** rule ensures that a **pre_defined_curve_font** is not independently instantiated.

EXPRESS specification:

```
*)
RULE pre_defined_curve_font_instantiation FOR (pre_defined_curve_font);
WHERE
  WR1: SIZEOF (QUERY (each <* pre_defined_curve_font |
         NOT (SIZEOF (USEDIN (each, ''))) >= 1
       ))) = 0;
END_RULE;
(*
```

Argument definitions:

pre_defined_curve_font: the set of all instances of the **pre_defined_curve_font** entity.

Formal propositions:

WR1: Each **pre_defined_curve_font** shall be used at least once.

5.2.3.36 pre_defined_symbol_instantiation

The **pre_defined_symbol_instantiation** rule ensures that a **pre_defined_symbol** is not independently instantiated.

EXPRESS specification:

```
*)
RULE pre_defined_symbol_instantiation FOR (pre_defined_symbol);
WHERE
  WR1: SIZEOF (QUERY (each <* pre_defined_symbol |
         NOT (SIZEOF (USEDIN (each, ''))) >= 1
       ))) = 0;
END_RULE;
(*
```

Argument definitions:

pre_defined_symbol: the set of all instances of the **pre_defined_symbol** entity.

Formal propositions:

WR1: Each **pre_defined_symbol** shall be used at least once.

5.2.3.37 pre_defined_text_font_instantiation

The **pre_defined_text_font_instantiation** rule ensures that a **pre_defined_text_font** is not independently instantiated.

EXPRESS specification:

```
*)
RULE pre_defined_text_font_instantiation FOR (pre_defined_text_font);
WHERE
  WR1: SIZEOF (QUERY (each <* pre_defined_text_font |
         NOT (SIZEOF (USEDIN (each, ''))) >= 1
       ))) = 0;
END_RULE;
(*
```

Argument definitions:

pre_defined_text_font: the set of all instances of the **pre_defined_text_font** entity.

Formal propositions:

WR1: Each **pre_defined_text_font** shall be used at least once.

5.2.3.38 product_definition_instantiation

The **product_definition_instantiation** rule ensures that a **product_definition** is not independently instantiated.

EXPRESS specification:

```
*)
RULE product_definition_instantiation FOR (product_definition);
WHERE
  WR1: SIZEOF (QUERY (each <* product_definition |
         NOT (SIZEOF (USEDIN (each, ''))) >= 1
       ))) = 0;
END_RULE;
(*
```

Argument definitions:

product_definition: the set of all instances of the **product_definition** entity.

Formal propositions:

WR1: Each **product_definition** shall be used at least once.

5.2.3.39 product_requires_version

The **product_requires_version** rule ensures that each instance of **product** is referenced by at least one instance of **product_definition_formation**. This rule enforces the requirement for every **product** to have one or more versions.

EXPRESS specification:

```
*)
RULE product_requires_version FOR
  (product,
   product_definition_formation);
WHERE
  WR1: SIZEOF (QUERY (e <* product |
       NOT (SIZEOF (QUERY (s <* product_definition_formation |
       e :=: s.of_product )) >= 1 ))) = 0;
END_RULE;
(*
```

Argument definitions:

product: the set of all instances of the **product** entity.

product_definition_formation: the set of all instances of the **product_definition_formation** entity.

Formal propositions:

WR1: For each instance of **product**, there shall be one or more instances of **product_definition_formation** that contains an **of_product** attribute equal to that instance of **product**.

5.2.3.40 product_shape_has_representations

The **product_shape_has_representations** rule specifies that each **product_definition_shape** is associated with one or more representations of the identified shape.

EXPRESS specification:

```
*)
RULE product_shape_has_representations FOR
  (product_definition_shape);
WHERE
  WR1: SIZEOF (QUERY (pds <* product_definition_shape |
       NOT (SIZEOF (USEDIN (pds, 'EXPLICIT_DRAUGHTING.' +
                                 'PROPERTY_DEFINITION_REPRESENTATION.' +
                                 'DEFINITION')) >= 1
       ))) = 0;
END_RULE;
(*
```

Attribute definitions:

product_definition_shape: the set of all instances of the **product_definition_shape** entity.

Formal propositions:

WR1: Each **product_definition_shape** shall be the definition for at least one **property_definition_representation**.

5.2.3.41 representation_context_global_units

The **representation_context_global_units** rule ensures that every **representation_context** is a **global_unit_assigned_context**.

EXPRESS specification:

```
*)
RULE representation_context_global_units FOR (representation_context);
WHERE
  WR1: SIZEOF (QUERY (rc <* representation_context |
       NOT ('EXPLICIT_DRAUGHTING.' +
            'GLOBAL_UNIT_ASSIGNED_CONTEXT' IN TYPEOF (rc)
       )))=0;
END_RULE;
(*
```

Argument definitions:

representation_context: the set of all instances of the **representation_context** entity.

Formal propositions:

WR1: Each **representation_context** shall be a **global_unit_assigned_context**.

5.2.3.42 representation_instantiation

The **representation_instantiation** rule ensures that a **representation** is not independently instantiated.

EXPRESS specification:

```
*)
RULE representation_instantiation FOR (representation);
WHERE
  WR1: SIZEOF (QUERY (each <* representation |
         NOT (SIZEOF (USEDIN (each, '')) >= 1
       ))) = 0;
END_RULE;
(*
```

Argument definitions:

representation: the set of all instances of the **representation** entity.

Formal propositions:

WR1: Each **representation** shall be used at least once.

5.2.3.43 sheets_belong_to_one_drawing

The **sheets_belong_to_one_drawing** rule ensures that each **drawing_sheet_revision** belongs to exactly one **drawing_revision**.

EXPRESS specification:

```
*)
RULE sheets_belong_to_one_drawing FOR
  (drawing_sheet_revision,
   drawing_sheet_revision_usage);
WHERE
  WR1: SIZEOF (QUERY (dsr <* drawing_sheet_revision |
       NOT (SIZEOF (QUERY (usage <* drawing_sheet_revision_usage |
       dsr :=: usage.area )) = 1 ))) = 0;
END_RULE;
(*
```

Argument definitions:

drawing_sheet_revision: the set of all instances of the **drawing_sheet_revision** entity.

drawing_sheet_revision_usage: the set of all instances of the **drawing_sheet_revision_usage** entity.

Formal propositions:

WR1: For each instance of **drawing_sheet_revision**, there shall be exactly one instance of **drawing_sheet_revision_usage** which contains the instance of **drawing_sheet_revision** as its **area** attribute.

5.2.3.44 styled_curve

The **styled_curve** rule ensures that a **styled_item** that gives the style for a **curve** specifies a single **curve_style**.

EXPRESS specification:

```
*)
RULE styled_curve FOR (styled_item);
WHERE
  WR1: SIZEOF (QUERY (crv <* QUERY (si <* styled_item |
         ('EXPLICIT_DRAUGHTING.CURVE' IN TYPEOF (si.item))) |
         NOT (SIZEOF (QUERY (sty <* crv.styles |
           NOT ((SIZEOF (sty.styles)=1) AND
           ('EXPLICIT_DRAUGHTING.CURVE_STYLE' IN TYPEOF (sty.styles[1]))
         )))=0
       )))=0;
END_RULE;
(*
```

Argument definitions:

styled_item: the set of all instances of the **styled_item** entity.

Formal propositions:

WR1: Every **styled_item** that provides the style for a **curve** shall have exactly one style that is a **curve_style**.

5.2.3.45 subtype_mandatory_annotation_occurrence

The **subtype_mandatory_annotation_occurrence** rule ensures that all **annotation_occurrence** entities are **draughting_annotation_occurrence** entities.

EXPRESS specification:

```
*)
RULE subtype_mandatory_annotation_occurrence FOR (annotation_occurrence);
WHERE
  WR1: SIZEOF (QUERY (ao <* annotation_occurrence |
       NOT ('EXPLICIT_DRAUGHTING.DRAUGHTING_ANNOTATION_OCCURRENCE'
              IN TYPEOF (ao))
       ))=0;
END_RULE;
(*
```

Argument definitions:

annotation_occurrence: the set of all instances of the **annotation_occurrence** entity.

Formal propositions:

WR1: Each **annotation_occurrence** shall be a **draughting_annotation_occurrence**.

5.2.3.46 subtype_mandatory_camera_image

The **subtype_mandatory_camera_image** rule ensures that all **camera_image** entities are **camera_image_2d_with_scale** entities.

EXPRESS specification:

```
*)
RULE subtype_mandatory_camera_image FOR (camera_image);
WHERE
  WR1: SIZEOF (QUERY (ci <* camera_image |
       NOT ('EXPLICIT_DRAUGHTING.CAMERA_IMAGE_2D_WITH_SCALE'
            IN TYPEOF (ci)
       )))=0;
END_RULE;
(*
```

Argument definitions:

camera_image: the set of all instances of the **camera_image** entity.

Formal propositions:

WR1: Each **camera_image** shall be a **camera_image_2d_with_scale.**

5.2.3.47 subtype_mandatory_draughting_callout

The **subtype_mandatory_draughting_callout** rule ensures that all **draughting_callout** entities are **draughting_elements** entities.

EXPRESS specification:

```
*)
RULE subtype_mandatory_draughting_callout FOR (draughting_callout);
WHERE
  WR1: SIZEOF (QUERY (dc <* draughting_callout |
       NOT ('EXPLICIT_DRAUGHTING.DRAUGHTING_ELEMENTS' IN
              TYPEOF (dc))
       ))=0;
END_RULE;
(*
```

Argument definitions:

draughting_callout: the set of all instances of the **draughting_callout** entity.

Formal propositions:

WR1: Each **draughting_callout** shall be a **draughting_elements.**

5.2.3.48 subtype_mandatory_drawing_revision

The **subtype_mandatory_drawing_revision** rule ensures that all **drawing_revision** entities are **draughting_drawing_revision** entities.

EXPRESS specification:

```
*)
RULE subtype_mandatory_drawing_revision FOR (drawing_revision);
WHERE
  WR1: SIZEOF (QUERY (dr <* drawing_revision |
       NOT ('EXPLICIT_DRAUGHTING.DRAUGHTING_DRAWING_REVISION'
              IN TYPEOF (dr))
       ))=0;
END_RULE;
(*
```

Attribute definitions:

drawing_revision: the set of all instances of the **drawing_revision** entity.

Formal propositions:

WR1: Each **drawing_revision** shall be a **draughting_drawing_revision.**

5.2.3.49 subtype_mandatory_pre_defined_colour

The **subtype_mandatory_pre_defined_colour** rule ensures that all **pre_defined_colour** entities are **draughting_pre_defined_colour** entities.

EXPRESS specification:

```
*)
RULE subtype_mandatory_pre_defined_colour FOR (pre_defined_colour);
WHERE
  WR1: SIZEOF (QUERY (pdc <* pre_defined_colour |
       NOT ('EXPLICIT_DRAUGHTING.DRAUGHTING_PRE_DEFINED_COLOUR'
              IN TYPEOF (pdc))))=0;
END_RULE;
(*
```

Argument definitions:

pre_defined_colour: the set of all instances of the **pre_defined_colour** entity.

Formal propositions:

WR1: Each **pre_defined_colour** shall be a **draughting_pre_defined_colour.**

5.2.3.50 subtype_mandatory_pre_defined_curve_font

The **subtype_mandatory_pre_defined_curve_font** rule ensures that all **pre_defined_curve_font** entities are **draughting_pre_defined_curve_font** entities.

EXPRESS specification:

```
*)
RULE subtype_mandatory_pre_defined_curve_font FOR (pre_defined_curve_font);
WHERE
  WR1: SIZEOF (QUERY (pdcf <* pre_defined_curve_font |
       NOT ('EXPLICIT_DRAUGHTING.DRAUGHTING_PRE_DEFINED_CURVE_FONT'
              IN TYPEOF (pdcf))))=0;
END_RULE;
(*
```

Argument definitions:

pre_defined_curve_font: the set of all instances of the **pre_defined_curve_font** entity.

Formal propositions:

WR1: Each **pre_defined_curve_font** shall be a **draughting_pre_defined_curve_font**.

5.2.3.51 subtype_mandatory_pre_defined_symbol

The **subtype_mandatory_pre_defined_symbol** rule ensures that all **pre_defined_symbol** entities are **pre_defined_geometrical_tolerance_symbol**, **pre_defined_dimension_symbol**, **pre_defined_point_marker_symbol**, or **pre_defined_terminator_symbol** entities.

EXPRESS specification:

```
*)
RULE subtype_mandatory_pre_defined_symbol FOR (pre_defined_symbol);
WHERE
  WR1: SIZEOF (QUERY (pds <* pre_defined_symbol |
       NOT (SIZEOF (TYPEOF (pds) *
                   ['EXPLICIT_DRAUGHTING.' +
                    'PRE_DEFINED_GEOMETRICAL_TOLERANCE_SYMBOL',
                    'EXPLICIT_DRAUGHTING.' +
                    'PRE_DEFINED_DIMENSION_SYMBOL',
                    'EXPLICIT_DRAUGHTING.' +
                    'PRE_DEFINED_POINT_MARKER_SYMBOL',
                    'EXPLICIT_DRAUGHTING.' +
                    'PRE_DEFINED_TERMINATOR_SYMBOL']) =1
        ))) = 0;
END_RULE;
(*
```

Argument definitions:

pre_defined_symbol: the set of all instances of the **pre_defined_symbol** entity.

Formal propositions:

WR1: Each **pre_defined_symbol** shall be either a **pre_defined_geometrical_tolerance_symbol**, **pre_defined_dimension_symbol**, **pre_defined_point_marker_symbol**, or **pre_defined_terminator_symbol**.

5.2.3.52 subtype_mandatory_pre_defined_text_font

The **subtype_mandatory_pre_defined_text_font** rule ensures that all **pre_defined_text_font** entities are **draughting_pre_defined_text_font** entities.

EXPRESS specification:

```
*)
```

```
RULE subtype_mandatory_pre_defined_text_font FOR (pre_defined_text_font);
WHERE
  WR1: SIZEOF (QUERY (pdtf <* pre_defined_text_font |
       NOT ('EXPLICIT_DRAUGHTING.DRAUGHTING_PRE_DEFINED_TEXT_FONT'
              IN TYPEOF (pdtf))))=0;
END_RULE;
(*
```

Argument definitions:

pre_defined_text_font: the set of all instances of the **pre_defined_text_font** entity.

Formal propositions:

WR1: Each **pre_defined_text_font** shall be a **draughting_pre_defined_text_font.**

5.2.3.53 subtype_mandatory_shape_representation

The **subtype_mandatory_shape_representation** rule ensures that all **shape_representation** entities are a **geometrically_bounded_2d_wireframe_representation** entities.

EXPRESS specification:

```
*)
RULE subtype_mandatory_shape_representation FOR
       (shape_representation);
WHERE
  WR1: SIZEOF (QUERY (sr <* shape_representation |
       NOT (('EXPLICIT_DRAUGHTING.' +
            'GEOMETRICALLY_BOUNDED_2D_WIREFRAME_REPRESENTATION')
            IN TYPEOF (sr)
       )))=0;
END_RULE;
(*
```

Argument definitions:

shape_representation: the set of all instances of the **shape_representation** entity.

Formal propositions:

WR1: Each **shape_representation** shall be a **geometrically_bounded_2d_wireframe_representation.**

5.2.3.54 symbol_colour_instantiation

The **symbol_colour_instantiation** rule ensures that a **symbol_colour** is not independently instantiated.

EXPRESS specification:

```
*)
RULE symbol_colour_instantiation FOR (symbol_colour);
WHERE
  WR1: SIZEOF (QUERY (each <* symbol_colour |
         NOT (SIZEOF (USEDIN (each, '')) >= 1
       ))) = 0;
END_RULE;
(*
```

Argument definitions:

symbol_colour: the set of all instances of the **symbol_colour** entity.

Formal propositions:

WR1: Each **symbol_colour** shall be used at least once.

5.2.3.55 terminator_symbol_constraint

The **terminator_symbol_constraint** rule ensures that a **defined_symbol** that is used as the definition of a **terminator_symbol** is described by a **pre_defined_terminator_symbol.**

EXPRESS specification:

```
*)
RULE terminator_symbol_constraint FOR (terminator_symbol);
WHERE
  WR1: SIZEOF (QUERY (ts <* terminator_symbol |
         (('EXPLICIT_DRAUGHTING.DEFINED_SYMBOL' IN TYPEOF(ts.item))
         AND
         NOT ('EXPLICIT_DRAUGHTING.PRE_DEFINED_TERMINATOR_SYMBOL' IN
                TYPEOF (ts.item\defined_symbol.definition)))
       )) = 0;
END_RULE;
(*
```

Argument definitions:

terminator_symbol: the set of all instances of the **terminator_symbol** entity.

Formal propositions:

WR1: Each **terminator_symbol** that has an implicit definition shall be defined by a **pre_defined_terminator_symbol.**

5.2.3.56 text_font_usage

The **text_font_usage** rule ensures that every **pre_defined_text_font** and **externally_defined_text_font** is used in the presentation of a **text_literal.**

EXPRESS specification:

```
*)
RULE text_font_usage FOR
  (pre_defined_text_font,
   externally_defined_text_font);
WHERE
  WR1: SIZEOF (QUERY (pdtf <* pre_defined_text_font |
       NOT (SIZEOF (USEDIN (pdtf, 'EXPLICIT_DRAUGHTING.' +
                                  'TEXT_LITERAL.FONT'))>=1
       ))=0;
  WR2: SIZEOF (QUERY (edtf <* externally_defined_text_font |
       NOT (SIZEOF (USEDIN (edtf, 'EXPLICIT_DRAUGHTING.' +
                                  'TEXT_LITERAL.FONT'))>=1
       ))=0;
END_RULE;
(*
```

Argument definitions:

pre_defined_text_font: the set of all instances of the **pre_defined_text_font** entity.

externally_defined_text_font: the set of all instances of the **externally_defined_text_font** entity.

Formal propositions:

WR1: Each **pre_defined_text_font** shall be the font for at least one **text_literal**.

WR2: Each **externally_defined_text_font** shall be the font for at least one **text_literal**.

5.2.3.57 text_style_for_defined_font_instantiation

The **text_style_for_defined_font_instantiation** rule ensures that a **text_style_for_defined_font** is not independently instantiated.

EXPRESS specification:

```
*)
RULE text_style_for_defined_font_instantiation FOR
  (text_style_for_defined_font);
WHERE
  WR1: SIZEOF (QUERY (each <* text_style_for_defined_font |
         NOT (SIZEOF (USEDIN (each, ''')) >= 1
       ))) = 0;
END_RULE;
(*
```

Argument definitions:

text_style_for_defined_font: the set of all instances of the **text_style_for_defined_font** entity.

Formal propositions:

WR1: Each **text_style_for_defined_font** shall be used at least once.

5.2.3.58 text_style_instantiation

The **text_style_instantiation** rule ensures that a **text_style** is not independently instantiated.

EXPRESS specification:

```
*)
RULE text_style_instantiation FOR (text_style);
WHERE
  WR1: SIZEOF (QUERY (each <* text_style |
         NOT (SIZEOF (USEDIN (each, '')) >= 1
       ))) = 0;
END_RULE;
(*
```

Argument definitions:

text_style: the set of all instances of the **text_style** entity.

Formal propositions:

WR1: Each **text_style** shall be used at least once.

5.2.3.59 use_of_geometrically_bounded_2d_wireframe_representation

The **use_of_geometrically_bounded_2d_wireframe_representation** rule specifies that each **geometrically_bounded_2d_wireframe_representation** shall be mapped into at least one other **representation**, and that the only **representation**s into which a **geometrically_bounded_2d_wireframe_representation** is mapped shall be **draughting_model** or another **geometrically_bounded_2d_wireframe_representation.**

EXPRESS specification:

```
*)
RULE use_of_geometrically_bounded_2d_wireframe_representation FOR
  (geometrically_bounded_2d_wireframe_representation,
   mapped_item);
WHERE
  WR1: SIZEOF (QUERY (gbwr <*
                      geometrically_bounded_2d_wireframe_representation |
       NOT (
```

```
      SIZEOF (QUERY (mi <* mapped_item |
      mi.mapping_source.mapped_representation :=: gbwr)) >= 1
      ))) = 0;
  WR2: SIZEOF (QUERY (gbwr_map <* QUERY (mi <* mapped_item |
              ('EXPLICIT_DRAUGHTING.' +
               'GEOMETRICALLY_BOUNDED_2D_WIREFRAME_REPRESENTATION') IN
              TYPEOF (mi.mapping_source.mapped_representation)) |
      NOT (
      SIZEOF (QUERY (rep <* USEDIN (gbwr_map, 'EXPLICIT_DRAUGHTING.' +
                                              'REPRESENTATION.ITEMS') |
      NOT (SIZEOF (['EXPLICIT_DRAUGHTING.DRAUGHTING_MODEL',
                    'EXPLICIT_DRAUGHTING.' +
                    'GEOMETRICALLY_BOUNDED_2D_WIREFRAME_REPRESENTATION'] *
                   TYPEOF (rep)) = 1
      ))) = 0
      ))) = 0;
END_RULE;
(*
```

Argument definitions:

geometrically_bounded_2d_wireframe_representation: the set of all instances of the **geometrically_bounded_2d_wireframe_representation** entity.

mapped_item: the set of all instances of the **mapped_item** entity.

Formal propositions:

WR1: For each instance of **geometrically_bounded_2d_wireframe_representation**, there shall be at least one instance of **mapped_item** which contains the instance of **geometrically_bounded_2d_wireframe_representation** as its **mapping_source.mapped_representation** attribute.

WR2: Each **mapped_item** that is the mapping of a **geometrically_bounded_2d_wireframe_representation** shall be an item in a **draughting_model** or a **geometrically_bounded_2d_wireframe_representation.**

5.2.4 explicit_draughting function definitions

5.2.4.1 acyclic_mapped_item_usage

The **acyclic_mapped_item_usage** boolean function returns true if the given **representation** contains one or more **mapped_items** that are a mapping of the **representation** itself.

EXPRESS specification:

```
*)
FUNCTION acyclic_mapped_item_usage (rep: representation) : BOOLEAN;
-- returns TRUE if the representation contains one or more mapped_items
```

```
-- that are a mapping of the representation itself
  LOCAL
    items : SET OF representation_item;
    i     : INTEGER;
  END_LOCAL;

  items := QUERY (item <* rep.items |
           'DRAUGHTING_ANNOTATION_AIC.MAPPED_ITEM' IN TYPEOF (item));
  IF SIZEOF (items) = 0
  THEN RETURN (FALSE);
  ELSE REPEAT i := 1 TO HIINDEX (items);
    IF items[i]\mapped_item.mapping_source.mapped_representation :=: rep
    THEN RETURN (TRUE);
    ELSE RETURN (acyclic_mapped_item_usage

(items[i]\mapped_item.mapping_source.mapped_representation));
    END_IF;
    END_REPEAT;
  RETURN (FALSE);
  END_IF;
END_FUNCTION;
(*
```

Argument definitions:

rep: the input **representation** to be checked.

5.2.4.2 aspect_ratio

The **aspect_ratio** function returns a **positive_ratio_measure** that is the ratio of length to height for a given **planar_box**.

EXPRESS specification:

```
*)
FUNCTION aspect_ratio (p : planar_box) : positive_ratio_measure;
  RETURN (p.size_in_x / p.size_in_y);
END_FUNCTION;
(*
```

Argument definitions:

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

5.2.4.3 bag_to_set

The **bag_to_set** function converts BAGs into SETs.

EXAMPLE 19 – It can be used to convert the BAGs returned by the USEDIN function into SETs that can be properly assigned to variables that are SETs.

EXPRESS specification:

```
*)
FUNCTION bag_to_set (the_bag : BAG OF GENERIC : intype) :
                       SET OF GENERIC : intype;

  LOCAL
    the_set: SET OF GENERIC : intype := [];
    i      : INTEGER;
  END_LOCAL;

  IF SIZEOF (the_bag) > 0 THEN
    REPEAT i := 1 to HIINDEX (the_bag);
      the_set := the_set + the_bag [i];
    END_REPEAT;
  END_IF;

  RETURN (the_set);

END_FUNCTION;
(*
```

Argument definitions:

the_bag: the input **BAG** that is to be converted into a **SET**.

5.2.4.4 check_text_alignment

The **check_text_alignment** boolean function returns true if the given **composite_text** has a common alignment value for each **text_literal** contained in the set of **collected_text**.

EXPRESS specification:

```
*)
FUNCTION check_text_alignment (ct : composite_text) : BOOLEAN;

  LOCAL
    a : SET OF text_alignment := [];
    i : INTEGER;
  END_LOCAL;

  -- create a set of all the alignments
  REPEAT i := 1 TO HIINDEX (ct.collected_text);
    a := a + [ct.collected_text[i].alignment];
  END_REPEAT;

  -- if there is more than one element in the set
  -- then not all alignments were the same
  RETURN (SIZEOF (a)=1);
END_FUNCTION;
(*
```

Argument definitions:

ct: the input **composite_text** to be checked.

5.2.4.5 check_text_font

The **check_text_font** boolean function returns true if the given **composite_text** has a common font value for each **text_literal** contained in the set of **collected_text**.

EXPRESS specification:

```
*)
FUNCTION check_text_font (ct : composite_text) : BOOLEAN;

  LOCAL
    f : SET OF font_select := [];
    i : INTEGER;
  END_LOCAL;

  -- build a set of all the fonts
  REPEAT i := 1 TO HIINDEX (ct.collected_text);
    f := f + [ct.collected_text[i].font];
  END_REPEAT;

  -- if there is more than one element in the set
  -- then not all fonts were the same
  RETURN (SIZEOF (f) <= 1);
END_FUNCTION;
(*
```

Argument definitions:

ct: the input **composite_text** to be checked.

5.2.4.6 using_representations

The **using_representations** function returns the set of representations in which a **representation_item** is used.

A **representation_item** is used in a **representation** if it is:

a) referenced in the set of items of the **representation**; or

b) referenced by a **representation_item** used in the **representation**.

NOTE – The second condition is a recursive check allowing for a **representation_item** to be used in a **representation** by being part of a tree of related **representation_items**. The tree is rooted in an entity used in a **representation** by virtue of compliance with the first condition.

EXPRESS specification:

```
*)
FUNCTION using_representations(item: representation_item) :
                SET OF representation;
  LOCAL
    results            : SET OF representation;
    result_bag         : BAG OF representation;
    intermediate_items : SET OF representation_item;
    i                  : INTEGER;
  END_LOCAL;

  -- Find the representations in which the item is used and add to the
  -- results set.

  result_bag := USEDIN(item,'REPRESENTATION_SCHEMA.REPRESENTATION.ITEMS');

  IF SIZEOF(result_bag) > 0 THEN
    REPEAT i := 1 TO HIINDEX(result_bag);
      results := results + result_bag[i];
    END_REPEAT;
  END_IF;

  -- Find the set of representation_items in which item is used.

  intermediate_items := QUERY(z <* bag_to_set( USEDIN(item , '')) |
    'REPRESENTATION_SCHEMA.REPRESENTATION_ITEM' IN TYPEOF(z));

  -- If the set of intermediate items is not empty;

  IF SIZEOF(intermediate_items) > 0 THEN

    -- For each element in the set, recursively add the
    -- using_representations of that element.

    REPEAT i := 1 TO HIINDEX(intermediate_items);
      results := results + using_representations(intermediate_items[i]);
    END_REPEAT;
  END_IF;

  -- Return the set of representation in which the input item is
  -- used directly and indirectly (through intervening representation_items).

  RETURN (results);
END_FUNCTION;
(*
```

Argument definitions:

item: (input) the **representation_item** for which using **representations** are determined.

EXPRESS specification:

```
*)
END_SCHEMA;
(*
```

6 Conformance requirements

Conformance to this part of ISO 10303 includes satisfying the requirements stated in this part, the requirements of the implementation method(s) supported, and the relevant requirements of the normative references.

An implementation shall support at least one of the following implementation methods:

- ISO 10303-21.

Requirements with respect to implementation are specified in annex C.

The Protocol Information Conformance Statement (PICS) proforma lists any conformance classes that may be used in claims of conformance and subsequent testing. The PICS proforma is provided in annex D.

NOTE – ISO 10303-1201[2)] defines the abstract test suite to be used in the assessment of conformance. ISO 10303-32[2)] describes the conformance assessment process.

[2)] To be published.

Annex A
(normative)

AIM EXPRESS expanded listing

The following EXPRESS is the expanded form of the short form schema given in 5.2. In the event of any discrepancy between the short form and this expanded listing, the expanded listing shall be used.

EXPRESS specification:

```
*)
SCHEMA explicit_draughting;

TYPE approved_item = SELECT
  (drawing_revision,
   drawing_sheet_revision);
END_TYPE; -- approved_item

TYPE area_or_view = SELECT
  (presentation_area,
   presentation_view);
END_TYPE; -- area_or_view

TYPE axis2_placement = SELECT
  (axis2_placement_2d);
END_TYPE; -- axis2_placement

TYPE b_spline_curve_form = ENUMERATION OF
  (elliptic_arc,
   polyline_form,
   parabolic_arc,
   circular_arc,
   unspecified,
   hyperbolic_arc);
END_TYPE; -- b_spline_curve_form

TYPE box_characteristic_select = SELECT
  (box_height,
   box_width,
   box_slant_angle,
   box_rotate_angle);
END_TYPE; -- box_characteristic_select

TYPE box_height = positive_ratio_measure;
END_TYPE; -- box_height

TYPE box_rotate_angle = plane_angle_measure;
END_TYPE; -- box_rotate_angle
```

```
TYPE box_slant_angle = plane_angle_measure;
END_TYPE; -- box_slant_angle

TYPE box_width = positive_ratio_measure;
END_TYPE; -- box_width

TYPE character_spacing_select = SELECT
  (length_measure,
   ratio_measure,
   measure_with_unit);
END_TYPE; -- character_spacing_select

TYPE character_style_select = SELECT
  (text_style_for_defined_font);
END_TYPE; -- character_style_select

TYPE characterized_definition = SELECT
  (characterized_product_definition,
   shape_definition);
END_TYPE; -- characterized_definition

TYPE characterized_product_definition = SELECT
  (product_definition);
END_TYPE; -- characterized_product_definition

TYPE classified_item = SELECT
  (drawing_revision,
   drawing_sheet_revision);
END_TYPE; -- classified_item

TYPE contracted_item = SELECT
  (drawing_revision);
END_TYPE; -- contracted_item

TYPE curve_font_or_scaled_curve_font_select = SELECT
  (curve_style_font_select);
END_TYPE; -- curve_font_or_scaled_curve_font_select

TYPE curve_or_annotation_curve_occurrence = SELECT
  (curve,
   annotation_curve_occurrence);
END_TYPE; -- curve_or_annotation_curve_occurrence

TYPE curve_or_render = SELECT
  (curve_style);
END_TYPE; -- curve_or_render

TYPE curve_style_font_select = SELECT
  (curve_style_font,
   pre_defined_curve_font,
   externally_defined_curve_font);
END_TYPE; -- curve_style_font_select
```

```
TYPE date_time_select = SELECT
  (date);
END_TYPE; -- date_time_select

TYPE day_in_month_number = INTEGER;
END_TYPE; -- day_in_month_number

TYPE defined_symbol_select = SELECT
  (pre_defined_symbol,
   externally_defined_symbol);
END_TYPE; -- defined_symbol_select

TYPE dimension_count = INTEGER;
WHERE
  wr1: SELF > 0;
END_TYPE; -- dimension_count

TYPE dimension_extent_usage = ENUMERATION OF
  (origin,
   target);
END_TYPE; -- dimension_extent_usage

TYPE draughting_callout_element = SELECT
  (annotation_text_occurrence,
   annotation_symbol_occurrence,
   annotation_curve_occurrence);
END_TYPE; -- draughting_callout_element

TYPE draughting_grouped_item = SELECT
  (annotation_occurrence,
   geometric_set_select);
END_TYPE; -- draughting_grouped_item

TYPE draughting_organization_item = SELECT
  (product_definition_formation,
   drawing_revision,
   drawing_sheet_revision);
END_TYPE; -- draughting_organization_item

TYPE draughting_presented_item_select = SELECT
  (product_definition_formation);
END_TYPE; -- draughting_presented_item_select

TYPE draughting_titled_item = SELECT
  (drawing_revision,
   drawing_sheet_revision);
END_TYPE; -- draughting_titled_item

TYPE fill_area_style_tile_shape_select = SELECT
  (fill_area_style_tile_symbol_with_style);
END_TYPE; -- fill_area_style_tile_shape_select
```

```
TYPE fill_style_select = SELECT
  (fill_area_style_colour,
   externally_defined_tile_style,
   fill_area_style_tiles,
   externally_defined_hatch_style,
   fill_area_style_hatching);
END_TYPE; -- fill_style_select

TYPE font_select = SELECT
  (pre_defined_text_font,
   externally_defined_text_font);
END_TYPE; -- font_select

TYPE geometric_set_select = SELECT
  (point,
   curve);
END_TYPE; -- geometric_set_select

TYPE hiding_or_blanking_select = SELECT
  (presentation_area,
   presentation_view,
   annotation_fill_area);
END_TYPE; -- hiding_or_blanking_select

TYPE identifier = STRING;
END_TYPE; -- identifier

TYPE invisibility_context = SELECT
  (presentation_layer_usage,
   presentation_representation,
   presentation_set);
END_TYPE; -- invisibility_context

TYPE invisible_item = SELECT
  (styled_item,
   presentation_layer_assignment,
   presentation_representation);
END_TYPE; -- invisible_item

TYPE knot_type = ENUMERATION OF
  (uniform_knots,
   quasi_uniform_knots,
   piecewise_bezier_knots,
   unspecified);
END_TYPE; -- knot_type

TYPE label = STRING;
END_TYPE; -- label

TYPE layered_item = SELECT
  (presentation_representation,
```

```
   representation_item);
END_TYPE; -- layered_item

TYPE length_measure = REAL;
END_TYPE; -- length_measure

TYPE measure_value = SELECT
  (length_measure,
   plane_angle_measure,
   ratio_measure,
   parameter_value,
   positive_length_measure,
   positive_ratio_measure);
END_TYPE; -- measure_value

TYPE month_in_year_number = INTEGER;
WHERE
  wr1: (1 <= SELF) AND (SELF <= 12);
END_TYPE; -- month_in_year_number

TYPE null_style = ENUMERATION OF
  (null);
END_TYPE; -- null_style

TYPE parameter_value = REAL;
END_TYPE; -- parameter_value

TYPE person_organization_select = SELECT
  (person,
   organization,
   person_and_organization);
END_TYPE; -- person_organization_select

TYPE plane_angle_measure = REAL;
END_TYPE; -- plane_angle_measure

TYPE positive_length_measure = length_measure;
WHERE
  wr1: SELF > 0;
END_TYPE; -- positive_length_measure

TYPE positive_ratio_measure = ratio_measure;
WHERE
  wr1: SELF > 0;
END_TYPE; -- positive_ratio_measure

TYPE presentable_text = STRING;
END_TYPE; -- presentable_text

TYPE presentation_representation_select = SELECT
  (presentation_representation,
```

```
    presentation_set);
END_TYPE; -- presentation_representation_select

TYPE presentation_size_assignment_select = SELECT
  (presentation_view,
   presentation_area,
   area_in_set);
END_TYPE; -- presentation_size_assignment_select

TYPE presentation_style_select = SELECT
  (curve_style,
   symbol_style,
   fill_area_style,
   text_style,
   null_style);
END_TYPE; -- presentation_style_select

TYPE ratio_measure = REAL;
END_TYPE; -- ratio_measure

TYPE shape_definition = SELECT
  (product_definition_shape);
END_TYPE; -- shape_definition

TYPE si_prefix = ENUMERATION OF
  (exa,
   pico,
   mega,
   femto,
   atto,
   centi,
   nano,
   hecto,
   micro,
   tera,
   giga,
   milli,
   peta,
   deci,
   kilo,
   deca);
END_TYPE; -- si_prefix

TYPE si_unit_name = ENUMERATION OF
  (hertz,
   degree_celsius,
   siemens,
   sievert,
   lux,
   watt,
   ohm,
   second,
```

```
   becquerel,
   pascal,
   henry,
   tesla,
   volt,
   joule,
   kelvin,
   ampere,
   gram,
   steradian,
   mole,
   lumen,
   gray,
   candela,
   farad,
   radian,
   newton,
   metre,
   weber,
   coulomb);
END_TYPE; -- si_unit_name

TYPE size_select = SELECT
  (positive_length_measure,
   measure_with_unit);
END_TYPE; -- size_select

TYPE source_item = SELECT
  (identifier);
END_TYPE; -- source_item

TYPE specified_item = SELECT
  (drawing_revision);
END_TYPE; -- specified_item

TYPE style_context_select = SELECT
  (representation,
   representation_item,
   presentation_set);
END_TYPE; -- style_context_select

TYPE symbol_style_select = SELECT
  (symbol_colour);
END_TYPE; -- symbol_style_select

TYPE text = STRING;
END_TYPE; -- text

TYPE text_alignment = label;
END_TYPE; -- text_alignment

TYPE text_delineation = label;
```

```
END_TYPE; -- text_delineation

TYPE text_or_character = SELECT
  (annotation_text,
   composite_text,
   text_literal);
END_TYPE; -- text_or_character

TYPE text_path = ENUMERATION OF
  (up,
   right,
   down,
   left);
END_TYPE; -- text_path

TYPE transition_code = ENUMERATION OF
  (discontinuous,
   cont_same_gradient_same_curvature,
   cont_same_gradient,
   continuous);
END_TYPE; -- transition_code

TYPE trimming_preference = ENUMERATION OF
  (parameter,
   unspecified,
   cartesian);
END_TYPE; -- trimming_preference

TYPE trimming_select = SELECT
  (cartesian_point,
   parameter_value);
END_TYPE; -- trimming_select

TYPE unit = SELECT
  (named_unit);
END_TYPE; -- unit

TYPE vector_or_direction = SELECT
  (vector,
   direction);
END_TYPE; -- vector_or_direction

TYPE year_number = INTEGER;
END_TYPE; -- year_number

ENTITY address;
    internal_location       : OPTIONAL label;
    street_number           : OPTIONAL label;
    street                  : OPTIONAL label;
    postal_box              : OPTIONAL label;
    town                    : OPTIONAL label;
```

```
    region                  : OPTIONAL label;
    postal_code             : OPTIONAL label;
    country                 : OPTIONAL label;
    facsimile_number        : OPTIONAL label;
    telephone_number        : OPTIONAL label;
    electronic_mail_address : OPTIONAL label;
    telex_number            : OPTIONAL label;
  WHERE
    wr1: EXISTS(internal_location) OR EXISTS(street_number) OR EXISTS(
             street) OR EXISTS(postal_box) OR EXISTS(town) OR EXISTS(
             region) OR EXISTS(postal_code) OR EXISTS(country) OR EXISTS(
             facsimile_number) OR EXISTS(telephone_number) OR EXISTS(
             electronic_mail_address) OR EXISTS(telex_number);
END_ENTITY; -- address

ENTITY angular_dimension
  SUBTYPE OF (dimension_curve_directed_callout);
END_ENTITY; -- angular_dimension

ENTITY annotation_curve_occurrence
  SUBTYPE OF (annotation_occurrence);
  WHERE
    wr1: 'EXPLICIT_DRAUGHTING.CURVE' IN TYPEOF(SELF\styled_item.item);
END_ENTITY; -- annotation_curve_occurrence

ENTITY annotation_fill_area
  SUBTYPE OF (geometric_representation_item);
    boundaries : SET [1:?] OF curve;
END_ENTITY; -- annotation_fill_area

ENTITY annotation_fill_area_occurrence
  SUBTYPE OF (annotation_occurrence);
    fill_style_target : point;
  WHERE
    wr1: 'EXPLICIT_DRAUGHTING.ANNOTATION_FILL_AREA' IN TYPEOF(SELF.item);
END_ENTITY; -- annotation_fill_area_occurrence

ENTITY annotation_occurrence
  SUPERTYPE OF (ONEOF (annotation_curve_occurrence,
      annotation_fill_area_occurrence,annotation_text_occurrence,
      annotation_symbol_occurrence))
  SUBTYPE OF (styled_item);
  WHERE
    wr1: 'EXPLICIT_DRAUGHTING.GEOMETRIC_REPRESENTATION_ITEM' IN TYPEOF(
             SELF);
END_ENTITY; -- annotation_occurrence

ENTITY annotation_subfigure_occurrence
  SUBTYPE OF (annotation_symbol_occurrence);
  WHERE
    wr1: SIZEOF(QUERY ( sty <* SELF\styled_item.styles | (NOT (SIZEOF(
             sty.styles) = 1)) )) = 0;
```

```
    wr2: SIZEOF(QUERY ( sty <* SELF\styled_item.styles | (NOT (
             'EXPLICIT_DRAUGHTING.NULL_STYLE' IN TYPEOF(sty.styles[1])))  ))
             = 0;
    wr3: 'EXPLICIT_DRAUGHTING.ANNOTATION_SYMBOL' IN TYPEOF(SELF\
             styled_item.item);
    wr4: 'EXPLICIT_DRAUGHTING.DRAUGHTING_SUBFIGURE_REPRESENTATION' IN
             TYPEOF(SELF\styled_item.item\mapped_item.mapping_source.
             mapped_representation);
END_ENTITY; -- annotation_subfigure_occurrence

ENTITY annotation_symbol
  SUBTYPE OF (mapped_item);
  WHERE
    wr1: 'EXPLICIT_DRAUGHTING.SYMBOL_REPRESENTATION_MAP' IN TYPEOF(SELF\
             mapped_item.mapping_source);
    wr2: 'EXPLICIT_DRAUGHTING.SYMBOL_TARGET' IN TYPEOF(SELF\mapped_item.
             mapping_target);
    wr3: 'EXPLICIT_DRAUGHTING.GEOMETRIC_REPRESENTATION_ITEM' IN TYPEOF(
             SELF);
END_ENTITY; -- annotation_symbol

ENTITY annotation_symbol_occurrence
  SUBTYPE OF (annotation_occurrence);
  WHERE
    wr1: SIZEOF(['EXPLICIT_DRAUGHTING.ANNOTATION_SYMBOL',
             'EXPLICIT_DRAUGHTING.DEFINED_SYMBOL'] * TYPEOF(SELF\
             styled_item.item)) > 0;
END_ENTITY; -- annotation_symbol_occurrence

ENTITY annotation_text
  SUBTYPE OF (mapped_item);
  WHERE
    wr1: 'EXPLICIT_DRAUGHTING.AXIS2_PLACEMENT' IN TYPEOF(SELF\
             mapped_item.mapping_target);
    wr2: 'EXPLICIT_DRAUGHTING.TEXT_STRING_REPRESENTATION' IN TYPEOF(SELF
             \mapped_item.mapping_source.mapped_representation);
    wr3: 'EXPLICIT_DRAUGHTING.GEOMETRIC_REPRESENTATION_ITEM' IN TYPEOF(
             SELF);
END_ENTITY; -- annotation_text

ENTITY annotation_text_occurrence
  SUBTYPE OF (annotation_occurrence);
  WHERE
    wr1: SIZEOF(['EXPLICIT_DRAUGHTING.TEXT_LITERAL',
             'EXPLICIT_DRAUGHTING.ANNOTATION_TEXT',
             'EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_CHARACTER',
             'EXPLICIT_DRAUGHTING.DEFINED_CHARACTER_GLYPH',
             'EXPLICIT_DRAUGHTING.COMPOSITE_TEXT'] * TYPEOF(SELF\
             styled_item.item)) > 0;
END_ENTITY; -- annotation_text_occurrence

ENTITY application_context;
```

```
    application : text;
  INVERSE
    context_elements : SET [1:?] OF application_context_element FOR
                       frame_of_reference;
END_ENTITY; -- application_context

ENTITY application_context_element
  SUPERTYPE OF (ONEOF (product_context,product_definition_context));
    name              : label;
    frame_of_reference : application_context;
END_ENTITY; -- application_context_element

ENTITY application_protocol_definition;
    status                                 : label;
    application_interpreted_model_schema_name : label;
    application_protocol_year              : year_number;
    application                            : application_context;
END_ENTITY; -- application_protocol_definition

ENTITY approval;
    status : approval_status;
    level  : label;
END_ENTITY; -- approval

ENTITY approval_assignment
  ABSTRACT SUPERTYPE;
    assigned_approval : approval;
END_ENTITY; -- approval_assignment

ENTITY approval_date_time;
    date_time      : date_time_select;
    dated_approval : approval;
END_ENTITY; -- approval_date_time

ENTITY approval_person_organization;
    person_organization : person_organization_select;
    authorized_approval : approval;
    role                : approval_role;
END_ENTITY; -- approval_person_organization

ENTITY approval_role;
    role : label;
END_ENTITY; -- approval_role

ENTITY approval_status;
    name : label;
END_ENTITY; -- approval_status

ENTITY area_in_set;
    area   : presentation_area;
    in_set : presentation_set;
END_ENTITY; -- area_in_set
```

```
ENTITY axis2_placement_2d
  SUBTYPE OF (placement);
    ref_direction : OPTIONAL direction;
  DERIVE
    p : LIST [2:2] OF direction := build_2axes(ref_direction);
  WHERE
    wr1: SELF\geometric_representation_item.dim = 2;
END_ENTITY; -- axis2_placement_2d

ENTITY b_spline_curve
  SUPERTYPE OF (ONEOF (uniform_curve,b_spline_curve_with_knots,
      quasi_uniform_curve,bezier_curve) ANDOR rational_b_spline_curve)
  SUBTYPE OF (bounded_curve);
    degree              : INTEGER;
    control_points_list : LIST [2:?] OF cartesian_point;
    curve_form          : b_spline_curve_form;
    closed_curve        : LOGICAL;
    self_intersect      : LOGICAL;
  DERIVE
    upper_index_on_control_points : INTEGER := SIZEOF(
                                       control_points_list) - 1;
    control_points                : ARRAY [0:
                                       upper_index_on_control_points] OF
                                        cartesian_point := list_to_array(
                                       control_points_list,0,
                                       upper_index_on_control_points);
  WHERE
    wr1: ('EXPLICIT_DRAUGHTING.UNIFORM_CURVE' IN TYPEOF(SELF)) OR (
             'EXPLICIT_DRAUGHTING.QUASI_UNIFORM_CURVE' IN TYPEOF(SELF))
            OR ('EXPLICIT_DRAUGHTING.BEZIER_CURVE' IN TYPEOF(SELF)) OR (
            'EXPLICIT_DRAUGHTING.B_SPLINE_CURVE_WITH_KNOTS' IN TYPEOF(
            SELF));
END_ENTITY; -- b_spline_curve

ENTITY b_spline_curve_with_knots
  SUBTYPE OF (b_spline_curve);
    knot_multiplicities : LIST [2:?] OF INTEGER;
    knots               : LIST [2:?] OF parameter_value;
    knot_spec           : knot_type;
  DERIVE
    upper_index_on_knots : INTEGER := SIZEOF(knots);
  WHERE
    wr1: constraints_param_b_spline(degree,upper_index_on_knots,
            upper_index_on_control_points,knot_multiplicities,knots);
    wr2: SIZEOF(knot_multiplicities) = upper_index_on_knots;
END_ENTITY; -- b_spline_curve_with_knots

ENTITY bezier_curve
  SUBTYPE OF (b_spline_curve);
END_ENTITY; -- bezier_curve

ENTITY bounded_curve
```

```
  SUPERTYPE OF (ONEOF (polyline,b_spline_curve,trimmed_curve,
      composite_curve))
  SUBTYPE OF (curve);
END_ENTITY; -- bounded_curve

ENTITY calendar_date
  SUBTYPE OF (date);
    day_component   : day_in_month_number;
    month_component : month_in_year_number;
  WHERE
    wr1: valid_calendar_date(SELF);
END_ENTITY; -- calendar_date

ENTITY camera_image
  SUBTYPE OF (mapped_item);
  WHERE
    wr1: 'EXPLICIT_DRAUGHTING.CAMERA_USAGE' IN TYPEOF(SELF\mapped_item.
             mapping_source);
    wr2: 'EXPLICIT_DRAUGHTING.PLANAR_BOX' IN TYPEOF(SELF\mapped_item.
             mapping_target);
    wr3: 'EXPLICIT_DRAUGHTING.GEOMETRIC_REPRESENTATION_ITEM' IN TYPEOF(
             SELF);
END_ENTITY; -- camera_image

ENTITY camera_image_2d_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_d2.view_window.size_in_x;
  WHERE
    wr1: 'EXPLICIT_DRAUGHTING.CAMERA_MODEL_D2' 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_d2.view_window);
    wr3: SELF\mapped_item.mapping_source.mapping_origin\camera_model_d2.
             view_window_clipping;
END_ENTITY; -- camera_image_2d_with_scale

ENTITY camera_model
  SUPERTYPE OF (camera_model_d2)
  SUBTYPE OF (geometric_representation_item);
  WHERE
    wr1: (SIZEOF(USEDIN(SELF,'EXPLICIT_DRAUGHTING.' +
             'ITEM_DEFINED_TRANSFORMATION.' + 'TRANSFORM_ITEM_1')) +
             SIZEOF(USEDIN(SELF,'EXPLICIT_DRAUGHTING.' +
             'REPRESENTATION_MAP.MAPPING_ORIGIN'))) > 0;
    wr2: SIZEOF(USEDIN(SELF,'EXPLICIT_DRAUGHTING.' + 'STYLED_ITEM.ITEM'))
             = 0;
END_ENTITY; -- camera_model
```

```
ENTITY camera_model_d2
  SUBTYPE OF (camera_model);
    view_window          : planar_box;
    view_window_clipping : BOOLEAN;
  WHERE
    wr1: SELF\geometric_representation_item.dim = 2;
END_ENTITY; -- camera_model_d2

ENTITY camera_usage
  SUBTYPE OF (representation_map);
  WHERE
    wr1: NOT ('EXPLICIT_DRAUGHTING.PRESENTATION_REPRESENTATION' IN
             TYPEOF(SELF\representation_map.mapped_representation));
    wr2: 'EXPLICIT_DRAUGHTING.CAMERA_MODEL' IN TYPEOF(SELF\
             representation_map.mapping_origin);
END_ENTITY; -- camera_usage

ENTITY cartesian_point
  SUBTYPE OF (point);
    coordinates : LIST [1:3] OF length_measure;
END_ENTITY; -- cartesian_point

ENTITY circle
  SUBTYPE OF (conic);
    radius : positive_length_measure;
END_ENTITY; -- circle

ENTITY colour;
END_ENTITY; -- colour

ENTITY colour_rgb
  SUBTYPE OF (colour_specification);
    red   : REAL;
    green : REAL;
    blue  : REAL;
  WHERE
    wr1: (0 <= red) AND (red <= 1);
    wr2: (0 <= green) AND (green <= 1);
    wr3: (0 <= blue) AND (blue <= 1);
END_ENTITY; -- colour_rgb

ENTITY colour_specification
  SUBTYPE OF (colour);
    name : label;
END_ENTITY; -- colour_specification

ENTITY composite_curve
  SUBTYPE OF (bounded_curve);
    segments       : LIST [1:?] OF composite_curve_segment;
    self_intersect : LOGICAL;
  DERIVE
    n_segments   : INTEGER := SIZEOF(segments);
```

```
    closed_curve : LOGICAL := segments[n_segments].transition <>
                        discontinuous;
  WHERE
    wr1: ((NOT closed_curve) AND (SIZEOF(QUERY ( temp <* segments | (
              temp.transition = discontinuous) )) = 1)) OR (closed_curve
              AND (SIZEOF(QUERY ( temp <* segments | (temp.transition =
              discontinuous) )) = 0));
END_ENTITY; -- composite_curve

ENTITY composite_curve_segment;
    transition   : transition_code;
    same_sense   : BOOLEAN;
    parent_curve : curve;
  INVERSE
    using_curves : BAG [1:?] OF composite_curve FOR segments;
  WHERE
    wr1: 'EXPLICIT_DRAUGHTING.BOUNDED_CURVE' IN TYPEOF(parent_curve);
END_ENTITY; -- composite_curve_segment

ENTITY composite_text
  SUBTYPE OF (geometric_representation_item);
    collected_text : SET [2:?] OF text_or_character;
  WHERE
    wr1: acyclic_composite_text(SELF,SELF.collected_text);
END_ENTITY; -- composite_text

ENTITY composite_text_with_associated_curves
  SUBTYPE OF (composite_text);
    associated_curves : SET [1:?] OF curve;
END_ENTITY; -- composite_text_with_associated_curves

ENTITY composite_text_with_blanking_box
  SUBTYPE OF (composite_text);
    blanking : planar_box;
END_ENTITY; -- composite_text_with_blanking_box

ENTITY composite_text_with_extent
  SUBTYPE OF (composite_text);
    extent : planar_extent;
END_ENTITY; -- composite_text_with_extent

ENTITY conic
  SUPERTYPE OF (ONEOF (circle,ellipse,hyperbola,parabola))
  SUBTYPE OF (curve);
    position : axis2_placement;
END_ENTITY; -- conic

ENTITY context_dependent_invisibility
  SUBTYPE OF (invisibility);
    presentation_context : invisibility_context;
END_ENTITY; -- context_dependent_invisibility
```

```
ENTITY contract;
    name    : label;
    purpose : text;
    kind    : contract_type;
END_ENTITY; -- contract

ENTITY contract_assignment
  ABSTRACT SUPERTYPE;
    assigned_contract : contract;
END_ENTITY; -- contract_assignment

ENTITY contract_type;
    description : label;
END_ENTITY; -- contract_type

ENTITY conversion_based_unit
  SUBTYPE OF (named_unit);
    name              : label;
    conversion_factor : measure_with_unit;
END_ENTITY; -- conversion_based_unit

ENTITY curve
  SUPERTYPE OF (ONEOF (line,conic,offset_curve_2d))
  SUBTYPE OF (geometric_representation_item);
END_ENTITY; -- curve

ENTITY curve_dimension
  SUBTYPE OF (dimension_curve_directed_callout);
END_ENTITY; -- curve_dimension

ENTITY curve_style;
    name         : label;
    curve_font   : curve_font_or_scaled_curve_font_select;
    curve_width  : size_select;
    curve_colour : colour;
END_ENTITY; -- curve_style

ENTITY curve_style_font;
    name         : label;
    pattern_list : LIST [1:?] OF curve_style_font_pattern;
END_ENTITY; -- curve_style_font

ENTITY curve_style_font_pattern;
    visible_segment_length   : positive_length_measure;
    invisible_segment_length : positive_length_measure;
END_ENTITY; -- curve_style_font_pattern

ENTITY date
  SUPERTYPE OF (calendar_date);
    year_component : year_number;
END_ENTITY; -- date
```

```
ENTITY datum_feature_callout
  SUBTYPE OF (draughting_callout);
END_ENTITY; -- datum_feature_callout

ENTITY datum_target_callout
  SUBTYPE OF (draughting_callout);
END_ENTITY; -- datum_target_callout

ENTITY defined_symbol
  SUBTYPE OF (geometric_representation_item);
    definition : defined_symbol_select;
    target  : symbol_target;
END_ENTITY; -- defined_symbol

ENTITY diameter_dimension
  SUBTYPE OF (dimension_curve_directed_callout);
END_ENTITY; -- diameter_dimension

ENTITY dimension_callout_component_relationship
  SUBTYPE OF (draughting_callout_relationship);
  WHERE
    wr1: SELF.name IN ['prefix','suffix'];
    wr2: 'EXPLICIT_DRAUGHTING.STRUCTURED_DIMENSION_CALLOUT' IN TYPEOF(
             SELF.relating_draughting_callout);
    wr3: SIZEOF(TYPEOF(SELF.related_draughting_callout) * [
              'EXPLICIT_DRAUGHTING.LEADER_DIRECTED_CALLOUT',
              'EXPLICIT_DRAUGHTING.PROJECTION_DIRECTED_CALLOUT',
              'EXPLICIT_DRAUGHTING.DIMENSION_CURVE_DIRECTED_CALLOUT',
              'EXPLICIT_DRAUGHTING.STRUCTURED_DIMENSION_CALLOUT']) = 0;
    wr4: (SELF.related_draughting_callout.contents * SELF.
             relating_draughting_callout.contents) = SELF.
             related_draughting_callout.contents;
END_ENTITY; -- dimension_callout_component_relationship

ENTITY dimension_callout_relationship
  SUBTYPE OF (draughting_callout_relationship);
  WHERE
    wr1: SELF.name IN ['primary','secondary'];
    wr2: SIZEOF(TYPEOF(SELF.relating_draughting_callout) * [
              'EXPLICIT_DRAUGHTING.ANGULAR_DIMENSION',
              'EXPLICIT_DRAUGHTING.CURVE_DIMENSION',
              'EXPLICIT_DRAUGHTING.DIAMETER_DIMENSION',
              'EXPLICIT_DRAUGHTING.LEADER_DIRECTED_DIMENSION',
              'EXPLICIT_DRAUGHTING.LINEAR_DIMENSION',
              'EXPLICIT_DRAUGHTING.ORDINATE_DIMENSION',
              'EXPLICIT_DRAUGHTING.RADIUS_DIMENSION']) = 1;
    wr3: SIZEOF(TYPEOF(SELF.related_draughting_callout) * [
              'EXPLICIT_DRAUGHTING.DIMENSION_CURVE_DIRECTED_CALLOUT',
              'EXPLICIT_DRAUGHTING.PROJECTION_DIRECTED_CALLOUT',
              'EXPLICIT_DRAUGHTING.LEADER_DIRECTED_CALLOUT']) = 0;
    wr4: (SELF.related_draughting_callout.contents * SELF.
             relating_draughting_callout.contents) = SELF.
```

```
            related_draughting_callout.contents;
END_ENTITY; -- dimension_callout_relationship

ENTITY dimension_curve
  SUBTYPE OF (annotation_curve_occurrence);
  WHERE
    wr1: SIZEOF(USEDIN(SELF,'EXPLICIT_DRAUGHTING.' +
            'DIMENSION_CURVE_TERMINATOR.ANNOTATED_DIMENSION_' + 'CURVE'))
            <= 2;
    wr2: SIZEOF(USEDIN(SELF,'EXPLICIT_DRAUGHTING.' +
            'DIMENSION_CURVE_DIRECTED_CALLOUT.CONTENTS')) >= 1;
    wr3: (SIZEOF(QUERY ( dct_1 <* USEDIN(SELF,'DRAUGHTING_ELEMENT_' +
            'SCHEMA.DIMENSION_CURVE_TERMINATOR.ANNOTATED_DIMENSION_' +
            'CURVE') | (dct_1.role = origin) )) <= 1) AND (SIZEOF(
            QUERY ( dct_2 <* USEDIN(SELF,'DRAUGHTING_ELEMENT_' +
            'SCHEMA.DIMENSION_CURVE_TERMINATOR.ANNOTATED_DIMENSION_' +
            'CURVE') | (dct_2.role = target) )) <= 1);
END_ENTITY; -- dimension_curve

ENTITY dimension_curve_directed_callout
  SUBTYPE OF (draughting_callout);
  WHERE
    wr1: SIZEOF(QUERY ( d_c <* SELF.contents | (
            'EXPLICIT_DRAUGHTING.DIMENSION_CURVE' IN TYPEOF(d_c)) )) = 1;
    wr2: SIZEOF(SELF.contents) >= 2;
END_ENTITY; -- dimension_curve_directed_callout

ENTITY dimension_curve_terminator
  SUBTYPE OF (terminator_symbol);
    role : dimension_extent_usage;
  WHERE
    wr1: 'EXPLICIT_DRAUGHTING.DIMENSION_CURVE' IN TYPEOF(SELF\
            terminator_symbol.annotated_curve);
END_ENTITY; -- dimension_curve_terminator

ENTITY dimension_pair
  SUBTYPE OF (draughting_callout_relationship);
  WHERE
    wr1: SELF.name IN ['chained','parallel'];
    wr2: SIZEOF(TYPEOF(SELF.relating_draughting_callout) * [
            'EXPLICIT_DRAUGHTING.ANGULAR_DIMENSION',
            'EXPLICIT_DRAUGHTING.CURVE_DIMENSION',
            'EXPLICIT_DRAUGHTING.DIAMETER_DIMENSION',
            'EXPLICIT_DRAUGHTING.LINEAR_DIMENSION',
            'EXPLICIT_DRAUGHTING.ORDINATE_DIMENSION',
            'EXPLICIT_DRAUGHTING.RADIUS_DIMENSION']) = 1;
    wr3: SIZEOF(TYPEOF(SELF.related_draughting_callout) * [
            'EXPLICIT_DRAUGHTING.ANGULAR_DIMENSION',
            'EXPLICIT_DRAUGHTING.CURVE_DIMENSION',
            'EXPLICIT_DRAUGHTING.DIAMETER_DIMENSION',
            'EXPLICIT_DRAUGHTING.LINEAR_DIMENSION',
            'EXPLICIT_DRAUGHTING.ORDINATE_DIMENSION',
```

```
            'EXPLICIT_DRAUGHTING.RADIUS_DIMENSION']) = 1;
END_ENTITY; -- dimension_pair

ENTITY dimensional_exponents;
    length_exponent                    : REAL;
    mass_exponent                      : REAL;
    time_exponent                      : REAL;
    electric_current_exponent          : REAL;
    thermodynamic_temperature_exponent : REAL;
    amount_of_substance_exponent       : REAL;
    luminous_intensity_exponent        : REAL;
END_ENTITY; -- dimensional_exponents

ENTITY direction
  SUBTYPE OF (geometric_representation_item);
    direction_ratios : LIST [2:3] OF REAL;
  WHERE
    wr1: SIZEOF(QUERY ( tmp <* direction_ratios | (tmp <> 0) )) > 0;
END_ENTITY; -- direction

ENTITY document;
    id          : identifier;
    name        : label;
    description : text;
    kind        : document_type;
  UNIQUE
    ur1 : id;
END_ENTITY; -- document

ENTITY document_reference
  ABSTRACT SUPERTYPE;
    assigned_document : document;
    source            : label;
END_ENTITY; -- document_reference

ENTITY document_type;
    product_data_type : label;
END_ENTITY; -- document_type

ENTITY draughting_annotation_occurrence
  SUBTYPE OF (annotation_occurrence);
  WHERE
    curve_has_curve_style :
      (NOT ('EXPLICIT_DRAUGHTING.ANNOTATION_CURVE_OCCURRENCE'
          IN TYPEOF(SELF))) OR (SIZEOF(
          QUERY ( sty <* SELF\
          styled_item.styles | (NOT ((
          SIZEOF(sty.styles) = 1) AND ('EXPLICIT_DRAUGHTING.CURVE_STYLE'
          IN TYPEOF(sty.styles[1])))) ))
          = 0);
    fill_area_has_fill_style :
      (NOT ('EXPLICIT_DRAUGHTING.ANNOTATION_FILL_AREA_OCCURRENCE'
```

```
      IN TYPEOF(SELF))) OR (SIZEOF(
      QUERY ( sty <* SELF\
      styled_item.styles | (NOT ((
      SIZEOF(sty.styles) = 1) AND ('EXPLICIT_DRAUGHTING.FILL_AREA_STYLE'
      IN TYPEOF(sty.styles[1])))) ))
      = 0);
  styled_fill_boundaries :
    (NOT ('EXPLICIT_DRAUGHTING.ANNOTATION_FILL_AREA_OCCURRENCE'
      IN TYPEOF(SELF))) OR (SIZEOF(
      QUERY ( bound <* SELF\
      styled_item.item\
      annotation_fill_area.
      boundaries | (NOT (SIZEOF(
      QUERY ( si <* USEDIN(bound,
      'EXPLICIT_DRAUGHTING.' +
      'STYLED_ITEM.ITEM') | ('EXPLICIT_DRAUGHTING.' +
      'ANNOTATION_CURVE_OCCURRENCE'
      IN TYPEOF(si)) )) > 0)) )) =
      0);
  symbol_has_symbol_style :
    (NOT ('EXPLICIT_DRAUGHTING.ANNOTATION_SYMBOL_OCCURRENCE'
      IN TYPEOF(SELF))) OR (SIZEOF(
      QUERY ( sty <* SELF\
      styled_item.styles | (NOT ((
      SIZEOF(sty.styles) = 1) AND (
      SIZEOF(TYPEOF(sty.styles[1])
      * ['EXPLICIT_DRAUGHTING.SYMBOL_STYLE',
      'EXPLICIT_DRAUGHTING.NULL_STYLE'])
      = 1))) )) = 0);
  allowable_symbol_representations :
    (NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_SYMBOL_OCCURRENCE'
      IN TYPEOF(SELF)) AND ('EXPLICIT_DRAUGHTING.ANNOTATION_SYMBOL'
      IN TYPEOF(SELF\styled_item.
      item)))) OR (SIZEOF([
      'EXPLICIT_DRAUGHTING.' + 'DRAUGHTING_SYMBOL_REPRESENTATION',
      'EXPLICIT_DRAUGHTING.' + 'DRAUGHTING_SUBFIGURE_REPRESENTATION']
      * TYPEOF(SELF\styled_item.
      item\mapped_item.
      mapping_source.
      mapped_representation)) = 1);
  text_has_text_style :
    (NOT ('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
      IN TYPEOF(SELF))) OR (SIZEOF(
      QUERY ( sty <* SELF\
      styled_item.styles | (NOT ((
      SIZEOF(sty.styles) = 1) AND ('EXPLICIT_DRAUGHTING.TEXT_STYLE'
      IN TYPEOF(sty.styles[1])))) ))
      = 0);
  text_not_nested :
    (NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
      IN TYPEOF(SELF)) AND ('EXPLICIT_DRAUGHTING.COMPOSITE_TEXT'
      IN TYPEOF(SELF\styled_item.
```

```
      item)))) OR (SIZEOF(
      QUERY ( text <* SELF\
      styled_item.item\
      composite_text.collected_text
       | (NOT ('EXPLICIT_DRAUGHTING.TEXT_LITERAL'
      IN TYPEOF(text))) )) = 0);
allowable_text :
  NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
          IN TYPEOF (SELF)) AND
       (SIZEOF (TYPEOF(SELF\styled_item.item) *
          ['EXPLICIT_DRAUGHTING.COMPOSITE_TEXT',
           'EXPLICIT_DRAUGHTING.TEXT_LITERAL']) = 1));
text_alignment_literals :
  (NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
      IN TYPEOF(SELF)) AND ('EXPLICIT_DRAUGHTING.TEXT_LITERAL'
      IN TYPEOF(SELF\styled_item.
      item)))) OR (SELF\styled_item
      .item\text_literal.alignment
      IN ['baseline left',
      'baseline centre',
      'baseline right']);
text_alignment_composites :
  (NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
      IN TYPEOF(SELF)) AND ('EXPLICIT_DRAUGHTING.COMPOSITE_TEXT'
      IN TYPEOF(SELF\styled_item.
      item)))) OR (SIZEOF(
      QUERY ( literal <*
      QUERY ( text <* SELF\
      styled_item.item\
      composite_text.collected_text
       | ('EXPLICIT_DRAUGHTING.TEXT_LITERAL'
      IN TYPEOF(text)) ) | (NOT (
      literal.alignment IN [
      'baseline left',
      'baseline centre',
      'baseline right'])) )) = 0);
text_literal_not_rotated :
  NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
     IN TYPEOF(SELF)) AND ('EXPLICIT_DRAUGHTING.TEXT_LITERAL'
     IN TYPEOF(SELF\styled_item.
     item)) AND EXISTS(SELF\
     styled_item.item\text_literal
     .placement\axis2_placement_2d
     .ref_direction));
literals_in_composites_not_rotated:
  (NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
      IN TYPEOF(SELF)) AND ('EXPLICIT_DRAUGHTING.COMPOSITE_TEXT'
      IN TYPEOF(SELF\styled_item.
      item)))) OR (SIZEOF(
      QUERY ( literal <*
      QUERY ( text <* SELF\
      styled_item.item\
```

```
      composite_text.collected_text
       | ('EXPLICIT_DRAUGHTING.TEXT_LITERAL'
      IN TYPEOF(text)) ) | EXISTS(
      literal.placement\
      axis2_placement_2d.
      ref_direction) )) = 0);
  single_text_alignment :
    (NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
      IN TYPEOF(SELF)) AND ('EXPLICIT_DRAUGHTING.COMPOSITE_TEXT'
      IN TYPEOF(SELF\styled_item.
      item)))) OR
      check_text_alignment(SELF\
      styled_item.item);
  single_text_font :
    (NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
      IN TYPEOF(SELF)) AND ('EXPLICIT_DRAUGHTING.COMPOSITE_TEXT'
      IN TYPEOF(SELF\styled_item.
      item)))) OR check_text_font(
      SELF\styled_item.item);
  allowable_text_literals :
    (NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
      IN TYPEOF(SELF)) AND ('EXPLICIT_DRAUGHTING.COMPOSITE_TEXT'
      IN TYPEOF(SELF\styled_item.
      item)))) OR (SIZEOF(
      QUERY ( literal <*
      QUERY ( text <* SELF\
      styled_item.item\
      composite_text.collected_text
       | ('EXPLICIT_DRAUGHTING.TEXT_LITERAL'
      IN TYPEOF(text)) ) | (NOT (
      SIZEOF(TYPEOF(literal) * [
      'EXPLICIT_DRAUGHTING.' + 'TEXT_LITERAL_WITH_BLANKING_BOX',
      'EXPLICIT_DRAUGHTING.' + 'TEXT_LITERAL_WITH_ASSOCIATED_CURVES'])
      = 0)) )) = 0);
  styled_text_literal_curves :
    (NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
      IN TYPEOF(SELF)) AND ('EXPLICIT_DRAUGHTING.' +
      'TEXT_LITERAL_WITH_ASSOCIATED_CURVES'
      IN TYPEOF(SELF\styled_item.
      item)))) OR (SIZEOF(
      QUERY ( crv <* SELF\
      styled_item.item\text_literal_with_associated_curves
      .associated_curves | (NOT (
      SIZEOF(QUERY ( si <* USEDIN(
      crv,'EXPLICIT_DRAUGHTING.' +
      'STYLED_ITEM.ITEM') | ('EXPLICIT_DRAUGHTING.' +
      'ANNOTATION_CURVE_OCCURRENCE'
      IN TYPEOF(si)) )) > 0)) )) =
      0);
  styled_composite_text_curves :
    (NOT (('EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE'
      IN TYPEOF(SELF)) AND ('EXPLICIT_DRAUGHTING.' +
```

```
        'COMPOSITE_TEXT_WITH_ASSOCIATED_CURVES'
        IN TYPEOF(SELF\styled_item.
        item)))) OR (SIZEOF(
        QUERY ( crv <* SELF\
        styled_item.item\composite_text_with_associated_curves
        .associated_curves | (NOT (
        SIZEOF(QUERY ( si <* USEDIN(
        crv,'EXPLICIT_DRAUGHTING.' +
        'STYLED_ITEM.ITEM') | ('EXPLICIT_DRAUGHTING.' +
        'ANNOTATION_CURVE_OCCURRENCE'
        IN TYPEOF(si)) )) > 0)) )) =
        0);
    curve_style_has_width :
      SIZEOF(QUERY ( cs <*
        QUERY ( sty <* SELF\
        styled_item.styles | ('EXPLICIT_DRAUGHTING.CURVE_STYLE'
        IN TYPEOF(sty.styles[1])) )
         | (NOT (('EXPLICIT_DRAUGHTING.LENGTH_MEASURE_WITH_UNIT'
        IN TYPEOF(cs.styles[1]\
        curve_style.curve_width)) AND
         ('EXPLICIT_DRAUGHTING.POSITIVE_LENGTH_MEASURE'
        IN TYPEOF(cs.styles[1]\
        curve_style.curve_width.
        value_component)))) )) = 0;
    tiling_constraints :
      SIZEOF(QUERY ( fas <*
        QUERY ( sty <* SELF\
        styled_item.styles | ('EXPLICIT_DRAUGHTING.FILL_AREA_STYLE'
        IN TYPEOF(sty.styles[1])) )
         | (NOT ((SIZEOF(
        QUERY ( fs <* fas.styles[1]\
        fill_area_style.fill_styles
         | ('EXPLICIT_DRAUGHTING.FILL_AREA_STYLE_TILES'
        IN TYPEOF(fs)) )) <= 1) AND (
        SIZEOF(QUERY ( fst <*
        QUERY ( fs <* fas.styles[1]\
        fill_area_style.fill_styles
         | ('EXPLICIT_DRAUGHTING.FILL_AREA_STYLE_TILES'
        IN TYPEOF(fs)) ) | (NOT (
        SIZEOF(fst.tiles) = 1)) )) =
        0))) )) = 0;
    hatching_constraints :
      SIZEOF(QUERY ( fas <*
        QUERY ( sty <* SELF\
        styled_item.styles | ('EXPLICIT_DRAUGHTING.FILL_AREA_STYLE'
        IN TYPEOF(sty.styles[1])) )
         | (NOT (SIZEOF(
        QUERY ( fst <* QUERY ( fs <*
        fas.styles[1]\fill_area_style
        .fill_styles | ('EXPLICIT_DRAUGHTING.FILL_AREA_STYLE_HATCHING'
        IN TYPEOF(fs)) ) | (NOT (fst.point_of_reference_hatch_line
        :=: fst.pattern_start)) )) =
```

```
          0)) )) = 0;
    text_style_constraint :
      SIZEOF(QUERY ( ts <*
          QUERY ( sty <* SELF\
          styled_item.styles | ('EXPLICIT_DRAUGHTING.TEXT_STYLE'
          IN TYPEOF(sty.styles[1])) )
           | (NOT ('EXPLICIT_DRAUGHTING.TEXT_STYLE_WITH_BOX_CHARACTERISTICS'
          IN TYPEOF(ts.styles[1])))  ))
          = 0;
    text_style_characteristics :
      SIZEOF(QUERY ( ts <*
          QUERY ( sty <* SELF\
          styled_item.styles | ((
          'EXPLICIT_DRAUGHTING.' + 'TEXT_STYLE_WITH_BOX_CHARACTERISTICS')
          IN TYPEOF(sty.styles[1])) )
           | (NOT (SIZEOF(ts.styles[1]\text_style_with_box_characteristics
          .box_characteristics) = 4)) ))
          = 0;
END_ENTITY; -- draughting_annotation_occurrence

ENTITY draughting_approval_assignment
  SUBTYPE OF (approval_assignment);
    approved_items : SET [1:?] OF approved_item;
  WHERE
    wr1: SIZEOF(QUERY ( item <* approved_items | (
             'EXPLICIT_DRAUGHTING.DRAWING_REVISION' IN TYPEOF(item)) ))
            <= 1;
END_ENTITY; -- draughting_approval_assignment

ENTITY draughting_callout
  SUBTYPE OF (geometric_representation_item);
    contents : SET [1:?] OF draughting_callout_element;
END_ENTITY; -- draughting_callout

ENTITY draughting_callout_relationship;
    name                        : label;
    description                 : text;
    relating_draughting_callout : draughting_callout;
    related_draughting_callout  : draughting_callout;
END_ENTITY; -- draughting_callout_relationship

ENTITY draughting_contract_assignment
  SUBTYPE OF (contract_assignment);
    items : SET [1:?] OF contracted_item;
END_ENTITY; -- draughting_contract_assignment

ENTITY draughting_drawing_revision
  SUBTYPE OF (drawing_revision);
  WHERE
    drawing_requires_sheets :
      (SIZEOF(USEDIN(SELF,'EXPLICIT_DRAUGHTING.AREA_IN_SET.IN_SET'))
          >= 1) AND (SIZEOF(
```

```
        QUERY ( ais <* USEDIN(SELF,
        'EXPLICIT_DRAUGHTING.' +
        'AREA_IN_SET.IN_SET') | (NOT
        ('EXPLICIT_DRAUGHTING.DRAWING_SHEET_REVISION'
        IN TYPEOF(ais.area))) )) = 0);
    drawing_approval_has_date :
      SIZEOF(QUERY ( app_ass <*
        USEDIN(SELF,
        'EXPLICIT_DRAUGHTING.' +
        'DRAUGHTING_APPROVAL_ASSIGNMENT.APPROVED_ITEMS')
         | (NOT (SIZEOF(USEDIN(
        app_ass.assigned_approval,
        'EXPLICIT_DRAUGHTING.APPROVAL_DATE_TIME.DATED_APPROVAL'))
        = 1)) )) = 0;
    sheet_approval_has_date :
      SIZEOF(QUERY ( ais <* USEDIN(
        SELF,'EXPLICIT_DRAUGHTING.' +
         'AREA_IN_SET.IN_SET') | (
        NOT (SIZEOF(
        QUERY ( app_ass <* USEDIN(ais
        .area,'EXPLICIT_DRAUGHTING.'
        + 'DRAUGHTING_APPROVAL_ASSIGNMENT.APPROVED_ITEMS')
         | (NOT (SIZEOF(USEDIN(
        app_ass.assigned_approval,
        'EXPLICIT_DRAUGHTING.APPROVAL_DATE_TIME.DATED_APPROVAL'))
        = 1)) )) = 0)) )) = 0;
    drawing_approval_has_authorization :
      SIZEOF(QUERY ( app_ass <*
        USEDIN(SELF,
        'EXPLICIT_DRAUGHTING.' +
        'DRAUGHTING_APPROVAL_ASSIGNMENT.APPROVED_ITEMS')
         | (NOT (SIZEOF(USEDIN(
        app_ass.assigned_approval,
        'EXPLICIT_DRAUGHTING.' +
        'APPROVAL_PERSON_ORGANIZATION.AUTHORIZED_APPROVAL'))
        >= 1)) )) = 0;
    sheet_approval_has_authorization :
      SIZEOF(QUERY ( ais <* USEDIN(
        SELF,'EXPLICIT_DRAUGHTING.' +
         'AREA_IN_SET.IN_SET') | (
        NOT (SIZEOF(
        QUERY ( app_ass <* USEDIN(ais
        .area,'EXPLICIT_DRAUGHTING.'
        + 'DRAUGHTING_APPROVAL_ASSIGNMENT.APPROVED_ITEMS')
         | (NOT (SIZEOF(USEDIN(
        app_ass.assigned_approval,
        'EXPLICIT_DRAUGHTING.' +
        'APPROVAL_PERSON_ORGANIZATION.AUTHORIZED_APPROVAL'))
        >= 1)) )) = 0)) )) = 0;
    drawing_title :
      SIZEOF(USEDIN(SELF,'EXPLICIT_DRAUGHTING.DRAUGHTING_TITLE.ITEMS'))
        <= 1;
```

```
    sheet_title :
      SIZEOF(QUERY ( ais <* USEDIN(
          SELF,'EXPLICIT_DRAUGHTING.' +
           'AREA_IN_SET.IN_SET') | (
          NOT (SIZEOF(USEDIN(ais.area,
          'EXPLICIT_DRAUGHTING.' +
          'DRAUGHTING_TITLE.ITEMS')) <=
           1)) )) = 0;
  allowable_items_of_sheet:
    SIZEOF (QUERY (ais <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                        'AREA_IN_SET.IN_SET') |
      NOT (SIZEOF (QUERY (item <* ais.area\representation.items |
        NOT (SIZEOF (['EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE',
                      'EXPLICIT_DRAUGHTING.MAPPED_ITEM',
                      'EXPLICIT_DRAUGHTING.AXIS2_PLACEMENT',
                      'EXPLICIT_DRAUGHTING.PLANAR_BOX'] *
                TYPEOF(item)) = 1)
      )) = 0)
    )) = 0;
  sheet_contains_view_or_annotation:
    SIZEOF (QUERY (ais <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                        'AREA_IN_SET.IN_SET') |
      NOT (SIZEOF (QUERY (item <* ais.area\representation.items |
          (SIZEOF (['EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE',
                    'EXPLICIT_DRAUGHTING.MAPPED_ITEM'] *
                   TYPEOF(item)) = 1)
      )) > 0)
    )) = 0;
    planar_box_in_sheet:
     SIZEOF (QUERY (ais <* USEDIN (SELF, 'EXPLICIT_DRAUGHTING.' +
                                         'AREA_IN_SET.IN_SET') |
      NOT (SIZEOF (QUERY (p_b <* QUERY (item <* ais.area\representation.items |
                  ('EXPLICIT_DRAUGHTING.PLANAR_BOX' IN TYPEOF(item))) |
        NOT (SIZEOF (USEDIN (p_b, 'EXPLICIT_DRAUGHTING.' +
                                  'PRESENTATION_SIZE.SIZE')) = 1)
      )) = 0)
     )) = 0;
    sheets_contain_views :
      SIZEOF(QUERY ( ais <* USEDIN(
          SELF,'EXPLICIT_DRAUGHTING.' +
           'AREA_IN_SET.IN_SET') | (
          NOT (SIZEOF(QUERY ( mi <*
          QUERY ( item <* ais.area\
          representation.items | ('EXPLICIT_DRAUGHTING.MAPPED_ITEM'
          IN TYPEOF(item)) ) | (NOT ('EXPLICIT_DRAUGHTING.PRESENTATION_VIEW'
          IN TYPEOF(mi.mapping_source.
          mapped_representation))) )) =
           0)) )) = 0;
    sheets_contain_placements :
      SIZEOF(QUERY ( ais <* USEDIN(
          SELF,'EXPLICIT_DRAUGHTING.' +
           'AREA_IN_SET.IN_SET') | (
```

```
        NOT (SIZEOF(QUERY ( a2p <*
        QUERY ( item <* ais.area\
        representation.items | ('EXPLICIT_DRAUGHTING.AXIS2_PLACEMENT'
        IN TYPEOF(item)) ) | (NOT (
        SIZEOF(USEDIN(a2p,
        'EXPLICIT_DRAUGHTING.' +
        'MAPPED_ITEM.MAPPING_TARGET'))
        > 0)) )) = 0)) )) = 0;
    sheets_have_unique_context :
      SIZEOF(QUERY ( ais <* USEDIN(
        SELF,'EXPLICIT_DRAUGHTING.AREA_IN_SET.IN_SET')
         | (NOT (SIZEOF(ais.area\
        representation.
        context_of_items.
        representations_in_context) =
         1)) )) = 0;
    views_presented_once :
      SIZEOF(QUERY ( ais <* USEDIN(
        SELF,'EXPLICIT_DRAUGHTING.' +
         'AREA_IN_SET.IN_SET') | (
        NOT (SIZEOF(QUERY ( mi <*
        QUERY ( item <* ais.area\
        representation.items | ('EXPLICIT_DRAUGHTING.MAPPED_ITEM'
        IN TYPEOF(item)) ) | (NOT (
        SIZEOF(USEDIN(mi.
        mapping_source.
        mapped_representation,'EXPLICIT_DRAUGHTING.REPRESENTATION_MAP.'
        + 'MAPPED_REPRESENTATION')) =
         1)) )) = 0)) )) = 0;
    allowable_items_of_view :
      SIZEOF(QUERY ( ais <* USEDIN(
        SELF,'EXPLICIT_DRAUGHTING.' +
         'AREA_IN_SET.IN_SET') | (
        NOT (SIZEOF(QUERY ( mi <*
        QUERY ( item <* ais.area\
        representation.items | ('EXPLICIT_DRAUGHTING.MAPPED_ITEM'
        IN TYPEOF(item)) ) | (NOT (
        SIZEOF(QUERY ( pv_item <* mi.
        mapping_source.
        mapped_representation\
        representation.items | (NOT (
        SIZEOF(['EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE',
        'EXPLICIT_DRAUGHTING.CAMERA_IMAGE',
        'EXPLICIT_DRAUGHTING.AXIS2_PLACEMENT']
        * TYPEOF(pv_item)) = 1)) )) =
         0)) )) = 0)) )) = 0;
    views_contain_one_projection :
      SIZEOF(QUERY ( ais <* USEDIN(
        SELF,'EXPLICIT_DRAUGHTING.' +
         'AREA_IN_SET.IN_SET') | (
        NOT (SIZEOF(QUERY ( mi <*
        QUERY ( item <* ais.area\
```

```
          representation.items | ('EXPLICIT_DRAUGHTING.MAPPED_ITEM'
          IN TYPEOF(item)) ) | (NOT (
          SIZEOF(QUERY ( pv_item <* mi.
          mapping_source.
          mapped_representation\
          representation.items | ('EXPLICIT_DRAUGHTING.CAMERA_IMAGE'
          IN TYPEOF(pv_item)) )) = 1)) ))
          = 0)) )) = 0;
    views_contain_one_placement :
      SIZEOF(QUERY ( ais <* USEDIN(
          SELF,'EXPLICIT_DRAUGHTING.' +
           'AREA_IN_SET.IN_SET') | (
          NOT (SIZEOF(QUERY ( mi <*
          QUERY ( item <* ais.area\
          representation.items | ('EXPLICIT_DRAUGHTING.MAPPED_ITEM'
          IN TYPEOF(item)) ) | (NOT (
          SIZEOF(QUERY ( a2p <*
          QUERY ( pv_item <* mi.
          mapping_source.
          mapped_representation\
          representation.items | ('EXPLICIT_DRAUGHTING.AXIS2_PLACEMENT'
          IN TYPEOF(pv_item)) ) |
          ('EXPLICIT_DRAUGHTING.MAPPING_SOURCE.MAPPING_ORIGIN'
          IN ROLESOF(a2p)) )) = 1)) ))
          = 0)) )) = 0;
    views_have_unique_context :
      SIZEOF(QUERY ( ais <* USEDIN(
          SELF,'EXPLICIT_DRAUGHTING.AREA_IN_SET.IN_SET')
           | (NOT (SIZEOF(
          QUERY ( mi <*
          QUERY ( item <* ais.area\
          representation.items | ('EXPLICIT_DRAUGHTING.MAPPED_ITEM'
          IN TYPEOF(item)) ) | (NOT (
          SIZEOF(mi.mapping_source.
          mapped_representation\
          representation.
          context_of_items.
          representations_in_context) =
           1)) )) = 0)) )) = 0;
END_ENTITY; -- draughting_drawing_revision

ENTITY draughting_elements
  SUBTYPE OF (draughting_callout);
  WHERE
    wr1: SIZEOF(QUERY ( l_c <* QUERY ( con <* SELF.contents | (
              'EXPLICIT_DRAUGHTING.LEADER_CURVE' IN TYPEOF(con)) ) | (NOT
              (SIZEOF(QUERY ( ldc <* USEDIN(l_c,'EXPLICIT_DRAUGHTING.' +
              'DRAUGHTING_CALLOUT.CONTENTS') | (
              'EXPLICIT_DRAUGHTING.LEADER_DIRECTED_CALLOUT' IN TYPEOF(ldc)) ))
              <= 1)) )) = 0;
    wr2: (NOT ('EXPLICIT_DRAUGHTING.DIMENSION_CURVE_DIRECTED_CALLOUT' IN
               TYPEOF(SELF))) OR (SIZEOF(QUERY ( con <* SELF.contents | (
```

```
            'EXPLICIT_DRAUGHTING.PROJECTION_CURVE' IN TYPEOF(con)) )) <=
             2);
    wr3: SIZEOF(QUERY ( rc <* USEDIN(SELF,
            'EXPLICIT_DRAUGHTING.DIMENSION_CALLOUT_' +
            'RELATIONSHIP.RELATING_DRAUGHTING_CALLOUT') | (rc.name =
            'primary') )) <= 1;
    wr4: SIZEOF(QUERY ( rc <* USEDIN(SELF,
            'EXPLICIT_DRAUGHTING.DIMENSION_CALLOUT_' +
            'RELATIONSHIP.RELATING_DRAUGHTING_CALLOUT') | (rc.name =
            'secondary') )) <= 1;
    wr5: SIZEOF(QUERY ( sec <* QUERY ( rc <* USEDIN(SELF,
            'EXPLICIT_DRAUGHTING.DIMENSION_CALLOUT_' +
            'RELATIONSHIP.RELATING_DRAUGHTING_CALLOUT') | (rc.name =
            'secondary') ) | (NOT (SIZEOF(QUERY ( prim <* USEDIN(SELF,
            'EXPLICIT_DRAUGHTING.DIMENSION_CALLOUT_' +
            'RELATIONSHIP.RELATING_DRAUGHTING_CALLOUT') | (prim.name =
            'primary') )) = 1)) )) = 0;
END_ENTITY; -- draughting_elements

ENTITY draughting_group_assignment
  SUBTYPE OF (group_assignment);
    items : SET [1:?] OF draughting_grouped_item;
END_ENTITY; -- draughting_group_assignment

ENTITY draughting_model
  SUBTYPE OF (representation);
  UNIQUE
    ur1 : name;
  WHERE
    wr1: SIZEOF(QUERY ( it <* SELF.items | (NOT (SIZEOF([
            'EXPLICIT_DRAUGHTING.MAPPED_ITEM',
            'EXPLICIT_DRAUGHTING.STYLED_ITEM',
            'EXPLICIT_DRAUGHTING.AXIS2_PLACEMENT'] * TYPEOF(it)) = 1)) ))
            = 0;
    wr2: SIZEOF(QUERY ( mi <* QUERY ( it <* SELF.items | (
            'EXPLICIT_DRAUGHTING.MAPPED_ITEM' IN TYPEOF(it)) ) | (NOT (
            SIZEOF(['EXPLICIT_DRAUGHTING.SHAPE_REPRESENTATION',
            'EXPLICIT_DRAUGHTING.DRAUGHTING_MODEL'] * TYPEOF(mi\
            mapped_item.mapping_source.mapped_representation)) = 1)) ))
            = 0;
    wr3: SIZEOF(QUERY ( smi <* QUERY ( si <* QUERY ( it <* SELF.items |
            ('EXPLICIT_DRAUGHTING.STYLED_ITEM' IN TYPEOF(it)) ) | (
            'EXPLICIT_DRAUGHTING.MAPPED_ITEM' IN TYPEOF(si\styled_item.
            item)) ) | (NOT (('EXPLICIT_DRAUGHTING.SHAPE_REPRESENTATION'
            IN TYPEOF(smi\styled_item.item\mapped_item.mapping_source.
            mapped_representation)) AND (SIZEOF(QUERY ( sty <* smi\
            styled_item.styles | (NOT (SIZEOF(QUERY ( psa <* sty.styles
             | (NOT ('EXPLICIT_DRAUGHTING.CURVE_STYLE' IN TYPEOF(psa))) ))
            = 1)) )) = 1))) )) = 0;
END_ENTITY; -- draughting_model

ENTITY draughting_organization_assignment
```

```
  SUBTYPE OF (organization_assignment);
    assigned_items : SET [1:?] OF draughting_organization_item;
END_ENTITY; -- draughting_organization_assignment

ENTITY draughting_person_and_organization_assignment
  SUBTYPE OF (person_and_organization_assignment);
    assigned_items : SET [1:?] OF draughting_organization_item;
END_ENTITY; -- draughting_person_and_organization_assignment

ENTITY draughting_person_assignment
  SUBTYPE OF (person_assignment);
    assigned_items : SET [1:?] OF draughting_organization_item;
END_ENTITY; -- draughting_person_assignment

ENTITY draughting_pre_defined_colour
  SUBTYPE OF (pre_defined_colour);
  WHERE
    wr1: SELF.name IN ['red','green','blue','yellow','magenta','cyan',
              'black','white'];
END_ENTITY; -- draughting_pre_defined_colour

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; -- draughting_pre_defined_curve_font

ENTITY draughting_pre_defined_text_font
  SUBTYPE OF (pre_defined_text_font);
  WHERE
    wr1: SELF.name IN ['ISO 3098-1 font'];
END_ENTITY; -- draughting_pre_defined_text_font

ENTITY draughting_presented_item
  SUBTYPE OF (presented_item);
    items        : SET [1:?] OF draughting_presented_item_select;
  WHERE
    presented_item_presentation:
      SIZEOF(QUERY ( pir <* USEDIN(SELF,
          'EXPLICIT_DRAUGHTING.' + 'PRESENTED_ITEM_REPRESENTATION.ITEM')
           | (NOT ('EXPLICIT_DRAUGHTING.DRAWING_REVISION'
          IN TYPEOF(pir.presentation))) )) = 0;
END_ENTITY; -- draughting_presented_item

ENTITY draughting_security_classification_assignment
  SUBTYPE OF (security_classification_assignment);
    assigned_items : SET [1:?] OF classified_item;
END_ENTITY; -- draughting_security_classification_assignment

ENTITY draughting_specification_reference
```

```
  SUBTYPE OF (document_reference);
    specified_items : SET [1:?] OF specified_item;
  WHERE
    wr1: SELF.assigned_document.kind.product_data_type =
             'draughting specification';
END_ENTITY; -- draughting_specification_reference

ENTITY draughting_subfigure_representation
  SUBTYPE OF (symbol_representation);
  WHERE
    wr1: SIZEOF(QUERY ( item <* SELF\representation.items | (NOT (
             SIZEOF(['EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE',
             'EXPLICIT_DRAUGHTING.DRAUGHTING_CALLOUT',
             'EXPLICIT_DRAUGHTING.AXIS2_PLACEMENT'] * TYPEOF(item)) = 1)) ))
             = 0;
    wr2: SIZEOF(QUERY ( item <* SELF\representation.items | (SIZEOF([
             'EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE',
             'EXPLICIT_DRAUGHTING.DRAUGHTING_CALLOUT'] * TYPEOF(item)) =
             1) )) >= 1;
      wr3: SIZEOF(QUERY ( srm <* QUERY ( rm <* USEDIN(SELF,
               'EXPLICIT_DRAUGHTING.' + 'REPRESENTATION_MAP.' +
               'MAPPED_REPRESENTATION') | (
               'EXPLICIT_DRAUGHTING.SYMBOL_REPRESENTATION_MAP' IN TYPEOF(rm)) )
                | (NOT (SIZEOF(QUERY ( a_s <* QUERY ( mi <* srm.map_usage
                | ('EXPLICIT_DRAUGHTING.ANNOTATION_SYMBOL' IN TYPEOF(mi)) )
                | (NOT (SIZEOF(QUERY ( aso <* USEDIN(a_s,
               'EXPLICIT_DRAUGHTING.' + 'STYLED_ITEM.ITEM') | (NOT ((
               'EXPLICIT_DRAUGHTING.' + 'ANNOTATION_SUBFIGURE_OCCURRENCE')
              IN TYPEOF(aso))) )) = 0)) )) = 0)) )) > 0;
    wr4: NOT acyclic_mapped_item_usage(SELF);
    wr5: SIZEOF(SELF.context_of_items.representations_in_context) = 1;
END_ENTITY; -- draughting_subfigure_representation

ENTITY draughting_symbol_representation
  SUBTYPE OF (symbol_representation);
  UNIQUE
    ur1 : name;
  WHERE
    wr1: SIZEOF(QUERY ( item <* SELF\representation.items | (NOT (
             SIZEOF(['EXPLICIT_DRAUGHTING.' +
             'ANNOTATION_CURVE_OCCURRENCE','EXPLICIT_DRAUGHTING.' +
             'ANNOTATION_SYMBOL_OCCURRENCE','EXPLICIT_DRAUGHTING.' +
             'ANNOTATION_FILL_AREA_OCCURRENCE','EXPLICIT_DRAUGHTING.' +
             'ANNOTATION_TEXT_OCCURRENCE','EXPLICIT_DRAUGHTING.' +
             'AXIS2_PLACEMENT'] * TYPEOF(item)) = 1)) )) = 0;
    wr2: SIZEOF(QUERY ( item <* SELF\representation.items | (SIZEOF([
             'EXPLICIT_DRAUGHTING.' + 'ANNOTATION_CURVE_OCCURRENCE',
             'EXPLICIT_DRAUGHTING.' + 'ANNOTATION_SYMBOL_OCCURRENCE',
             'EXPLICIT_DRAUGHTING.' + 'ANNOTATION_FILL_AREA_OCCURRENCE',
             'EXPLICIT_DRAUGHTING.' + 'ANNOTATION_TEXT_OCCURRENCE'] *
             TYPEOF(item)) = 1) )) >= 1;
      wr3: SIZEOF(QUERY ( item <* SELF\representation.items | (
```

```
            'EXPLICIT_DRAUGHTING.ANNOTATION_SUBFIGURE_OCCURRENCE' IN
            TYPEOF(item)) )) = 0;
      wr4: SIZEOF(QUERY ( srm <* QUERY ( rm <* USEDIN(SELF,
            'EXPLICIT_DRAUGHTING.' + 'REPRESENTATION_MAP.' +
            'MAPPED_REPRESENTATION') | (
            'EXPLICIT_DRAUGHTING.SYMBOL_REPRESENTATION_MAP' IN TYPEOF(rm)) )
             | (NOT (SIZEOF(QUERY ( a_s <* QUERY ( mi <* srm.map_usage
             | ('EXPLICIT_DRAUGHTING.ANNOTATION_SYMBOL' IN TYPEOF(mi)) )
             | (NOT (SIZEOF(QUERY ( aso <* USEDIN(a_s,
            'EXPLICIT_DRAUGHTING.' + 'STYLED_ITEM.ITEM') | (NOT ((
            'EXPLICIT_DRAUGHTING.' + 'ANNOTATION_SYMBOL_OCCURRENCE') IN
            TYPEOF(aso))) )) = 0)) )) = 0)) )) > 0;
    wr5: NOT acyclic_mapped_item_usage(SELF);
    wr6: SIZEOF(SELF.context_of_items.representations_in_context) = 1;
END_ENTITY; -- draughting_symbol_representation

ENTITY draughting_text_literal_with_delineation
  SUBTYPE OF (text_literal_with_delineation);
  WHERE
    wr1: SELF\text_literal_with_delineation.delineation IN ['underline',
           'overline'];
END_ENTITY; -- draughting_text_literal_with_delineation

ENTITY draughting_title;
    items    : SET [1:?] OF draughting_titled_item;
    language : label;
    contents : text;
END_ENTITY; -- draughting_title

ENTITY drawing_definition;
    drawing_number : identifier;
    drawing_type   : OPTIONAL label;
END_ENTITY; -- drawing_definition

ENTITY drawing_revision
  SUBTYPE OF (presentation_set);
    revision_identifier : identifier;
    drawing_identifier  : drawing_definition;
    intended_scale      : OPTIONAL text;
  UNIQUE
    ur1 : revision_identifier, drawing_identifier;
END_ENTITY; -- drawing_revision

ENTITY drawing_sheet_layout
  SUBTYPE OF (draughting_symbol_representation);
END_ENTITY; -- drawing_sheet_layout

ENTITY drawing_sheet_revision
  SUBTYPE OF (presentation_area);
    revision_identifier : identifier;
  WHERE
    wr1: SIZEOF(QUERY ( item <* SELF.items | ((
```

```
            'EXPLICIT_DRAUGHTING.MAPPED_ITEM' IN TYPEOF(item)) AND (
            'EXPLICIT_DRAUGHTING.DRAWING_SHEET_REVISION' IN TYPEOF(item\
            mapped_item.mapping_source.mapped_representation))) )) = 0;
END_ENTITY; -- drawing_sheet_revision

ENTITY drawing_sheet_revision_usage
  SUBTYPE OF (area_in_set);
    sheet_number : identifier;
  UNIQUE
    ur1 : sheet_number, in_set;
  WHERE
    wr1: ('EXPLICIT_DRAUGHTING.DRAWING_SHEET_REVISION' IN TYPEOF(SELF\
             area_in_set.area)) AND (
             'EXPLICIT_DRAUGHTING.DRAWING_REVISION' IN TYPEOF(SELF\
             area_in_set.in_set));
END_ENTITY; -- drawing_sheet_revision_usage

ENTITY ellipse
  SUBTYPE OF (conic);
    semi_axis_1 : positive_length_measure;
    semi_axis_2 : positive_length_measure;
END_ENTITY; -- ellipse

ENTITY external_source;
    source_id : source_item;
END_ENTITY; -- external_source

ENTITY externally_defined_curve_font
  SUBTYPE OF (externally_defined_item);
END_ENTITY; -- externally_defined_curve_font

ENTITY externally_defined_hatch_style
  SUBTYPE OF (externally_defined_item, geometric_representation_item);
END_ENTITY; -- externally_defined_hatch_style

ENTITY externally_defined_item;
    item_id : source_item;
    source  : external_source;
END_ENTITY; -- externally_defined_item

ENTITY externally_defined_symbol
  SUBTYPE OF (externally_defined_item);
END_ENTITY; -- externally_defined_symbol

ENTITY externally_defined_text_font
  SUBTYPE OF (externally_defined_item);
END_ENTITY; -- externally_defined_text_font

ENTITY externally_defined_tile_style
  SUBTYPE OF (externally_defined_item, geometric_representation_item);
END_ENTITY; -- externally_defined_tile_style
```

```
ENTITY fill_area_style;
    name        : label;
    fill_styles : SET [1:?] OF fill_style_select;
  WHERE
    wr1: SIZEOF(QUERY ( fill_style <* SELF.fill_styles | ((
             'EXPLICIT_DRAUGHTING.' + 'FILL_AREA_STYLE_COLOUR') IN
             TYPEOF(fill_style)) )) <= 1;
END_ENTITY; -- fill_area_style

ENTITY fill_area_style_colour;
    name        : label;
    fill_colour : colour;
END_ENTITY; -- fill_area_style_colour

ENTITY fill_area_style_hatching
  SUBTYPE OF (geometric_representation_item);
    hatch_line_appearance         : curve_style;
    start_of_next_hatch_line      : one_direction_repeat_factor;
    point_of_reference_hatch_line : cartesian_point;
    pattern_start                 : cartesian_point;
    hatch_line_angle              : plane_angle_measure;
END_ENTITY; -- fill_area_style_hatching

ENTITY fill_area_style_tile_symbol_with_style
  SUBTYPE OF (geometric_representation_item);
    symbol : annotation_symbol_occurrence;
END_ENTITY; -- fill_area_style_tile_symbol_with_style

ENTITY fill_area_style_tiles
  SUBTYPE OF (geometric_representation_item);
    tiling_pattern : two_direction_repeat_factor;
    tiles          : SET [1:?] OF fill_area_style_tile_shape_select;
    tiling_scale   : positive_ratio_measure;
END_ENTITY; -- fill_area_style_tiles

ENTITY geometric_curve_set
  SUBTYPE OF (geometric_set);
  WHERE
    wr1: SIZEOF(QUERY ( temp <* SELF\geometric_set.elements | (
             'EXPLICIT_DRAUGHTING.SURFACE' IN TYPEOF(temp)) )) = 0;
END_ENTITY; -- geometric_curve_set

ENTITY geometric_representation_context
  SUBTYPE OF (representation_context);
    coordinate_space_dimension : dimension_count;
END_ENTITY; -- geometric_representation_context

ENTITY geometric_representation_item
  SUPERTYPE OF (ONEOF (point,direction,vector,placement,curve,
      geometric_set))
  SUBTYPE OF (representation_item);
  DERIVE
```

```
    dim : dimension_count := dimension_of(SELF);
  WHERE
    wr1: SIZEOF(QUERY ( using_rep <* using_representations(SELF) | (NOT
            ('EXPLICIT_DRAUGHTING.GEOMETRIC_REPRESENTATION_CONTEXT' IN
            TYPEOF(using_rep.context_of_items))) )) = 0;
END_ENTITY; -- geometric_representation_item

ENTITY geometric_set
  SUPERTYPE OF (geometric_curve_set)
  SUBTYPE OF (geometric_representation_item);
    elements : SET [1:?] OF geometric_set_select;
END_ENTITY; -- geometric_set

ENTITY geometrical_tolerance_callout
  SUBTYPE OF (draughting_callout);
END_ENTITY; -- geometrical_tolerance_callout

ENTITY geometrically_bounded_2d_wireframe_representation
  SUBTYPE OF (shape_representation);
  WHERE
    wr1: SELF.context_of_items\geometric_representation_context.
            coordinate_space_dimension = 2;
    wr2: SIZEOF(QUERY ( item <* SELF\representation.items | (NOT (
            SIZEOF(TYPEOF(item) * [
            'EXPLICIT_DRAUGHTING.GEOMETRIC_CURVE_SET',
            'EXPLICIT_DRAUGHTING.AXIS2_PLACEMENT_2D',
            'EXPLICIT_DRAUGHTING.MAPPED_ITEM']) = 1)) )) = 0;
    wr3: SIZEOF(QUERY ( gcs <* QUERY ( item <* SELF\representation.items
             | ('EXPLICIT_DRAUGHTING.GEOMETRIC_CURVE_SET' IN TYPEOF(item)) )
             | (NOT (SIZEOF(QUERY ( elem <* gcs\geometric_set.elements
             | (NOT (SIZEOF(TYPEOF(elem) * ['EXPLICIT_DRAUGHTING.LINE',
            'EXPLICIT_DRAUGHTING.HYPERBOLA',
            'EXPLICIT_DRAUGHTING.PARABOLA']) = 0)) )) = 0)) )) = 0;
    wr4: SIZEOF(QUERY ( mi <* QUERY ( item <* SELF\representation.items
             | ('EXPLICIT_DRAUGHTING.MAPPED_ITEM' IN TYPEOF(item)) ) | (
            NOT ('EXPLICIT_DRAUGHTING.' +
            'GEOMETRICALLY_BOUNDED_2D_WIREFRAME_REPRESENTATION'
            IN TYPEOF(mi\mapped_item.mapping_source.
            mapped_representation))) )) = 0;
    wr5: SIZEOF(QUERY ( gcs <* QUERY ( item <* SELF\representation.items
             | ('EXPLICIT_DRAUGHTING.GEOMETRIC_CURVE_SET' IN TYPEOF(item)) )
             | (NOT (SIZEOF(QUERY ( pl <* QUERY ( elem <* gcs\
            geometric_set.elements | ('EXPLICIT_DRAUGHTING.POLYLINE' IN
            TYPEOF(elem)) ) | (NOT (SIZEOF(pl\polyline.points) > 2)) ))
            = 0)) )) = 0;
    wr6: SIZEOF(QUERY ( item <* SELF\representation.items | (SIZEOF(
            TYPEOF(item) * ['EXPLICIT_DRAUGHTING.GEOMETRIC_CURVE_SET',
            'EXPLICIT_DRAUGHTING.MAPPED_ITEM']) = 1) )) >= 1;
    wr7: SIZEOF(QUERY ( gcs <* QUERY ( item <* SELF.items | (
            'EXPLICIT_DRAUGHTING.GEOMETRIC_CURVE_SET' IN TYPEOF(item)) )
             | (NOT (SIZEOF(QUERY ( pnt <* QUERY ( elem <* gcs\
            geometric_set.elements | ('EXPLICIT_DRAUGHTING.POINT' IN
```

```
            TYPEOF(elem)) ) | (NOT (SIZEOF(TYPEOF(pnt) * [
            'EXPLICIT_DRAUGHTING.CARTESIAN_POINT',
            'EXPLICIT_DRAUGHTING.POINT_ON_CURVE']) = 1)) )) = 0)) )) = 0;
END_ENTITY; -- geometrically_bounded_2d_wireframe_representation

ENTITY global_unit_assigned_context
  SUBTYPE OF (representation_context);
    units : SET [1:?] OF unit;
END_ENTITY; -- global_unit_assigned_context

ENTITY group;
    name        : label;
    description : text;
END_ENTITY; -- group

ENTITY group_assignment
  ABSTRACT SUPERTYPE;
    assigned_group : group;
END_ENTITY; -- group_assignment

ENTITY group_relationship;
    name          : label;
    description   : text;
    relating_group : group;
    related_group  : group;
END_ENTITY; -- group_relationship

ENTITY hyperbola
  SUBTYPE OF (conic);
    semi_axis      : positive_length_measure;
    semi_imag_axis : positive_length_measure;
END_ENTITY; -- hyperbola

ENTITY invisibility;
    invisible_items : SET [1:?] OF invisible_item;
END_ENTITY; -- invisibility

ENTITY leader_curve
  SUBTYPE OF (annotation_curve_occurrence);
WHERE
  wr1: SIZEOF( USEDIN( SELF, 'EXPLICIT_DRAUGHTING.'+
       'LEADER_DIRECTED_CALLOUT.CONTENTS')) >= 1;
END_ENTITY; -- leader_curve

ENTITY leader_directed_callout
  SUBTYPE OF (draughting_callout);
  WHERE
    wr1: SIZEOF(QUERY ( l_1 <* SELF.contents | (('EXPLICIT_DRAUGHTING.'
            + 'LEADER_CURVE') IN TYPEOF(l_1)) )) >= 1;
    wr2: SIZEOF(SELF.contents) >= 2;
END_ENTITY; -- leader_directed_callout
```

```
ENTITY leader_directed_dimension
  SUBTYPE OF (leader_directed_callout);
  WHERE
    wr1: SIZEOF(QUERY ( con <* SELF.contents | (
             'EXPLICIT_DRAUGHTING.LEADER_CURVE' IN TYPEOF(con)) )) = 1;
END_ENTITY; -- leader_directed_dimension

ENTITY leader_terminator
  SUBTYPE OF (terminator_symbol);
  WHERE
    wr1: 'EXPLICIT_DRAUGHTING.LEADER_CURVE' IN TYPEOF(SELF\
             terminator_symbol.annotated_curve);
END_ENTITY; -- leader_terminator

ENTITY length_measure_with_unit
  SUBTYPE OF (measure_with_unit);
  WHERE
    wr1: 'EXPLICIT_DRAUGHTING.LENGTH_UNIT' IN TYPEOF(SELF\
             measure_with_unit.unit_component);
END_ENTITY; -- length_measure_with_unit

ENTITY length_unit
  SUBTYPE OF (named_unit);
  WHERE
    wr1: (SELF\named_unit.dimensions.length_exponent = 1) AND (SELF\
             named_unit.dimensions.mass_exponent = 0) AND (SELF\
             named_unit.dimensions.time_exponent = 0) AND (SELF\
             named_unit.dimensions.electric_current_exponent = 0) AND (
             SELF\named_unit.dimensions.
             thermodynamic_temperature_exponent = 0) AND (SELF\named_unit
             .dimensions.amount_of_substance_exponent = 0) AND (SELF\
             named_unit.dimensions.luminous_intensity_exponent = 0);
END_ENTITY; -- length_unit

ENTITY line
  SUBTYPE OF (curve);
    pnt : cartesian_point;
    dir : vector;
  WHERE
    wr1: dir.dim = pnt.dim;
END_ENTITY; -- line

ENTITY linear_dimension
  SUBTYPE OF (dimension_curve_directed_callout);
END_ENTITY; -- linear_dimension

ENTITY mapped_item
  SUBTYPE OF (representation_item);
    mapping_source : representation_map;
    mapping_target : representation_item;
  WHERE
    wr1: acyclic_mapped_representation(using_representations(SELF),[SELF]);
```

```
END_ENTITY; -- mapped_item

ENTITY measure_with_unit
  SUPERTYPE OF (ONEOF (length_measure_with_unit,
      plane_angle_measure_with_unit));
    value_component : measure_value;
    unit_component  : unit;
  WHERE
    wr1: valid_units(SELF);
END_ENTITY; -- measure_with_unit

ENTITY named_unit
  SUPERTYPE OF (ONEOF (si_unit,conversion_based_unit) ANDOR ONEOF (
      length_unit,plane_angle_unit));
    dimensions : dimensional_exponents;
END_ENTITY; -- named_unit

ENTITY offset_curve_2d
  SUBTYPE OF (curve);
    basis_curve    : curve;
    distance       : length_measure;
    self_intersect : LOGICAL;
  WHERE
    wr1: basis_curve.dim = 2;
END_ENTITY; -- offset_curve_2d

ENTITY one_direction_repeat_factor
  SUBTYPE OF (geometric_representation_item);
    repeat_factor : vector;
END_ENTITY; -- one_direction_repeat_factor

ENTITY ordinate_dimension
  SUBTYPE OF (projection_directed_callout);
END_ENTITY; -- ordinate_dimension

ENTITY organization;
    id          : OPTIONAL identifier;
    name        : label;
    description : text;
END_ENTITY; -- organization

ENTITY organization_assignment
  ABSTRACT SUPERTYPE;
    assigned_organization : organization;
    role                  : organization_role;
END_ENTITY; -- organization_assignment

ENTITY organization_role;
    name : label;
END_ENTITY; -- organization_role
```

```
ENTITY organizational_address
  SUBTYPE OF (address);
    organizations : SET [1:?] OF organization;
    description   : text;
END_ENTITY; -- organizational_address

ENTITY parabola
  SUBTYPE OF (conic);
    focal_dist : length_measure;
  WHERE
    wr1: focal_dist <> 0;
END_ENTITY; -- parabola

ENTITY person;
    id            : identifier;
    last_name     : OPTIONAL label;
    first_name    : OPTIONAL label;
    middle_names  : OPTIONAL LIST [1:?] OF label;
    prefix_titles : OPTIONAL LIST [1:?] OF label;
    suffix_titles : OPTIONAL LIST [1:?] OF label;
  UNIQUE
    ur1 : id;
  WHERE
    wr1: EXISTS(last_name) OR EXISTS(first_name);
END_ENTITY; -- person

ENTITY person_and_organization;
    the_person       : person;
    the_organization : organization;
END_ENTITY; -- person_and_organization

ENTITY person_and_organization_assignment
  ABSTRACT SUPERTYPE;
    assigned_person_and_organization : person_and_organization;
    role                             : person_and_organization_role;
END_ENTITY; -- person_and_organization_assignment

ENTITY person_and_organization_role;
    name : label;
END_ENTITY; -- person_and_organization_role

ENTITY person_assignment
  ABSTRACT SUPERTYPE;
    assigned_person : person;
    role            : person_role;
END_ENTITY; -- person_assignment

ENTITY person_role;
    name : label;
END_ENTITY; -- person_role

ENTITY personal_address
```

```
  SUBTYPE OF (address);
    people      : SET [1:?] OF person;
    description : text;
END_ENTITY; -- personal_address

ENTITY placement
  SUPERTYPE OF (axis2_placement_2d)
  SUBTYPE OF (geometric_representation_item);
    location : cartesian_point;
END_ENTITY; -- placement

ENTITY planar_box
  SUBTYPE OF (planar_extent);
    placement : axis2_placement;
END_ENTITY; -- planar_box

ENTITY planar_extent
  SUBTYPE OF (geometric_representation_item);
    size_in_x : length_measure;
    size_in_y : length_measure;
END_ENTITY; -- planar_extent

ENTITY plane_angle_measure_with_unit
  SUBTYPE OF (measure_with_unit);
  WHERE
    wr1: 'EXPLICIT_DRAUGHTING.PLANE_ANGLE_UNIT' IN TYPEOF(SELF\
              measure_with_unit.unit_component);
END_ENTITY; -- plane_angle_measure_with_unit

ENTITY plane_angle_unit
  SUBTYPE OF (named_unit);
  WHERE
    wr1: (SELF\named_unit.dimensions.length_exponent = 0) AND (SELF\
              named_unit.dimensions.mass_exponent = 0) AND (SELF\
              named_unit.dimensions.time_exponent = 0) AND (SELF\
              named_unit.dimensions.electric_current_exponent = 0) AND (
              SELF\named_unit.dimensions.
              thermodynamic_temperature_exponent = 0) AND (SELF\named_unit
              .dimensions.amount_of_substance_exponent = 0) AND (SELF\
              named_unit.dimensions.luminous_intensity_exponent = 0);
END_ENTITY; -- plane_angle_unit

ENTITY point
  SUPERTYPE OF (ONEOF (cartesian_point,point_on_curve))
  SUBTYPE OF (geometric_representation_item);
END_ENTITY; -- point

ENTITY point_on_curve
  SUBTYPE OF (point);
    basis_curve     : curve;
    point_parameter : parameter_value;
END_ENTITY; -- point_on_curve
```

```
ENTITY polyline
  SUBTYPE OF (bounded_curve);
    points : LIST [2:?] OF cartesian_point;
END_ENTITY; -- polyline

ENTITY pre_defined_colour
  SUBTYPE OF (pre_defined_item, colour);
END_ENTITY; -- pre_defined_colour

ENTITY pre_defined_curve_font
  SUBTYPE OF (pre_defined_item);
END_ENTITY; -- pre_defined_curve_font

ENTITY pre_defined_dimension_symbol
  SUBTYPE OF (pre_defined_symbol);
  WHERE
    wr1: SELF.name IN ['arc length','conical taper','counterbore',
              'countersink','depth','diameter','plus minus','radius',
              'slope','spherical diameter','spherical radius','square'];
END_ENTITY; -- pre_defined_dimension_symbol

ENTITY pre_defined_geometrical_tolerance_symbol
  SUBTYPE OF (pre_defined_symbol);
  WHERE
    wr1: SELF.name IN ['angularity','circular runout','circularity',
              'concentricity','cylindricity','diameter','flatness',
              'least material condition','maximum material condition',
              'parallelism','perpendicularity','position',
              'profile of a line','profile of a surface',
              'projected tolerance zone','regardless of feature size',
              'straightness','symmetry','target point','total runout'];
END_ENTITY; -- pre_defined_geometrical_tolerance_symbol

ENTITY pre_defined_item;
    name : label;
END_ENTITY; -- pre_defined_item

ENTITY pre_defined_point_marker_symbol
  SUBTYPE OF (pre_defined_symbol);
  WHERE
    wr1: SELF.name IN ['asterisk','circle','dot','plus','square',
              'triangle','x'];
END_ENTITY; -- pre_defined_point_marker_symbol

ENTITY pre_defined_symbol
  SUBTYPE OF (pre_defined_item);
END_ENTITY; -- pre_defined_symbol

ENTITY pre_defined_terminator_symbol
  SUBTYPE OF (pre_defined_symbol);
  WHERE
```

```
    wr1: SELF.name IN ['blanked arrow','blanked box','blanked dot',
              'dimension origin','filled arrow','filled box','filled dot',
              'integral symbol','open arrow','slash','unfilled arrow'];
END_ENTITY; -- pre_defined_terminator_symbol

ENTITY pre_defined_text_font
  SUBTYPE OF (pre_defined_item);
END_ENTITY; -- pre_defined_text_font

ENTITY presentation_area
  SUBTYPE OF (presentation_representation);
  WHERE
    wr1: (SIZEOF(QUERY ( ais <* USEDIN(SELF,'EXPLICIT_DRAUGHTING.' +
              'AREA_IN_SET.AREA') | (SIZEOF(USEDIN(ais,
              'EXPLICIT_DRAUGHTING.' + 'PRESENTATION_SIZE.UNIT')) = 1) ))
              > 0) OR (SIZEOF(USEDIN(SELF,'EXPLICIT_DRAUGHTING.' +
              'PRESENTATION_SIZE.UNIT')) = 1);
END_ENTITY; -- presentation_area

ENTITY presentation_layer_assignment;
    name           : label;
    description    : text;
    assigned_items : SET [1:?] OF layered_item;
END_ENTITY; -- presentation_layer_assignment

ENTITY presentation_layer_usage;
    assignment   : presentation_layer_assignment;
    presentation : presentation_representation;
  UNIQUE
    ur1 : assignment, presentation;
END_ENTITY; -- presentation_layer_usage

ENTITY presentation_representation
  SUBTYPE OF (representation);
  WHERE
    wr1: SELF\representation.context_of_items\
              geometric_representation_context.coordinate_space_dimension
              = 2;
    wr2: 'EXPLICIT_DRAUGHTING.GEOMETRIC_REPRESENTATION_CONTEXT' IN
              TYPEOF(SELF\representation.context_of_items);
END_ENTITY; -- presentation_representation

ENTITY presentation_set;
  INVERSE
    areas : SET [1:?] OF area_in_set FOR in_set;
END_ENTITY; -- presentation_set

ENTITY presentation_size;
    unit : presentation_size_assignment_select;
    size : planar_box;
  WHERE
    wr1: (('EXPLICIT_DRAUGHTING.PRESENTATION_REPRESENTATION' IN TYPEOF(
```

```
            SELF.unit)) AND item_in_context(SELF.size,SELF.unit\
            representation.context_of_items)) OR ((
            'EXPLICIT_DRAUGHTING.AREA_IN_SET' IN TYPEOF(SELF.unit)) AND
            (SIZEOF(QUERY ( ais <* SELF.unit\area_in_set.in_set.areas |
            (NOT item_in_context(SELF.size,ais.area\representation.
            context_of_items)) )) = 0));
END_ENTITY; -- presentation_size

ENTITY presentation_style_assignment;
    styles : SET [1:?] OF presentation_style_select;
  WHERE
    wr1: SIZEOF(QUERY ( style1 <* SELF.styles | (NOT (SIZEOF(
            QUERY ( style2 <* (SELF.styles - style1) | (NOT ((TYPEOF(
            style1) <> TYPEOF(style2)) OR (SIZEOF([
            'EXPLICIT_DRAUGHTING.' + 'SURFACE_STYLE_USAGE',
            'EXPLICIT_DRAUGHTING.' + 'EXTERNALLY_DEFINED_STYLE'] *
            TYPEOF(style1)) = 1))) )) = 0)) )) = 0;
    wr2: SIZEOF(QUERY ( style1 <* SELF.styles | (
            'EXPLICIT_DRAUGHTING.SURFACE_STYLE_USAGE' IN TYPEOF(style1)) ))
            <= 2;
END_ENTITY; -- presentation_style_assignment

ENTITY presentation_style_by_context
  SUBTYPE OF (presentation_style_assignment);
    style_context : style_context_select;
END_ENTITY; -- presentation_style_by_context

ENTITY presentation_view
  SUBTYPE OF (presentation_representation);
END_ENTITY; -- presentation_view

ENTITY presented_item
  ABSTRACT SUPERTYPE;
END_ENTITY; -- presented_item

ENTITY presented_item_representation;
    presentation : presentation_representation_select;
    item         : presented_item;
END_ENTITY; -- presented_item_representation

ENTITY product;
    id                : identifier;
    name              : label;
    description       : text;
    frame_of_reference : SET [1:?] OF product_context;
  UNIQUE
    ur1 : id;
END_ENTITY; -- product

ENTITY product_context
  SUBTYPE OF (application_context_element);
    discipline_type : label;
```

```
END_ENTITY; -- product_context

ENTITY product_definition;
    id                 : identifier;
    description        : text;
    formation          : product_definition_formation;
    frame_of_reference : product_definition_context;
END_ENTITY; -- product_definition

ENTITY product_definition_context
  SUBTYPE OF (application_context_element);
    life_cycle_stage : label;
END_ENTITY; -- product_definition_context

ENTITY product_definition_formation;
    id          : identifier;
    description : text;
    of_product  : product;
  UNIQUE
    ur1 : id, of_product;
END_ENTITY; -- product_definition_formation

ENTITY product_definition_shape
  SUBTYPE OF (property_definition);
  UNIQUE
    ur1 : definition;
  WHERE
    wr1: 'EXPLICIT_DRAUGHTING.CHARACTERIZED_PRODUCT_DEFINITION' IN
             TYPEOF(SELF\property_definition.definition);
END_ENTITY; -- product_definition_shape

ENTITY projection_curve
  SUBTYPE OF (annotation_curve_occurrence);
END_ENTITY; -- projection_curve

ENTITY projection_directed_callout
  SUBTYPE OF (draughting_callout);
  WHERE
    wr1: SIZEOF(QUERY ( p_1 <* SELF.contents | (
             'EXPLICIT_DRAUGHTING.PROJECTION_CURVE' IN TYPEOF(p_1)) )) =
             1;
    wr2: SIZEOF(SELF.contents) >= 2;
END_ENTITY; -- projection_directed_callout

ENTITY property_definition;
    name        : label;
    description : text;
    definition  : characterized_definition;
END_ENTITY; -- property_definition

ENTITY property_definition_representation;
```

```
    definition          : property_definition;
    used_representation : representation;
END_ENTITY; -- property_definition_representation

ENTITY quasi_uniform_curve
  SUBTYPE OF (b_spline_curve);
END_ENTITY; -- quasi_uniform_curve

ENTITY radius_dimension
  SUBTYPE OF (dimension_curve_directed_callout);
  WHERE
    wr1: SIZEOF(QUERY ( con <* SELF.contents | (
              'EXPLICIT_DRAUGHTING.PROJECTION_CURVE' IN TYPEOF(con)) )) <=
               1;
END_ENTITY; -- radius_dimension

ENTITY rational_b_spline_curve
  SUBTYPE OF (b_spline_curve);
    weights_data : LIST [2:?] OF REAL;
  DERIVE
    weights : ARRAY [0:upper_index_on_control_points] OF REAL :=
                list_to_array(weights_data,0,
                upper_index_on_control_points);
  WHERE
    wr1: SIZEOF(weights_data) = SIZEOF(SELF\b_spline_curve.
             control_points_list);
    wr2: curve_weights_positive(SELF);
END_ENTITY; -- rational_b_spline_curve

ENTITY representation;
    name             : label;
    items            : SET [1:?] OF representation_item;
    context_of_items : representation_context;
END_ENTITY; -- representation

ENTITY representation_context;
    context_identifier : identifier;
    context_type       : text;
  INVERSE
    representations_in_context : SET [1:?] OF representation FOR
                                     context_of_items;
END_ENTITY; -- representation_context

ENTITY representation_item;
    name : label;
  WHERE
    wr1: SIZEOF(using_representations(SELF)) > 0;
END_ENTITY; -- representation_item

ENTITY representation_map;
    mapping_origin        : representation_item;
    mapped_representation : representation;
```

```
  INVERSE
    map_usage : SET [1:?] OF mapped_item FOR mapping_source;
  WHERE
    wr1: item_in_context(SELF.mapping_origin,SELF.mapped_representation.
             context_of_items);
END_ENTITY; -- representation_map

ENTITY security_classification;
    name           : label;
    purpose        : text;
    security_level : security_classification_level;
END_ENTITY; -- security_classification

ENTITY security_classification_assignment
  ABSTRACT SUPERTYPE;
    assigned_security_classification : security_classification;
END_ENTITY; -- security_classification_assignment

ENTITY security_classification_level;
    name : label;
END_ENTITY; -- security_classification_level

ENTITY shape_definition_representation
  SUBTYPE OF (property_definition_representation);
  WHERE
    wr1: ('EXPLICIT_DRAUGHTING.SHAPE_DEFINITION' IN TYPEOF(SELF.
             definition.definition)) OR (
             'EXPLICIT_DRAUGHTING.PRODUCT_DEFINITION_SHAPE' IN TYPEOF(
             SELF.definition));
    wr2: 'EXPLICIT_DRAUGHTING.SHAPE_REPRESENTATION' IN TYPEOF(SELF.
             used_representation);
END_ENTITY; -- shape_definition_representation

ENTITY shape_representation
  SUBTYPE OF (representation);
END_ENTITY; -- shape_representation

ENTITY si_unit
  SUBTYPE OF (named_unit);
    prefix : OPTIONAL si_prefix;
    name   : si_unit_name;
  DERIVE
    SELF\named_unit.dimensions : dimensional_exponents :=
                                    dimensions_for_si_unit(SELF.name);
END_ENTITY; -- si_unit

ENTITY structured_dimension_callout
  SUBTYPE OF (draughting_callout);
  WHERE
    wr1: SIZEOF(TYPEOF(SELF) * [
             'EXPLICIT_DRAUGHTING.DATUM_FEATURE_CALLOUT',
             'EXPLICIT_DRAUGHTING.DATUM_TARGET_CALLOUT',
```

```
            'EXPLICIT_DRAUGHTING.GEOMETRICAL_TOLERANCE_CALLOUT',
            'EXPLICIT_DRAUGHTING.LEADER_DIRECTED_CALLOUT',
            'EXPLICIT_DRAUGHTING.PROJECTION_DIRECTED_CALLOUT',
            'EXPLICIT_DRAUGHTING.DIMENSION_CURVE_DIRECTED_CALLOUT']) = 0;
    wr2: SIZEOF(QUERY ( ato <* QUERY ( con <* SELF.contents | (
            'EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE' IN TYPEOF(
            con)) ) | (NOT (ato\representation_item.name IN
            ['dimension value', 'tolerance value','unit text'])) )) = 0;
    wr3: SIZEOF(QUERY ( ato <* QUERY ( con <* SELF.contents | (
            'EXPLICIT_DRAUGHTING.ANNOTATION_TEXT_OCCURRENCE' IN TYPEOF(
            con)) ) | (ato\representation_item.name = 'dimension value')
            )) >= 1;
    wr4: SIZEOF(QUERY ( dcr <* USEDIN(SELF,'EXPLICIT_DRAUGHTING.' +
            'DIMENSION_CALLOUT_COMPONENT_RELATIONSHIP.' +
            'RELATING_DRAUGHTING_CALLOUT') | (dcr.name = 'prefix') )) <=
             1;
    wr5: SIZEOF(QUERY ( dcr <* USEDIN(SELF,'EXPLICIT_DRAUGHTING.' +
            'DIMENSION_CALLOUT_COMPONENT_RELATIONSHIP.' +
            'RELATING_DRAUGHTING_CALLOUT') | (dcr.name = 'suffix') )) <=
             1;
END_ENTITY; -- structured_dimension_callout

ENTITY styled_item
  SUBTYPE OF (representation_item);
    styles : SET [1:?] OF presentation_style_assignment;
    item   : representation_item;
  WHERE
    wr1: (SIZEOF(SELF.styles) = 1) XOR (SIZEOF(QUERY ( pres_style <*
            SELF.styles | (NOT (
            'EXPLICIT_DRAUGHTING.PRESENTATION_STYLE_BY_CONTEXT' IN
            TYPEOF(pres_style))) )) = 0);
END_ENTITY; -- styled_item

ENTITY symbol_colour;
    colour_of_symbol : colour;
END_ENTITY; -- symbol_colour

ENTITY symbol_representation
  SUBTYPE OF (representation);
END_ENTITY; -- symbol_representation

ENTITY symbol_representation_map
  SUBTYPE OF (representation_map);
  WHERE
    wr1: 'EXPLICIT_DRAUGHTING.SYMBOL_REPRESENTATION' IN TYPEOF(SELF\
            representation_map.mapped_representation);
    wr2: 'EXPLICIT_DRAUGHTING.AXIS2_PLACEMENT' IN TYPEOF(SELF\
            representation_map.mapping_origin);
END_ENTITY; -- symbol_representation_map

ENTITY symbol_style;
    name            : label;
```

```
    style_of_symbol : symbol_style_select;
END_ENTITY; -- symbol_style

ENTITY symbol_target
  SUBTYPE OF (geometric_representation_item);
    placement : axis2_placement;
    x_scale   : positive_ratio_measure;
    y_scale   : positive_ratio_measure;
END_ENTITY; -- symbol_target

ENTITY terminator_symbol
  SUBTYPE OF (annotation_symbol_occurrence);
    annotated_curve : annotation_curve_occurrence;
END_ENTITY; -- terminator_symbol

ENTITY text_literal
  SUBTYPE OF (geometric_representation_item);
    literal   : presentable_text;
    placement : axis2_placement;
    alignment : text_alignment;
    path      : text_path;
    font      : font_select;
END_ENTITY; -- text_literal

ENTITY text_literal_with_associated_curves
  SUBTYPE OF (text_literal);
    associated_curves : SET [1:?] OF curve;
END_ENTITY; -- text_literal_with_associated_curves

ENTITY text_literal_with_blanking_box
  SUBTYPE OF (text_literal);
    blanking : planar_box;
END_ENTITY; -- text_literal_with_blanking_box

ENTITY text_literal_with_delineation
  SUBTYPE OF (text_literal);
    delineation : text_delineation;
END_ENTITY; -- text_literal_with_delineation

ENTITY text_literal_with_extent
  SUBTYPE OF (text_literal);
    extent : planar_extent;
END_ENTITY; -- text_literal_with_extent

ENTITY text_style;
    name                : label;
    character_appearance : character_style_select;
END_ENTITY; -- text_style

ENTITY text_style_for_defined_font;
    text_colour : colour;
END_ENTITY; -- text_style_for_defined_font
```

```
ENTITY text_style_with_box_characteristics
  SUBTYPE OF (text_style);
    characteristics : SET [1:4] OF box_characteristic_select;
  WHERE
    wr1: SIZEOF(QUERY ( c1 <* SELF.characteristics | (SIZEOF(
             QUERY ( c2 <* (SELF.characteristics - c1) | (TYPEOF(c1) =
             TYPEOF(c2)) )) > 0) )) = 0;
END_ENTITY; -- text_style_with_box_characteristics

ENTITY text_style_with_mirror
  SUBTYPE OF (text_style);
    mirror_placement : axis2_placement;
END_ENTITY; -- text_style_with_mirror

ENTITY trimmed_curve
  SUBTYPE OF (bounded_curve);
    basis_curve           : curve;
    trim_1                : SET [1:2] OF trimming_select;
    trim_2                : SET [1:2] OF trimming_select;
    sense_agreement       : BOOLEAN;
    master_representation : trimming_preference;
  WHERE
    wr1: (HIINDEX(trim_1) = 1) XOR (TYPEOF(trim_1[1]) <> TYPEOF(trim_1[2]));
    wr2: (HIINDEX(trim_2) = 1) XOR (TYPEOF(trim_2[1]) <> TYPEOF(trim_2[2]));
END_ENTITY; -- trimmed_curve

ENTITY two_direction_repeat_factor
  SUBTYPE OF (one_direction_repeat_factor);
    second_repeat_factor : vector;
END_ENTITY; -- two_direction_repeat_factor

ENTITY uniform_curve
  SUBTYPE OF (b_spline_curve);
END_ENTITY; -- uniform_curve

ENTITY vector
  SUBTYPE OF (geometric_representation_item);
    orientation : direction;
    magnitude   : length_measure;
  WHERE
    wr1: magnitude >= 0;
END_ENTITY; -- vector

RULE application_context_requires_ap_definition FOR (application_context,
             application_protocol_definition);
WHERE
  wr1: SIZEOF(QUERY ( ac <* application_context | (NOT (SIZEOF(
           QUERY ( apd <* application_protocol_definition | ((ac :=: apd.
           application) AND (apd.
           application_interpreted_model_schema_name =
           'EXPLICIT_DRAUGHTING') AND (ac.application = 'draughting')) ))
           = 1)) )) = 0;
```

```
END_RULE; -- application_context_requires_ap_definition

RULE approvals_are_assigned FOR (approval);
WHERE
  wr1: SIZEOF(QUERY ( a <* approval | (NOT (SIZEOF(USEDIN(a,
          'EXPLICIT_DRAUGHTING.' + 'APPROVAL_ASSIGNMENT.' +
          'ASSIGNED_APPROVAL')) >= 1)) )) = 0;
END_RULE; -- approvals_are_assigned

RULE calendar_date_instantiation FOR (calendar_date);
WHERE
  wr1: SIZEOF(QUERY ( each <* calendar_date | (NOT (SIZEOF(USEDIN(each,
          ''')) >= 1)) )) = 0;
END_RULE; -- calendar_date_instantiation

RULE camera_usage_mapped_representation FOR (camera_usage);
WHERE
  wr1: SIZEOF(QUERY ( cu <* camera_usage | (NOT (
          'EXPLICIT_DRAUGHTING.DRAUGHTING_MODEL' IN TYPEOF(cu.
          mapped_representation))) )) = 0;
END_RULE; -- camera_usage_mapped_representation

RULE colour_rgb_instantiation FOR (colour_rgb);
WHERE
  wr1: SIZEOF(QUERY ( each <* colour_rgb | (NOT (SIZEOF(USEDIN(each,''))
          >= 1)) )) = 0;
END_RULE; -- colour_rgb_instantiation

RULE compatible_dimension FOR (cartesian_point, direction,
          representation_context, geometric_representation_context);
WHERE
  wr1: SIZEOF(QUERY ( x <* cartesian_point | (SIZEOF(QUERY ( y <*
          geometric_representation_context | (item_in_context(x,y) AND (
          HIINDEX(x.coordinates) <> y.coordinate_space_dimension)) )) >
          0) )) = 0;
  wr2: SIZEOF(QUERY ( x <* direction | (SIZEOF(QUERY ( y <*
          geometric_representation_context | (item_in_context(x,y) AND (
          HIINDEX(x.direction_ratios) <> y.coordinate_space_dimension)) ))
          > 0) )) = 0;
END_RULE; -- compatible_dimension

RULE curve_font_usage FOR (curve_style_font, pre_defined_curve_font,
          externally_defined_curve_font);
WHERE
  wr1: SIZEOF(QUERY ( csf <* curve_style_font | (NOT (SIZEOF(USEDIN(csf,
          'EXPLICIT_DRAUGHTING.' + 'CURVE_STYLE.CURVE_FONT')) >= 1)) ))
          = 0;
  wr2: SIZEOF(QUERY ( pdcf <* pre_defined_curve_font | (NOT (SIZEOF(
          USEDIN(pdcf,'EXPLICIT_DRAUGHTING.' + 'CURVE_STYLE.CURVE_FONT'))
          >= 1)) )) = 0;
  wr3: SIZEOF(QUERY ( edcf <* externally_defined_curve_font | (NOT (
          SIZEOF(USEDIN(edcf,'EXPLICIT_DRAUGHTING.' +
```

```
          'CURVE_STYLE.CURVE_FONT')) >= 1)) )) = 0;
END_RULE; -- curve_font_usage

RULE curve_style_font_instantiation FOR (curve_style_font);
WHERE
  wr1: SIZEOF(QUERY ( each <* curve_style_font | (NOT (SIZEOF(USEDIN(
          each,'')) >= 1)) )) = 0;
END_RULE; -- curve_style_font_instantiation

RULE curve_style_instantiation FOR (curve_style);
WHERE
  wr1: SIZEOF(QUERY ( each <* curve_style | (NOT (SIZEOF(USEDIN(each,''))
          >= 1)) )) = 0;
END_RULE; -- curve_style_instantiation

RULE dimensionality_is_two FOR (geometric_representation_context);
WHERE
  wr1: SIZEOF(QUERY ( g <* geometric_representation_context | (NOT (g.
          coordinate_space_dimension = 2)) )) = 0;
END_RULE; -- dimensionality_is_two

RULE draughting_model_annotation_layers FOR (draughting_model);
WHERE
  wr1: SIZEOF(QUERY ( dm <* draughting_model | (NOT (SIZEOF(
          QUERY ( ao <* QUERY ( it <* dm\representation.items | (
          'EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE' IN TYPEOF(it)) )
           | (NOT (SIZEOF(USEDIN(ao,'EXPLICIT_DRAUGHTING.' +
          'PRESENTATION_LAYER_ASSIGNMENT.' + 'ASSIGNED_ITEMS')) >= 1)) ))
          = 0)) )) = 0;
END_RULE; -- draughting_model_annotation_layers

RULE draughting_sub_model_mapped_item FOR (mapped_item);
WHERE
  wr1: SIZEOF(QUERY ( sr <* QUERY ( mi <* mapped_item | (
          'EXPLICIT_DRAUGHTING.SHAPE_REPRESENTATION' IN TYPEOF(mi.
          mapping_source.mapped_representation)) ) | (NOT (SIZEOF(
          USEDIN(sr,'')) = SIZEOF(QUERY ( mr <* USEDIN(sr,
          'EXPLICIT_DRAUGHTING.' + 'REPRESENTATION.ITEMS') | (SIZEOF([
          'EXPLICIT_DRAUGHTING.DRAUGHTING_MODEL',
          'EXPLICIT_DRAUGHTING.SHAPE_REPRESENTATION'] * TYPEOF(mr)) = 1)
          )))) ))
          = 0;
END_RULE; -- draughting_sub_model_mapped_item

RULE draughting_subfigure_representation_layers FOR (
          draughting_subfigure_representation);
WHERE
  wr1: SIZEOF(QUERY ( dsr <* draughting_subfigure_representation | (NOT
          (SIZEOF(QUERY ( item <* dsr\representation.items | ((
          'EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE' IN TYPEOF(item))
          AND (SIZEOF(USEDIN(item,'EXPLICIT_DRAUGHTING.' +
          'PRESENTATION_LAYER_ASSIGNMENT.' + 'ASSIGNED_ITEMS')) = 0)) ))
```

```
          = 0)) )) = 0;
END_RULE; -- draughting_subfigure_representation_layers

RULE drawing_sheet_annotation_layers FOR (drawing_sheet_revision);
WHERE
  wr1: SIZEOF(QUERY ( dsr <* drawing_sheet_revision | (NOT (SIZEOF(
          QUERY ( ao <* QUERY ( it <* dsr\representation.items | (
          'EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE' IN TYPEOF(it)) )
           | (NOT (SIZEOF(USEDIN(ao,'EXPLICIT_DRAUGHTING.' +
          'PRESENTATION_LAYER_ASSIGNMENT.' + 'ASSIGNED_ITEMS')) >= 1)) ))
          = 0)) )) = 0;
END_RULE; -- drawing_sheet_annotation_layers

RULE drawing_sheet_layout_usage FOR (mapped_item);
WHERE
  wr1: SIZEOF(QUERY ( dsl <* QUERY ( mi <* mapped_item | (
          'EXPLICIT_DRAUGHTING.DRAWING_SHEET_LAYOUT' IN TYPEOF(mi.
          mapping_source.mapped_representation)) ) | (NOT (SIZEOF(
          USEDIN(dsl,'')) = SIZEOF(QUERY ( dsr <* USEDIN(dsl,
          'EXPLICIT_DRAUGHTING.' + 'REPRESENTATION.ITEMS') | (
          'EXPLICIT_DRAUGHTING.DRAWING_SHEET_REVISION' IN TYPEOF(dsr)) )))) ))
          = 0;
END_RULE; -- drawing_sheet_layout_usage

RULE drawing_view_annotation_layers FOR (presentation_view);
WHERE
  wr1: SIZEOF(QUERY ( pv <* presentation_view | (NOT (SIZEOF(
          QUERY ( ao <* QUERY ( it <* pv\representation.items | (
          'EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE' IN TYPEOF(it)) )
           | (NOT (SIZEOF(USEDIN(ao,'EXPLICIT_DRAUGHTING.' +
          'PRESENTATION_LAYER_ASSIGNMENT.' + 'ASSIGNED_ITEMS')) >= 1)) ))
          = 0)) )) = 0;
END_RULE; -- drawing_view_annotation_layers

RULE externally_defined_curve_font_instantiation FOR (
          externally_defined_curve_font);
WHERE
  wr1: SIZEOF(QUERY ( each <* externally_defined_curve_font | (NOT (
          SIZEOF(USEDIN(each,'')) >= 1)) )) = 0;
END_RULE; -- externally_defined_curve_font_instantiation

RULE externally_defined_symbol_instantiation FOR (
          externally_defined_symbol);
WHERE
  wr1: SIZEOF(QUERY ( each <* externally_defined_symbol | (NOT (SIZEOF(
          USEDIN(each,'')) >= 1)) )) = 0;
END_RULE; -- externally_defined_symbol_instantiation

RULE externally_defined_text_font_instantiation FOR (
          externally_defined_text_font);
WHERE
  wr1: SIZEOF(QUERY ( each <* externally_defined_text_font | (NOT (
```

```
          SIZEOF(USEDIN(each,'')) >= 1)) )) = 0;
END_RULE; -- externally_defined_text_font_instantiation

RULE fill_area_style_colour_instantiation FOR (fill_area_style_colour);
WHERE
  wr1: SIZEOF(QUERY ( each <* fill_area_style_colour | (NOT (SIZEOF(
          USEDIN(each,'')) >= 1)) )) = 0;
END_RULE; -- fill_area_style_colour_instantiation

RULE fill_area_style_instantiation FOR (fill_area_style);
WHERE
  wr1: SIZEOF(QUERY ( each <* fill_area_style | (NOT (SIZEOF(USEDIN(each,
          '')) >= 1)) )) = 0;
END_RULE; -- fill_area_style_instantiation

RULE fill_area_style_tile_symbol_constraint FOR (
          fill_area_style_tile_symbol_with_style);
WHERE
  wr1: SIZEOF(QUERY ( fast <* fill_area_style_tile_symbol_with_style | (
          ('EXPLICIT_DRAUGHTING.DEFINED_SYMBOL' IN TYPEOF(fast.symbol\
          styled_item.item)) AND (NOT (
          'EXPLICIT_DRAUGHTING.EXTERNALLY_DEFINED_SYMBOL' IN TYPEOF(fast
          .symbol\styled_item.item\defined_symbol.definition)))) )) = 0;
END_RULE; -- fill_area_style_tile_symbol_constraint

RULE geometric_set_layers FOR (geometric_set);
WHERE
  wr1: SIZEOF(QUERY ( gs <* geometric_set | (NOT (SIZEOF(
          QUERY ( element <* gs.elements | (NOT (SIZEOF(USEDIN(element,
          'EXPLICIT_DRAUGHTING.' + 'PRESENTATION_LAYER_ASSIGNMENT.' +
          'ASSIGNED_ITEMS')) >= 1)) )) = 0)) )) = 0;
END_RULE; -- geometric_set_layers

RULE global_length_and_angle_units FOR (global_unit_assigned_context);
WHERE
  wr1: SIZEOF(QUERY ( guac <* global_unit_assigned_context | (NOT (
          SIZEOF(guac.units) = 2)) )) = 0;
  wr2: SIZEOF(QUERY ( guac <* global_unit_assigned_context | (NOT ((
          SIZEOF(QUERY ( x <* guac.units | (
          'EXPLICIT_DRAUGHTING.LENGTH_UNIT' IN TYPEOF(x)) )) = 1) AND (
          SIZEOF(QUERY ( x <* guac.units | (
          'EXPLICIT_DRAUGHTING.PLANE_ANGLE_UNIT' IN TYPEOF(x)) )) = 1))) ))
          = 0;
END_RULE; -- global_length_and_angle_units

RULE group_usage_constraint FOR (group, group_assignment);
WHERE
  wr1: SIZEOF(QUERY ( g <* group | (NOT (SIZEOF(QUERY ( ga <*
          group_assignment | (g :=: ga.assigned_group) )) = 1)) )) = 0;
END_RULE; -- group_usage_constraint

RULE measure_with_unit_instantiation FOR (measure_with_unit);
```

```
WHERE
  wr1: SIZEOF(QUERY ( m <* measure_with_unit | (NOT (SIZEOF(USEDIN(m,''))
          >= 1)) )) = 0;
  wr2: SIZEOF(QUERY ( m <* measure_with_unit | (NOT (SIZEOF(TYPEOF(m) *
          ['EXPLICIT_DRAUGHTING.LENGTH_MEASURE_WITH_UNIT',
          'EXPLICIT_DRAUGHTING.PLANE_ANGLE_MEASURE_WITH_UNIT']) = 1)) ))
          = 0;
END_RULE; -- measure_with_unit_instantiation

RULE named_unit_instantiation FOR (named_unit);
WHERE
  wr1: SIZEOF(QUERY ( n <* named_unit | (NOT (SIZEOF(USEDIN(n,'')) >= 1)) ))
          = 0;
  wr2: SIZEOF(QUERY ( n <* named_unit | (NOT (SIZEOF(TYPEOF(n) * [
          'EXPLICIT_DRAUGHTING.CONVERSION_BASED_UNIT',
          'EXPLICIT_DRAUGHTING.LENGTH_UNIT',
          'EXPLICIT_DRAUGHTING.PLANE_ANGLE_UNIT',
          'EXPLICIT_DRAUGHTING.SI_UNIT']) = 2)) )) = 0;
END_RULE; -- named_unit_instantiation

RULE pre_defined_colour_instantiation FOR (pre_defined_colour);
WHERE
  wr1: SIZEOF(QUERY ( each <* pre_defined_colour | (NOT (SIZEOF(USEDIN(
          each,'')) >= 1)) )) = 0;
END_RULE; -- pre_defined_colour_instantiation

RULE pre_defined_curve_font_instantiation FOR (pre_defined_curve_font);
WHERE
  wr1: SIZEOF(QUERY ( each <* pre_defined_curve_font | (NOT (SIZEOF(
          USEDIN(each,'')) >= 1)) )) = 0;
END_RULE; -- pre_defined_curve_font_instantiation

RULE pre_defined_symbol_instantiation FOR (pre_defined_symbol);
WHERE
  wr1: SIZEOF(QUERY ( each <* pre_defined_symbol | (NOT (SIZEOF(USEDIN(
          each,'')) >= 1)) )) = 0;
END_RULE; -- pre_defined_symbol_instantiation

RULE pre_defined_text_font_instantiation FOR (pre_defined_text_font);
WHERE
  wr1: SIZEOF(QUERY ( each <* pre_defined_text_font | (NOT (SIZEOF(
          USEDIN(each,'')) >= 1)) )) = 0;
END_RULE; -- pre_defined_text_font_instantiation

RULE presentation_layer_assignment_constraint FOR (
          presentation_layer_assignment);
WHERE
  wr1: SIZEOF(QUERY ( pla <* presentation_layer_assignment | (NOT (
          SIZEOF(QUERY ( item <* pla.assigned_items | (NOT (SIZEOF(
          TYPEOF(item) * ['EXPLICIT_DRAUGHTING.POINT',
          'EXPLICIT_DRAUGHTING.CURVE',
          'EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE']) = 1)) )) = 0)) ))
```

```
          = 0;
  wr2: SIZEOF(QUERY ( pla <* presentation_layer_assignment | (NOT (
          SIZEOF(QUERY ( pnt <* QUERY ( item <* pla.assigned_items | (
          'EXPLICIT_DRAUGHTING.POINT' IN TYPEOF(item)) ) | (NOT (SIZEOF(
          USEDIN(pnt,'EXPLICIT_DRAUGHTING.' + 'GEOMETRIC_SET.ELEMENTS'))
          = 0)) )) = 0)) )) = 0;
  wr3: SIZEOF(QUERY ( pla <* presentation_layer_assignment | (NOT (
          SIZEOF(QUERY ( crv <* QUERY ( item <* pla.assigned_items | (
          'EXPLICIT_DRAUGHTING.CURVE' IN TYPEOF(item)) ) | (NOT (SIZEOF(
          USEDIN(crv,'EXPLICIT_DRAUGHTING.' + 'GEOMETRIC_SET.ELEMENTS'))
          = 0)) )) = 0)) )) = 0;
  wr4: SIZEOF(QUERY ( pla <* presentation_layer_assignment | (NOT (
          SIZEOF(QUERY ( ao <* QUERY ( item <* pla.assigned_items | (
          'EXPLICIT_DRAUGHTING.ANNOTATION_OCCURRENCE' IN TYPEOF(item)) )
           | ((NOT SIZEOF(QUERY ( ur <* using_representations(ao) | (
          NOT (SIZEOF(TYPEOF(ur) * ['EXPLICIT_DRAUGHTING.' +
          'DRAUGHTING_SUBFIGURE_REPRESENTATION',
          'EXPLICIT_DRAUGHTING.DRAUGHTING_MODEL',
          'EXPLICIT_DRAUGHTING.DRAWING_SHEET_REVISION',
          'EXPLICIT_DRAUGHTING.PRESENTATION_VIEW']) = 1)) ))) = 0) )) =
          0)) )) = 0;
END_RULE; -- presentation_layer_assignment_constraint

RULE presentation_layer_assignment_requires_usage FOR (
          presentation_layer_assignment, presentation_layer_usage);
WHERE
  wr1: SIZEOF(QUERY ( e <* presentation_layer_assignment | (NOT (SIZEOF(
          QUERY ( s <* presentation_layer_usage | (e :=: s.assignment) ))
          = 1)) )) = 0;
END_RULE; -- presentation_layer_assignment_requires_usage

RULE presentation_of_product_definition_formation FOR (
          product_definition_formation);
WHERE
  wr1: SIZEOF(QUERY ( pdf <* product_definition_formation | (NOT ((
          SIZEOF(USEDIN(pdf,'EXPLICIT_DRAUGHTING.' +
          'DRAUGHTING_PRESENTED_ITEM.ITEMS')) >= 1) OR (SIZEOF(
          QUERY ( pd <* USEDIN(pdf,'EXPLICIT_DRAUGHTING.' +
          'PRODUCT_DEFINITION.FORMATION') | (NOT (SIZEOF(QUERY ( pds <*
          QUERY ( prop_def <* USEDIN(pd,'EXPLICIT_DRAUGHTING.' +
          'PROPERTY_DEFINITION.DEFINITION') | (
          'EXPLICIT_DRAUGHTING.PRODUCT_DEFINITION_SHAPE' IN TYPEOF(
          prop_def)) ) | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pds,
          'EXPLICIT_DRAUGHTING.' + 'PROPERTY_DEFINITION_REPRESENTATION.'
          + 'DEFINITION') | (
          'EXPLICIT_DRAUGHTING.SHAPE_DEFINITION_REPRESENTATION' IN
          TYPEOF(pdr)) )) >= 1)) )) = 0)) )) = 0))) )) = 0;
END_RULE; -- presentation_of_product_definition_formation

RULE presentation_size_constraint FOR (presentation_size);
WHERE
  wr1: SIZEOF(QUERY ( ps <* presentation_size | (NOT ((
```

```
          'EXPLICIT_DRAUGHTING.' + 'DRAWING_SHEET_REVISION') IN TYPEOF(
          ps.unit))) )) = 0;
END_RULE; -- presentation_size_constraint

RULE presentation_style_by_context_constraint FOR (
          presentation_style_by_context);
WHERE
  wr1: SIZEOF(QUERY ( psbc <* presentation_style_by_context | (NOT (
          'EXPLICIT_DRAUGHTING.PRESENTATION_VIEW' IN TYPEOF(psbc.
          style_context))) )) = 0;
END_RULE; -- presentation_style_by_context_constraint

RULE presentation_style_by_context_instantiation FOR (
          presentation_style_by_context);
WHERE
  wr1: SIZEOF(QUERY ( each <* presentation_style_by_context | (NOT (
          SIZEOF(USEDIN(each,'')) >= 1)) )) = 0;
END_RULE; -- presentation_style_by_context_instantiation

RULE presentation_view_presented_once FOR (presentation_view,
          drawing_sheet_revision);
WHERE
  wr1: SIZEOF(QUERY ( pv <* presentation_view | (NOT (SIZEOF(
          QUERY ( rm <* USEDIN(pv,'EXPLICIT_DRAUGHTING.' +
          'REPRESENTATION_MAP.' + 'MAPPED_REPRESENTATION') | (NOT (
          SIZEOF(QUERY ( mi <* rm.map_usage | (NOT (SIZEOF(
          QUERY ( dsr <* drawing_sheet_revision | (mi IN dsr\
          representation.items) )) = 1)) )) = 0)) )) = 0)) )) = 0;
END_RULE; -- presentation_view_presented_once

RULE product_definition_instantiation FOR (product_definition);
WHERE
  wr1: SIZEOF(QUERY ( each <* product_definition | (NOT (SIZEOF(USEDIN(
          each,'')) >= 1)) )) = 0;
END_RULE; -- product_definition_instantiation

RULE product_requires_version FOR (product, product_definition_formation);
WHERE
  wr1: SIZEOF(QUERY ( e <* product | (NOT (SIZEOF(QUERY ( s <*
          product_definition_formation | (e :=: s.of_product) )) >= 1)) ))
          = 0;
END_RULE; -- product_requires_version

RULE product_shape_has_representations FOR (product_definition_shape);
WHERE
  wr1: SIZEOF(QUERY ( pds <* product_definition_shape | (NOT (SIZEOF(
          USEDIN(pds,'EXPLICIT_DRAUGHTING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.' + 'DEFINITION')) >= 1)) ))
          = 0;
END_RULE; -- product_shape_has_representations

RULE representation_context_global_units FOR (representation_context);
```

```
WHERE
  wr1: SIZEOF(QUERY ( rc <* representation_context | (NOT ((
           'EXPLICIT_DRAUGHTING.' + 'GLOBAL_UNIT_ASSIGNED_CONTEXT') IN
           TYPEOF(rc))) )) = 0;
END_RULE; -- representation_context_global_units

RULE representation_instantiation FOR (representation);
WHERE
  wr1: SIZEOF(QUERY ( each <* representation | (NOT (SIZEOF(USEDIN(each,
           '')) >= 1)) )) = 0;
END_RULE; -- representation_instantiation

RULE sheets_belong_to_one_drawing FOR (drawing_sheet_revision,
           drawing_sheet_revision_usage);
WHERE
  wr1: SIZEOF(QUERY ( dsr <* drawing_sheet_revision | (NOT (SIZEOF(
           QUERY ( usage <* drawing_sheet_revision_usage | (dsr :=: usage
           .area) )) = 1)) )) = 0;
END_RULE; -- sheets_belong_to_one_drawing

RULE styled_curve FOR (styled_item);
WHERE
  wr1: SIZEOF(QUERY ( crv <* QUERY ( si <* styled_item | (
           'EXPLICIT_DRAUGHTING.CURVE' IN TYPEOF(si.item)) ) | (NOT (
           SIZEOF(QUERY ( sty <* crv.styles | (NOT ((SIZEOF(sty.styles) =
            1) AND ('EXPLICIT_DRAUGHTING.CURVE_STYLE' IN TYPEOF(sty.
           styles[1])))) )) = 0)) )) = 0;
END_RULE; -- styled_curve

RULE subtype_mandatory_annotation_occurrence FOR (annotation_occurrence);
WHERE
  wr1: SIZEOF(QUERY ( ao <* annotation_occurrence | (NOT (
           'EXPLICIT_DRAUGHTING.DRAUGHTING_ANNOTATION_OCCURRENCE' IN
           TYPEOF(ao))) )) = 0;
END_RULE; -- subtype_mandatory_annotation_occurrence

RULE subtype_mandatory_camera_image FOR (camera_image);
WHERE
  wr1: SIZEOF(QUERY ( ci <* camera_image | (NOT (
           'EXPLICIT_DRAUGHTING.CAMERA_IMAGE_2D_WITH_SCALE' IN TYPEOF(ci))) ))
           = 0;
END_RULE; -- subtype_mandatory_camera_image

RULE subtype_mandatory_draughting_callout FOR (draughting_callout);
WHERE
  wr1: SIZEOF(QUERY ( dc <* draughting_callout | (NOT (
           'EXPLICIT_DRAUGHTING.DRAUGHTING_ELEMENTS' IN TYPEOF(dc))) )) =
            0;
END_RULE; -- subtype_mandatory_draughting_callout

RULE subtype_mandatory_drawing_revision FOR (drawing_revision);
WHERE
```

```
  wr1: SIZEOF(QUERY ( dr <* drawing_revision | (NOT (
          'EXPLICIT_DRAUGHTING.DRAUGHTING_DRAWING_REVISION' IN TYPEOF(dr))) ))
          = 0;
END_RULE; -- subtype_mandatory_drawing_revision

RULE subtype_mandatory_pre_defined_colour FOR (pre_defined_colour);
WHERE
  wr1: SIZEOF(QUERY ( pdc <* pre_defined_colour | (NOT (
          'EXPLICIT_DRAUGHTING.DRAUGHTING_PRE_DEFINED_COLOUR' IN TYPEOF(
          pdc))) )) = 0;
END_RULE; -- subtype_mandatory_pre_defined_colour

RULE subtype_mandatory_pre_defined_curve_font FOR (
          pre_defined_curve_font);
WHERE
  wr1: SIZEOF(QUERY ( pdcf <* pre_defined_curve_font | (NOT (
          'EXPLICIT_DRAUGHTING.DRAUGHTING_PRE_DEFINED_CURVE_FONT' IN
          TYPEOF(pdcf))) )) = 0;
END_RULE; -- subtype_mandatory_pre_defined_curve_font

RULE subtype_mandatory_pre_defined_symbol FOR (pre_defined_symbol);
WHERE
  wr1: SIZEOF(QUERY ( pds <* pre_defined_symbol | (NOT (SIZEOF(TYPEOF(
          pds) * ['EXPLICIT_DRAUGHTING.' +
          'PRE_DEFINED_GEOMETRICAL_TOLERANCE_SYMBOL',
          'EXPLICIT_DRAUGHTING.' + 'PRE_DEFINED_DIMENSION_SYMBOL',
          'EXPLICIT_DRAUGHTING.' + 'PRE_DEFINED_POINT_MARKER_SYMBOL',
          'EXPLICIT_DRAUGHTING.' + 'PRE_DEFINED_TERMINATOR_SYMBOL']) = 1)) ))
          = 0;
END_RULE; -- subtype_mandatory_pre_defined_symbol

RULE subtype_mandatory_pre_defined_text_font FOR (pre_defined_text_font);
WHERE
  wr1: SIZEOF(QUERY ( pdtf <* pre_defined_text_font | (NOT (
          'EXPLICIT_DRAUGHTING.DRAUGHTING_PRE_DEFINED_TEXT_FONT' IN
          TYPEOF(pdtf))) )) = 0;
END_RULE; -- subtype_mandatory_pre_defined_text_font

RULE subtype_mandatory_shape_representation FOR (shape_representation);
WHERE
  wr1: SIZEOF(QUERY ( sr <* shape_representation | (NOT ((
          'EXPLICIT_DRAUGHTING.' +
          'GEOMETRICALLY_BOUNDED_2D_WIREFRAME_REPRESENTATION') IN
          TYPEOF(sr))) )) = 0;
END_RULE; -- subtype_mandatory_shape_representation

RULE symbol_colour_instantiation FOR (symbol_colour);
WHERE
  wr1: SIZEOF(QUERY ( each <* symbol_colour | (NOT (SIZEOF(USEDIN(each,
          ''')) >= 1)) )) = 0;
END_RULE; -- symbol_colour_instantiation
```

```
RULE terminator_symbol_constraint FOR (terminator_symbol);
WHERE
  wr1: SIZEOF(QUERY ( ts <* terminator_symbol | ((
           'EXPLICIT_DRAUGHTING.DEFINED_SYMBOL' IN TYPEOF(ts.item)) AND (
           NOT ('EXPLICIT_DRAUGHTING.PRE_DEFINED_TERMINATOR_SYMBOL' IN
           TYPEOF(ts.item\defined_symbol.definition)))) )) = 0;
END_RULE; -- terminator_symbol_constraint

RULE text_font_usage FOR (pre_defined_text_font,
           externally_defined_text_font);
WHERE
  wr1: SIZEOF(QUERY ( pdtf <* pre_defined_text_font | (NOT (SIZEOF(
           USEDIN(pdtf,'EXPLICIT_DRAUGHTING.' + 'TEXT_LITERAL.FONT')) >=
           1)) )) = 0;
  wr2: SIZEOF(QUERY ( edtf <* externally_defined_text_font | (NOT (
           SIZEOF(USEDIN(edtf,'EXPLICIT_DRAUGHTING.' +
           'TEXT_LITERAL.FONT')) >= 1)) )) = 0;
END_RULE; -- text_font_usage

RULE text_style_for_defined_font_instantiation FOR (
           text_style_for_defined_font);
WHERE
  wr1: SIZEOF(QUERY ( each <* text_style_for_defined_font | (NOT (
           SIZEOF(USEDIN(each,'')) >= 1)) )) = 0;
END_RULE; -- text_style_for_defined_font_instantiation

RULE text_style_instantiation FOR (text_style);
WHERE
  wr1: SIZEOF(QUERY ( each <* text_style | (NOT (SIZEOF(USEDIN(each,''))
           >= 1)) )) = 0;
END_RULE; -- text_style_instantiation

RULE use_of_geometrically_bounded_2d_wireframe_representation FOR (
           geometrically_bounded_2d_wireframe_representation, mapped_item);
WHERE
  wr1: SIZEOF(QUERY ( gbwr <*
           geometrically_bounded_2d_wireframe_representation | (NOT (
           SIZEOF(QUERY ( mi <* mapped_item | (mi.mapping_source.
           mapped_representation :=: gbwr) )) >= 1)) )) = 0;
  wr2: SIZEOF(QUERY ( gbwr_map <* QUERY ( mi <* mapped_item | ((
           'EXPLICIT_DRAUGHTING.' +
           'GEOMETRICALLY_BOUNDED_2D_WIREFRAME_REPRESENTATION') IN
           TYPEOF(mi.mapping_source.mapped_representation)) ) | (NOT (
           SIZEOF(QUERY ( rep <* USEDIN(gbwr_map,'EXPLICIT_DRAUGHTING.' +
            'REPRESENTATION.ITEMS') | (NOT (SIZEOF([
           'EXPLICIT_DRAUGHTING.DRAUGHTING_MODEL','EXPLICIT_DRAUGHTING.'
           + 'GEOMETRICALLY_BOUNDED_2D_WIREFRAME_REPRESENTATION'] *
           TYPEOF(rep)) = 1)) )) = 0)) )) = 0;
END_RULE; -- use_of_geometrically_bounded_2d_wireframe_representation

FUNCTION acyclic_composite_text(
             start_composite: composite_text;
```

```
            child_text: SET [1:?] OF text_or_character
    ): LOGICAL;

  LOCAL
   i : INTEGER;
   local_composite_text : SET [0:?] OF composite_text;
   local_annotation_text : SET [0:?] OF annotation_text;
   local_children : SET [0:?] OF text_or_character;
  END_LOCAL;
  local_composite_text := QUERY (child <* child_text |
                          ('PRESENTATION_DEFINITION_SCHEMA.COMPOSITE_TEXT'
                           IN TYPEOF (child)));
  IF (SIZEOF (local_composite_text) > 0) THEN
    REPEAT i := 1 TO HIINDEX (local_composite_text);
      IF (start_composite :=: local_composite_text[i]) THEN
        RETURN (FALSE);
      END_IF;
    END_REPEAT;
  END_IF;
  local_children := child_text;
  IF (SIZEOF (local_composite_text)) > 0 THEN
    REPEAT i := 1 TO HIINDEX (local_composite_text);
      local_children := local_children +
                        local_composite_text[i].collected_text;
    END_REPEAT;
  END_IF;
  local_annotation_text := QUERY (child <* child_text |
                          ('PRESENTATION_DEFINITION_SCHEMA.ANNOTATION_TEXT'
                           IN TYPEOF (child)));
  IF (SIZEOF (local_annotation_text) > 0) THEN
    REPEAT i := 1 TO HIINDEX (local_annotation_text);
      local_children := local_children +
      QUERY (item <* local_annotation_text[i]\mapped_item.
                     mapping_source.mapped_representation.items |
        SIZEOF(['PRESENTATION_DEFINITION_SCHEMA.ANNOTATION_TEXT',
                'PRESENTATION_DEFINITION_SCHEMA.COMPOSITE_TEXT'] *
               TYPEOF(item)) > 0);
    END_REPEAT;
  END_IF;
  IF (local_children :<>: child_text) THEN
    RETURN (acyclic_composite_text (start_composite, local_children));
  ELSE
    RETURN (TRUE);
  END_IF;
END_FUNCTION; -- acyclic_composite_text

FUNCTION acyclic_mapped_item_usage(
            rep: representation
    ): BOOLEAN;

  LOCAL
    i     : INTEGER;
```

```
    items : SET OF representation_item;
  END_LOCAL;
  items := QUERY ( item <* rep.items | (
      'DRAUGHTING_ANNOTATION_AIC.MAPPED_ITEM' IN TYPEOF(item)) );
  IF SIZEOF(items) = 0 THEN
    RETURN(FALSE);
  ELSE
    REPEAT i := 1 TO HIINDEX(items) BY 1;
      IF items[i]\mapped_item.mapping_source.mapped_representation :=:
          rep THEN
        RETURN(TRUE);
      ELSE
        RETURN(acyclic_mapped_item_usage(items[i]\mapped_item.
            mapping_source.mapped_representation));
      END_IF;
    END_REPEAT;
    RETURN(FALSE);
  END_IF;
END_FUNCTION; -- acyclic_mapped_item_usage

FUNCTION acyclic_mapped_representation(
             parent_set: SET OF representation;
             children_set: SET OF representation_item
    ): BOOLEAN;

  LOCAL
    i : INTEGER;
    x : SET OF representation_item;
    y : SET OF representation_item;
  END_LOCAL;
  x := QUERY ( z <* children_set | ('EXPLICIT_DRAUGHTING.MAPPED_ITEM' IN
       TYPEOF(z)) );
  IF SIZEOF(x) > 0 THEN
    REPEAT i := 1 TO HIINDEX(x) BY 1;
      IF x[i]\mapped_item.mapping_source.mapped_representation IN
          parent_set THEN
        RETURN(FALSE);
      END_IF;
      IF NOT acyclic_mapped_representation(parent_set + x[i]\mapped_item
          .mapping_source.mapped_representation,x[i]\mapped_item.
          mapping_source.mapped_representation.items) THEN
        RETURN(FALSE);
      END_IF;
    END_REPEAT;
  END_IF;
  x := children_set - x;
  IF SIZEOF(x) > 0 THEN
    REPEAT i := 1 TO HIINDEX(x) BY 1;
      y := QUERY ( z <* bag_to_set(USEDIN(x[i],'')) | (
          'EXPLICIT_DRAUGHTING.REPRESENTATION_ITEM' IN TYPEOF(z)) );
      IF NOT acyclic_mapped_representation(parent_set,y) THEN
        RETURN(FALSE);
```

```
      END_IF;
    END_REPEAT;
  END_IF;
  RETURN(TRUE);
END_FUNCTION; -- acyclic_mapped_representation

FUNCTION aspect_ratio(
             p: planar_box
    ): positive_ratio_measure;
  RETURN(p.size_in_x / p.size_in_y);
END_FUNCTION; -- aspect_ratio

FUNCTION bag_to_set(
             the_bag: BAG OF GENERIC:intype
    ): SET OF GENERIC:intype;

  LOCAL
    i       : INTEGER;
    the_set : SET OF GENERIC:intype := [];
  END_LOCAL;
  IF SIZEOF(the_bag) > 0 THEN
    REPEAT i := 1 TO HIINDEX(the_bag) BY 1;
      the_set := the_set + the_bag[i];
    END_REPEAT;
  END_IF;
  RETURN(the_set);
END_FUNCTION; -- bag_to_set

FUNCTION build_2axes(
             ref_direction: direction
    ): LIST [2:2] OF direction;

  LOCAL
    u : LIST [2:2] OF direction;
  END_LOCAL;
  u[1] := NVL(normalise(ref_direction),direction([1,0]));
  u[2] := orthogonal_complement(u[1]);
  RETURN(u);
END_FUNCTION; -- build_2axes

FUNCTION check_text_alignment(
             ct: composite_text
    ): BOOLEAN;

  LOCAL
    a : SET OF text_alignment := [];
    i : INTEGER;
  END_LOCAL;
  REPEAT i := 1 TO HIINDEX(ct.collected_text) BY 1;
    a := a + [ct.collected_text[i].alignment];
  END_REPEAT;
  RETURN(SIZEOF(a) = 1);
```

```
END_FUNCTION; -- check_text_alignment

FUNCTION check_text_font(
            ct: composite_text
    ): BOOLEAN;

  LOCAL
    f : SET OF font_select := [];
    i : INTEGER;
  END_LOCAL;
  REPEAT i := 1 TO HIINDEX(ct.collected_text) BY 1;
    f := f + [ct.collected_text[i].font];
  END_REPEAT;
  RETURN(SIZEOF(f) <= 1);
END_FUNCTION; -- check_text_font

FUNCTION constraints_param_b_spline(
            degree, up_knots, up_cp: INTEGER;
            knot_mult: LIST OF INTEGER;
            knots: LIST OF parameter_value
    ): BOOLEAN;

  LOCAL
    k      : INTEGER;
    l      : INTEGER;
    sum    : INTEGER;
    result : BOOLEAN := TRUE;
  END_LOCAL;
  sum := knot_mult[1];
  REPEAT i := 2 TO up_knots BY 1;
    sum := sum + knot_mult[i];
  END_REPEAT;
  IF (degree < 1) OR (up_knots < 2) OR (up_cp < degree) OR (sum <> (
      degree + up_cp + 2)) THEN
    result := FALSE;
    RETURN(result);
  END_IF;
  k := knot_mult[1];
  IF (k < 1) OR (k > (degree + 1)) THEN
    result := FALSE;
    RETURN(result);
  END_IF;
  REPEAT i := 2 TO up_knots BY 1;
    IF (knot_mult[i] < 1) OR (knots[i] <= knots[i - 1]) THEN
      result := FALSE;
      RETURN(result);
    END_IF;
    k := knot_mult[i];
    IF (i < up_knots) AND (k > degree) THEN
      result := FALSE;
      RETURN(result);
    END_IF;
```

```
    IF (i = up_knots) AND (k > (degree + 1)) THEN
      result := FALSE;
      RETURN(result);
    END_IF;
  END_REPEAT;
  RETURN(result);
END_FUNCTION; -- constraints_param_b_spline

FUNCTION curve_weights_positive(
            b: rational_b_spline_curve
    ): BOOLEAN;

  LOCAL
    result : BOOLEAN := TRUE;
  END_LOCAL;
  REPEAT i := 0 TO b.upper_index_on_control_points BY 1;
    IF b.weights[i] <= 0 THEN
      result := FALSE;
      RETURN(result);
    END_IF;
  END_REPEAT;
  RETURN(result);
END_FUNCTION; -- curve_weights_positive

FUNCTION derive_dimensional_exponents(
            x: unit
    ): dimensional_exponents;

  LOCAL
    i      : INTEGER;
    result : dimensional_exponents := dimensional_exponents(0,0,0,0,0,0,
             0);
  END_LOCAL;
  IF 'EXPLICIT_DRAUGHTING.DERIVED_UNIT' IN TYPEOF(x) THEN
    REPEAT i := LOINDEX(x.elements) TO HIINDEX(x.elements) BY 1;
      result.length_exponent := result.length_exponent + (x.elements[i].
          exponent * x.elements[i].unit.dimensions.length_exponent);
      result.mass_exponent := result.mass_exponent + (x.elements[i].
          exponent * x.elements[i].unit.dimensions.mass_exponent);
      result.time_exponent := result.time_exponent + (x.elements[i].
          exponent * x.elements[i].unit.dimensions.time_exponent);
      result.electric_current_exponent := result.
          electric_current_exponent + (x.elements[i].exponent * x.
          elements[i].unit.dimensions.electric_current_exponent);
      result.thermodynamic_temperature_exponent := result.
          thermodynamic_temperature_exponent + (x.elements[i].exponent *
          x.elements[i].unit.dimensions.
          thermodynamic_temperature_exponent);
      result.amount_of_substance_exponent := result.
          amount_of_substance_exponent + (x.elements[i].exponent * x.
          elements[i].unit.dimensions.amount_of_substance_exponent);
      result.luminous_intensity_exponent := result.
```

```
          luminous_intensity_exponent + (x.elements[i].exponent * x.
          elements[i].unit.dimensions.luminous_intensity_exponent);
    END_REPEAT;
  ELSE
    result := x.dimensions;
  END_IF;
  RETURN(result);
END_FUNCTION; -- derive_dimensional_exponents

FUNCTION dimension_of(
             item: geometric_representation_item
    ): dimension_count;

  LOCAL
    x : SET OF representation;
    y : representation_context;
  END_LOCAL;
  x := using_representations(item);
  y := x[1].context_of_items;
  RETURN(y\geometric_representation_context.coordinate_space_dimension);
END_FUNCTION; -- dimension_of

FUNCTION dimensions_for_si_unit(
            n: si_unit_name
    ): dimensional_exponents;
  CASE n OF
    metre          :          RETURN(dimensional_exponents(1,0,0,0,0,0,0));
    gram           :          RETURN(dimensional_exponents(0,1,0,0,0,0,0));
    second         :          RETURN(dimensional_exponents(0,0,1,0,0,0,0));
    ampere         :          RETURN(dimensional_exponents(0,0,0,1,0,0,0));
    kelvin         :          RETURN(dimensional_exponents(0,0,0,0,1,0,0));
    mole           :          RETURN(dimensional_exponents(0,0,0,0,0,1,0));
    candela        :          RETURN(dimensional_exponents(0,0,0,0,0,0,1));
    radian         :          RETURN(dimensional_exponents(0,0,0,0,0,0,0));
    steradian      :          RETURN(dimensional_exponents(0,0,0,0,0,0,0));
    hertz          :          RETURN(dimensional_exponents(0,0,-1,0,0,0,0));
    newton         :          RETURN(dimensional_exponents(1,1,-2,0,0,0,0));
    pascal         :          RETURN(dimensional_exponents(-1,1,-2,0,0,0,0));
    joule          :          RETURN(dimensional_exponents(2,1,-2,0,0,0,0));
    watt           :          RETURN(dimensional_exponents(2,1,-3,0,0,0,0));
    coulomb        :          RETURN(dimensional_exponents(0,0,1,1,0,0,0));
    volt           :          RETURN(dimensional_exponents(2,1,-3,-1,0,0,0));
    farad          :          RETURN(dimensional_exponents(-2,-1,4,1,0,0,0));
    ohm            :          RETURN(dimensional_exponents(2,1,-3,-2,0,0,0));
    siemens        :          RETURN(dimensional_exponents(-2,-1,3,2,0,0,0));
    weber          :          RETURN(dimensional_exponents(2,1,-2,-1,0,0,0));
    tesla          :          RETURN(dimensional_exponents(0,1,-2,-1,0,0,0));
    henry          :          RETURN(dimensional_exponents(2,1,-2,-2,0,0,0));
    degree_celsius  :           RETURN(dimensional_exponents(0,0,0,0,1,0,0));
    lumen          :          RETURN(dimensional_exponents(0,0,0,0,0,0,1));
    lux            :          RETURN(dimensional_exponents(-2,0,0,0,0,0,1));
    becquerel      :          RETURN(dimensional_exponents(0,0,-1,0,0,0,0));
```

```
    gray          :         RETURN(dimensional_exponents(2,0,-2,0,0,0,0));
    sievert       :         RETURN(dimensional_exponents(2,0,-2,0,0,0,0));
    END_CASE;
END_FUNCTION; -- dimensions_for_si_unit

FUNCTION item_in_context(
             item: representation_item;
             cntxt: representation_context
    ): BOOLEAN;

  LOCAL
    i : INTEGER;
    y : BAG OF representation_item;
  END_LOCAL;
  IF SIZEOF(USEDIN(item,'EXPLICIT_DRAUGHTING.REPRESENTATION.ITEMS') *
      cntxt.representations_in_context) > 0 THEN
    RETURN(TRUE);
  ELSE
    y := QUERY ( z <* USEDIN(item,'') | (
        'EXPLICIT_DRAUGHTING.REPRESENTATION_ITEM' IN TYPEOF(z)) );
    IF SIZEOF(y) > 0 THEN
      REPEAT i := 1 TO HIINDEX(y) BY 1;
        IF item_in_context(y[i],cntxt) THEN
          RETURN(TRUE);
        END_IF;
      END_REPEAT;
    END_IF;
  END_IF;
  RETURN(FALSE);
END_FUNCTION; -- item_in_context

FUNCTION leap_year(
             year: year_number
    ): BOOLEAN;
  IF (((year MOD 4) = 0) AND ((year MOD 100) <> 0)) OR ((year MOD 400) =
       0) THEN
    RETURN(TRUE);
  ELSE
    RETURN(FALSE);
  END_IF;
END_FUNCTION; -- leap_year

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
    REPEAT i := 1 TO n BY 1;
      res[(low + i) - 1] := lis[i];
    END_REPEAT;
    RETURN(res);
  END_IF;
END_FUNCTION; -- list_to_array

FUNCTION normalise(
            arg: vector_or_direction
    ): vector_or_direction;

  LOCAL
    ndim   : INTEGER;
    v      : direction;
    vec    : vector;
    mag    : REAL;
    result : vector_or_direction;
  END_LOCAL;
  IF NOT EXISTS(arg) THEN
    result := ?;
  ELSE
    ndim := arg.dim;
    IF 'EXPLICIT_DRAUGHTING.VECTOR' IN TYPEOF(arg) THEN
      BEGIN
        vec := arg;
        v := arg.orientation;
        IF arg.magnitude = 0 THEN
          RETURN(?);
        ELSE
          vec.magnitude := 1;
        END_IF;
      END;
    ELSE
      v := arg;
    END_IF;
    mag := 0;
    REPEAT i := 1 TO ndim BY 1;
      mag := mag + (v.direction_ratios[i] * v.direction_ratios[i]);
    END_REPEAT;
    IF mag > 0 THEN
      mag := SQRT(mag);
      REPEAT i := 1 TO ndim BY 1;
        v.direction_ratios[i] := v.direction_ratios[i] / mag;
      END_REPEAT;
      IF 'EXPLICIT_DRAUGHTING.VECTOR' IN TYPEOF(arg) THEN
        vec.orientation := v;
        result := vec;
      ELSE
        result := v;
```

```
      END_IF;
    ELSE
      RETURN(?);
    END_IF;
  END_IF;
  RETURN(result);
END_FUNCTION; -- normalise

FUNCTION orthogonal_complement(
            vec: direction
    ): direction;

  LOCAL
    result : direction;
  END_LOCAL;
  IF (vec.dim <> 2) OR (NOT EXISTS(vec)) THEN
    RETURN(?);
  ELSE
    result.direction_ratios[1] := -vec.direction_ratios[2];
    result.direction_ratios[2] := vec.direction_ratios[1];
    RETURN(result);
  END_IF;
END_FUNCTION; -- orthogonal_complement

FUNCTION using_representations(
            item: representation_item
    ): SET OF representation;

  LOCAL
    results            : SET OF representation;
    result_bag         : BAG OF representation;
    intermediate_items : SET OF representation_item;
    i                  : INTEGER;
  END_LOCAL;
  result_bag := USEDIN(item,'REPRESENTATION_SCHEMA.REPRESENTATION.ITEMS');
  IF SIZEOF(result_bag) > 0 THEN
    REPEAT i := 1 TO HIINDEX(result_bag);
      results := results + result_bag[i];
    END_REPEAT;
  END_IF;
  intermediate_items := QUERY(z <* bag_to_set( USEDIN(item , '')) |
    'REPRESENTATION_SCHEMA.REPRESENTATION_ITEM' IN TYPEOF(z));
  IF SIZEOF(intermediate_items) > 0 THEN
    REPEAT i := 1 TO HIINDEX(intermediate_items);
      results := results + using_representations(intermediate_items[i]);
    END_REPEAT;
  END_IF;
  RETURN (results);
END_FUNCTION; -- using_representations

FUNCTION valid_calendar_date(
            date: calendar_date
```

```
    ): LOGICAL;
  IF NOT ((1 <= date.day_component) AND (date.day_component <= 31))
       THEN
    RETURN(FALSE);
  END_IF;
  CASE date.month_component OF
    4 :         RETURN((1 <= date.day_component) AND (date.day_component
          <= 30));
    6 :         RETURN((1 <= date.day_component) AND (date.day_component
          <= 30));
    9 :         RETURN((1 <= date.day_component) AND (date.day_component
          <= 30));
    11  :         RETURN((1 <= date.day_component) AND (date.
          day_component <= 30));
    2 :         BEGIN
        IF leap_year(date.year_component) THEN
          RETURN((1 <= date.day_component) AND (date.day_component <= 29));
        ELSE
          RETURN((1 <= date.day_component) AND (date.day_component <= 28));
        END_IF;
      END;
    END_CASE;
END_FUNCTION; -- valid_calendar_date

FUNCTION valid_units(
             m: measure_with_unit
    ): BOOLEAN;
  IF 'EXPLICIT_DRAUGHTING.LENGTH_MEASURE' IN TYPEOF(m.value_component)
       THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(1,0,0,0,0,0,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  IF 'EXPLICIT_DRAUGHTING.MASS_MEASURE' IN TYPEOF(m.value_component)
       THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(0,1,0,0,0,0,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  IF 'EXPLICIT_DRAUGHTING.TIME_MEASURE' IN TYPEOF(m.value_component)
       THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(0,0,1,0,0,0,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  IF 'EXPLICIT_DRAUGHTING.ELECTRIC_CURRENT_MEASURE' IN TYPEOF(m.
      value_component) THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(0,0,0,1,0,0,0) THEN
```

```
    RETURN(FALSE);
  END_IF;
END_IF;
IF 'EXPLICIT_DRAUGHTING.THERMODYNAMIC_TEMPERATURE_MEASURE' IN TYPEOF(m
    .value_component) THEN
  IF derive_dimensional_exponents(m.unit_component) <>
      dimensional_exponents(0,0,0,0,1,0,0) THEN
    RETURN(FALSE);
  END_IF;
END_IF;
IF 'EXPLICIT_DRAUGHTING.AMOUNT_OF_SUBSTANCE_MEASURE' IN TYPEOF(m.
    value_component) THEN
  IF derive_dimensional_exponents(m.unit_component) <>
      dimensional_exponents(0,0,0,0,0,1,0) THEN
    RETURN(FALSE);
  END_IF;
END_IF;
IF 'EXPLICIT_DRAUGHTING.LUMINOUS_INTENSITY_MEASURE' IN TYPEOF(m.
    value_component) THEN
  IF derive_dimensional_exponents(m.unit_component) <>
      dimensional_exponents(0,0,0,0,0,0,1) THEN
    RETURN(FALSE);
  END_IF;
END_IF;
IF 'EXPLICIT_DRAUGHTING.PLANE_ANGLE_MEASURE' IN TYPEOF(m.
    value_component) THEN
  IF derive_dimensional_exponents(m.unit_component) <>
      dimensional_exponents(0,0,0,0,0,0,0) THEN
    RETURN(FALSE);
  END_IF;
END_IF;
IF 'EXPLICIT_DRAUGHTING.SOLID_ANGLE_MEASURE' IN TYPEOF(m.
    value_component) THEN
  IF derive_dimensional_exponents(m.unit_component) <>
      dimensional_exponents(0,0,0,0,0,0,0) THEN
    RETURN(FALSE);
  END_IF;
END_IF;
IF 'EXPLICIT_DRAUGHTING.AREA_MEASURE' IN TYPEOF(m.value_component)
    THEN
  IF derive_dimensional_exponents(m.unit_component) <>
      dimensional_exponents(2,0,0,0,0,0,0) THEN
    RETURN(FALSE);
  END_IF;
END_IF;
IF 'EXPLICIT_DRAUGHTING.VOLUME_MEASURE' IN TYPEOF(m.value_component)
    THEN
  IF derive_dimensional_exponents(m.unit_component) <>
      dimensional_exponents(3,0,0,0,0,0,0) THEN
    RETURN(FALSE);
  END_IF;
END_IF;
```

```
  IF 'EXPLICIT_DRAUGHTING.RATIO_MEASURE' IN TYPEOF(m.value_component)
       THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(0,0,0,0,0,0,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  IF 'EXPLICIT_DRAUGHTING.POSITIVE_LENGTH_MEASURE' IN TYPEOF(m.
      value_component) THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(1,0,0,0,0,0,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  IF 'EXPLICIT_DRAUGHTING.POSITIVE_PLANE_ANGLE_MEASURE' IN TYPEOF(m.
      value_component) THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(0,0,0,0,0,0,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  RETURN(TRUE);
END_FUNCTION; -- valid_units

END_SCHEMA; -- explicit_draughting
(*
```

Annex B
(normative)

AIM short names

Table B.1 provides the short names of entities specified in the AIM of this part of ISO 10303. Requirements on the use of the short names are found in the implementation methods included in ISO 10303.

Table B.1 – AIM short names

Entity names	Short names
ADDRESS	ADDRSS
ANGULAR_DIMENSION	ANGDMN
ANNOTATION_CURVE_OCCURRENCE	ANCROC
ANNOTATION_FILL_AREA	ANFLAR
ANNOTATION_FILL_AREA_OCCURRENCE	AFAO
ANNOTATION_OCCURRENCE	ANNOCC
ANNOTATION_SUBFIGURE_OCCURRENCE	ANSBOC
ANNOTATION_SYMBOL	ANNSYM
ANNOTATION_SYMBOL_OCCURRENCE	ANSYOC
ANNOTATION_TEXT	ANNTXT
ANNOTATION_TEXT_OCCURRENCE	ANTXOC
APPLICATION_CONTEXT	APPCNT
APPLICATION_CONTEXT_ELEMENT	APCNEL
APPLICATION_PROTOCOL_DEFINITION	APPRDF
APPROVAL	APPRVL
APPROVAL_ASSIGNMENT	APPASS
APPROVAL_DATE_TIME	APDTTM
APPROVAL_PERSON_ORGANIZATION	APPROR
APPROVAL_ROLE	APPRL
APPROVAL_STATUS	APPSTT
AREA_IN_SET	ARINST

Table B.1 (continued)

Entity names	Short names
AXIS2_PLACEMENT_2D	A2PL2D
BEZIER_CURVE	BZRCRV
BOUNDED_CURVE	BNDCRV
B_SPLINE_CURVE	BSPCR
B_SPLINE_CURVE_WITH_KNOTS	BSCWK
CALENDAR_DATE	CLNDT
CAMERA_IMAGE	CMRIMG
CAMERA_IMAGE_2D_WITH_SCALE	CI2WS
CAMERA_MODEL	CMRMDL
CAMERA_MODEL_D2	CMMDD2
CAMERA_USAGE	CMRUSG
CARTESIAN_POINT	CRTPNT
CIRCLE	CIRCLE
COLOUR	COLOUR
COLOUR_RGB	CLRRGB
COLOUR_SPECIFICATION	CLRSPC
COMPOSITE_CURVE	CMPCRV
COMPOSITE_CURVE_SEGMENT	CMCRSG
COMPOSITE_TEXT	CMPTXT
COMPOSITE_TEXT_WITH_ASSOCIATED_CURVES	CTWAC
COMPOSITE_TEXT_WITH_BLANKING_BOX	CTWBB
COMPOSITE_TEXT_WITH_EXTENT	CTWE
CONIC	CONIC
CONTEXT_DEPENDENT_INVISIBILITY	CNDPIN

Table B.1 (continued)

Entity names	Short names
CONTRACT	CNTRCT
CONTRACT_ASSIGNMENT	CNTASS
CONTRACT_TYPE	CNTTYP
CONVERSION_BASED_UNIT	CNBSUN
CURVE	CURVE
CURVE_DIMENSION	CRVDMN
CURVE_STYLE	CRVSTY
CURVE_STYLE_FONT	CRSTFN
CURVE_STYLE_FONT_PATTERN	CSFP
DATE	DATE
DATUM_FEATURE_CALLOUT	DTFTCL
DATUM_TARGET_CALLOUT	DTTRCL
DEFINED_SYMBOL	DFNSYM
DIAMETER_DIMENSION	DMTDMN
DIMENSIONAL_EXPONENTS	DMNEXP
DIMENSION_CALLOUT_COMPONENT_RELATIONSHIP	DCCR
DIMENSION_CALLOUT_RELATIONSHIP	DMCLRL
DIMENSION_CURVE	DMNCRV
DIMENSION_CURVE_DIRECTED_CALLOUT	DCDC
DIMENSION_CURVE_TERMINATOR	DMCRTR
DIMENSION_PAIR	DMNPR
DIRECTION	DRCTN
DOCUMENT	DCMNT
DOCUMENT_REFERENCE	DCMRFR
DOCUMENT_TYPE	DCMTYP

Table B.1 (continued)

Entity names	Short names
DRAUGHTING_ANNOTATION_OCCURRENCE	DRANOC
DRAUGHTING_APPROVAL_ASSIGNMENT	DRAPAS
DRAUGHTING_CALLOUT	DRGCLL
DRAUGHTING_CALLOUT_RELATIONSHIP	DRCLRL
DRAUGHTING_CONTRACT_ASSIGNMENT	DRCNAS
DRAUGHTING_DRAWING_REVISION	DRDRRV
DRAUGHTING_ELEMENTS	DRGELM
DRAUGHTING_GROUP_ASSIGNMENT	DRGRAS
DRAUGHTING_MODEL	DRGMDL
DRAUGHTING_ORGANIZATION_ASSIGNMENT	DRORAS
DRAUGHTING_PERSON_AND_ORGANIZATION_ASSIGNMENT	DPAOA
DRAUGHTING_PERSON_ASSIGNMENT	DRPRAS
DRAUGHTING_PRESENTED_ITEM	DRPRIT
DRAUGHTING_PRE_DEFINED_COLOUR	DPDC
DRAUGHTING_PRE_DEFINED_CURVE_FONT	DPDCF
DRAUGHTING_PRE_DEFINED_TEXT_FONT	DPDTF
DRAUGHTING_SECURITY_CLASSIFICATION_ASSIGNMENT	DSCA
DRAUGHTING_SPECIFICATION_REFERENCE	DRSPRF
DRAUGHTING_SUBFIGURE_REPRESENTATION	DRSBRP
DRAUGHTING_SYMBOL_REPRESENTATION	DRSYRP
DRAUGHTING_TEXT_LITERAL_WITH_DELINEATION	DTLWD
DRAUGHTING_TITLE	DRGTTL
DRAWING_DEFINITION	DRWDFN
DRAWING_REVISION	DRWRVS
DRAWING_SHEET_LAYOUT	DRSHLY

Table B.1 (continued)

Entity names	Short names
DRAWING_SHEET_REVISION	DRSHRV
DRAWING_SHEET_REVISION_USAGE	DSRU
ELLIPSE	ELLPS
EXTERNALLY_DEFINED_CURVE_FONT	EDCF
EXTERNALLY_DEFINED_HATCH_STYLE	EDHS
EXTERNALLY_DEFINED_ITEM	EXDFIT
EXTERNALLY_DEFINED_SYMBOL	EXDFSY
EXTERNALLY_DEFINED_TEXT_FONT	EDTF
EXTERNALLY_DEFINED_TILE_STYLE	EDTS
EXTERNAL_SOURCE	EXTSRC
FILL_AREA_STYLE	FLARST
FILL_AREA_STYLE_COLOUR	FASC
FILL_AREA_STYLE_HATCHING	FASH
FILL_AREA_STYLE_TILES	FAST
FILL_AREA_STYLE_TILE_SYMBOL_WITH_STYLE	FASTSW
GEOMETRICALLY_BOUNDED_2D_WIREFRAME_REPRESENTATION	GB2WR
GEOMETRICAL_TOLERANCE_CALLOUT	GMTLCL
GEOMETRIC_CURVE_SET	GMCRST
GEOMETRIC_REPRESENTATION_CONTEXT	GMRPCN
GEOMETRIC_REPRESENTATION_ITEM	GMRPIT
GEOMETRIC_SET	GMTST
GLOBAL_UNIT_ASSIGNED_CONTEXT	GUAC
GROUP	GROUP
GROUP_ASSIGNMENT	GRPASS
GROUP_RELATIONSHIP	GRPRLT

Table B.1 (continued)

Entity names	Short names
HYPERBOLA	HYPRBL
INVISIBILITY	INVSBL
LEADER_CURVE	LDRCRV
LEADER_DIRECTED_CALLOUT	LDDRCL
LEADER_DIRECTED_DIMENSION	LDDRDM
LEADER_TERMINATOR	LDRTRM
LENGTH_MEASURE_WITH_UNIT	LMWU
LENGTH_UNIT	LNGUNT
LINE	LINE
LINEAR_DIMENSION	LNRDMN
MAPPED_ITEM	MPPITM
MEASURE_WITH_UNIT	MSWTUN
NAMED_UNIT	NMDUNT
OFFSET_CURVE_2D	OFCR2D
ONE_DIRECTION_REPEAT_FACTOR	ODRF
ORDINATE_DIMENSION	ORDDMN
ORGANIZATION	ORGNZT
ORGANIZATIONAL_ADDRESS	ORGADD
ORGANIZATION_ASSIGNMENT	ORGASS
ORGANIZATION_ROLE	ORGRL
PARABOLA	PRBL
PERSON	PERSON
PERSONAL_ADDRESS	PRSADD
PERSON_AND_ORGANIZATION	PRANOR
PERSON_AND_ORGANIZATION_ASSIGNMENT	PAOA
PERSON_AND_ORGANIZATION_ROLE	PAOR

Table B.1 (continued)

Entity names	Short names
PERSON_ASSIGNMENT	PRSASS
PERSON_ROLE	PRSRL
PLACEMENT	PLCMNT
PLANAR_BOX	PLNBX
PLANAR_EXTENT	PLNEXT
PLANE_ANGLE_MEASURE_WITH_UNIT	PAMWU
PLANE_ANGLE_UNIT	PLANUN
POINT	POINT
POINT_ON_CURVE	PNONCR
POLYLINE	PLYLN
PRESENTATION_AREA	PRSAR
PRESENTATION_LAYER_ASSIGNMENT	PRLYAS
PRESENTATION_LAYER_USAGE	PRLYUS
PRESENTATION_REPRESENTATION	PRSRPR
PRESENTATION_SET	PRSST
PRESENTATION_SIZE	PRSSZ
PRESENTATION_STYLE_ASSIGNMENT	PRSTAS
PRESENTATION_STYLE_BY_CONTEXT	PSBC
PRESENTATION_VIEW	PRSVW
PRESENTED_ITEM	PRSITM
PRESENTED_ITEM_REPRESENTATION	PRITRP
PRE_DEFINED_COLOUR	PRDFCL
PRE_DEFINED_CURVE_FONT	PDCF
PRE_DEFINED_DIMENSION_SYMBOL	PDDS
PRE_DEFINED_GEOMETRICAL_TOLERANCE_SYMBOL	PDGTS

Table B.1 (continued)

Entity names	Short names
PRE_DEFINED_ITEM	PRDFIT
PRE_DEFINED_POINT_MARKER_SYMBOL	PDPMS
PRE_DEFINED_SYMBOL	PRDFSY
PRE_DEFINED_TERMINATOR_SYMBOL	PDT
PRE_DEFINED_TEXT_FONT	PDTF
PRODUCT	PRDCT
PRODUCT_CONTEXT	PRDCNT
PRODUCT_DEFINITION	PRDDFN
PRODUCT_DEFINITION_CONTEXT	PRDFCN
PRODUCT_DEFINITION_FORMATION	PRDFFR
PRODUCT_DEFINITION_SHAPE	PRDFSH
PROJECTION_CURVE	PRJCRV
PROJECTION_DIRECTED_CALLOUT	PRDRCL
PROPERTY_DEFINITION	PRPDFN
PROPERTY_DEFINITION_REPRESENTATION	PRDFRP
QUASI_UNIFORM_CURVE	QSUNCR
RADIUS_DIMENSION	RDSDMN
RATIONAL_B_SPLINE_CURVE	RBSC
REPRESENTATION	RPRSNT
REPRESENTATION_CONTEXT	RPRCNT
REPRESENTATION_ITEM	RPRITM
REPRESENTATION_MAP	RPRMP
SECURITY_CLASSIFICATION	SCRCLS
SECURITY_CLASSIFICATION_ASSIGNMENT	SCCLAS
SECURITY_CLASSIFICATION_LEVEL	SCCLLV

Table B.1 (concluded)

Entity names	Short names
SHAPE_DEFINITION_REPRESENTATION	SHDFRP
SHAPE_REPRESENTATION	SHPRPR
SI_UNIT	SUNT
STRUCTURED_DIMENSION_CALLOUT	STDMCL
STYLED_ITEM	STYITM
SYMBOL_COLOUR	SYMCLR
SYMBOL_REPRESENTATION	SYMRPR
SYMBOL_REPRESENTATION_MAP	SYRPMP
SYMBOL_STYLE	SYMSTY
SYMBOL_TARGET	SYMTRG
TERMINATOR_SYMBOL	TRMSYM
TEXT_LITERAL	TXTLTR
TEXT_LITERAL_WITH_ASSOCIATED_CURVES	TLWAC
TEXT_LITERAL_WITH_BLANKING_BOX	TLWBB
TEXT_LITERAL_WITH_DELINEATION	TLWD
TEXT_LITERAL_WITH_EXTENT	TLWE
TEXT_STYLE	TXTSTY
TEXT_STYLE_FOR_DEFINED_FONT	TSFDF
TEXT_STYLE_WITH_BOX_CHARACTERISTICS	TSWBC
TEXT_STYLE_WITH_MIRROR	TSWM
TRIMMED_CURVE	TRMCRV
TWO_DIRECTION_REPEAT_FACTOR	TDRF
UNIFORM_CURVE	UNFCRV
VECTOR	VECTOR

Annex C
(normative)

Implementation method specific requirements

The implementation method defines what types of exchange behaviour are required with respect to this part of ISO 10303. Conformance to this part of ISO 10303 shall be realized in an exchange structure. The file format shall be encoded according to the syntax and EXPRESS language mapping defined in ISO 10303-21 and the AIM defined in annex A of this part of ISO 10303. The header of the exchange structure shall identify use of this part of ISO 10303 by the schema name 'explicit_draughting'.

Annex D
(normative)

Protocol Implementation Conformance Statement (PICS) proforma

The Protocol Implementation Conformance Statement (PICS) proforma is supplied for completion by the person or organization (the client) requesting conformance testing. Its purpose is to ascertain the scope of claimed conformance to a particular application protocol by an implementation under test (IUT) using a defined implementation method. Through the completion of this form, the PICS Proforma becomes a PICS.

The information contained in the PICS is used to configure an appropriate executable test suite for use by the client.

1. Please indicate the implementation method adopted in this choice of conformance class:

   ☐ ISO 10303-21 Exchange Structure – preprocessor

   ☐ ISO 10303-21 Exchange Structure – postprocessor

2. Please provide an identifier for the product or system for which conformance is claimed:

   – Product name and current version number: _____

   – Preprocessor name and current version number: _____

   – Postprocessor name and current version number: _____

Annex E
(normative)

Information object registration

E.1 Document identification

In order to provide for unambiguous identification of an information object in an open system, the object identifier

{ iso standard 10303 part(201) version(1) }

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

E.2 Schema identification

In order to provide for unambiguous identification of the schema specifications given in this application protocol in an open information system, object identifiers are assigned as follows:

{ iso standard 10303 part(201) version(1) object(1) explicit-draughting-schema(1) }

is assigned to the Explicit_draughting_schema expanded schema (see Annex A).

{ iso standard 10303 part(201) version(1) object(1) explicit-draughting-schema(2) }

is assigned to the Explicit_draughting_schema short form schema (see 5.2).

The meaning of these values is defined in ISO 8824-1, and is described in ISO 10303-1.

Annex F
(informative)

Application activity model

The application activity model (AAM) is provided to aid in understanding the scope and information requirements defined in this application protocol. The model is presented as a set of definitions of the activities and the data and a set of activity figures. The AAM covers activities which go beyond the scope of this application protocol.

The figures use IDEF0 function modelling. Each activity may be decomposed to provide more detail. If an activity has been decomposed, a separate figure is included.

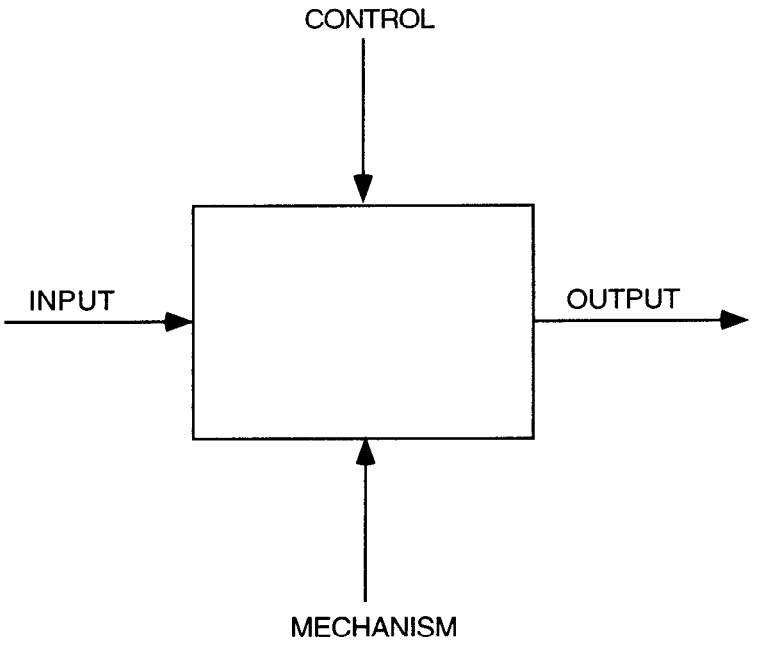

Figure F.1 – IDEF0 basic notation

As with any IDEF0 model, the application activity model is dependent on a particular viewpoint and purpose. The viewpoint of the application activity model is from a design engineer.

F.1 Application activity model definitions and abbreviations

The following terms are used in the application activity model. Terms marked with an asterisk are outside the scope of this application protocol.

The definitions given in this annex do not supersede the definitions given in the main body of the text.

F.1.1 2D draughting shape model: the two-dimensional representation, at any stage of development, of the physical shape and size of a product that has been produced or created for the purpose of presentation in a drawing.

F.1.2 annotate drawing sheet: select and apply a drawing sheet layout, as required, to the drawing sheet and annotate this drawing sheet with related administrative product information and any necessary additional annotation that is not related to a specific view.i

F.1.3 annotated drawing sheet: a drawing sheet that may contain drawing views and their view-related annotation, as well as annotation not related to a specific view, such as the drawing sheet layout and administrative product information.

F.1.4 annotated drawing view: a view of the 2D draughting shape model that contains annotation related to the geometric data presented in the view.

F.1.5 approved drawing sheet: a drawing sheet to which an organization or project specific approval has been assigned.

F.1.6 CAD system*: a computer system used for creation, revision, storage, and control of CAD drawings created from a 2D draughting shape model with explicit annotation.

F.1.7 change requests*: detailed information and instructions issued to a draughtsman to create or modify a drawing.

F.1.8 configure 2D draughting shape model : analyze the view requirements and modify, as necessary, the 2D draughting shape model to support these views.

F.1.9 configure drawing: order the sheets of the drawing, add product information that is relative to the entire drawing to each sheet in the form of annotation, and approve the drawing.

F.1.10 configure drawing sheet: select, locate, and orient drawing views with respect to a drawing sheet, annotate the drawing sheet, and approve the drawing sheet as required by any organization-specific practices.

F.1.11 configure drawing view: create and annotate the views of the 2D draughting shape model that are presented on a drawing sheet.

F.1.12 create drawing view: scale and place a view of a 2D draughting shape model on the drawing sheet and configure and place annotation on the view as required to present product information. The appearance of any elements of the 2D draughting shape model presented in the view maybe modified to accurately portray product shape.

F.1.13 create model view: define the perspective from which the model will be viewed and specify any clipping required to limit the view to only those desired elements.

F.1.14 document drawing sheet approval: annotate a drawing sheet with an organization-

or project-specific approval .

F.1.15 document product: develop the presentation of the form, fit, and functional requirements of a product in the form of a drawing.

F.1.16 draughting standards*: organization, project, national, or international standards that specify the manner in which product information is depicted in a drawing.

F.1.17 draughtsman*: a person responsible for documenting a product who makes use of a CAD system to create or modify a drawing.

F.1.18 drawing requirements*: constraints on the configuration, contents, or appearance of a drawing, from the organization, project or both.

F.1.19 drawing sheet: a subdivision of a drawing that is presented as an image on a single piece of paper or other suitable output medium.

F.1.20 existing CAD drawing: a CAD drawing that is to be modified in response to a change request.

F.1.21 existing drawing sheet: a drawing sheet that is to be modified in response to a change request.

F.1.22 existing drawing view: a drawing view that is to be modified in response to a change request.

F.1.23 model view: a two-dimensional planar projection of a 2D draughting shape model from a specified position within its coordinate system. A view may include clipping in order to capture only certain elements of the object being viewed.

F.1.24 non-shape product information*: information that describes physical aspects, functional aspects, or both, of a product other than its physical shape.

EXAMPLE 20 – Colour and surface finish are examples of non-shape product information.

F.1.25 organization/project requirements*: requirements imposed by an organization or a project that define the creation or modification of a drawing.

F.1.26 position drawing views: locate and orient drawing views on a drawing sheet.

F.1.27 product information: information that describes the physical or functional aspects of a product.

F.1.28 refined 2D draughting shape model: the representation of the physical shape of a product that supports all draughting requirements for the presentation of that shape on a drawing.

F.1.29 revised CAD drawing: a CAD drawing which has been modified in response to a change request. An organization or project specific approval may have been assigned to the CAD drawing as well.

F.1.30 select drawing sheet layout and size: specify the size of a drawing sheet to be used in the drawing and any standard drawing layout to be included on that sheet.

F.1.31 sheet approval requirements*: requirements that govern the approval of drawing sheets for an organization, a project, or both.

F.1.32 sheet layout and size requirements*: requirements that constrain the layout, the sheet size, or both, of drawing sheets for an organization, a project, or both.

F.1.33 sheet layout symbol: a symbol used to present standard drawing layouts.

F.1.34 sheet requirements*: requirements that constrain the configuration, the content, the appearance, or all three of Drawing_sheets for an organization, a project, or both.

F.1.35 sheet with layout: a drawing sheet that has been annotated with a sheet layout symbol.

F.1.36 sheet with views: a drawing sheet that contains one or more drawing views annotated as required.

F.1.37 sheet without views: a drawing sheet that does not contain any drawing views.

F.1.38 symbols and fonts: collections of text fonts, symbol fonts, and other symbol definitions that are used as annotation by a draughtsman in the creation or modification of a CAD drawing.

F.1.39 unapproved drawing sheet: a drawing sheet to which no organization or project specific approval has been assigned.

F.1.40 view location requirements*: requirements that constrain the positioning of drawing views on a drawing sheet for an organization, a project, or both.

F.1.41 view requirements*: requirements that constrain the contents, the appearance, or both of drawing views for an organization, a project, or both.

F.2 Application activity model diagrams

The application activity model is given in F.2 through F.5. The graphical form of the application activity model is presented in the IDEF0 activity modelling format. Activities and data flows that are out of scope are marked with asterisks.

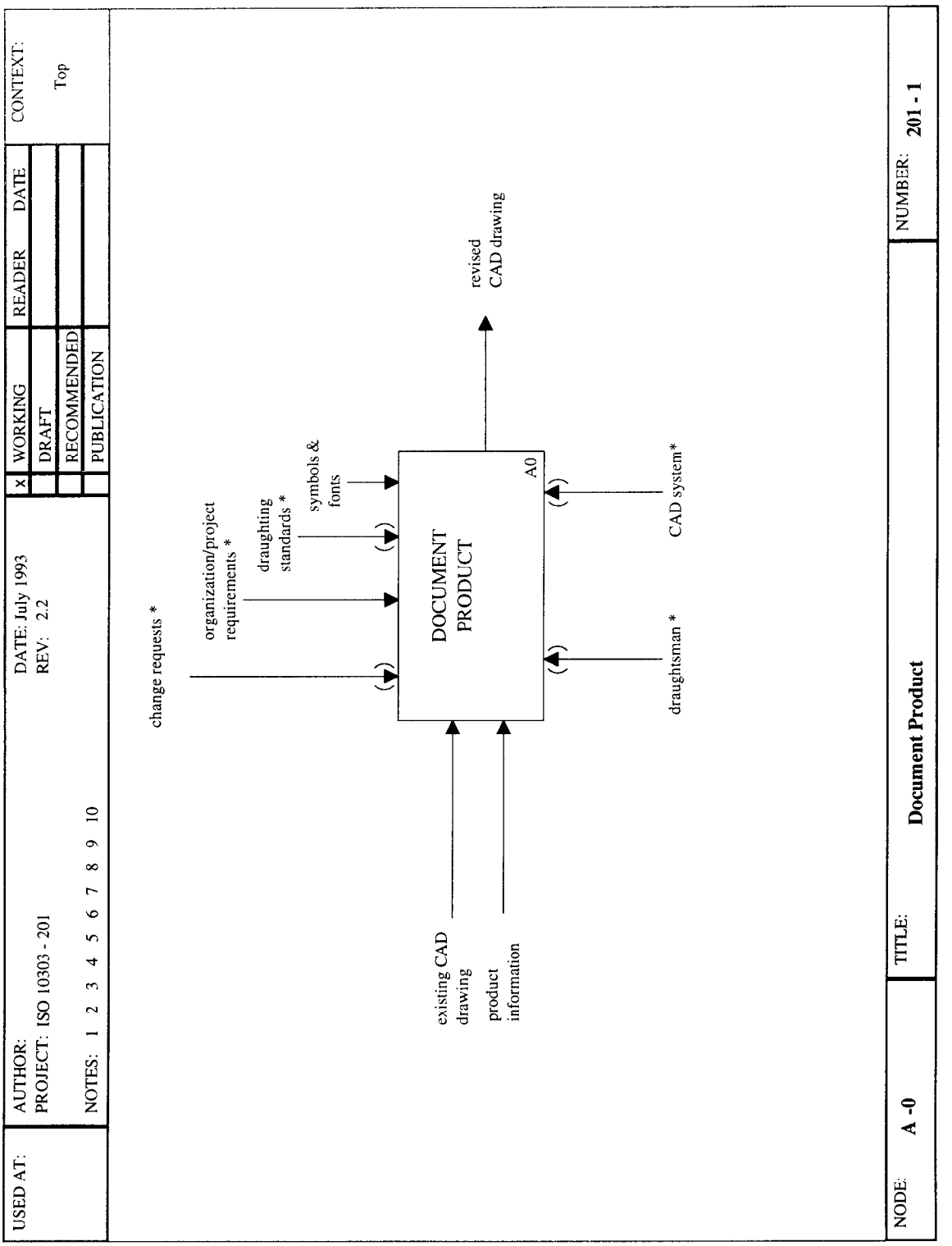

Figure F.2 – A-0: document product

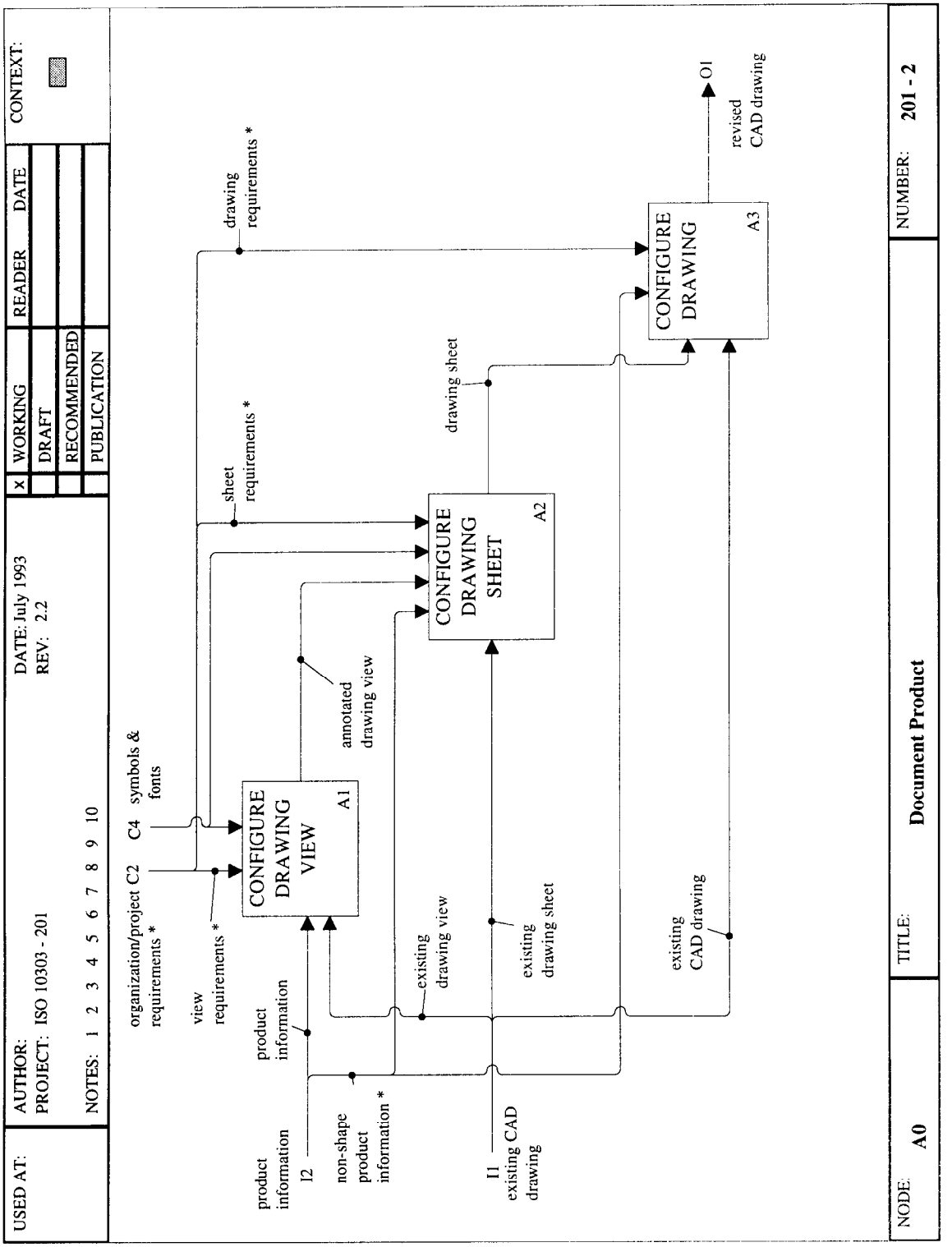

Figure F.3 – A0: document product

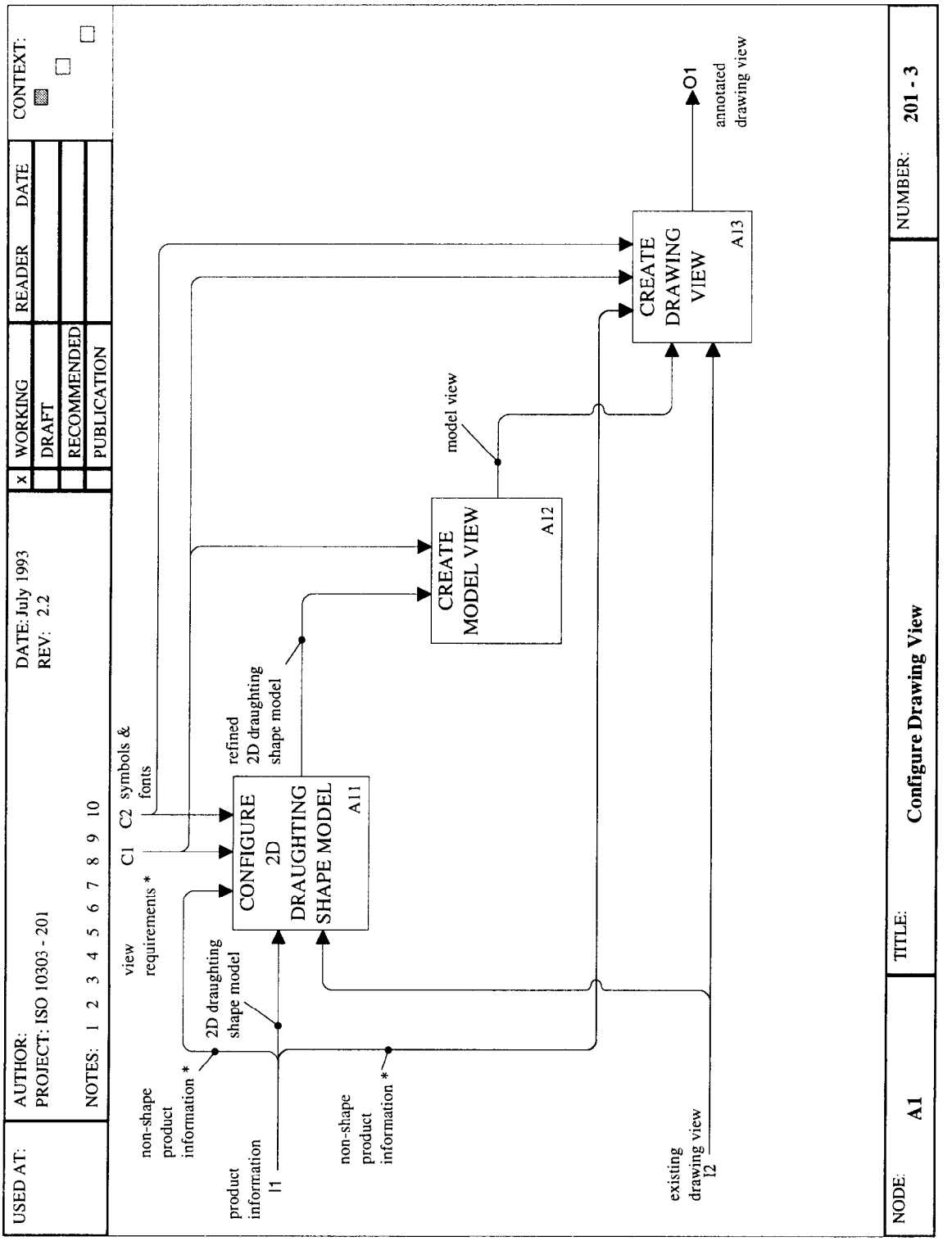

Figure F.4 – A1: configure drawing view

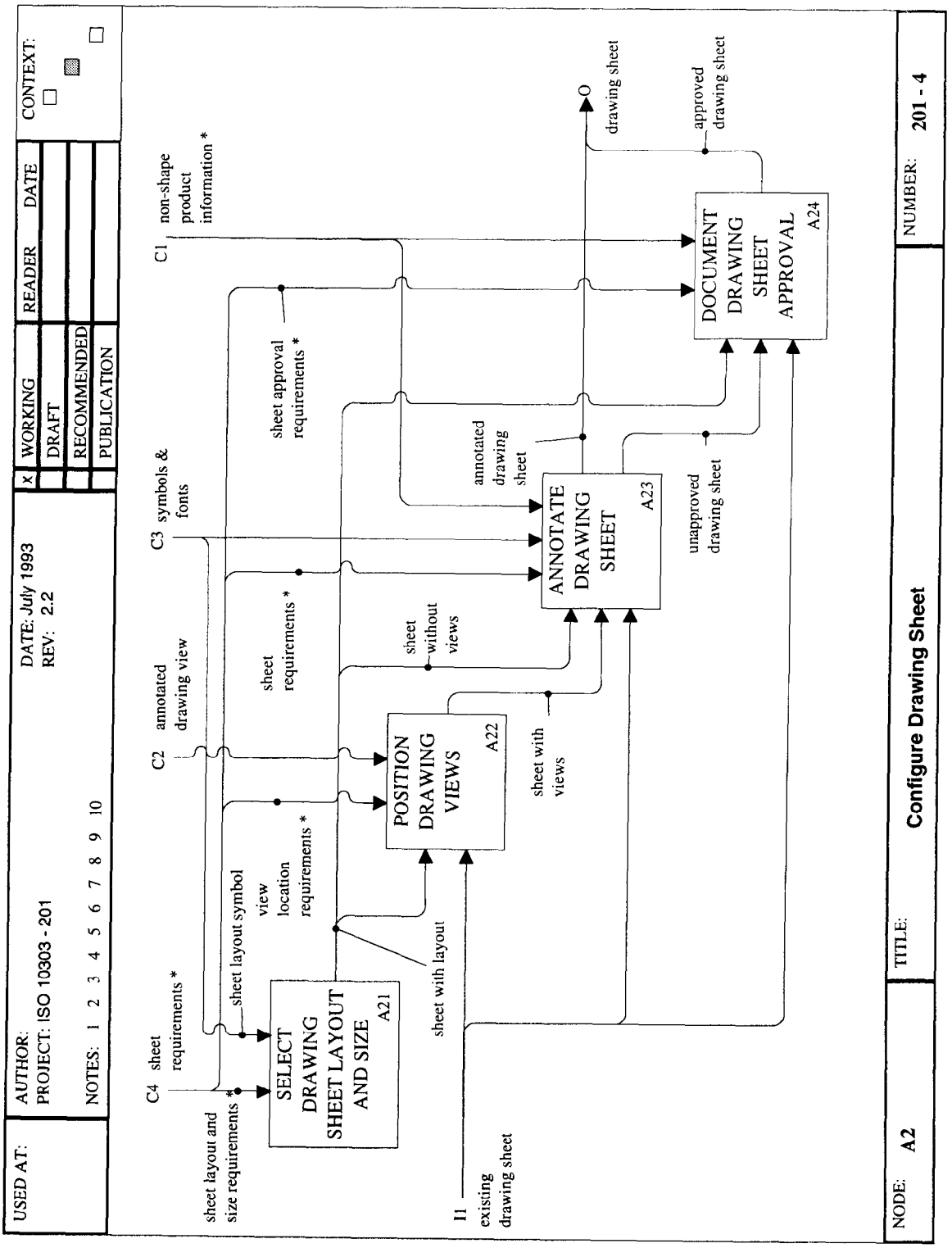

Figure F.5 – A2: configure drawing sheet

Annex G
(informative)

Application reference model

This annex provides the application reference model for this part of ISO 10303 and is given in G.1 through G.15. The application reference model is a graphical representation of the structure and constraints of the application objects specified in clause 4. The graphical form of the application reference model is presented in the IDEF1X format. The application reference model is independent from any implementation method.

The graphical form of the application reference model is presented in IDEF1X. Extentions to the IDEF1X notation are used within the diagrams. The "-O" notation designates that an attribute is optional. The "-*" notation designates that an attribute can be aggregated. The different types of predefined objects that are supported by this part of ISO 10303 are listed in capital letters under the application object. Application objects shown with a dashed line font are defined on the diagram indicated within the object. Attributes which do not fulfill user requirements and serve only to identify an object are excluded from the model.

NOTE – The table G.1, which follows the diagrams, provides an index to these diagrams and lists all the cross references between them.

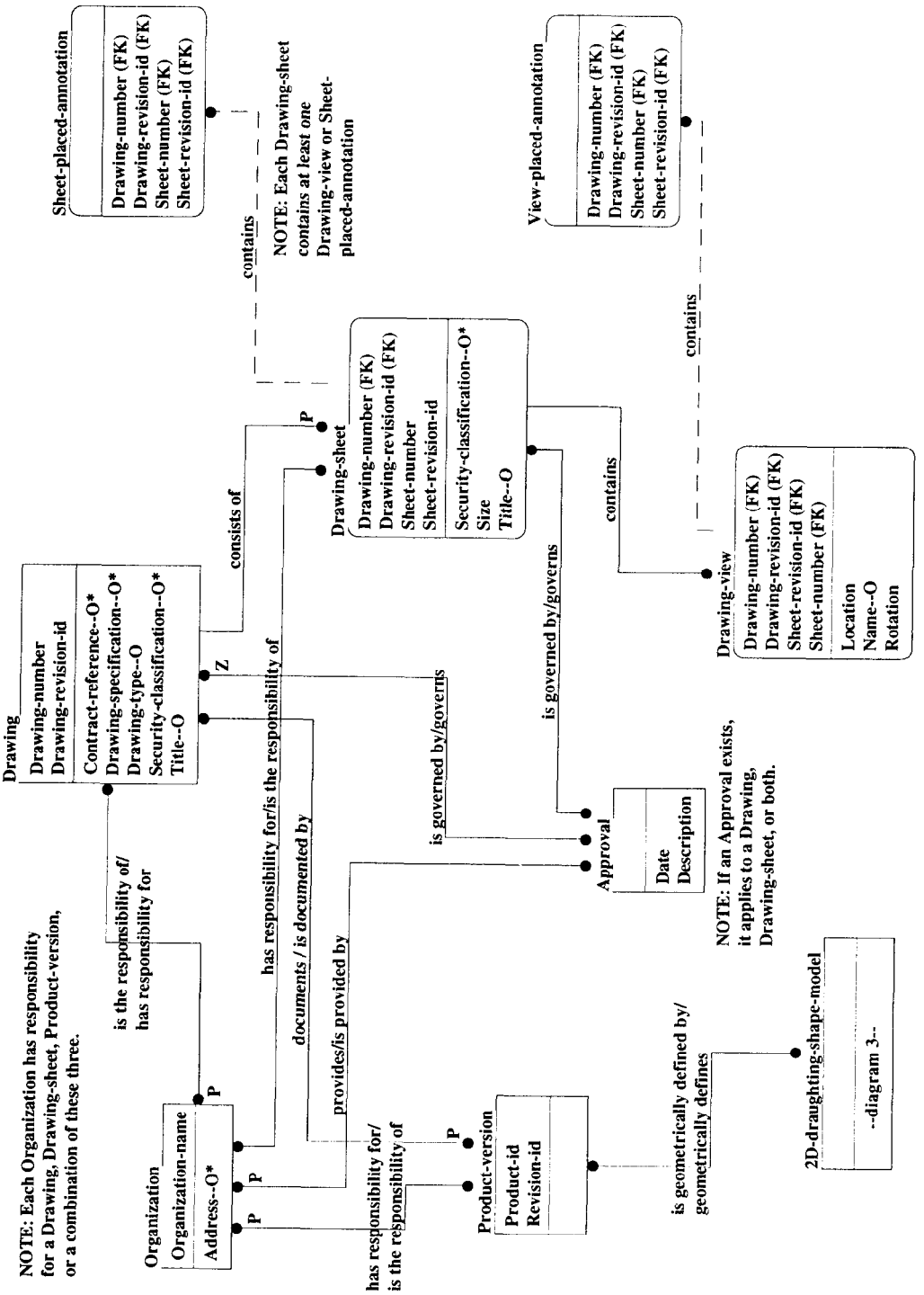

Figure G.1 – ARM diagram 1 of 15

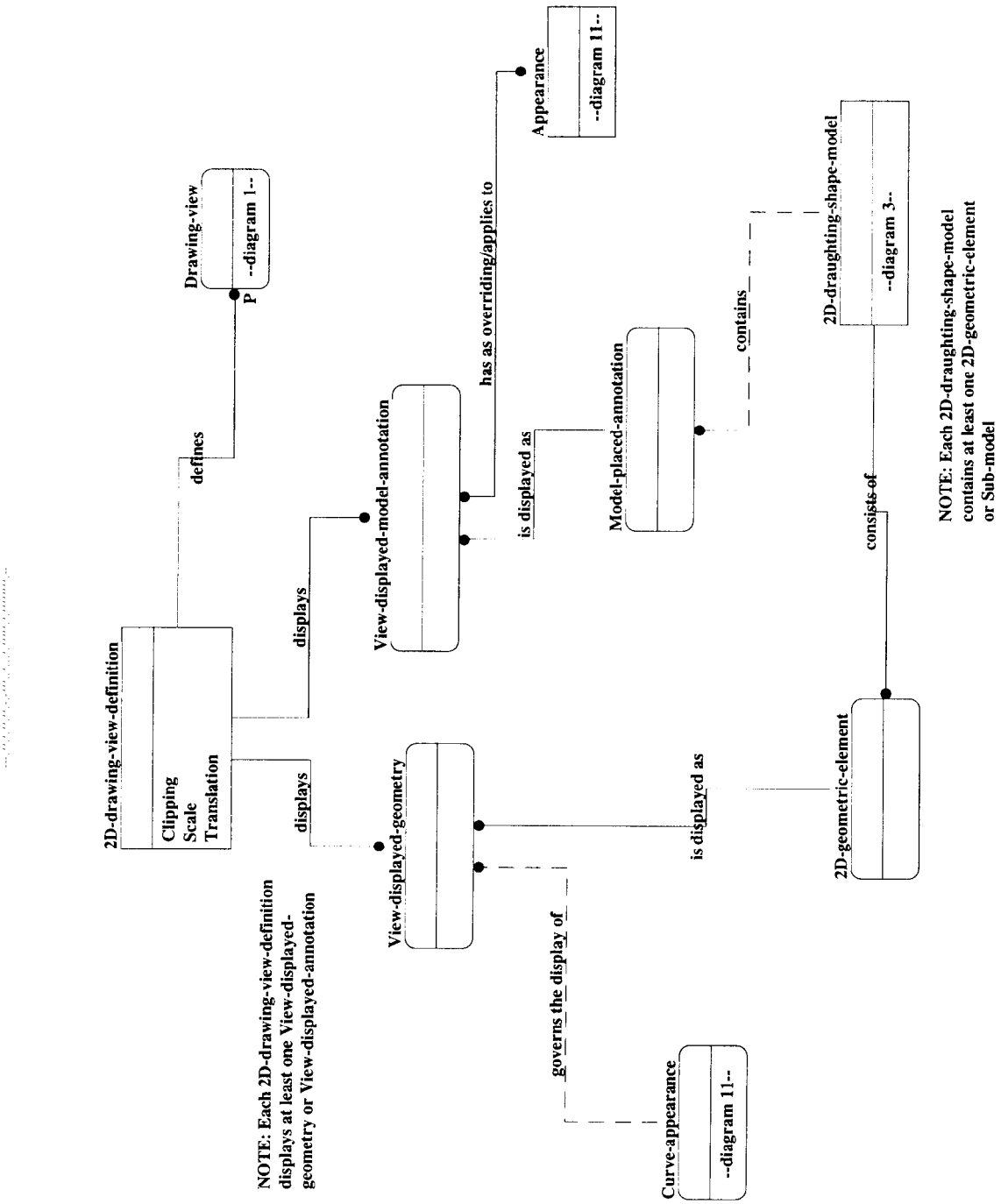

Figure G.2 – ARM diagram 2 of 15

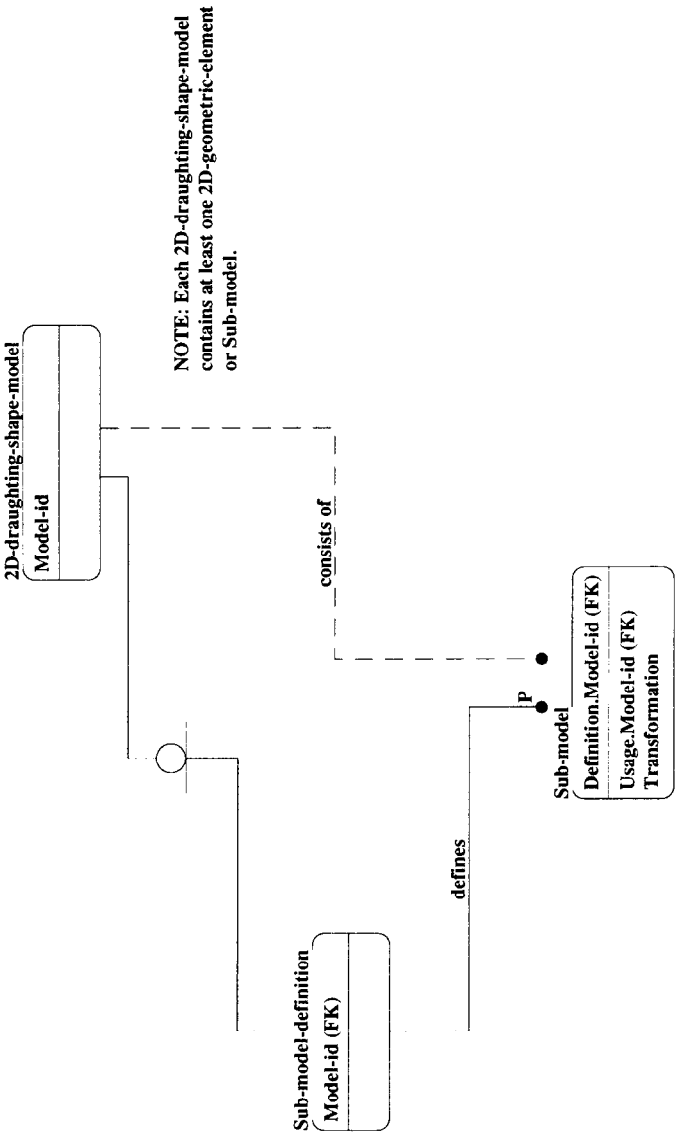

Figure G.3 – ARM diagram 3 of 15

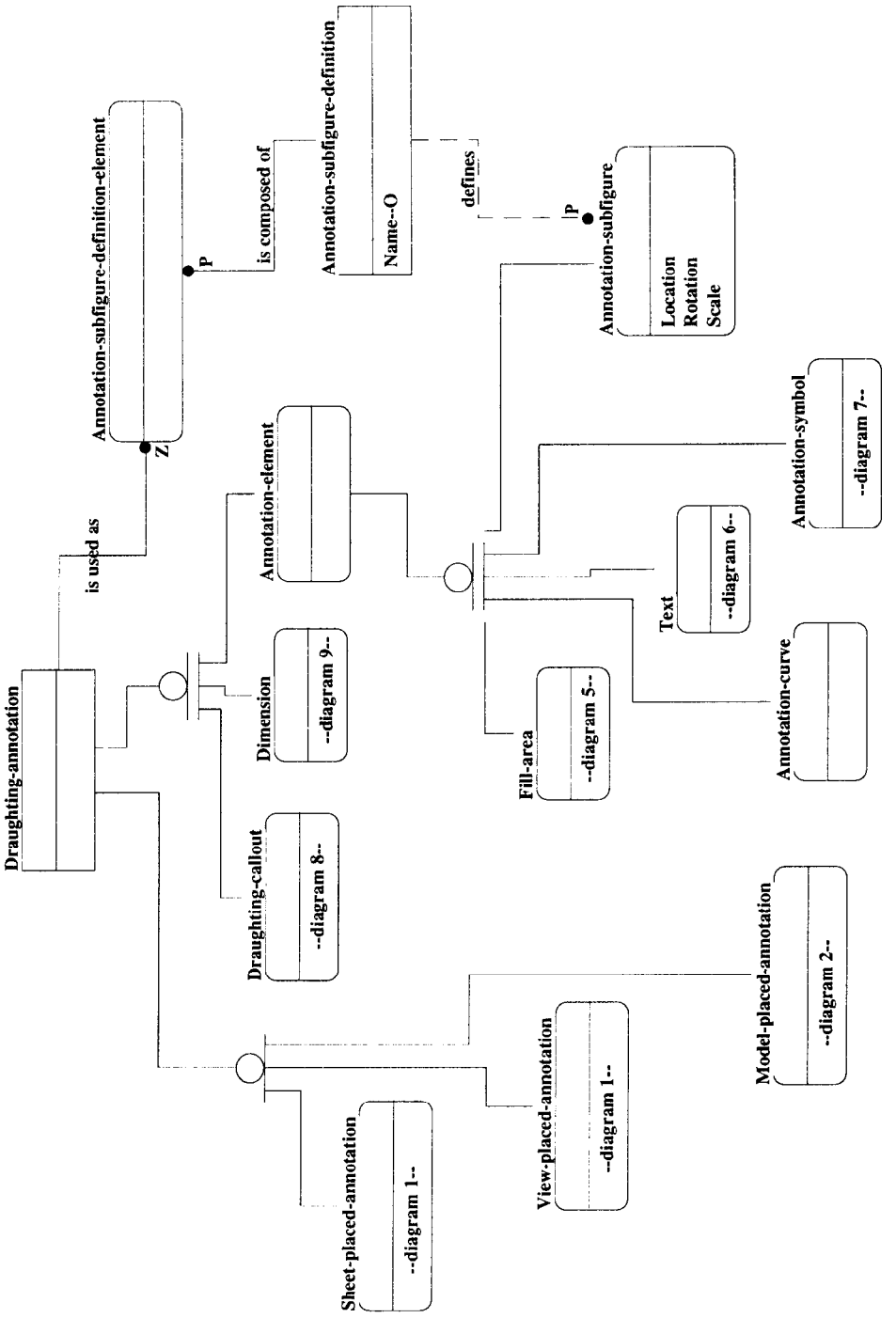

Figure G.4 – ARM diagram 4 of 15

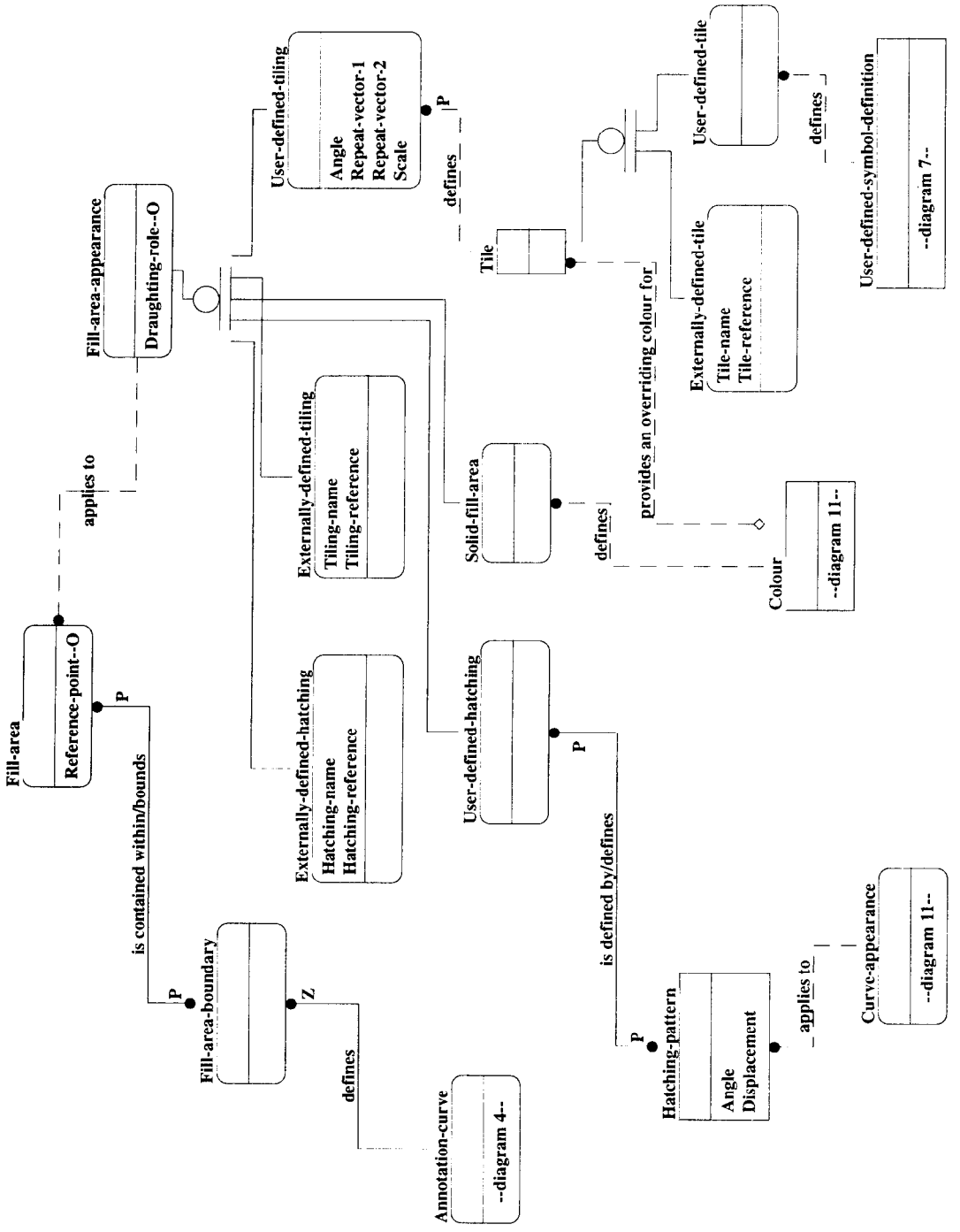

Figure G.5 – ARM diagram 5 of 15

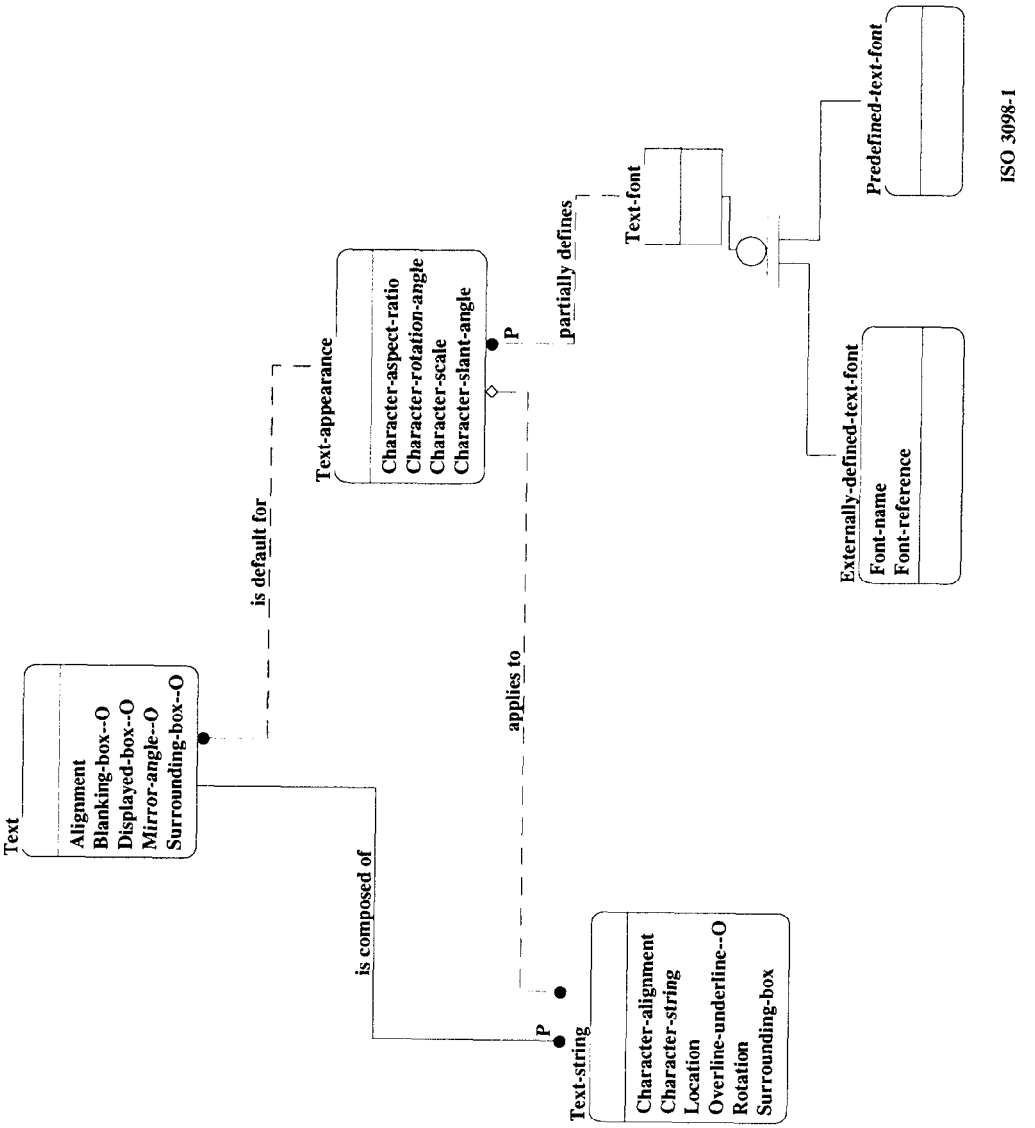

Figure G.6 – ARM diagram 6 of 15

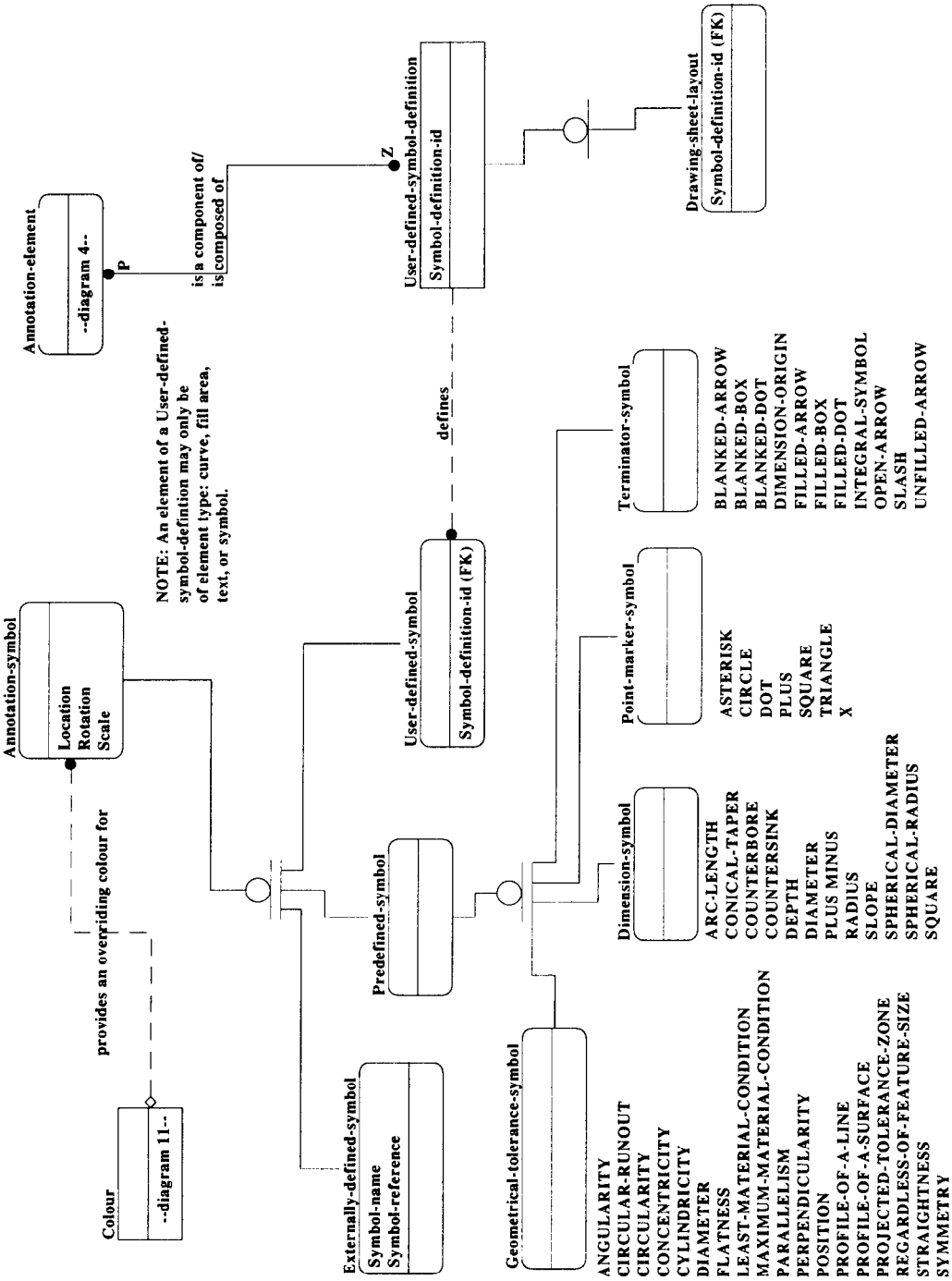

Figure G.7 – ARM diagram 7 of 15

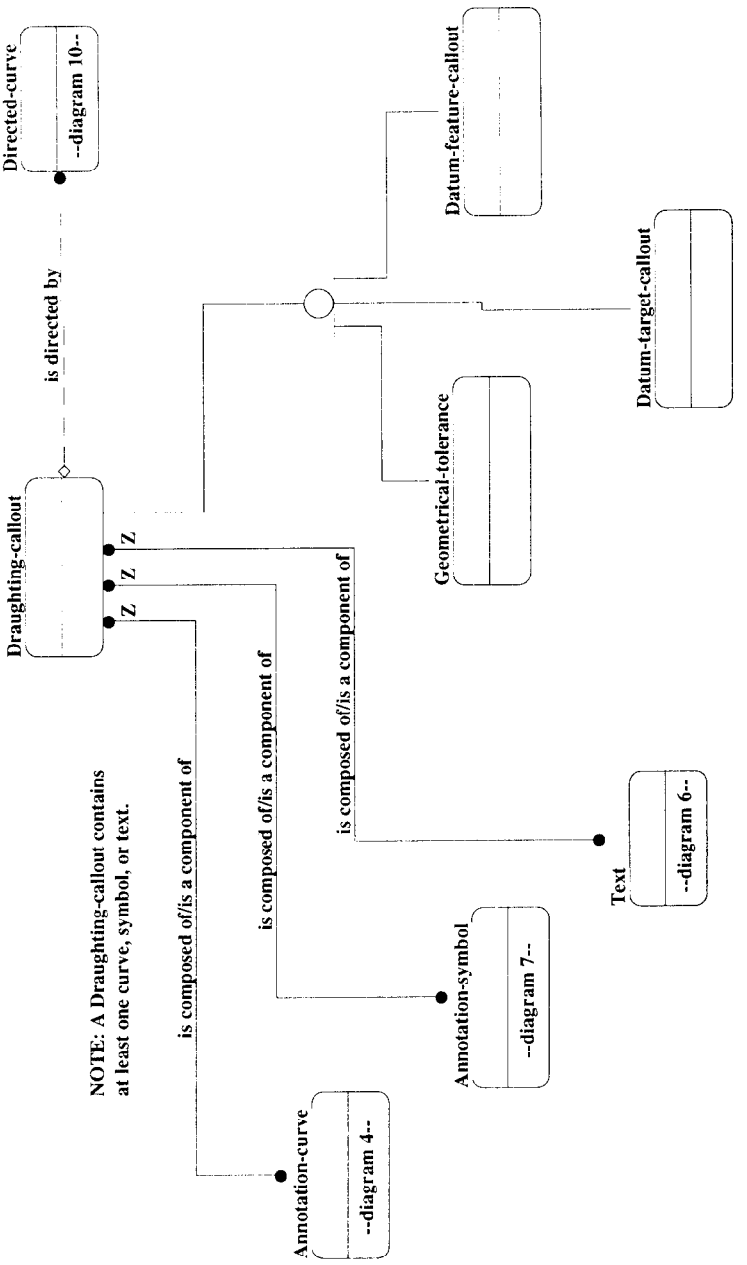

Figure G.8 – ARM diagram 8 of 15

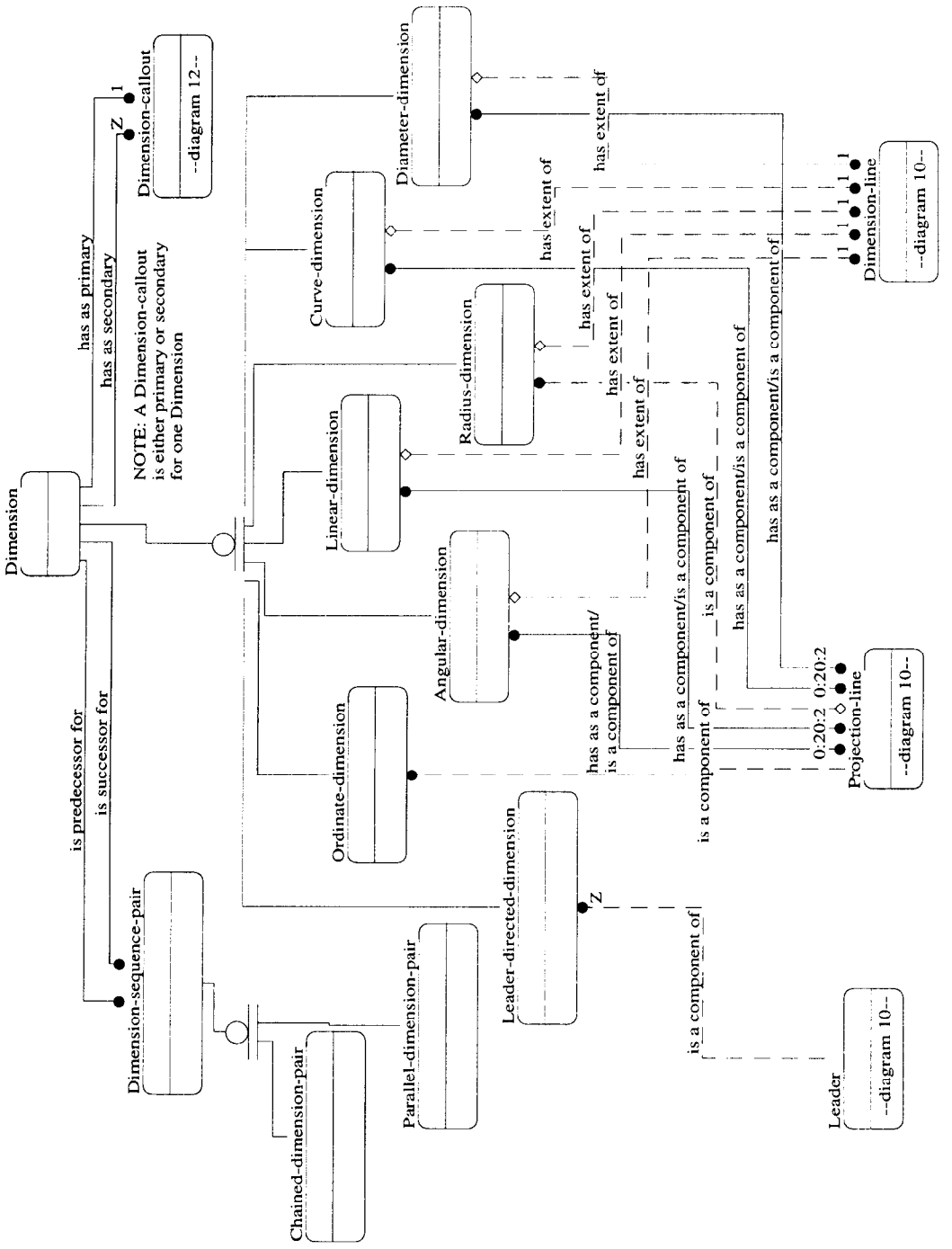

Figure G.9 – ARM diagram 9 of 15

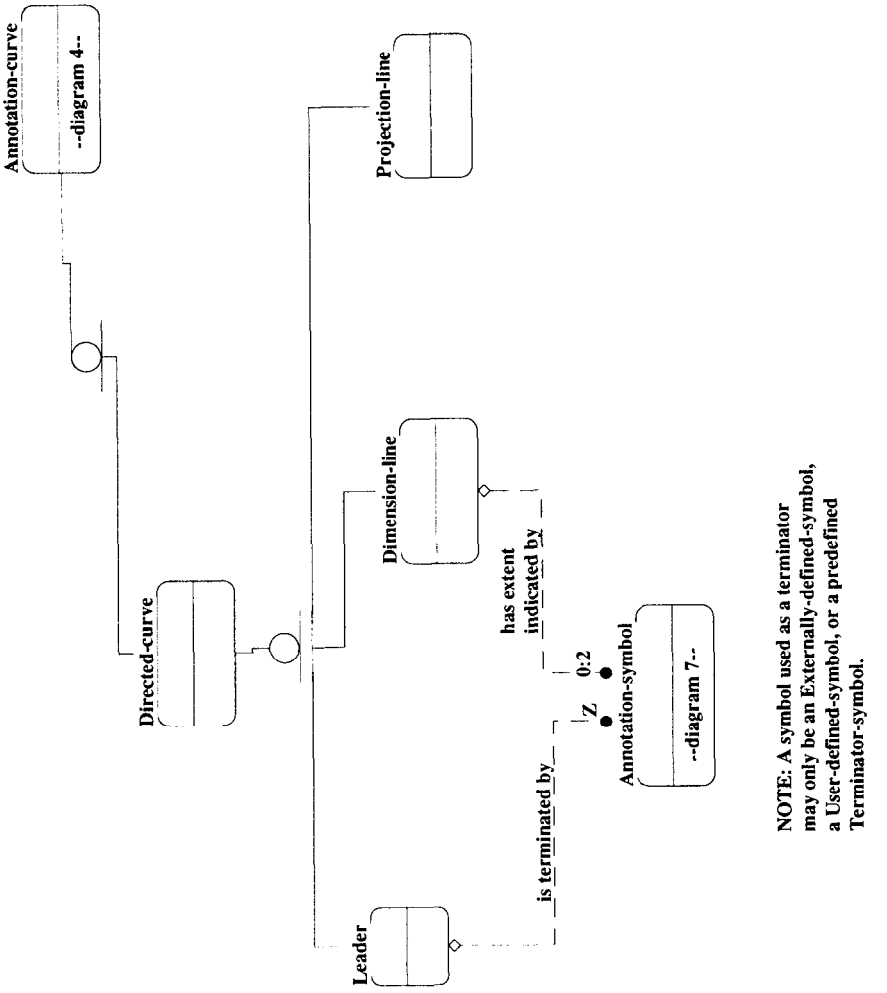

Figure G.10 – ARM diagram 10 of 15

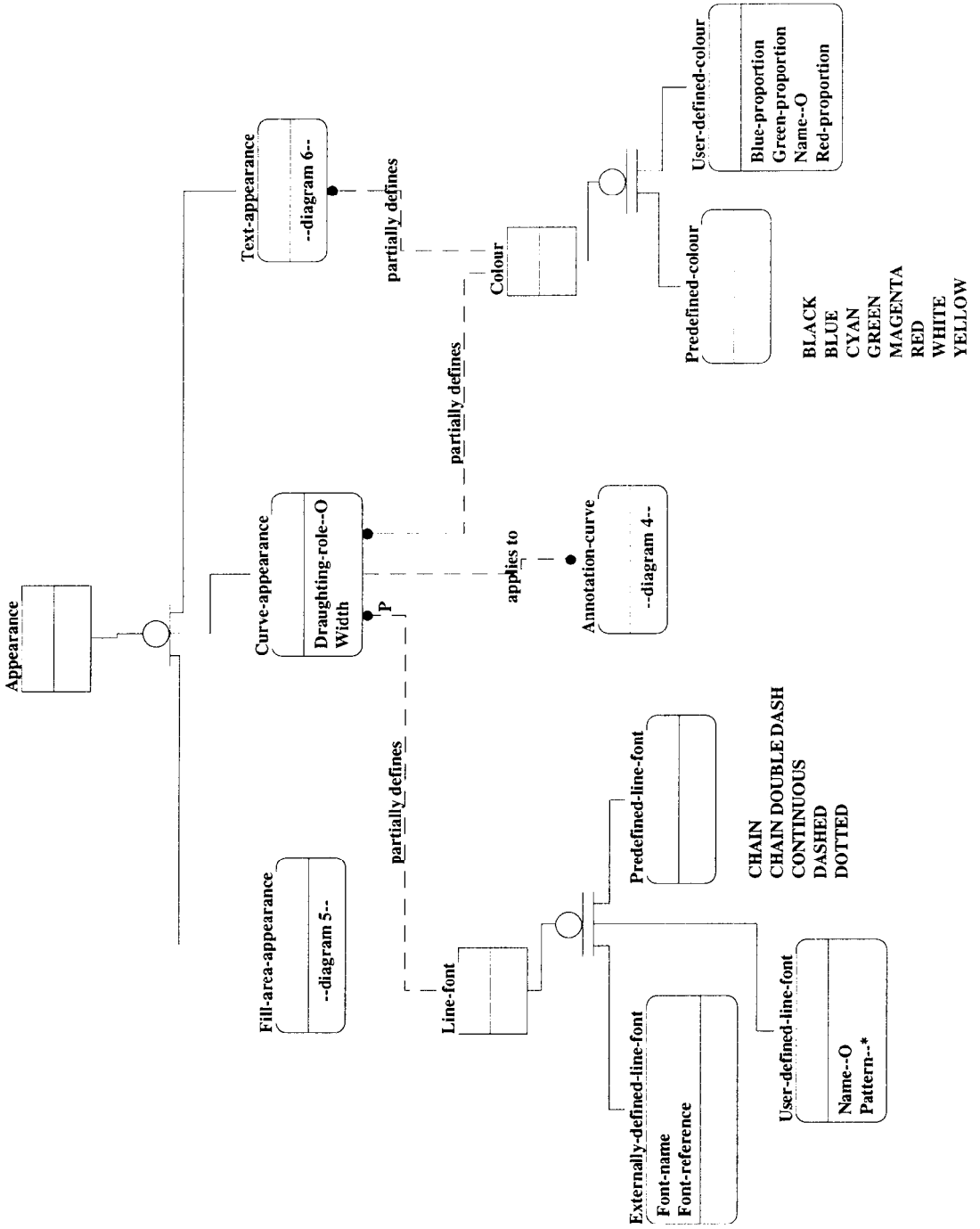

Figure G.11 – ARM diagram 11 of 15

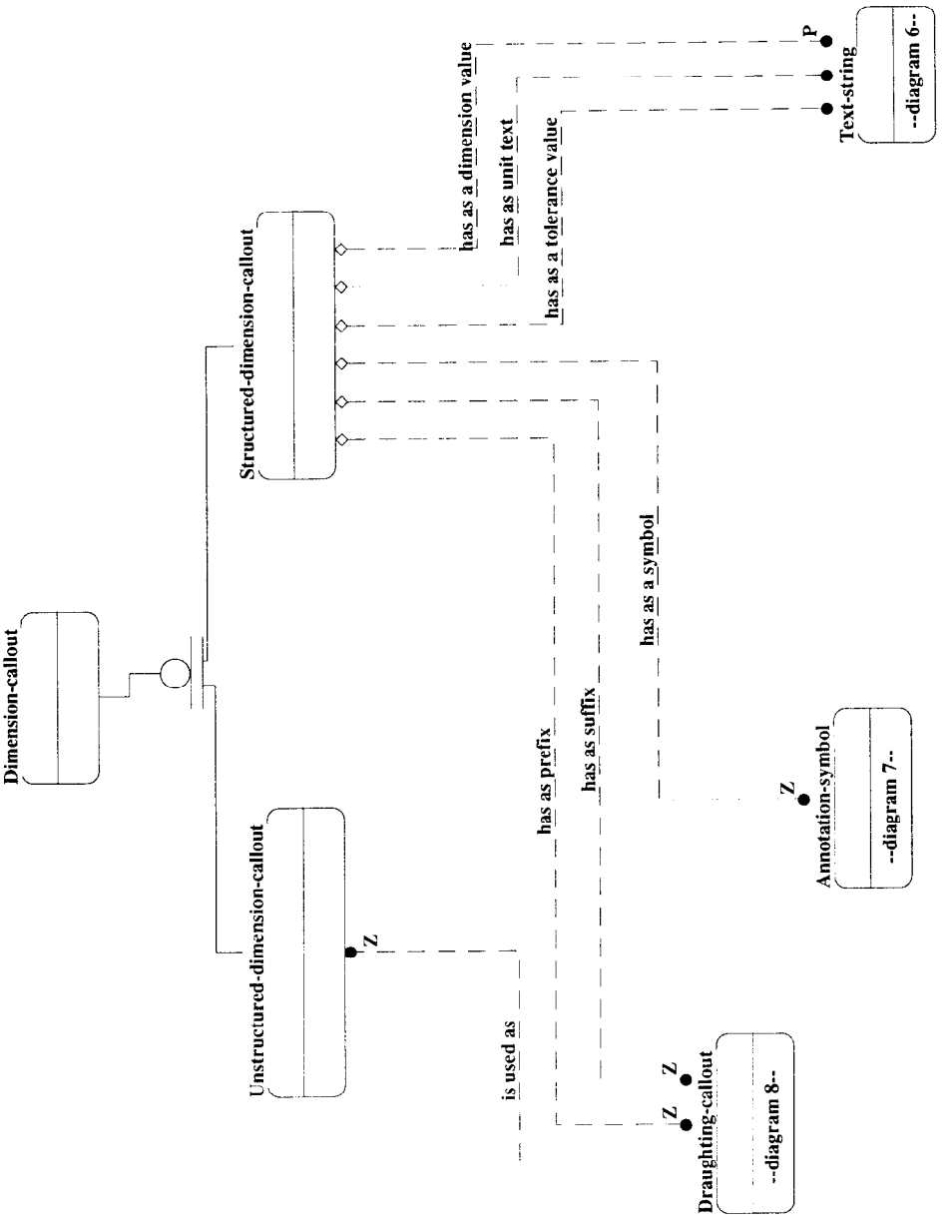

Figure G.12 – ARM diagram 12 of 15

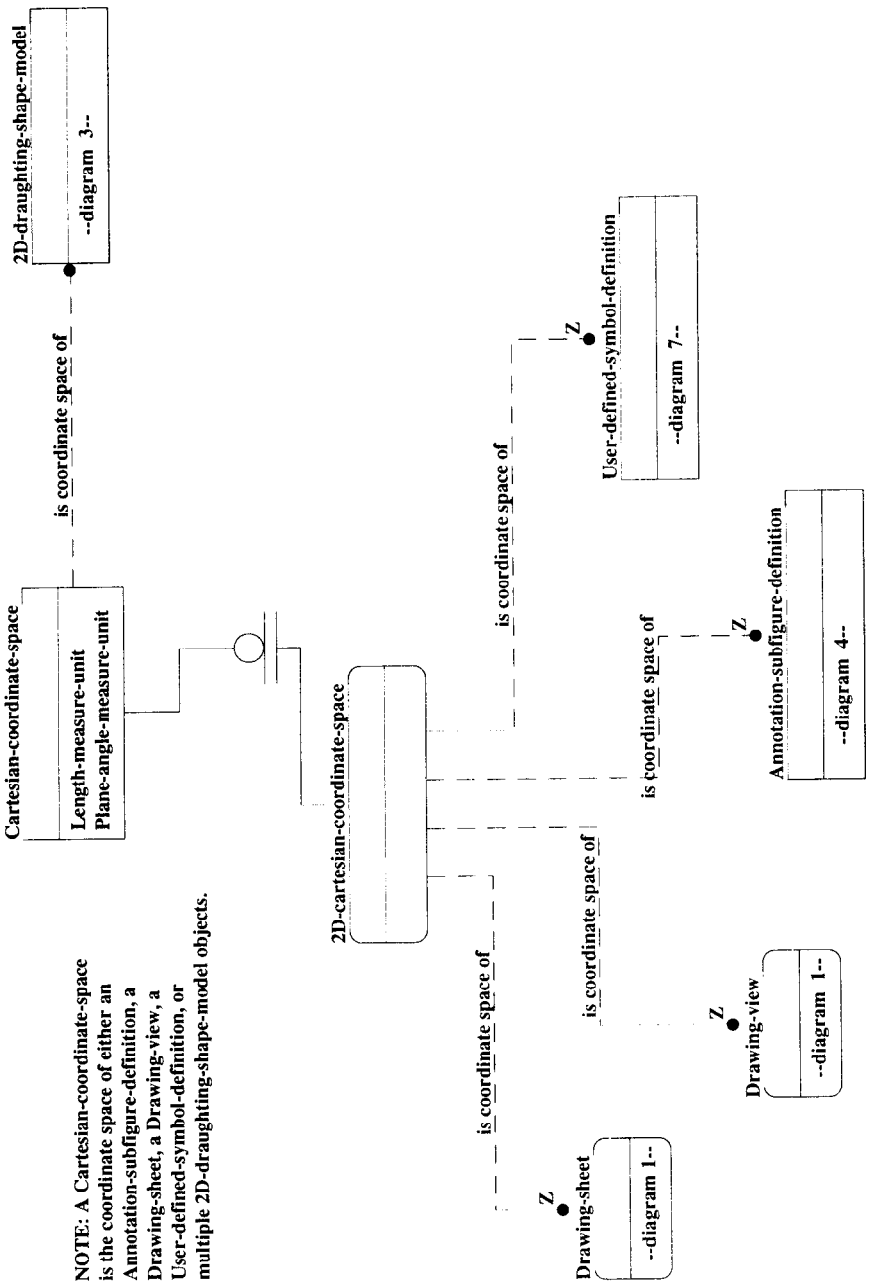

Figure G.13 – ARM diagram 13 of 15

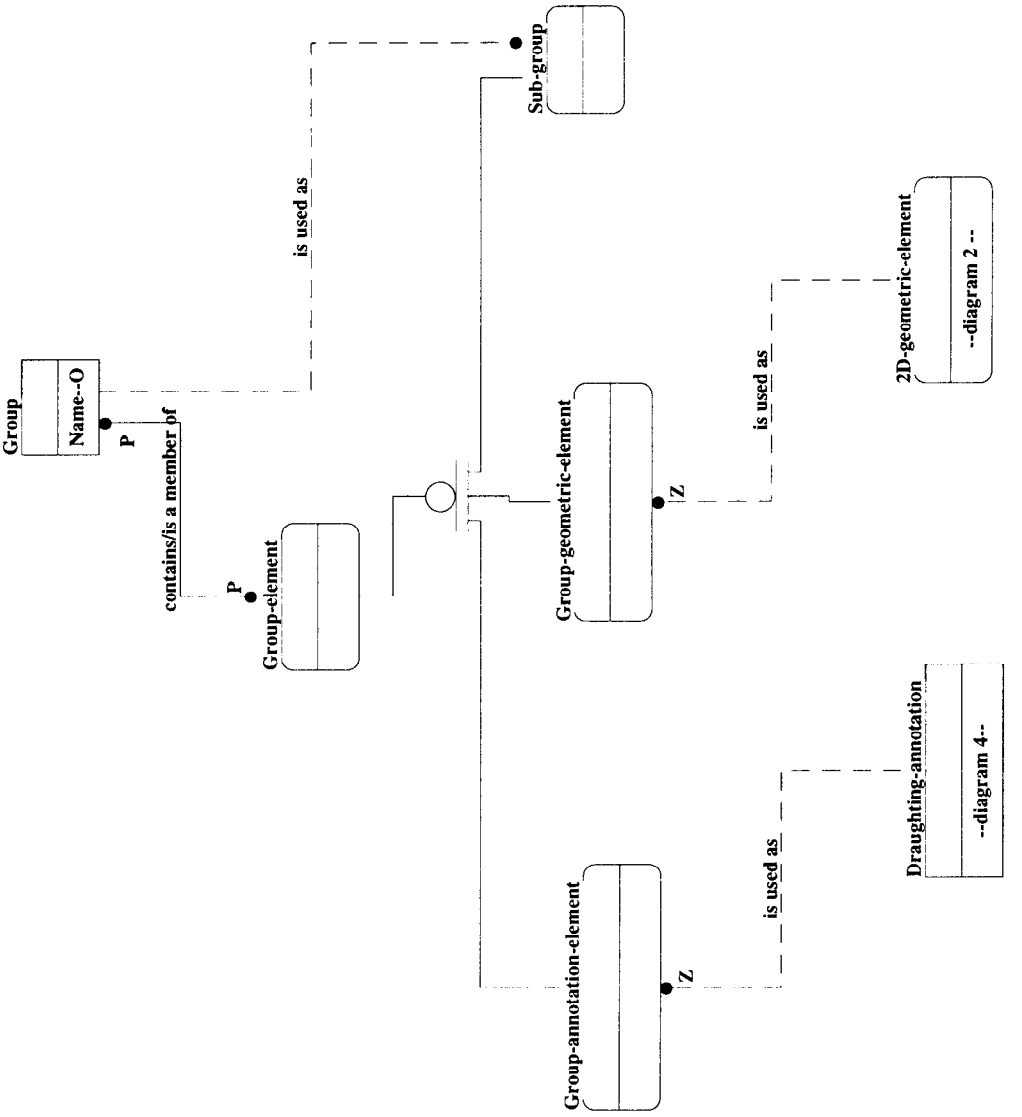

Figure G.14 – ARM diagram 14 of 15

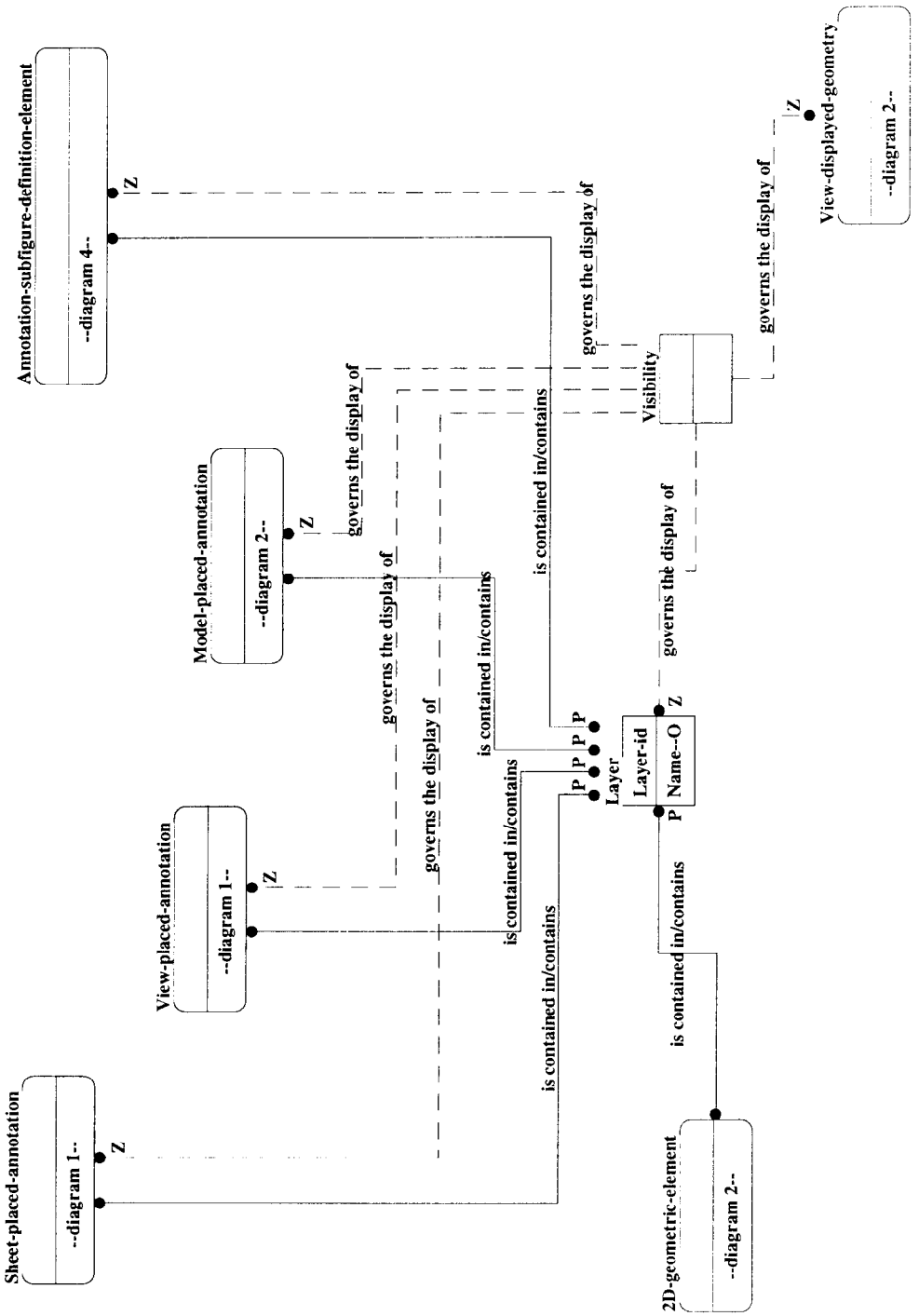

Figure G.15 – ARM diagram 15 of 15

Table G.1 – Cross reference table for ARM diagrams

Figure	Application object	Referenced from
F.1	Approval	
	Drawing	
	Drawing_sheet	F.13
	Drawing_view	F.2, F.13
	Organization	
	Product_version	
	Sheet_placed_annotation	F.4, F.15
	View_placed_annotation	F.4, F.15
F.2	2D_drawing_view_definition	
	2D_geometric_element	F.14, F.15
	Model_placed_annotation	F.4, F.15
	View_displayed_geometry	F.15
	View_displayed_model_annotation	
F.3	2D_draughting_shape_model	F.1, F.2, F.13
	Sub_model	
	Sub_model_definition	
F.4	Annotation_curve	F.5, F.8, F.10, F.11
	Annotation_element	F.7
	Annotation_subfigure	
	Annotation_subfigure_definition	F.13
	Annotation_subfigure_definition_element	F.15
	Draughting_annotation	F.14
F.5	Externally_defined_hatching	
	Externally_defined_tile	
	Externally_defined_tiling	
	Fill_area	F.4
	Fill_area_appearance	F.11
	Fill_area_boundary	
	Hatching_pattern	
	Solid_fill_area	
	Tile	
	User_defined_hatching	
	User_defined_tile	
	User_defined_tiling	
F.6	Externally_defined_text_font	
	Predefined_text_font	
	Text	F.4, F.8
	Text_appearance	F.11
	Text_font	
	Text_string	F.12

Cross reference table for ARM diagrams (continued)

Figure	Application object	Referenced from
F.7	Annotation_symbol	F.4, F.8, F.10, F.12
	Dimension_symbol	
	Drawing_sheet_layout	
	Externally_defined_symbol	
	Geometrical_tolerance_symbol	
	Point_marker_symbol	
	Predefined_symbol	
	Terminator_symbol	
	User_defined_symbol	
	User_defined_symbol_definition	F.5, F.13
F.8	Datum_feature_callout	
	Datum_target_callout	
	Draughting_callout	F.4, F.12
	Geometrical_tolerance	
F.9	Angular_dimension	
	Chained_dimension_pair	
	Curve_dimension	
	Diameter_dimension	
	Dimension	F.4
	Dimension_sequence_pair	
	Leader_directed_dimension	
	Linear_dimension	
	Ordinate_dimension	
	Parallel_dimension_pair	
	Radius_dimension	
F.10	Dimension_line	F.9
	Directed_curve	F.8
	Leader	F.9
	Projection_line	F.9
F.11	Appearance	F.2
	Colour	F.5, F.7
	Curve_appearance	F.2, F.5
	Externally_defined_line_font	
	Line_font	
	Predefined_colour	
	Predefined_line_font	
	User_defined_colour	
	User_defined_line_font	

Cross reference table for ARM diagrams (concluded)

Figure	Application object	Referenced from
F.12	Dimension_callout	F.9
	Structured_dimension_callout	
	Unstructured_dimension_callout	
F.13	2D_cartesian_coordinate_space	
	Cartesian_coordinate_space	
F.14	Group	
	Group_annotation_element	
	Group_element	
	Group_geometric_element	
	Sub_group	
F.15	Layer	
	Visibility	

Annex H
(informative)

AIM EXPRESS-G

H.1 through H.52 correspond to the AIM *EXPRESS* listing given in annex A. The figures use the *EXPRESS-G* graphical notation for the *EXPRESS* language. *EXPRESS-G* is defined in annex D of ISO 10303-11.

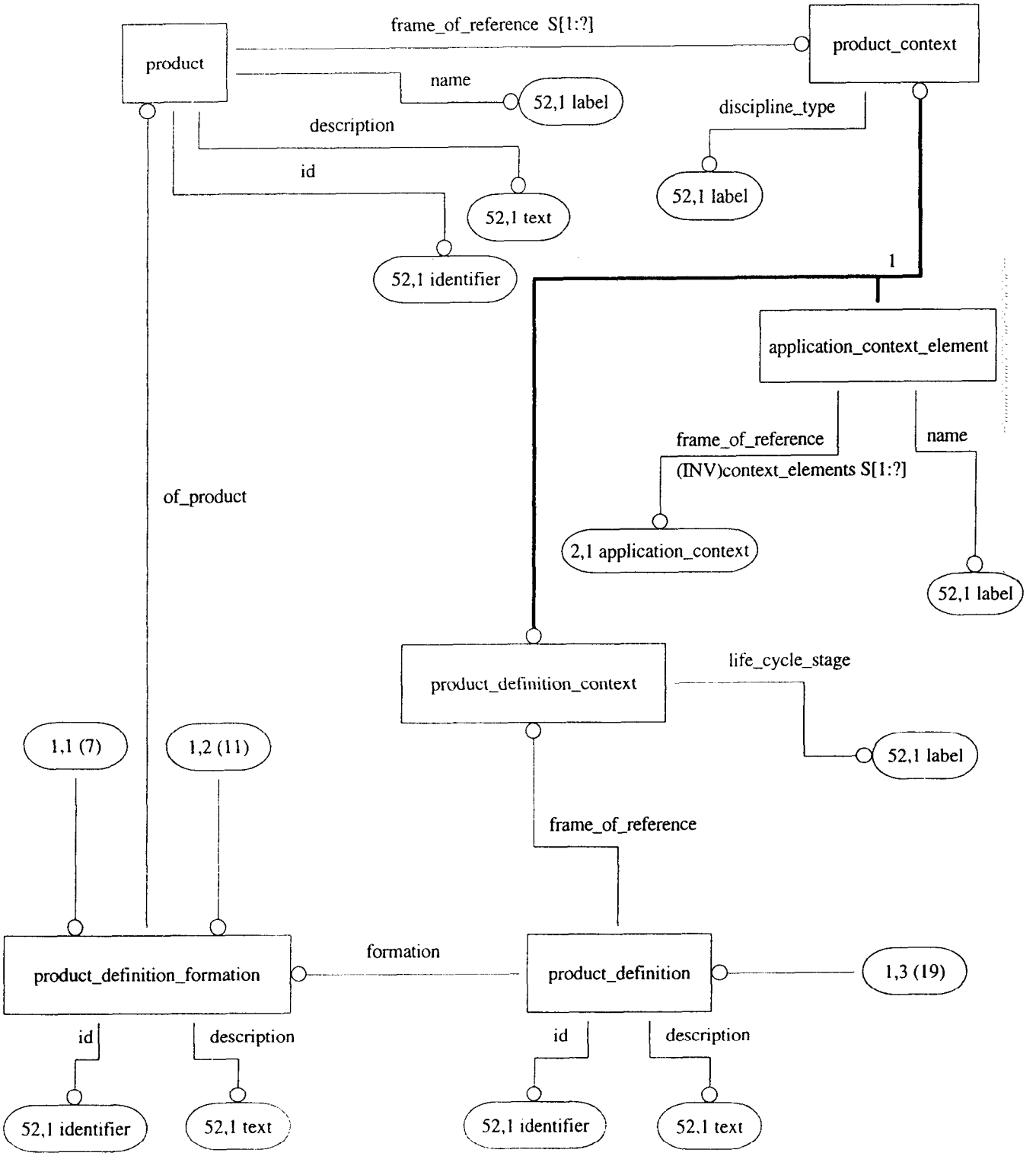

Figure H.1 – AIM EXPRESS-G diagram 1 of 52

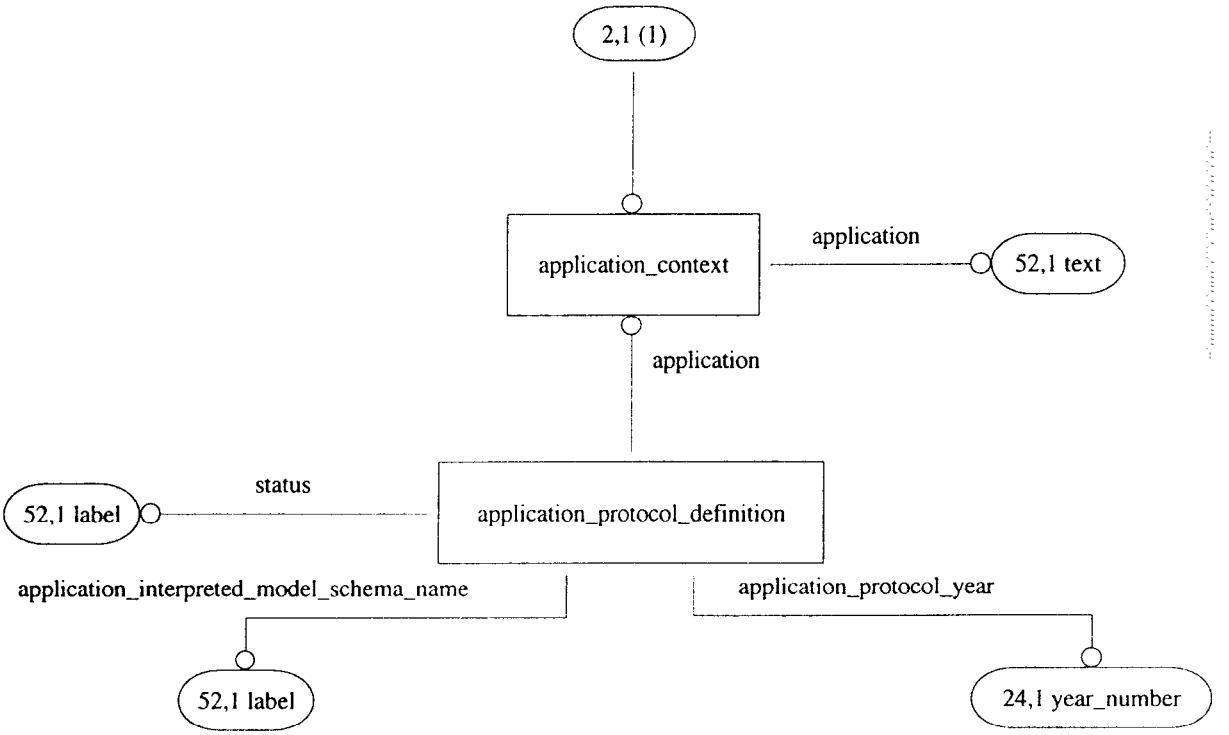

Figure H.2 – AIM EXPRESS-G diagram 2 of 52

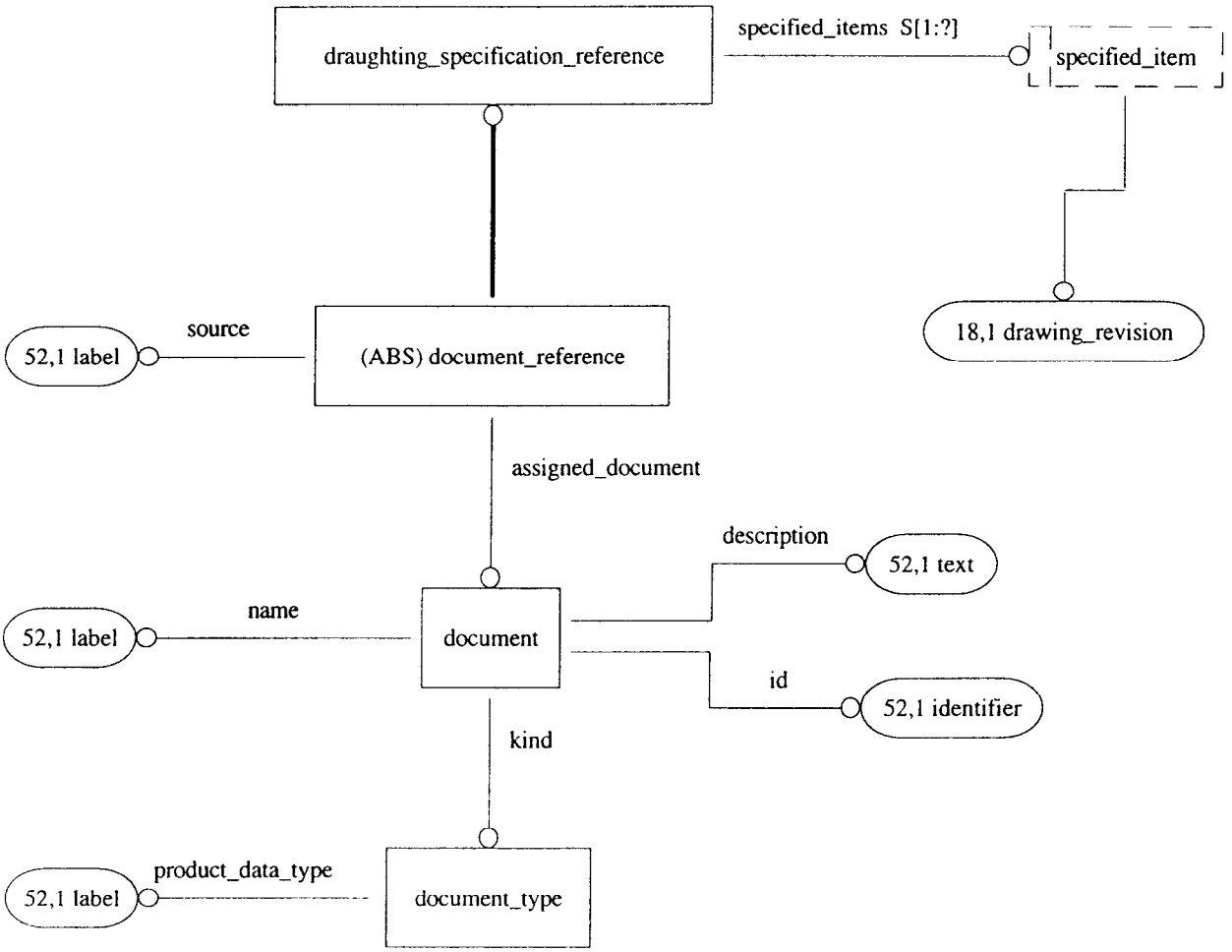

Figure H.3 – AIM EXPRESS-G diagram 3 of 52

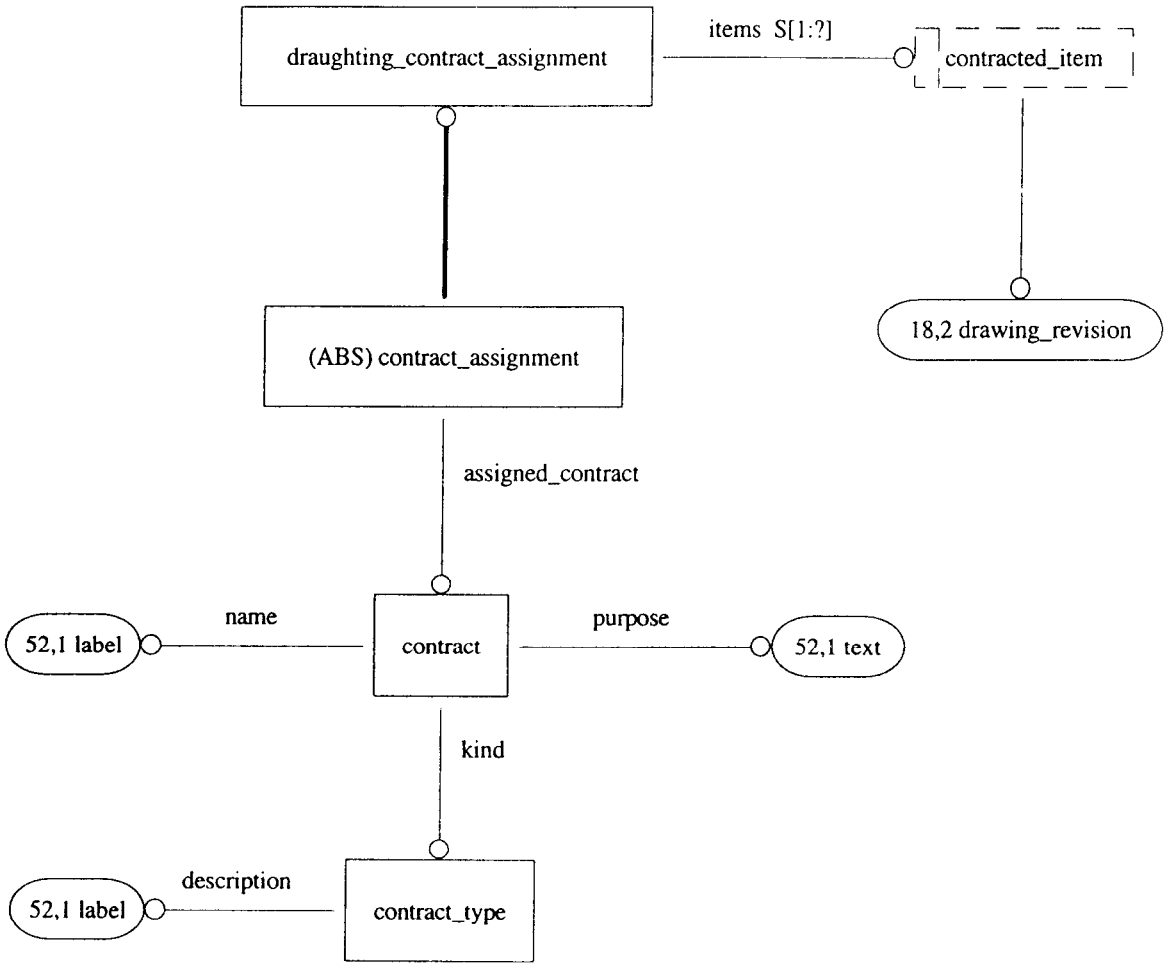

Figure H.4 – AIM EXPRESS-G diagram 4 of 52

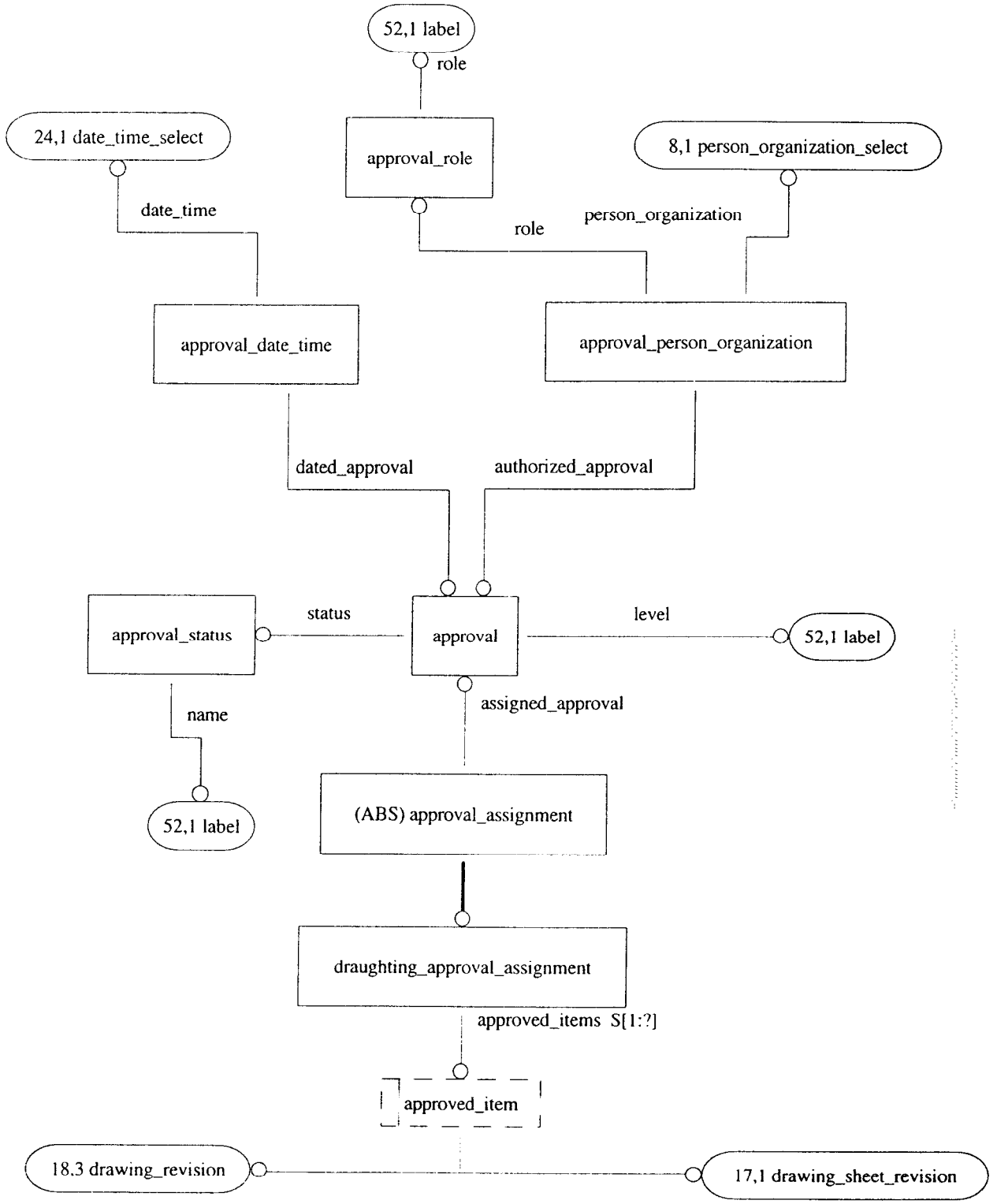

Figure H.5 – AIM EXPRESS-G diagram 5 of 52

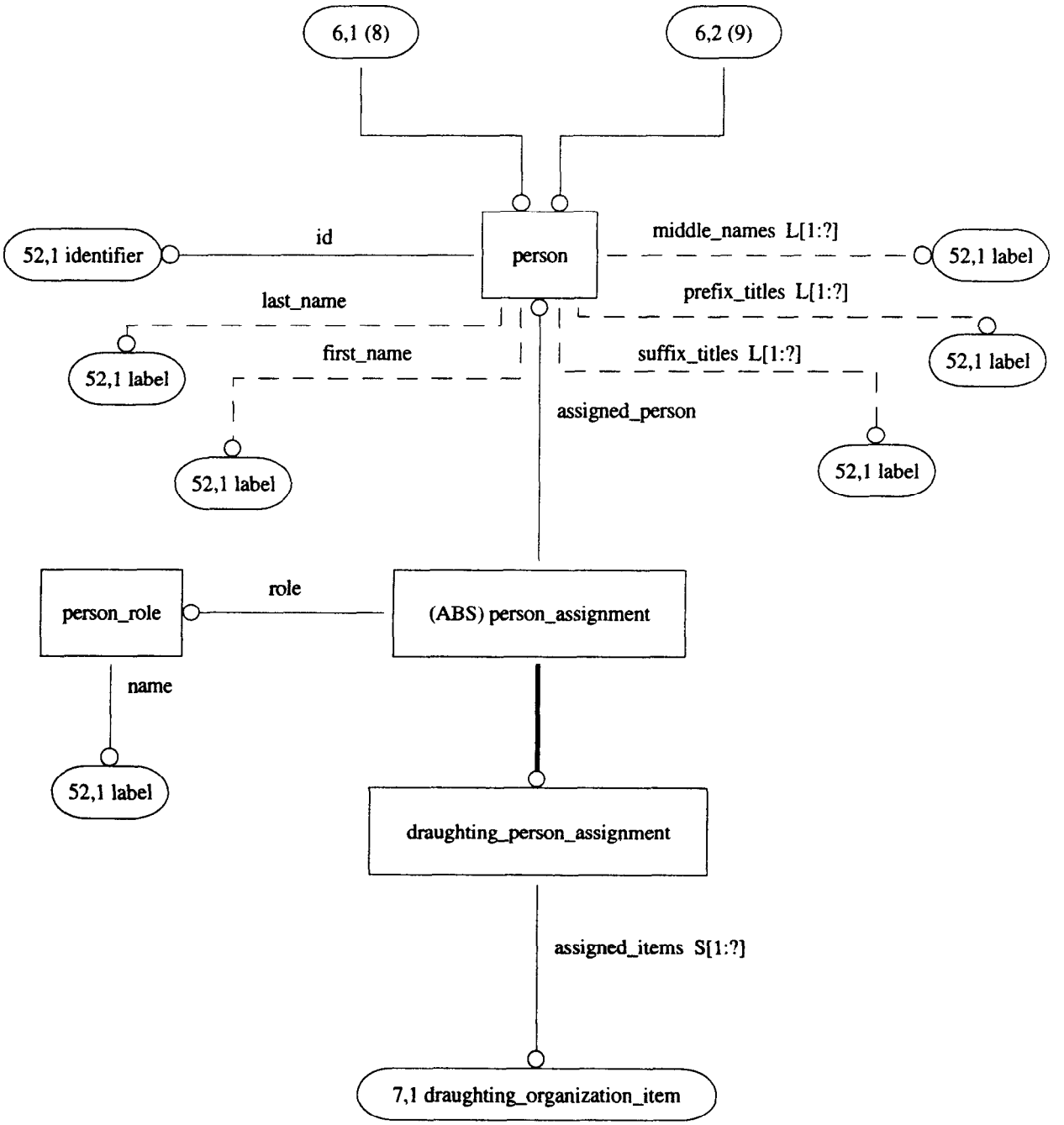

Figure H.6 – AIM EXPRESS-G diagram 6 of 52

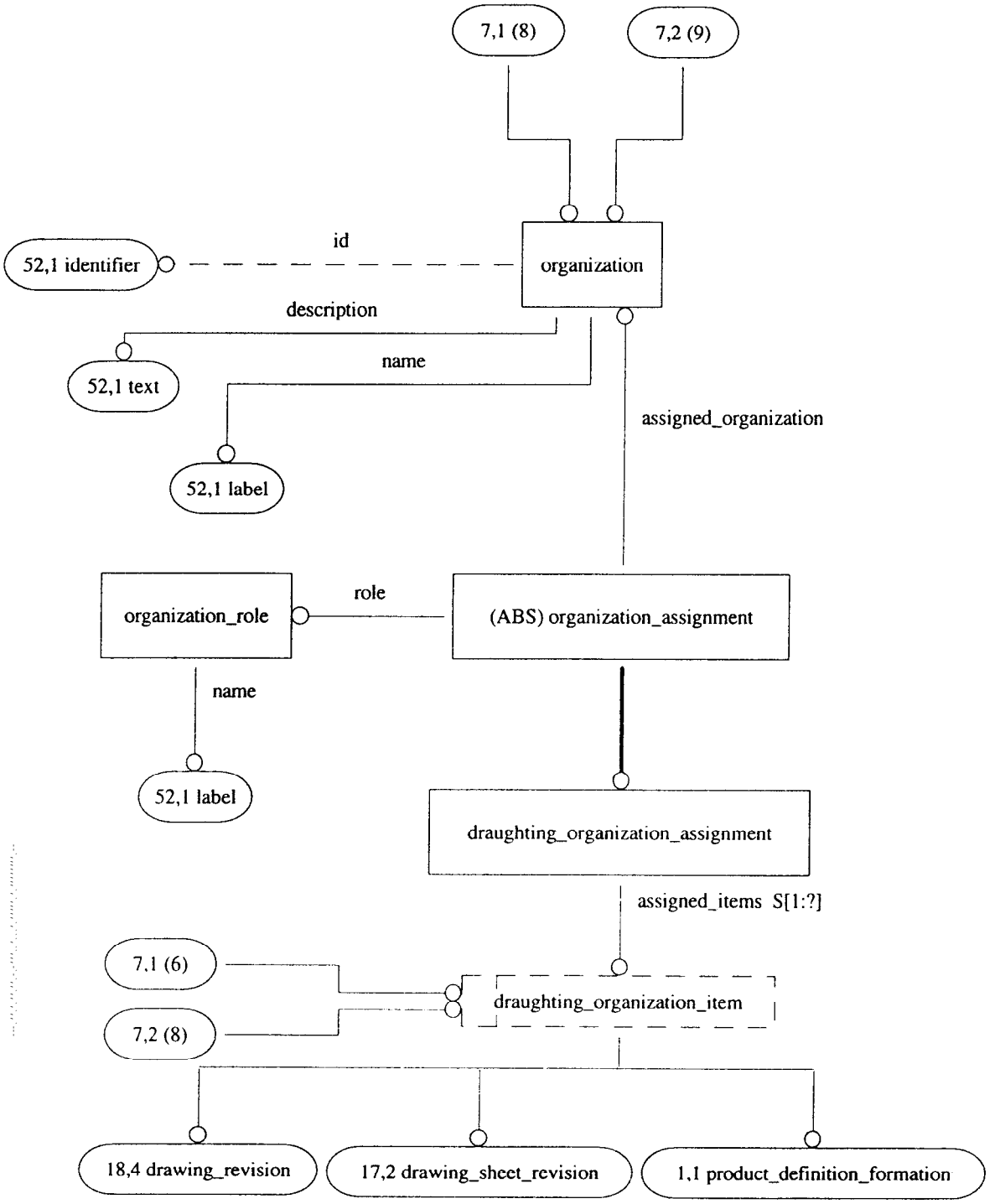

Figure H.7 – AIM EXPRESS-G diagram 7 of 52

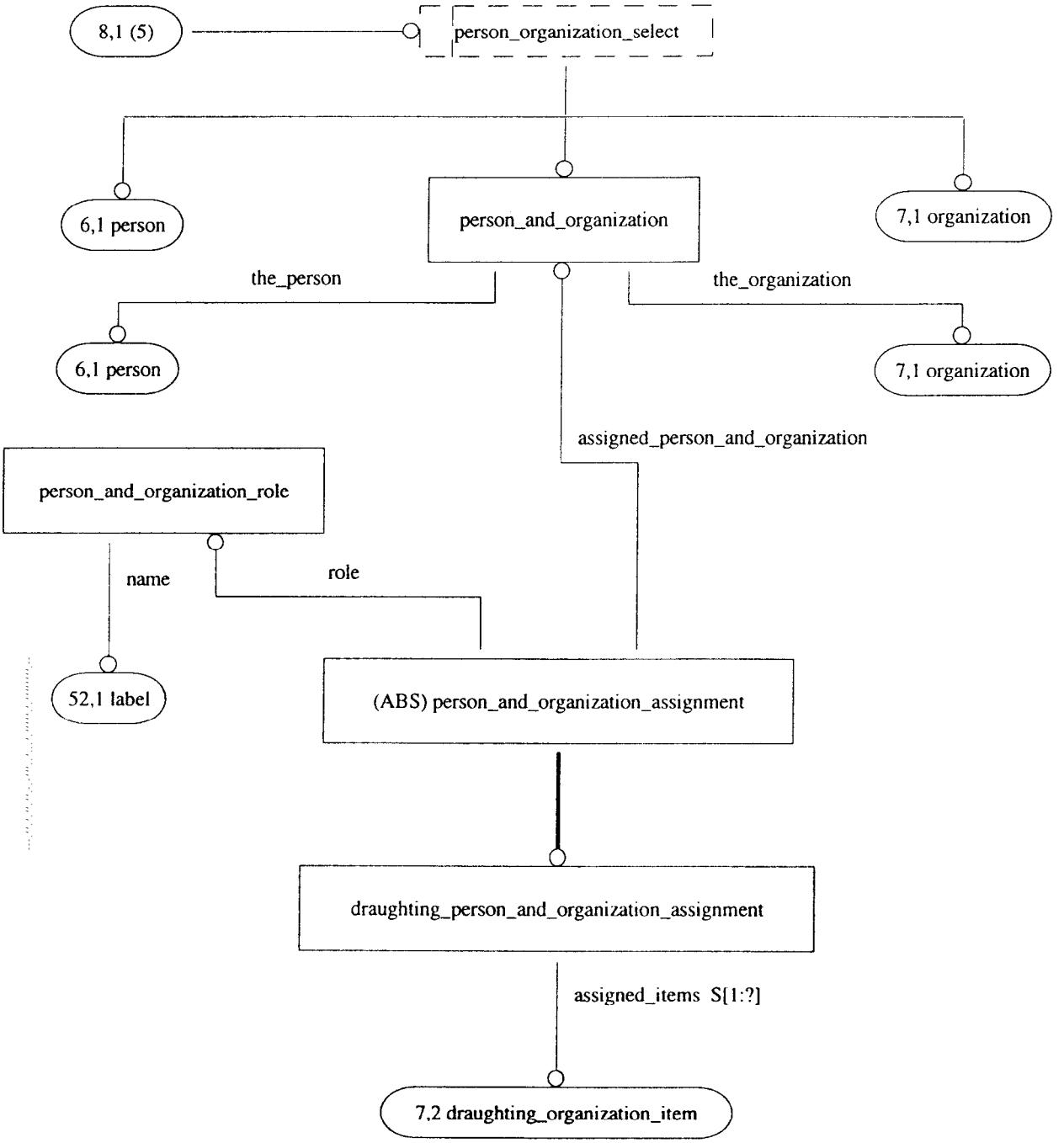

Figure H.8 – AIM EXPRESS-G diagram 8 of 52

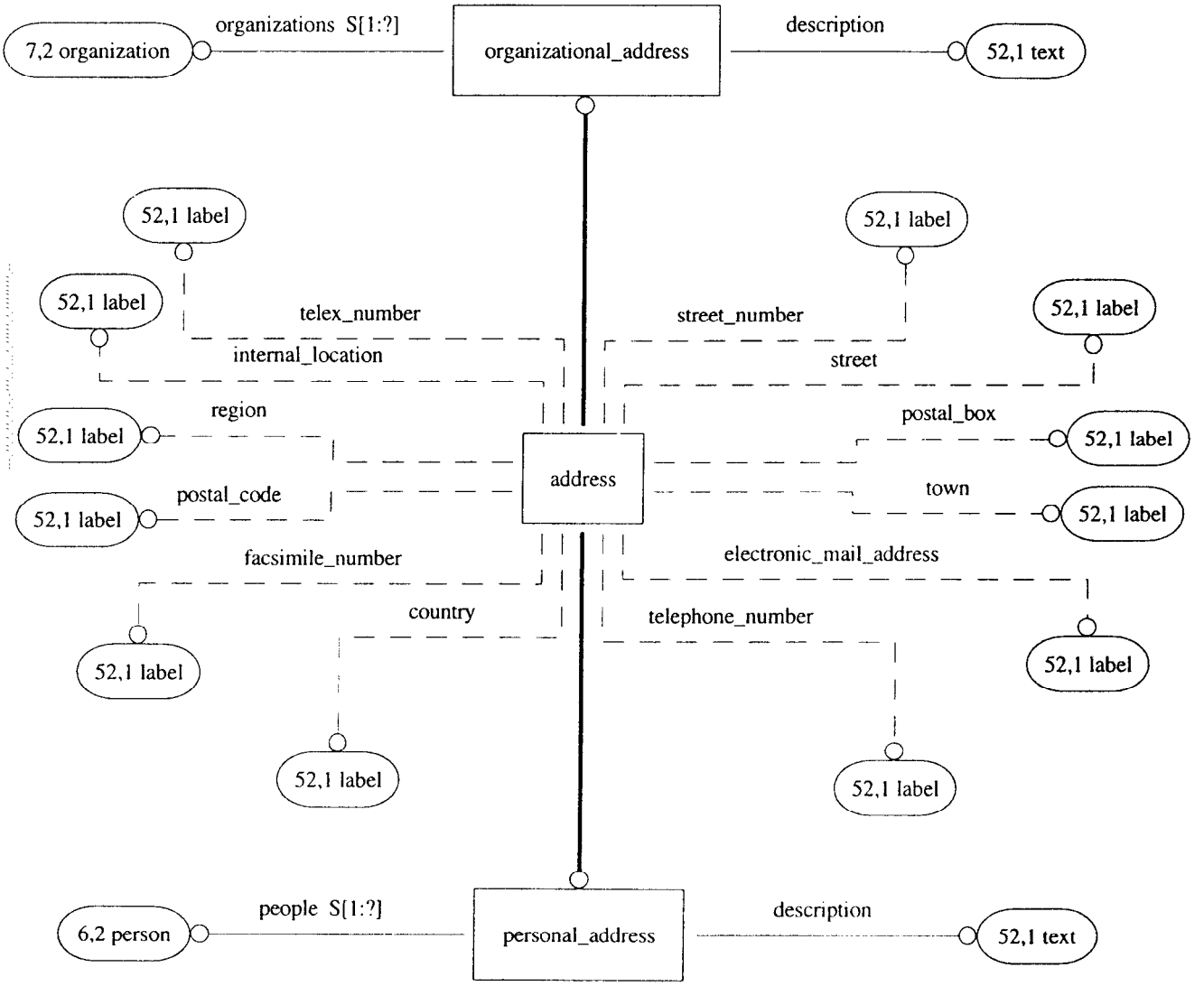

Figure H.9 – AIM EXPRESS-G diagram 9 of 52

Figure H.10 – AIM EXPRESS-G diagram 10 of 52

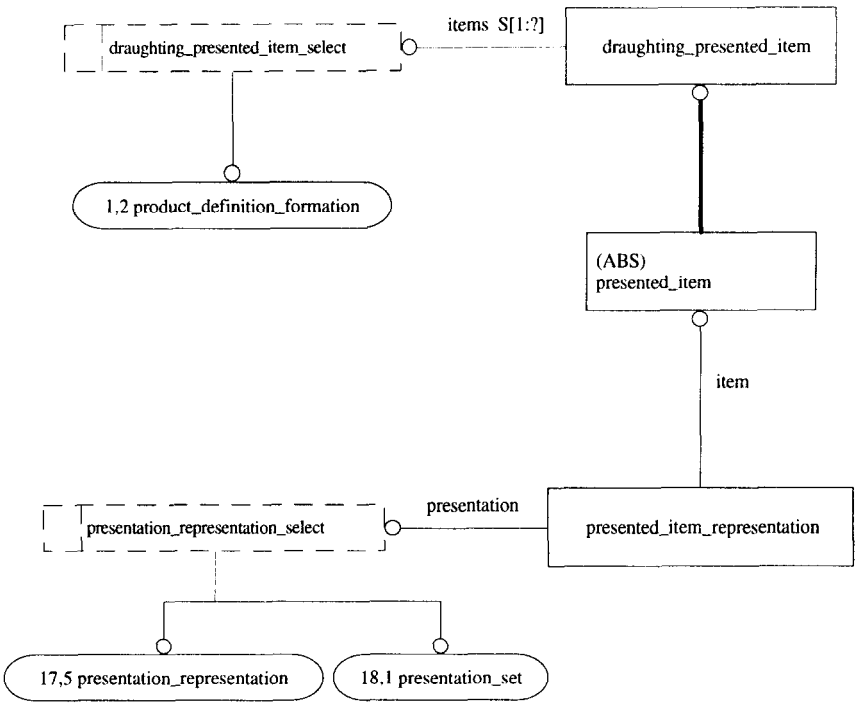

Figure H.11 – AIM EXPRESS-G diagram 11 of 52

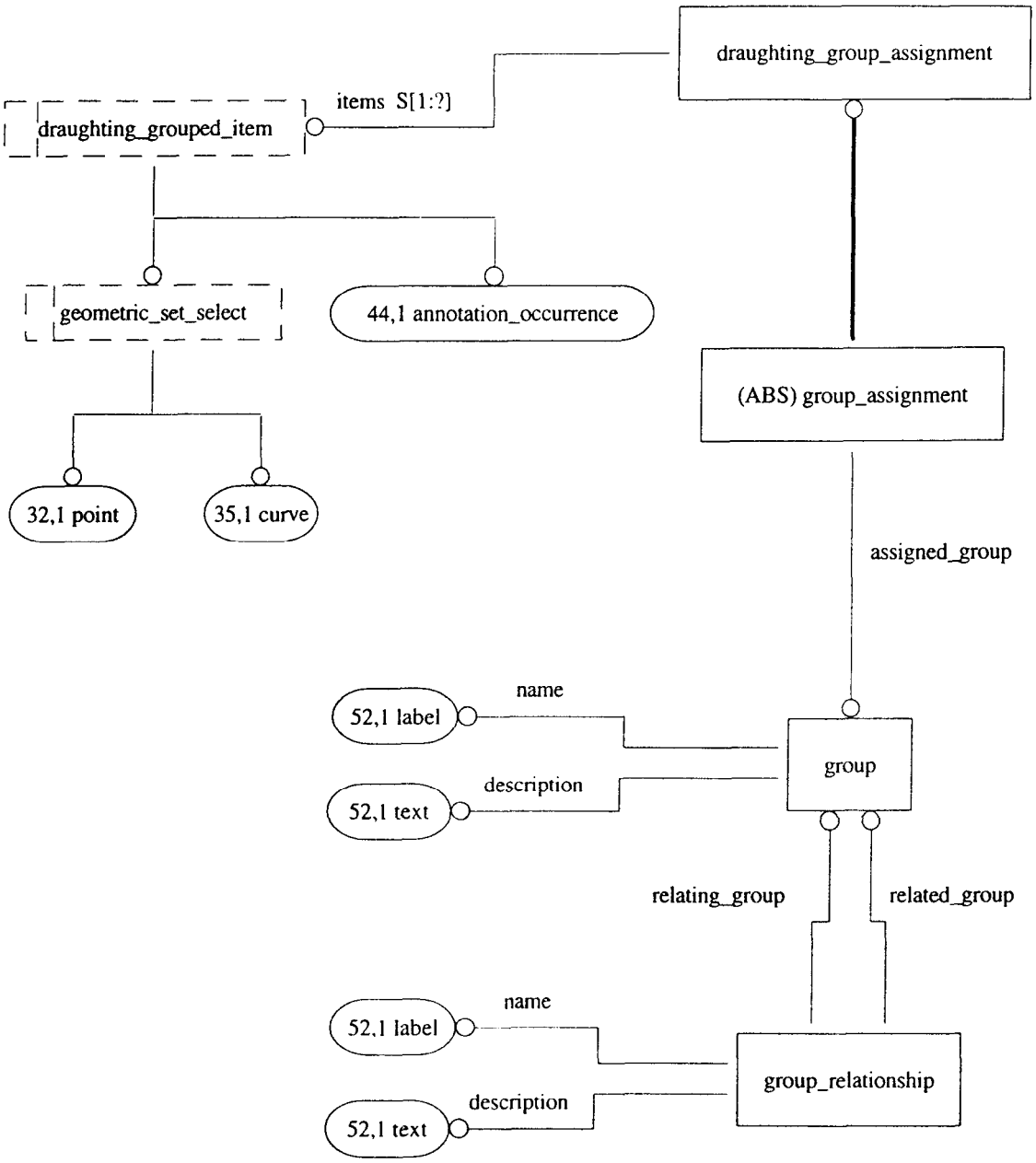

Figure H.12 – AIM EXPRESS-G diagram 12 of 52

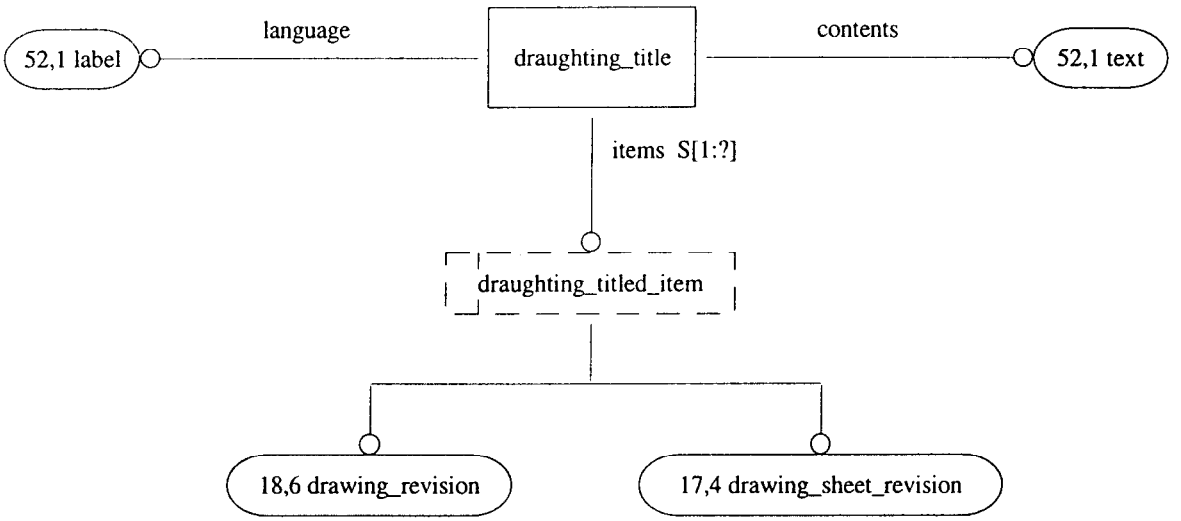

Figure H.13 – AIM EXPRESS-G diagram 13 of 52

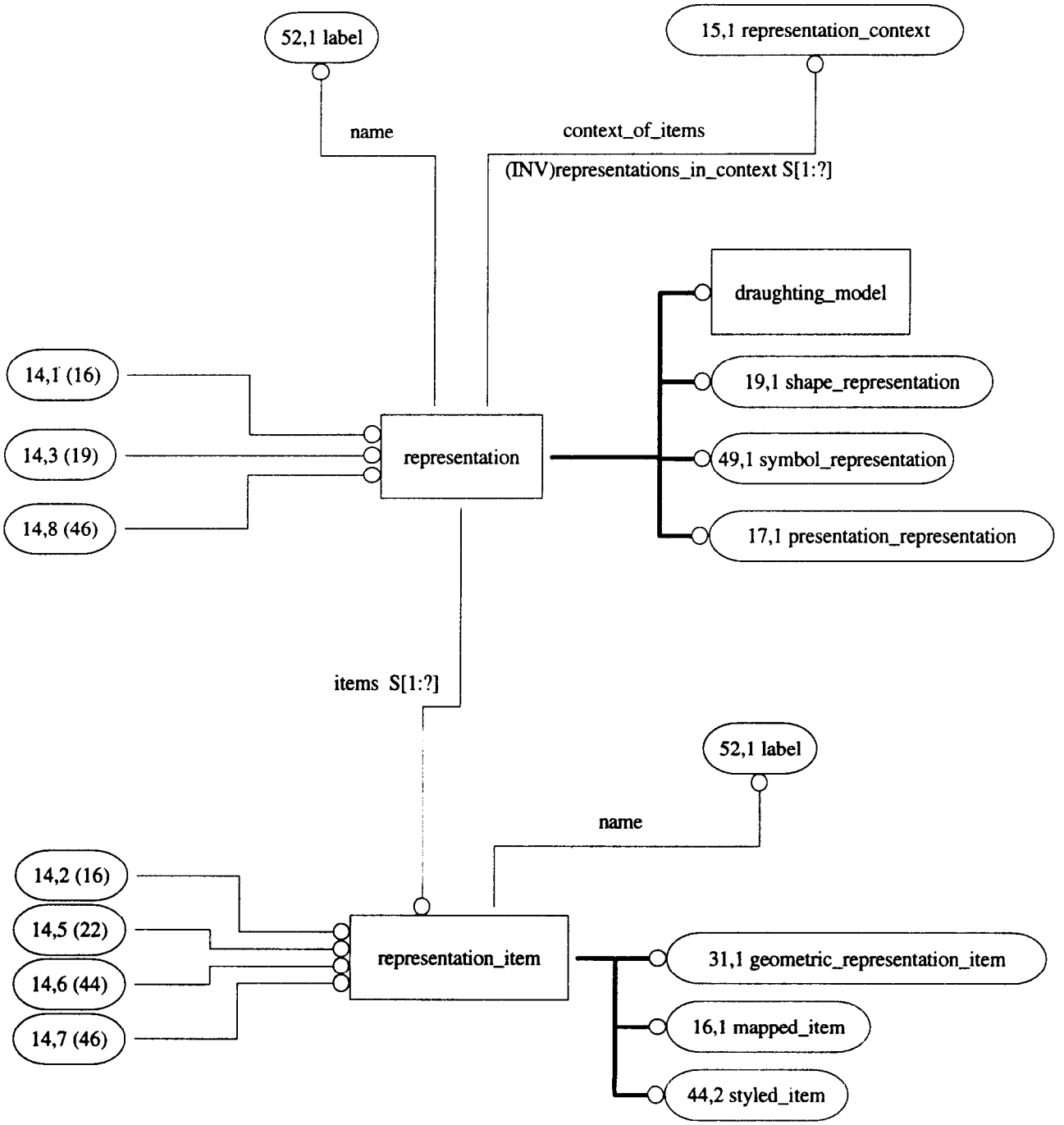

Figure H.14 – AIM EXPRESS-G diagram 14 of 52

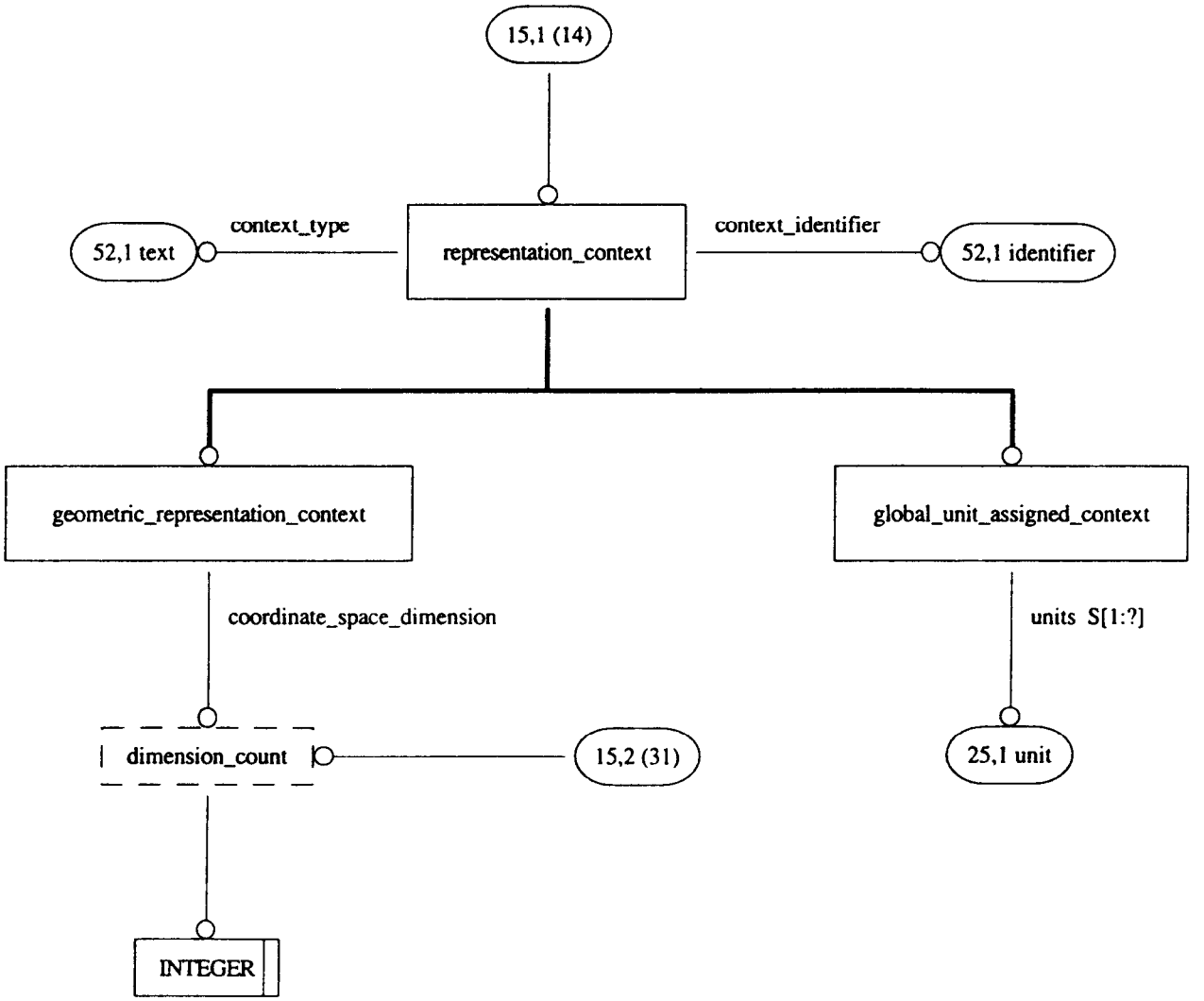

Figure H.15 – AIM EXPRESS-G diagram 15 of 52

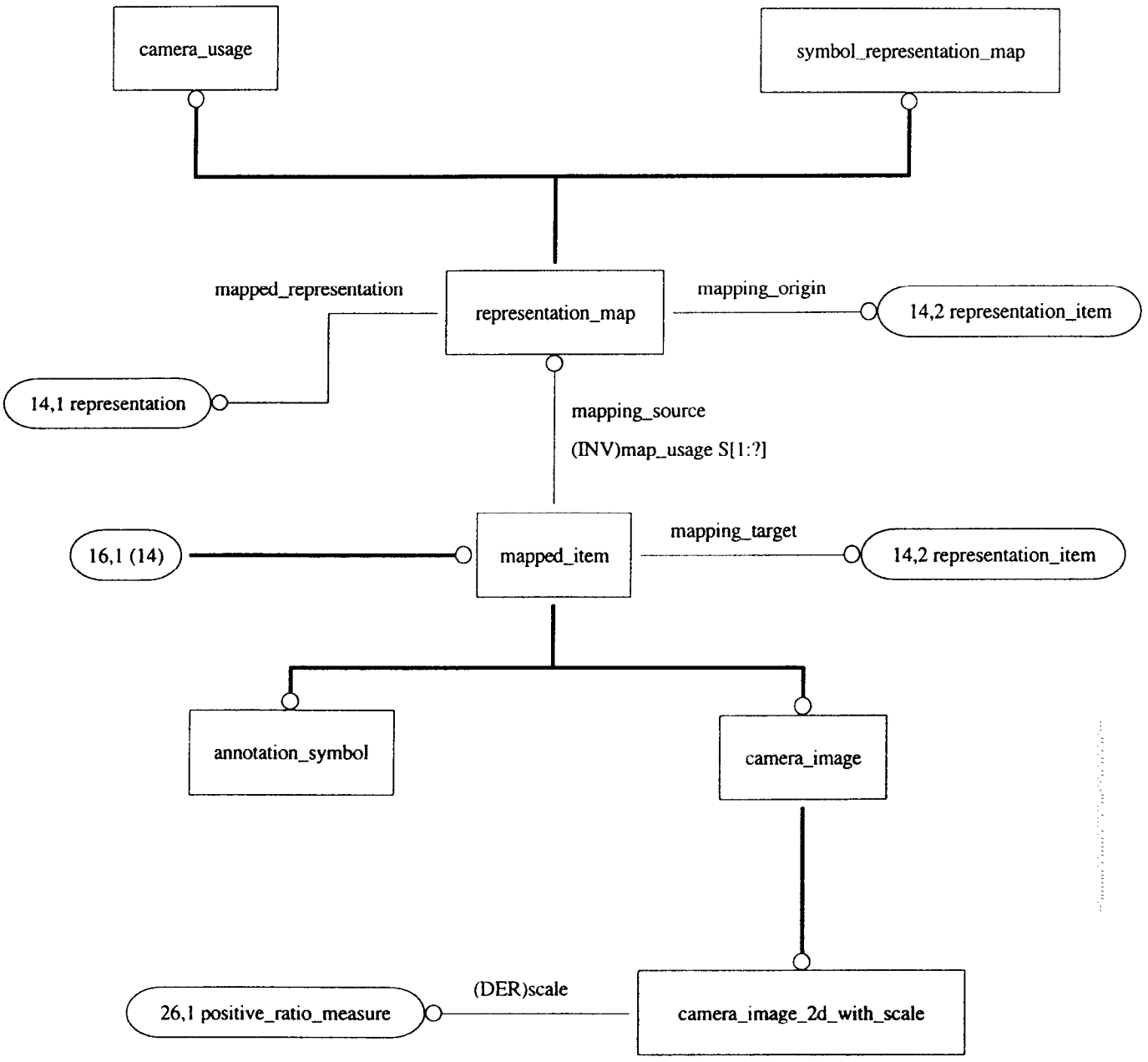

Figure H.16 – AIM EXPRESS-G diagram 16 of 52

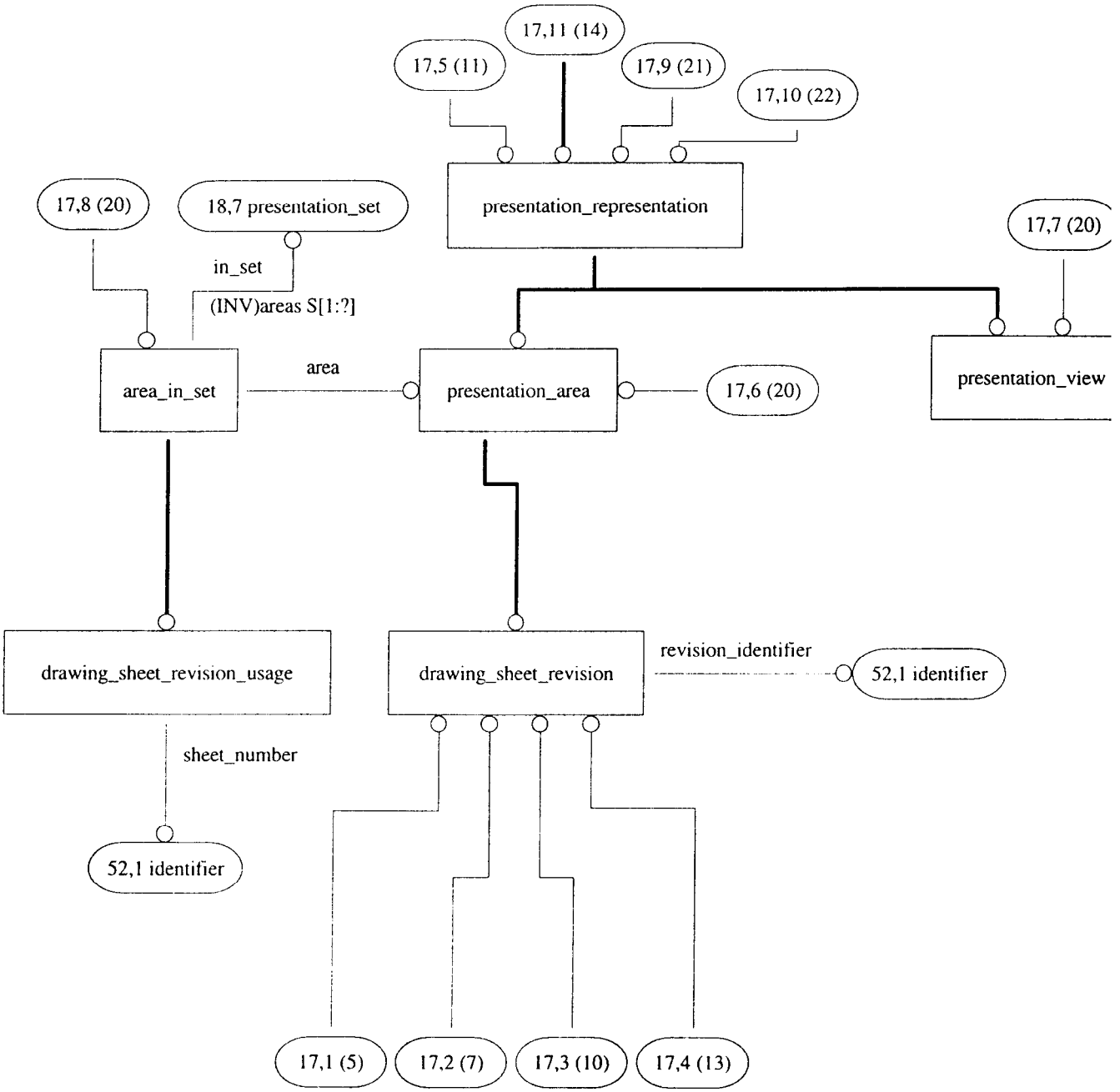

Figure H.17 – AIM EXPRESS-G diagram 17 of 52

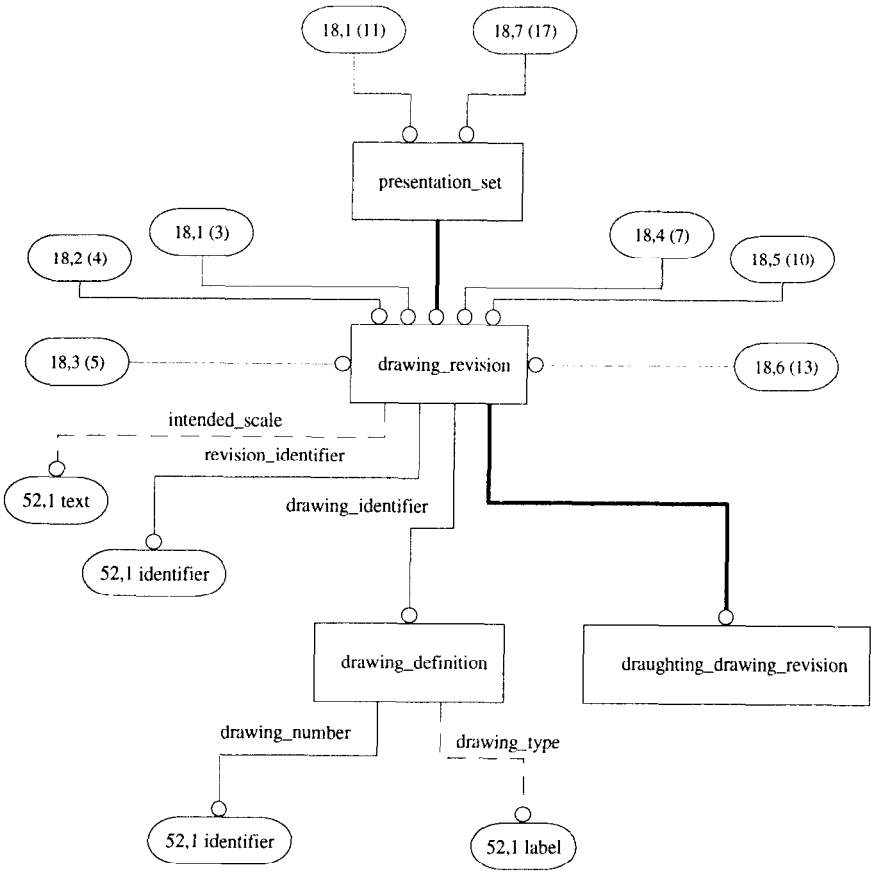

Figure H.18 – AIM EXPRESS-G diagram 18 of 52

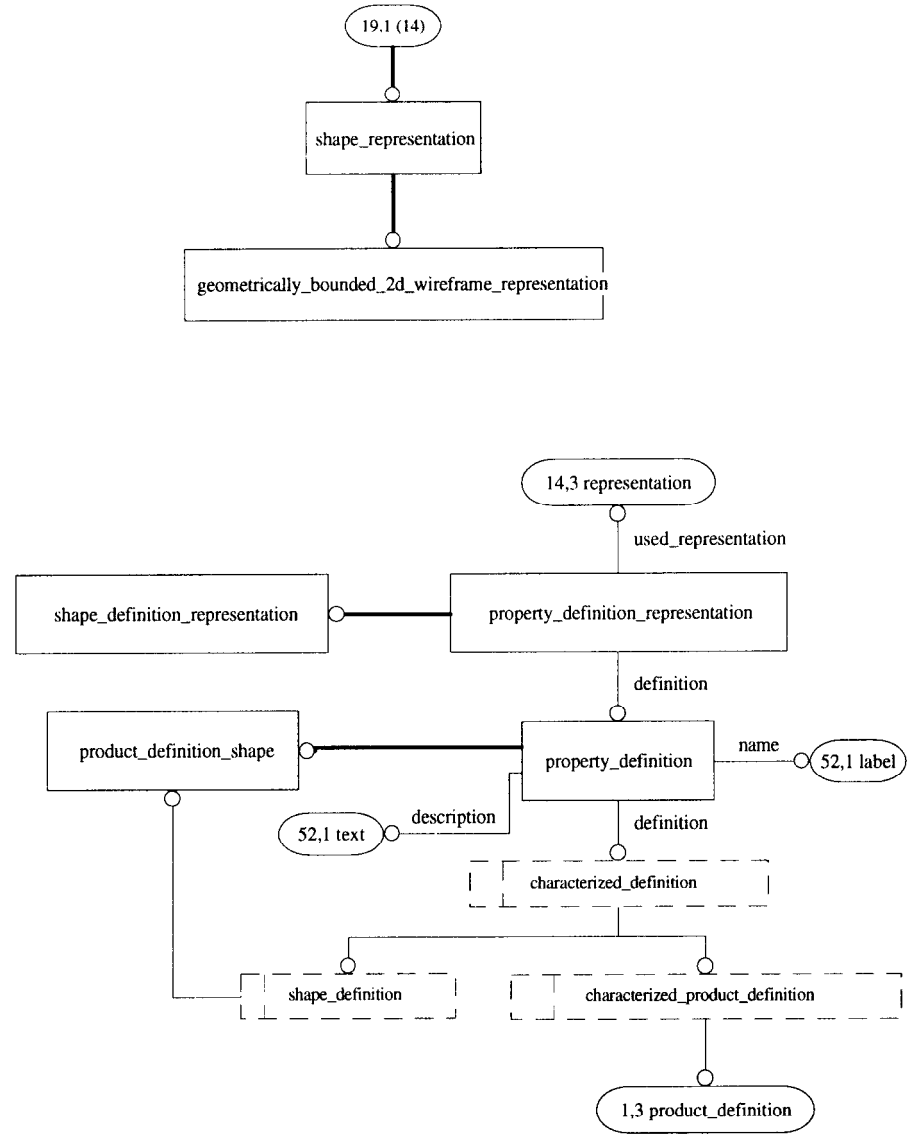

Figure H.19 – AIM EXPRESS-G diagram 19 of 52

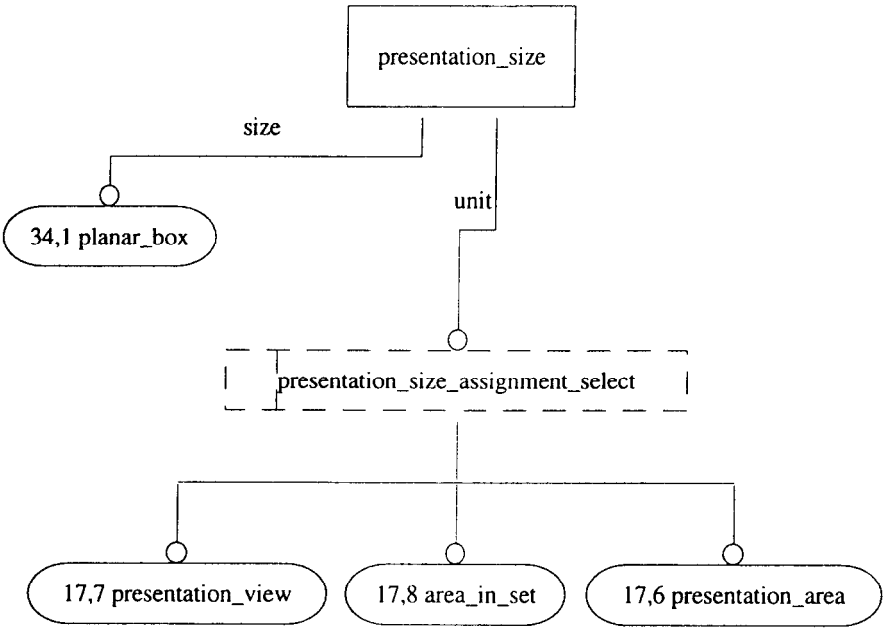

Figure H.20 – AIM EXPRESS-G diagram 20 of 52

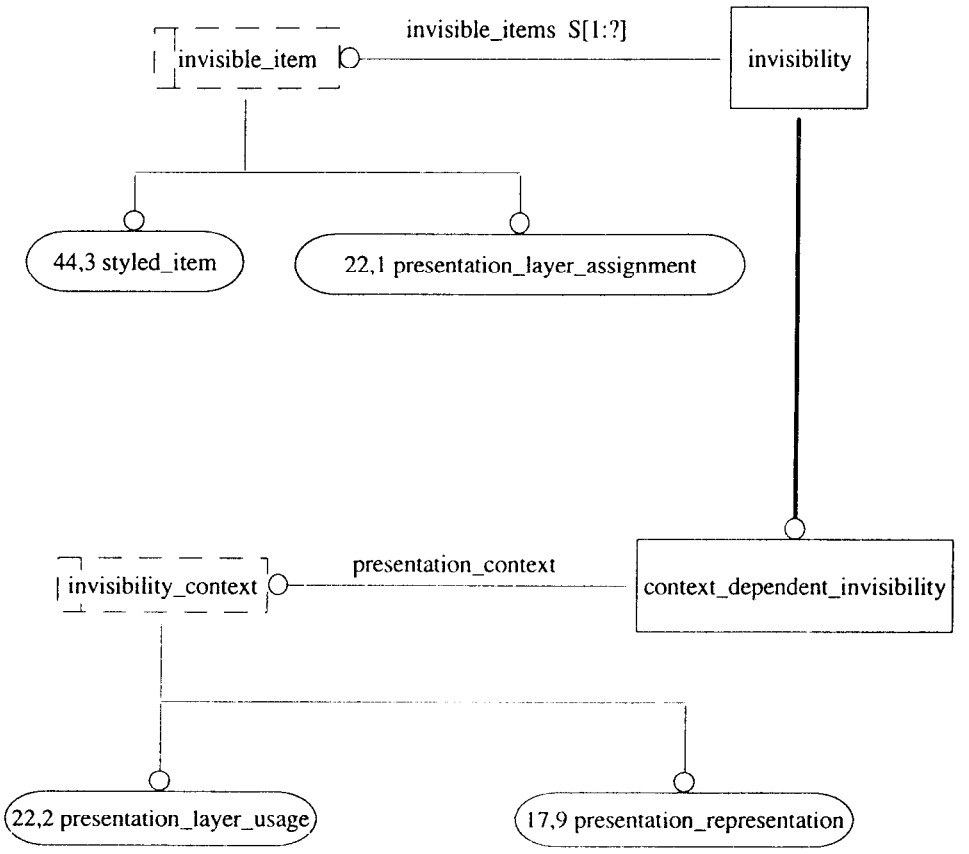

Figure H.21 – AIM EXPRESS-G diagram 21 of 52

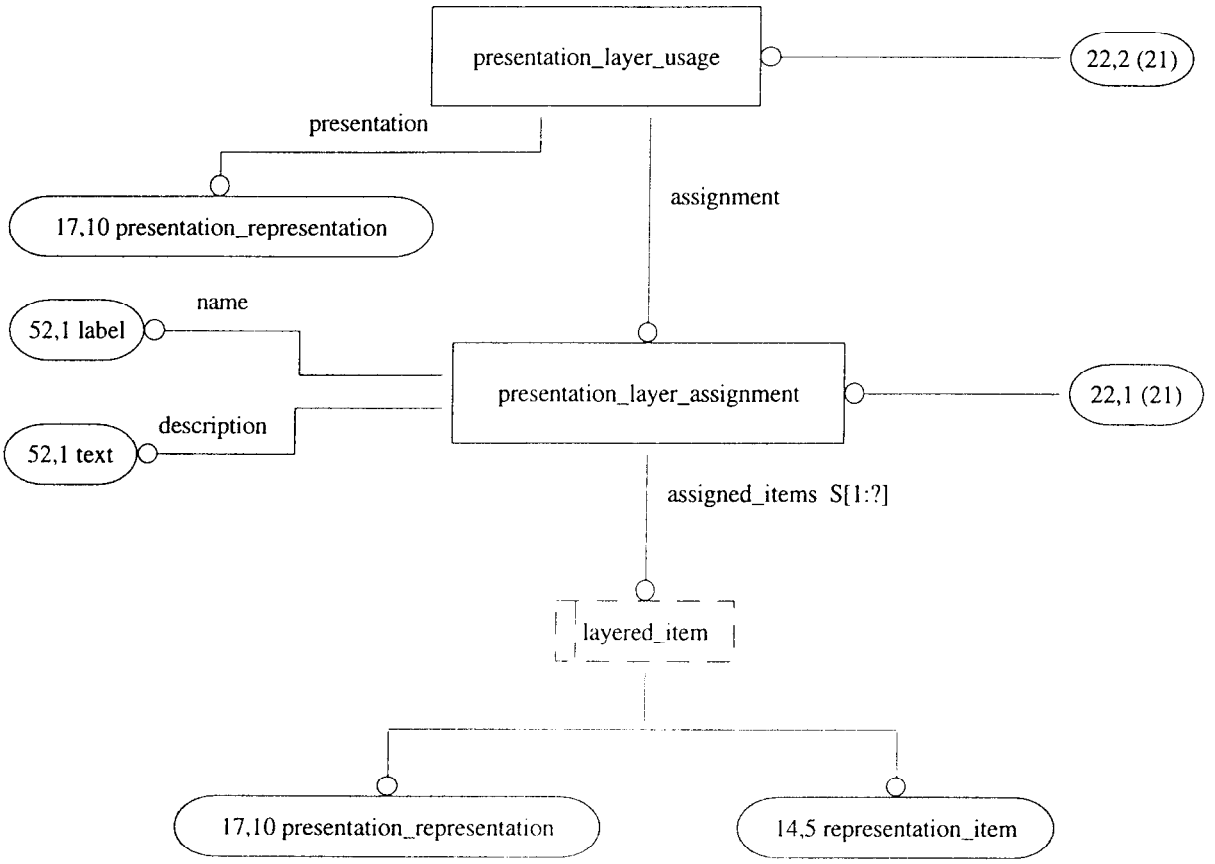

Figure H.22 – AIM EXPRESS-G diagram 22 of 52

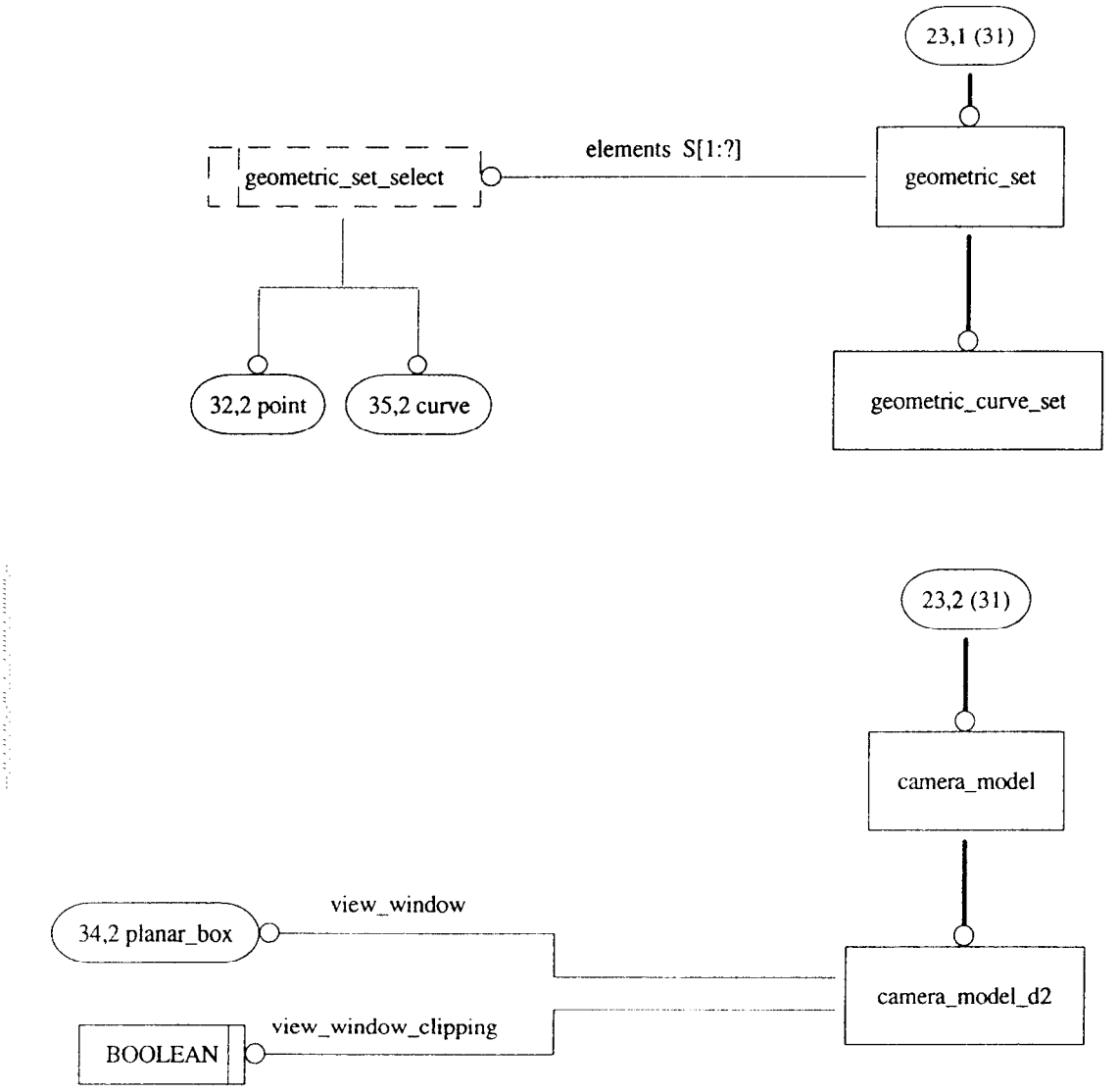

Figure H.23 – AIM EXPRESS-G diagram 23 of 52

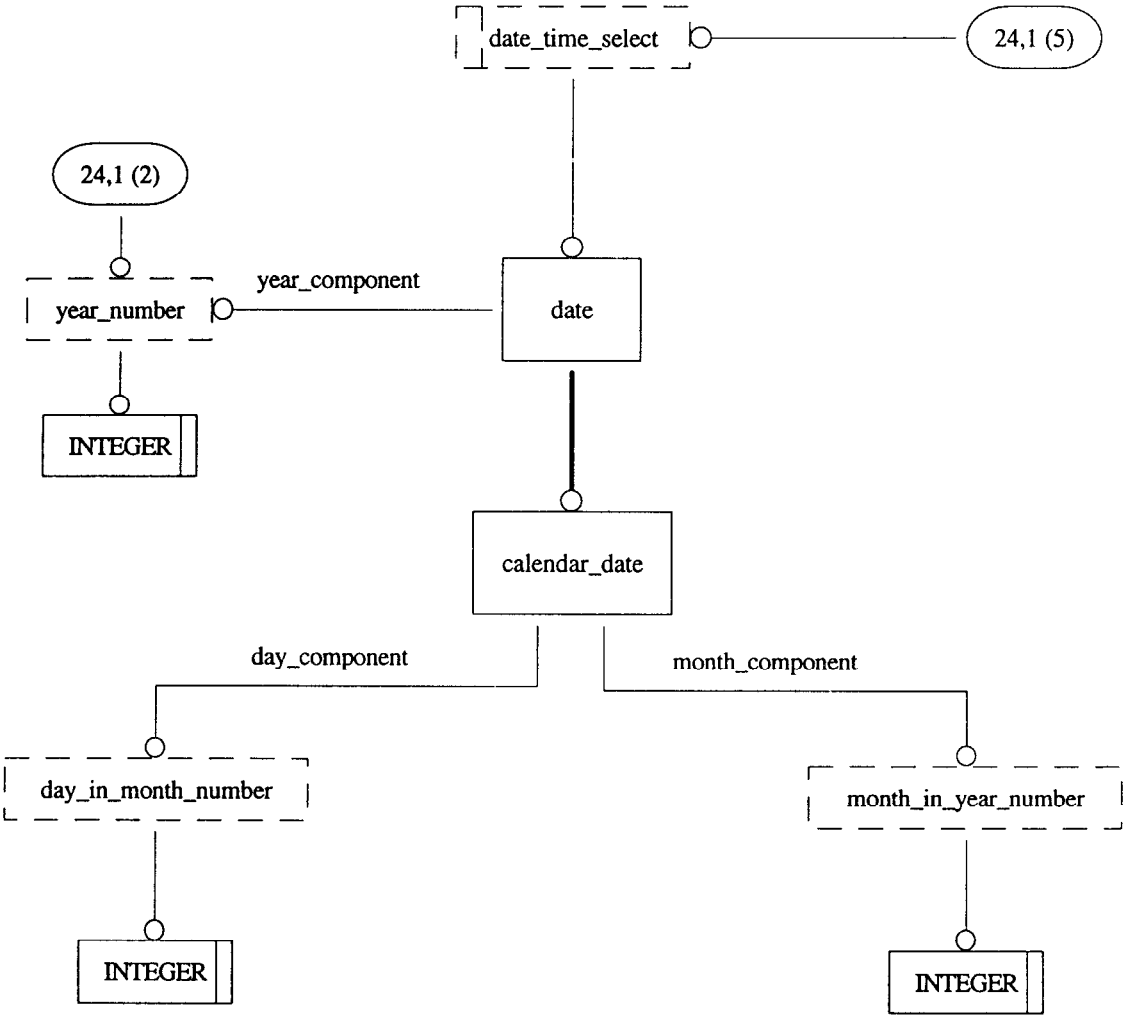

Figure H.24 – AIM EXPRESS-G diagram 24 of 52

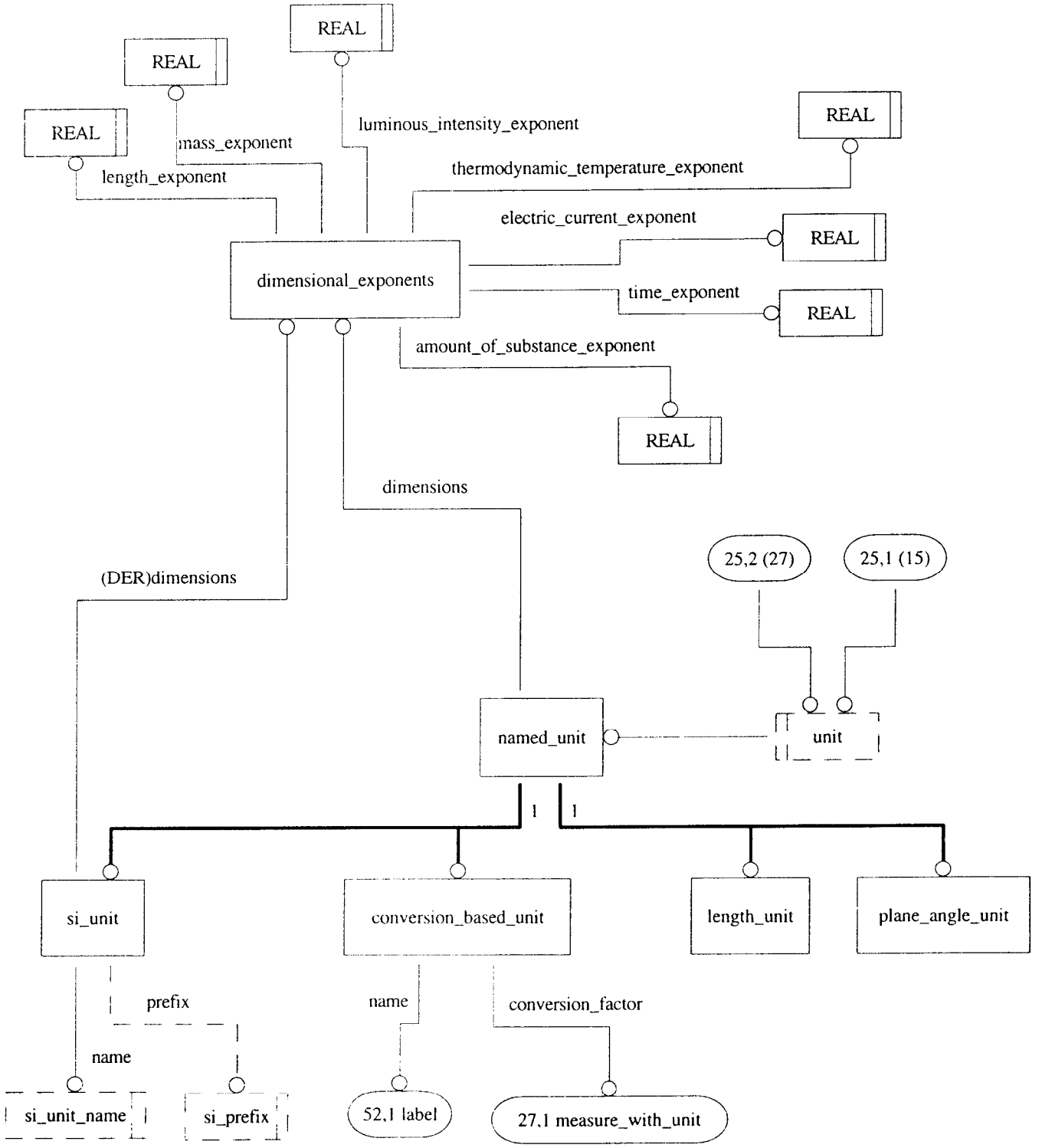

Figure H.25 – AIM EXPRESS-G diagram 25 of 52

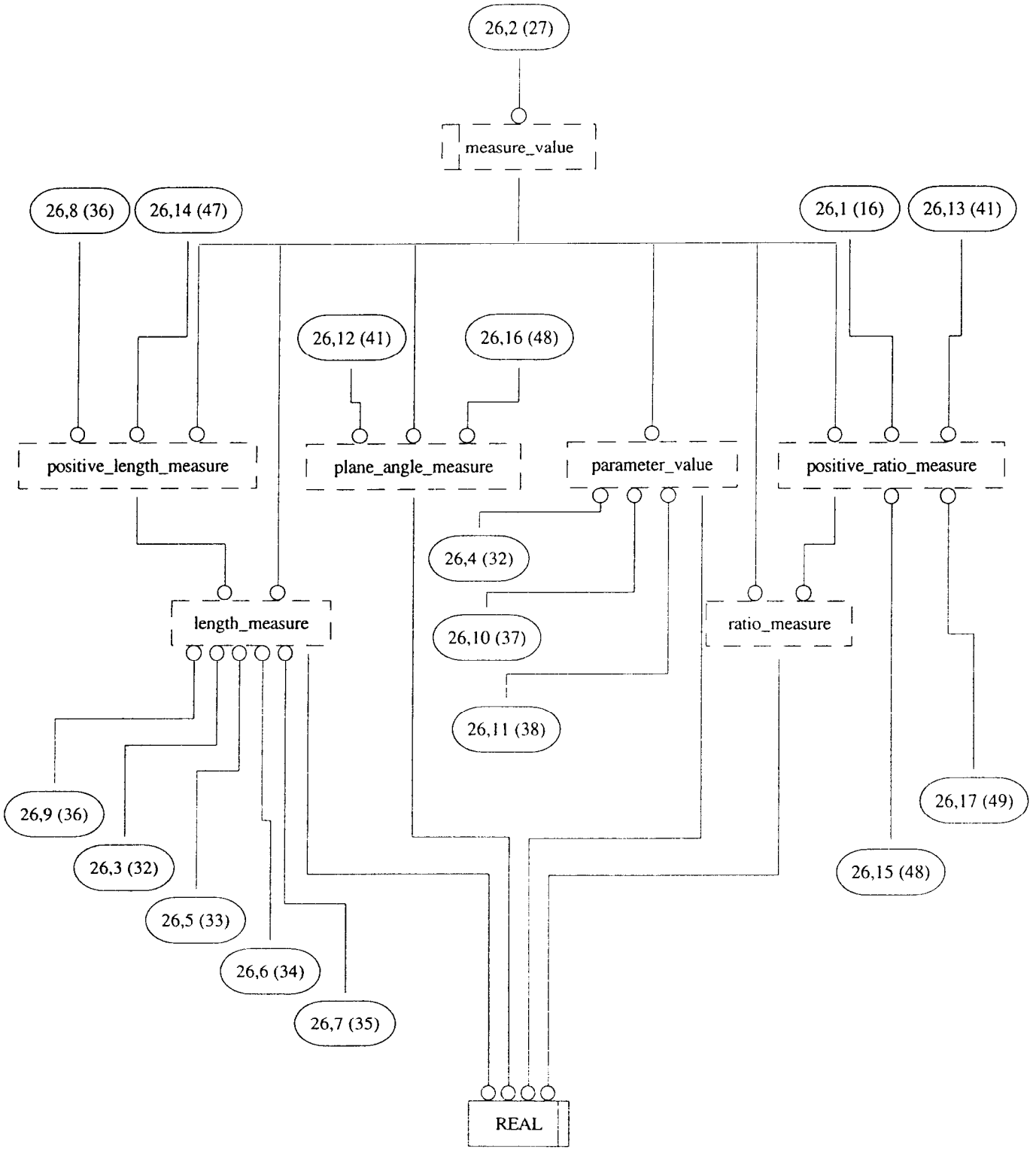

Figure H.26 – AIM EXPRESS-G diagram 26 of 52

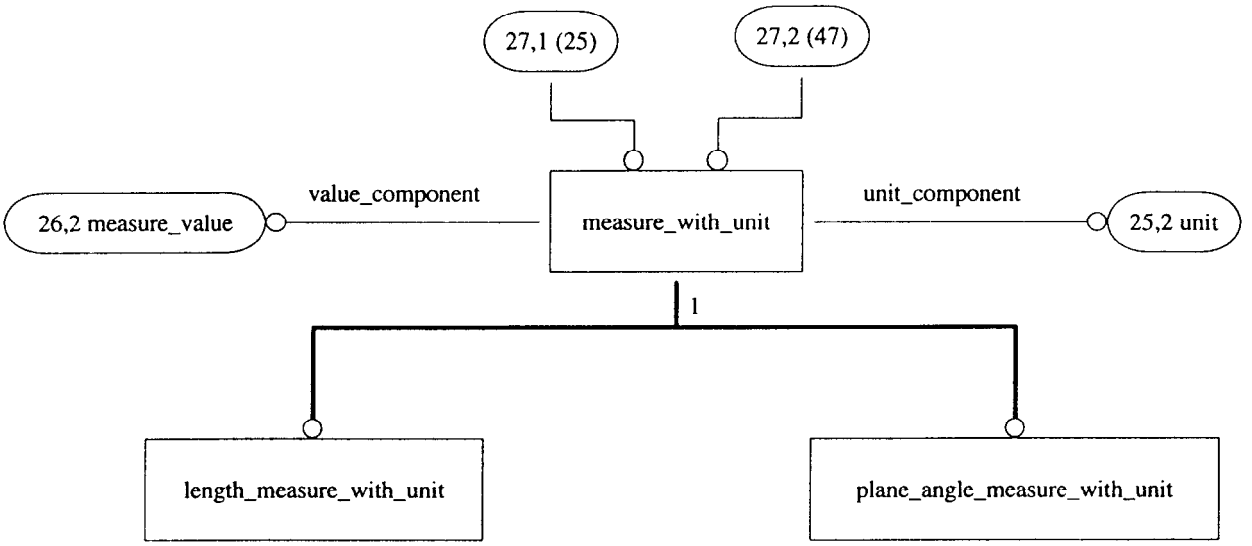

Figure H.27 – AIM EXPRESS-G diagram 27 of 52

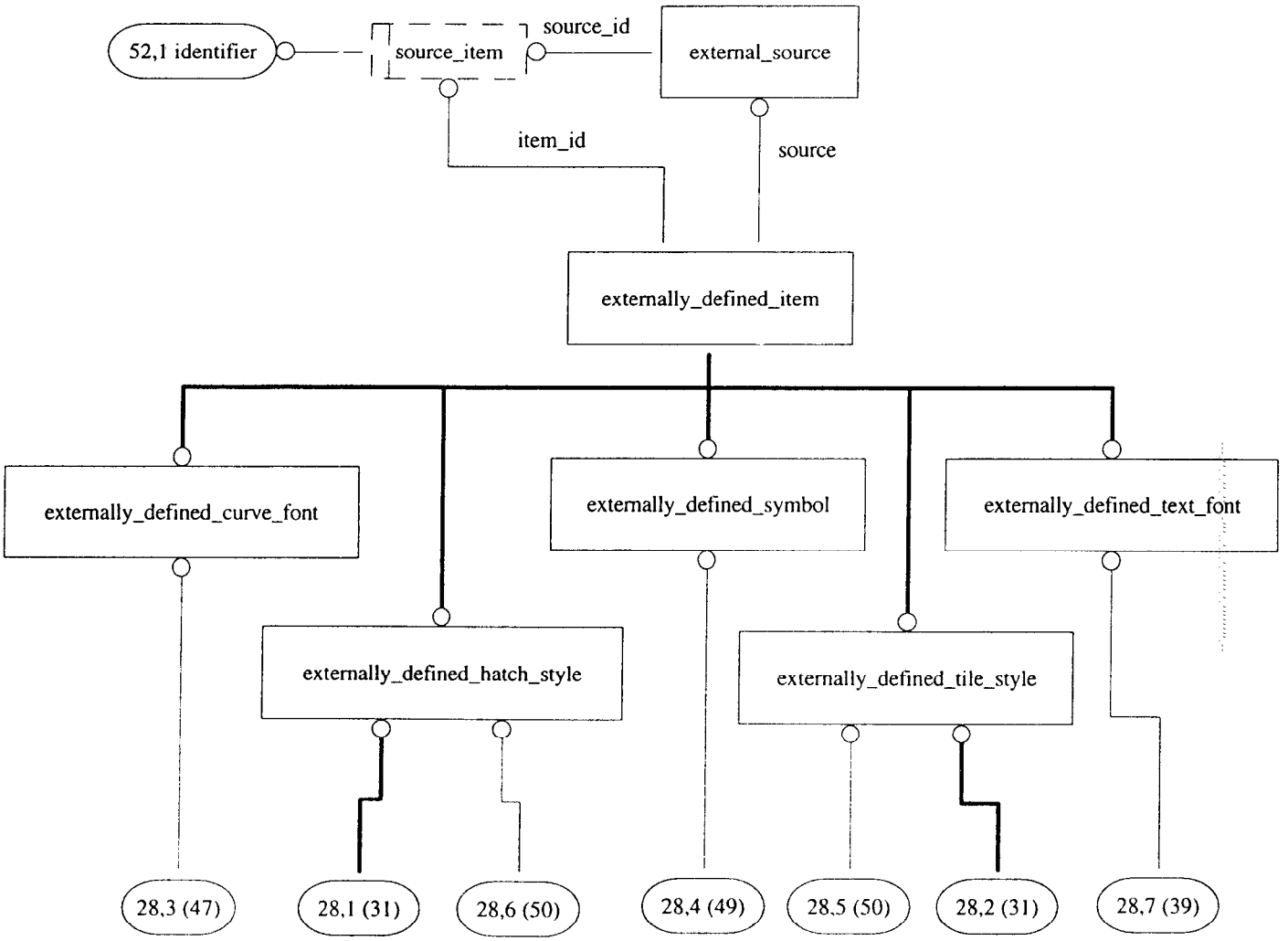

Figure H.28 – AIM EXPRESS-G diagram 28 of 52

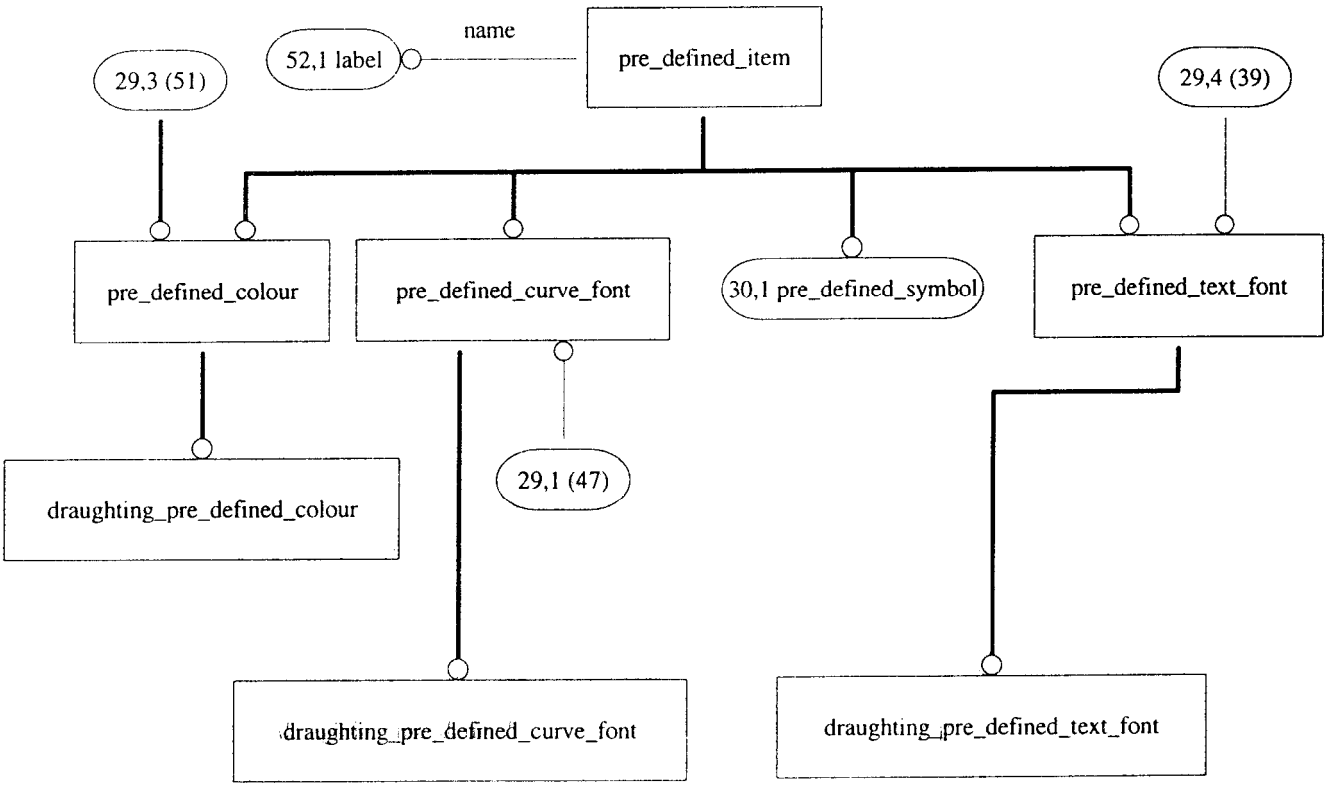

Figure H.29 – AIM EXPRESS-G diagram 29 of 52

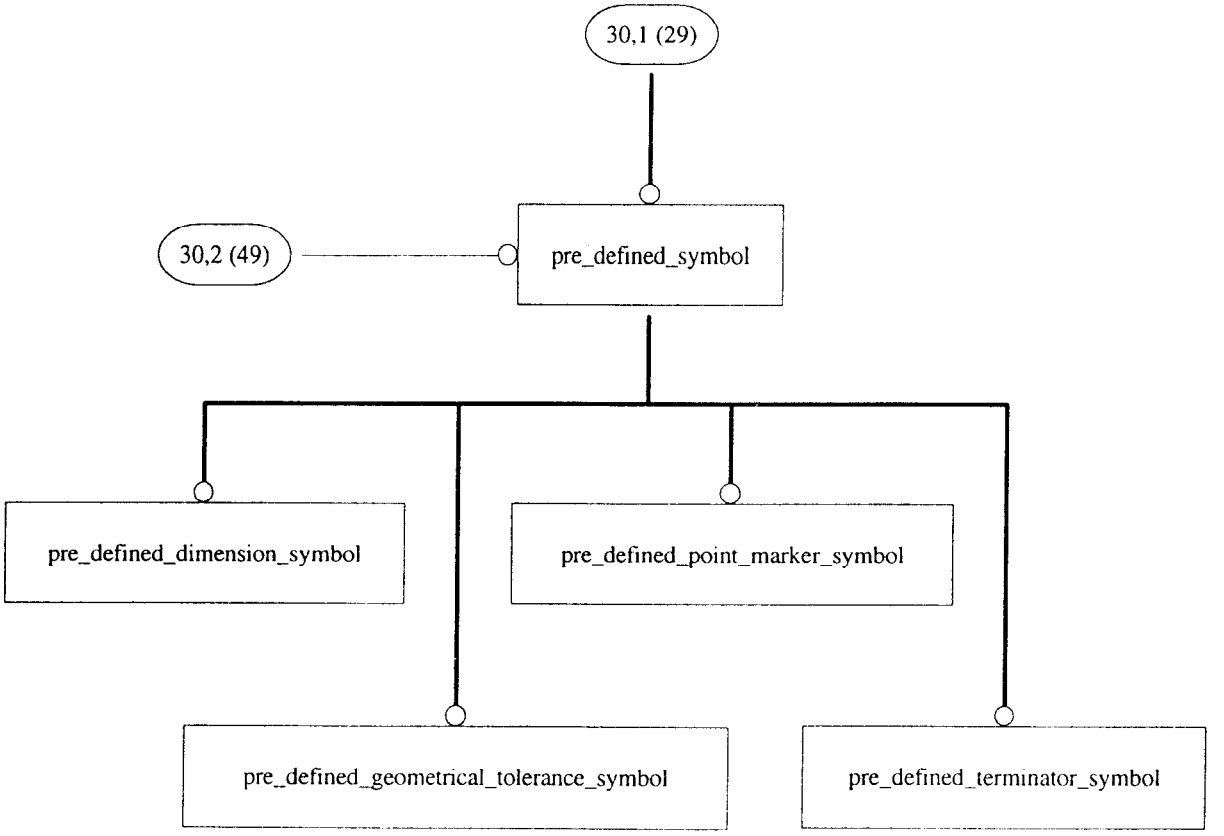

Figure H.30 – AIM EXPRESS-G diagram 30 of 52

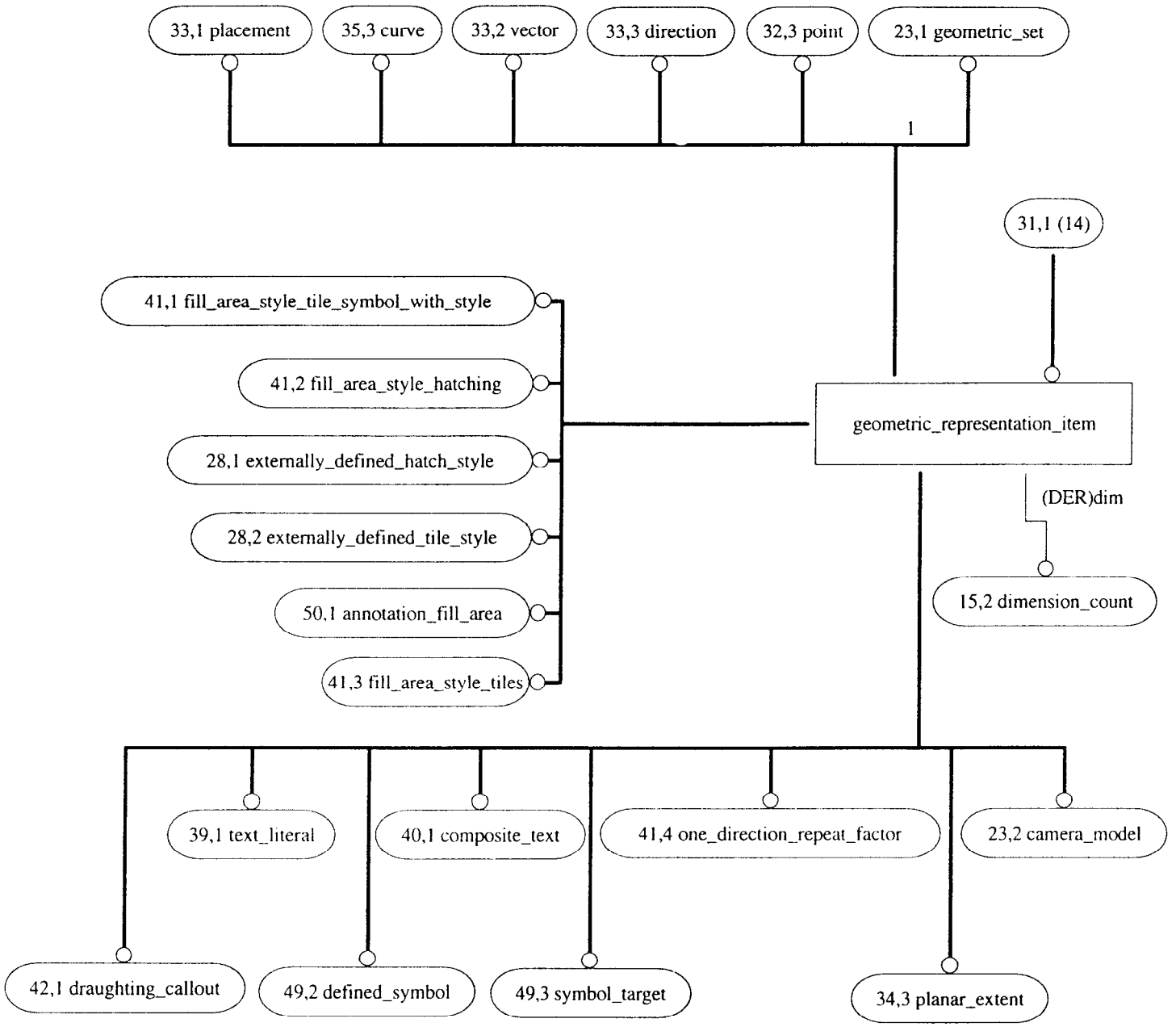

Figure H.31 – AIM EXPRESS-G diagram 31 of 52

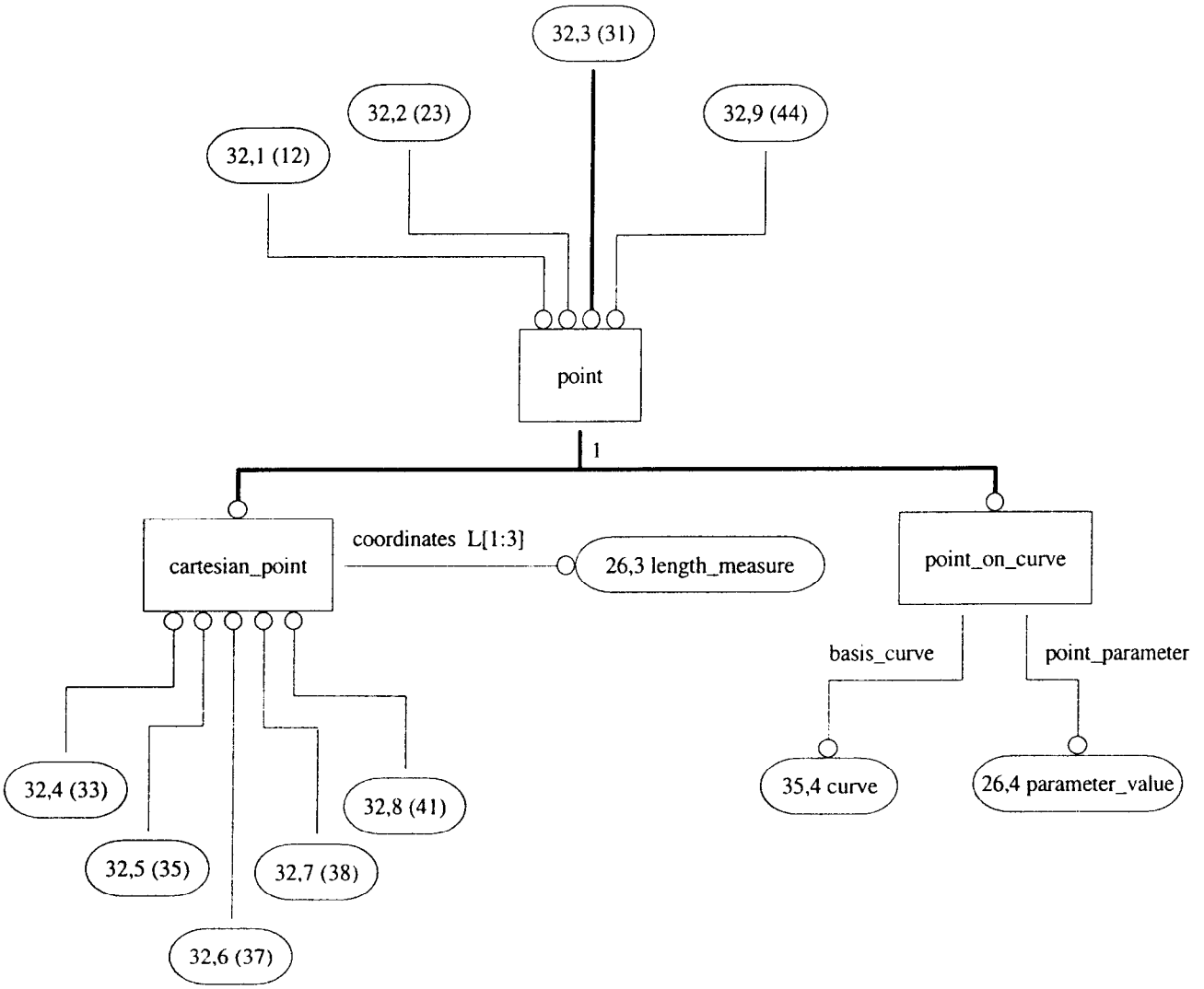

Figure H.32 – AIM EXPRESS-G diagram 32 of 52

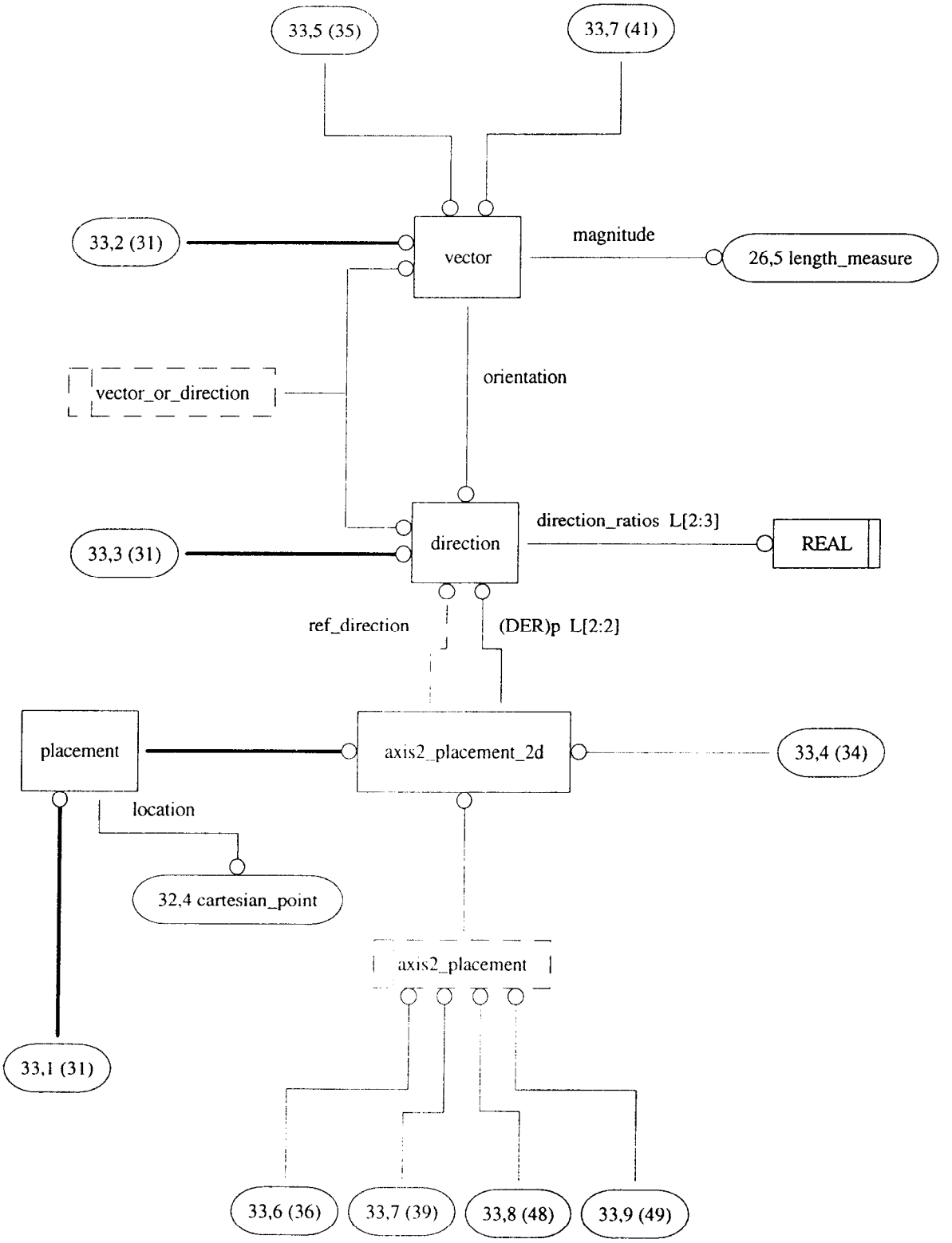

Figure H.33 – AIM EXPRESS-G diagram 33 of 52

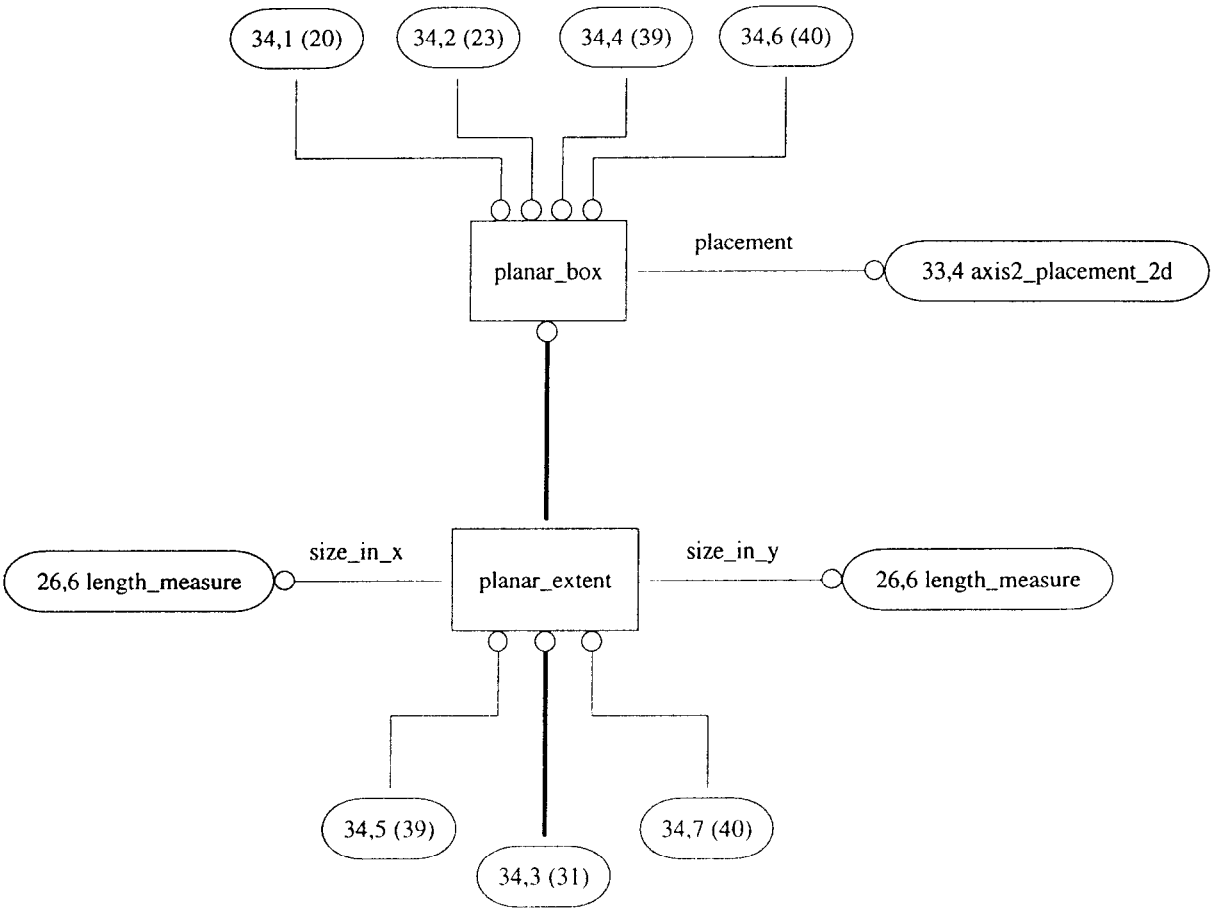

Figure H.34 – AIM EXPRESS-G diagram 34 of 52

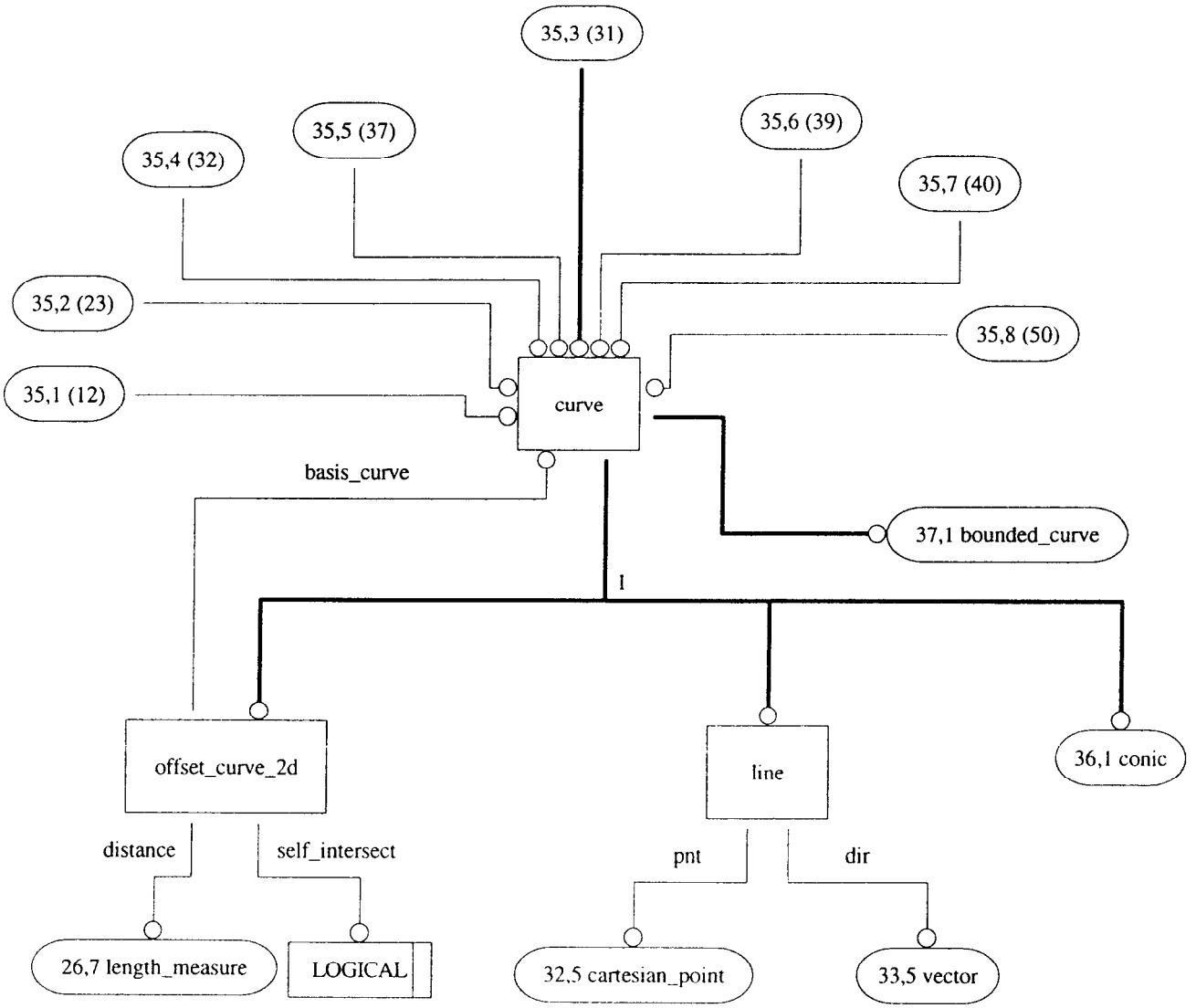

Figure H.35 – AIM EXPRESS-G diagram 35 of 52

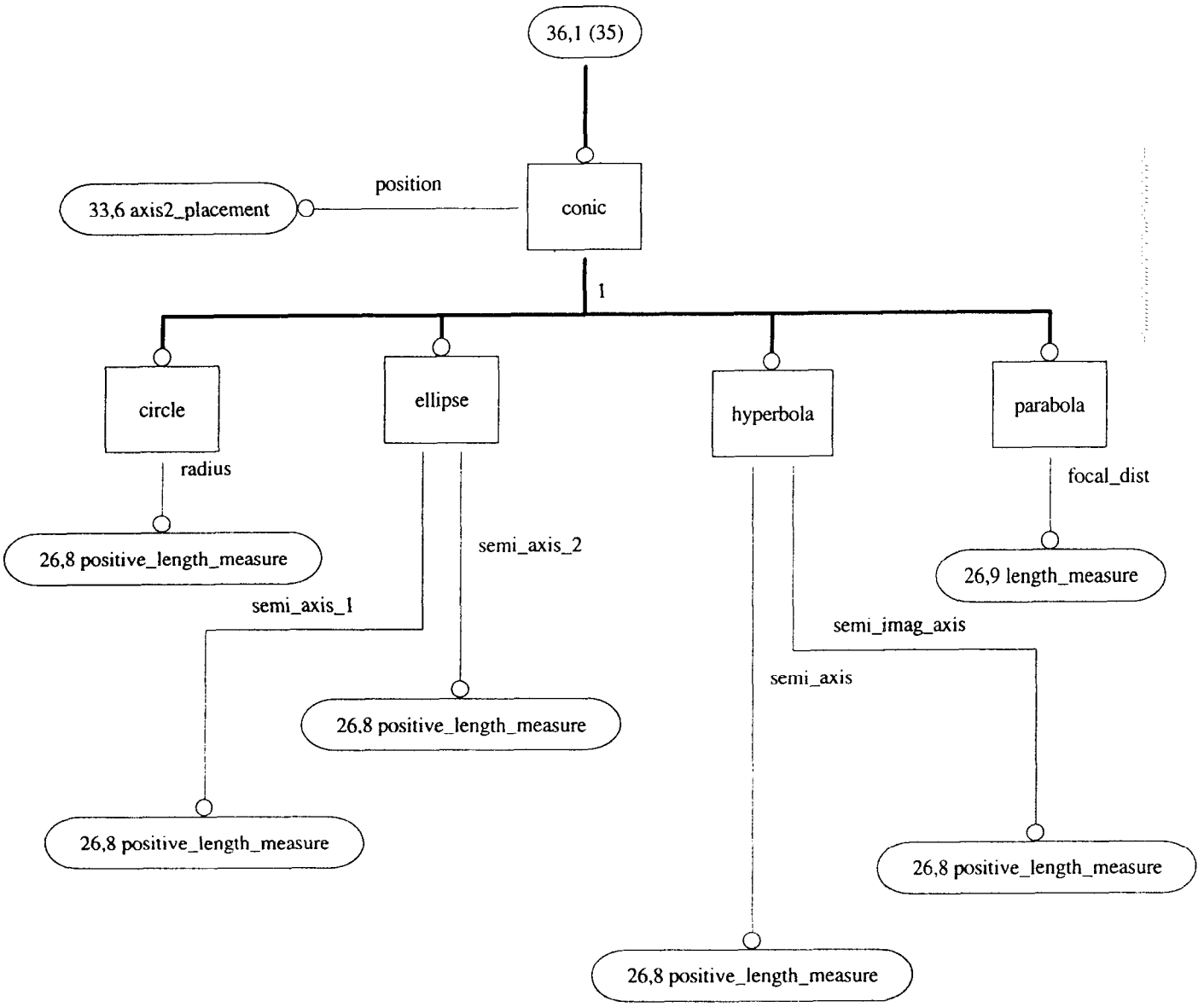

Figure H.36 – AIM EXPRESS-G diagram 36 of 52

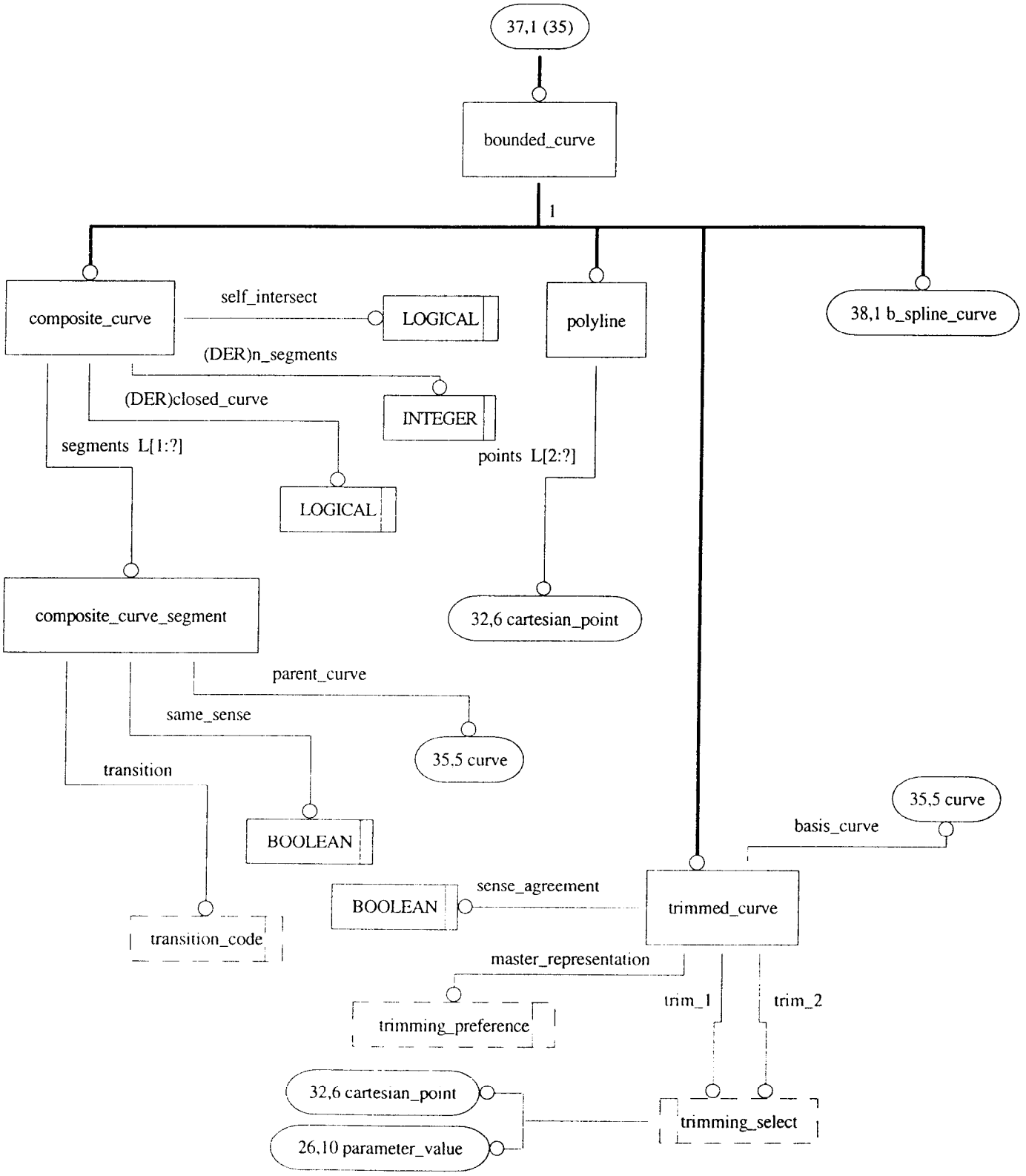

Figure H.37 – AIM EXPRESS-G diagram 37 of 52

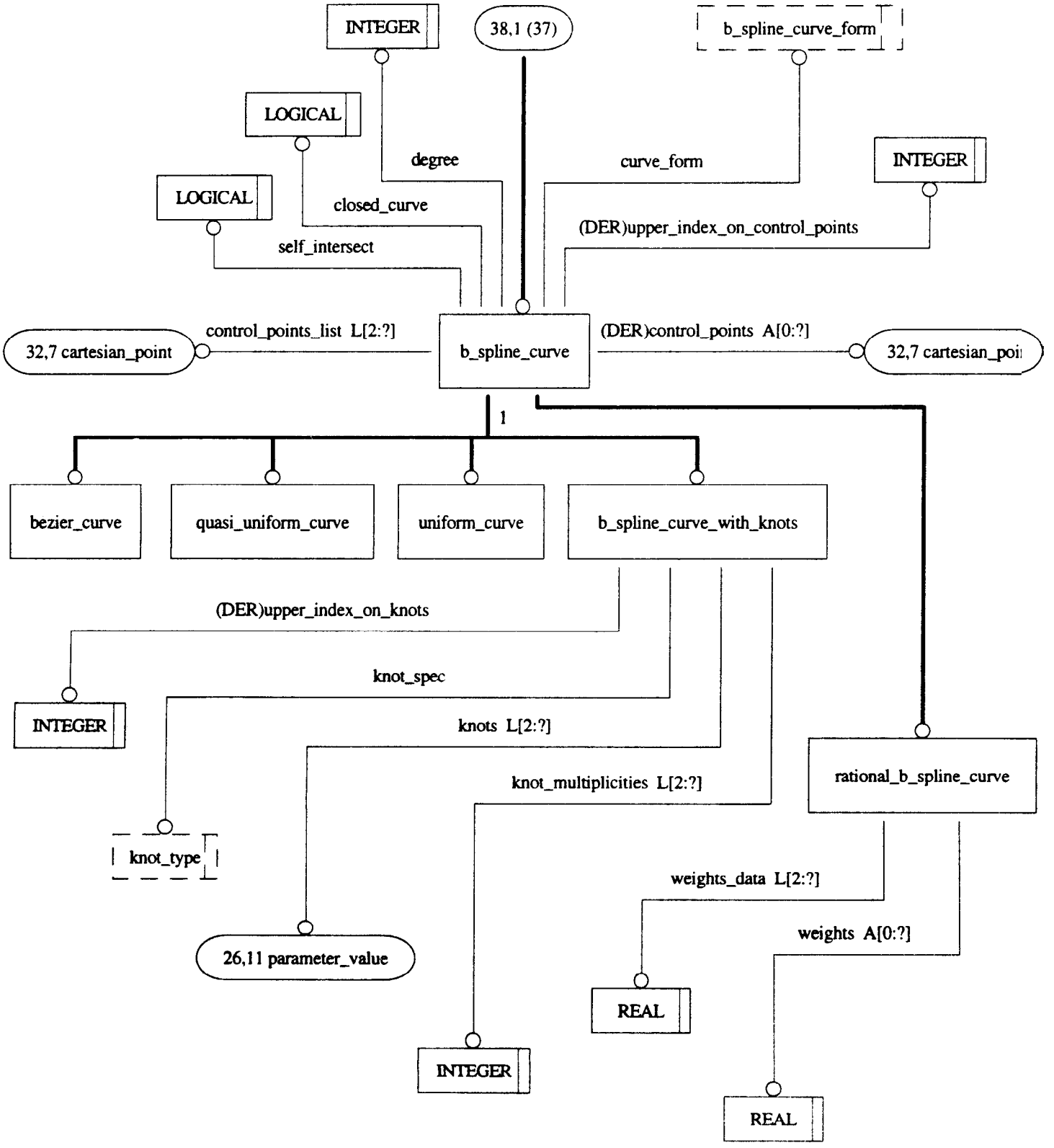

Figure H.38 – AIM EXPRESS-G diagram 38 of 52

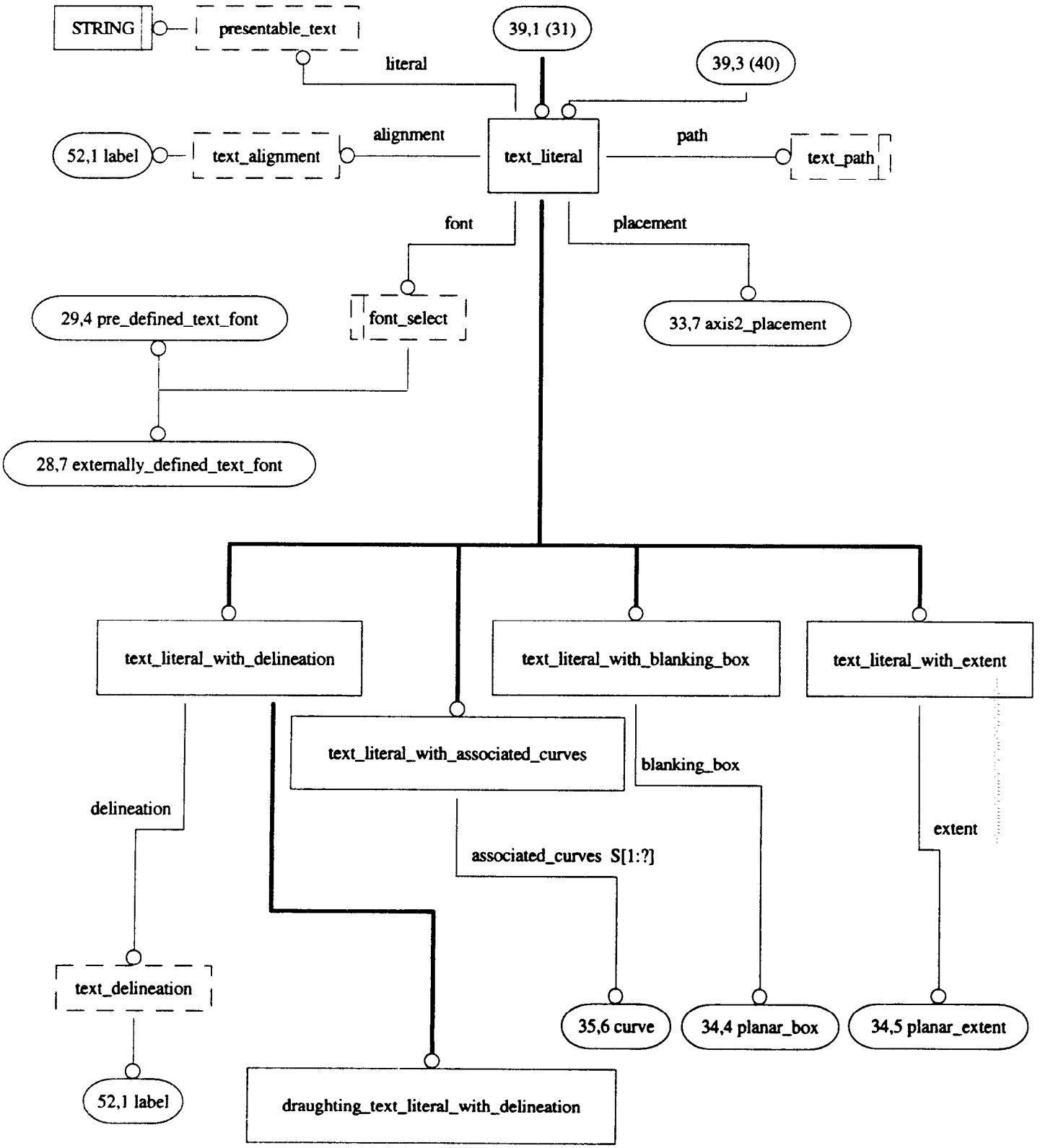

Figure H.39 – AIM EXPRESS-G diagram 39 of 52

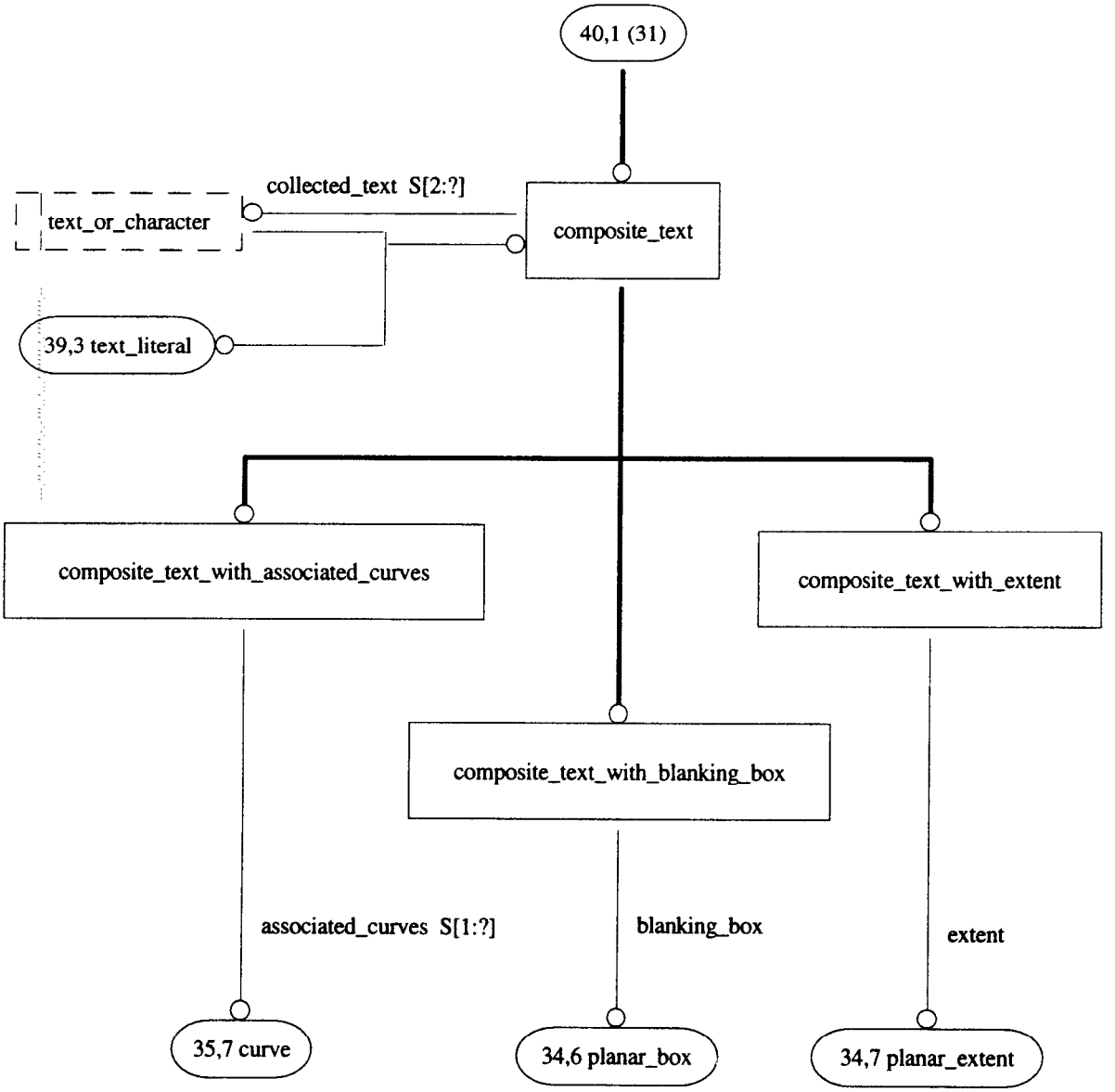

Figure H.40 – AIM EXPRESS-G diagram 40 of 52

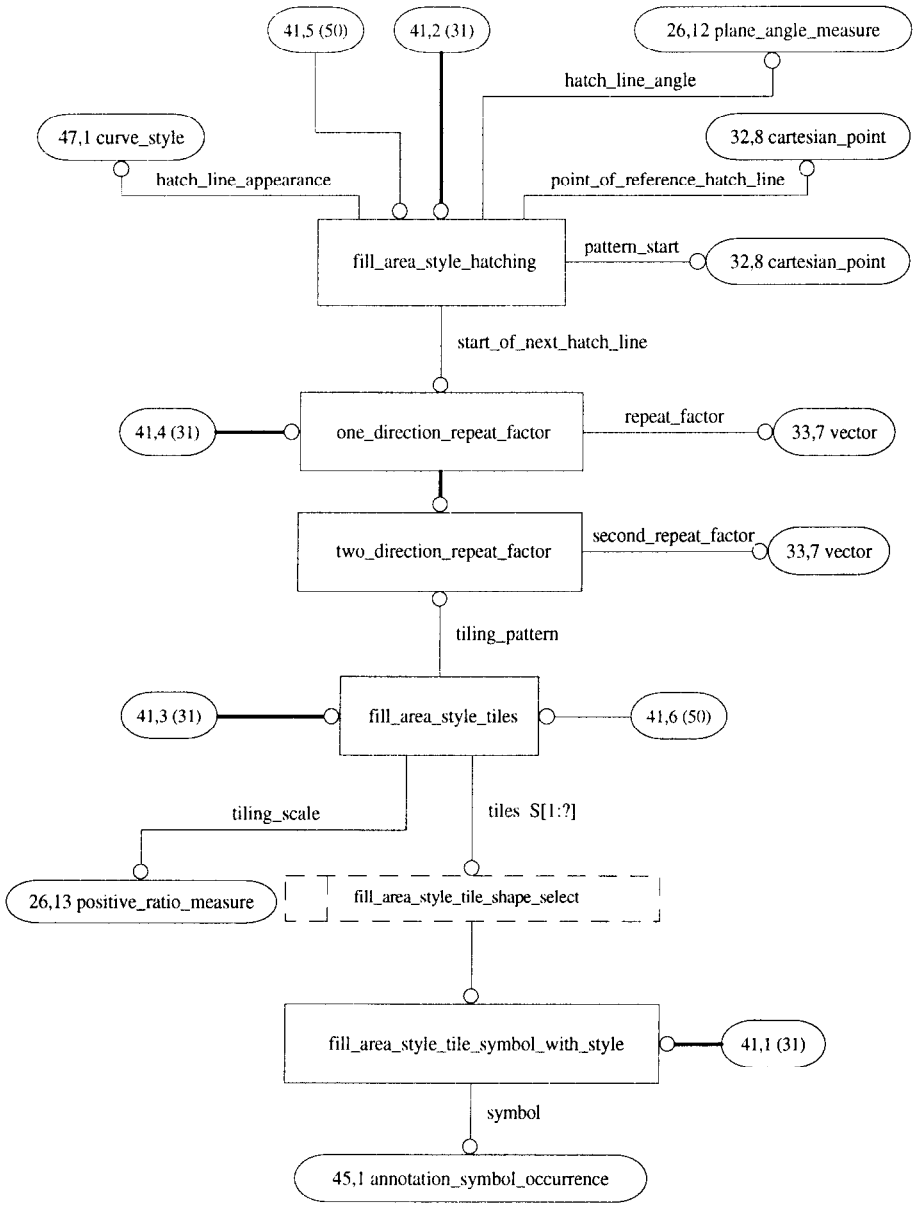

Figure H.41 – AIM EXPRESS-G diagram 41 of 52

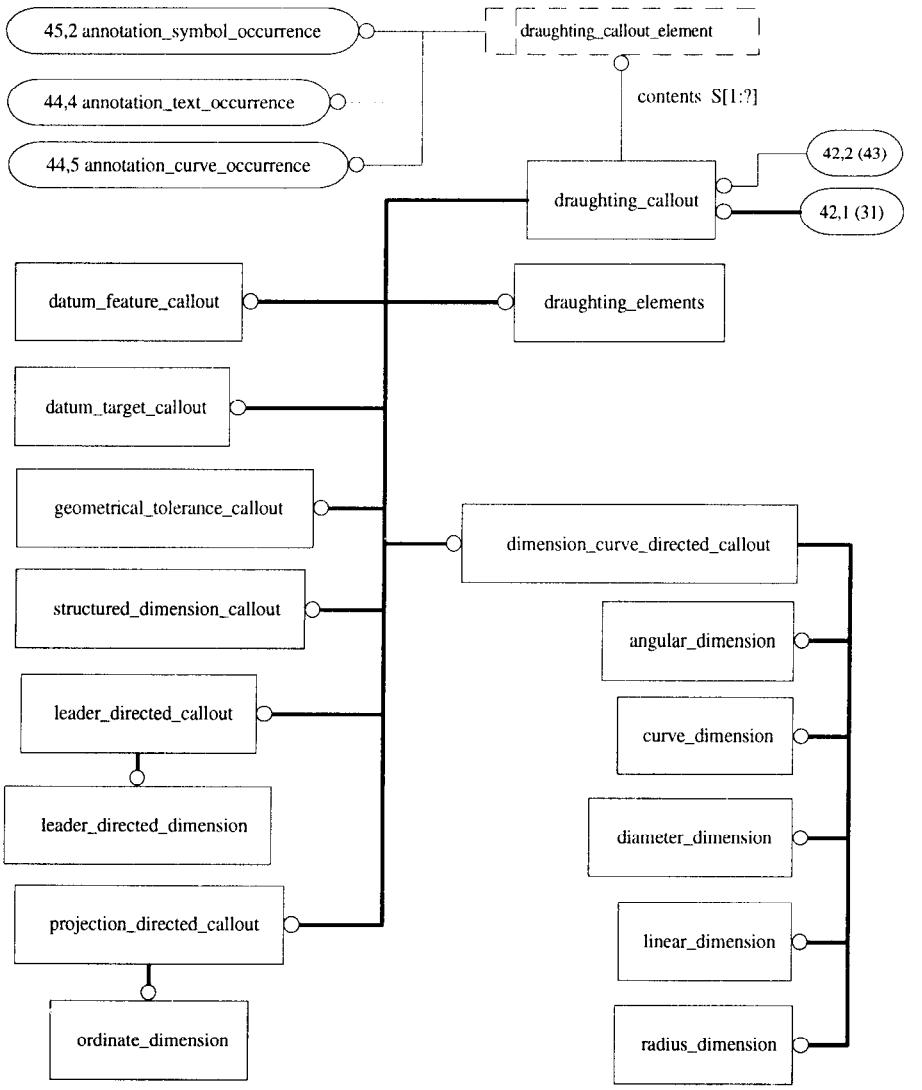

Figure H.42 – AIM EXPRESS-G diagram 42 of 52

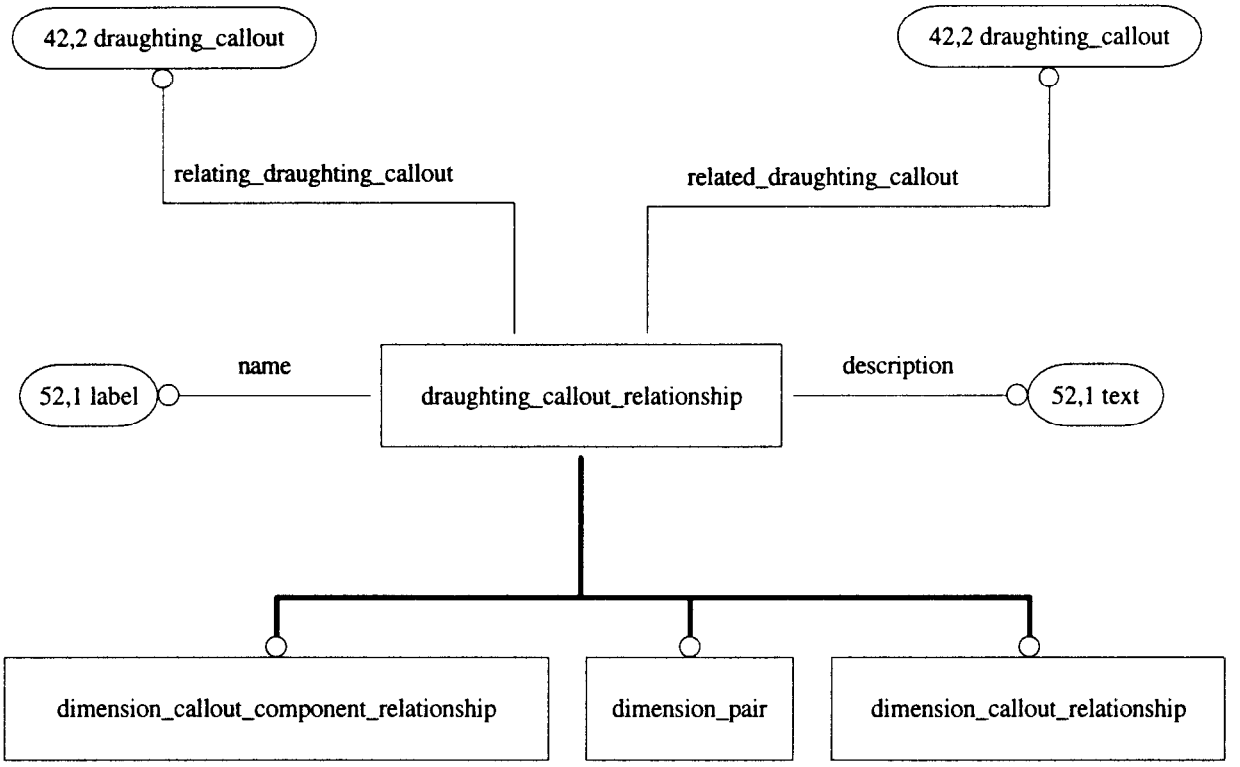

Figure H.43 – AIM EXPRESS-G diagram 43 of 52

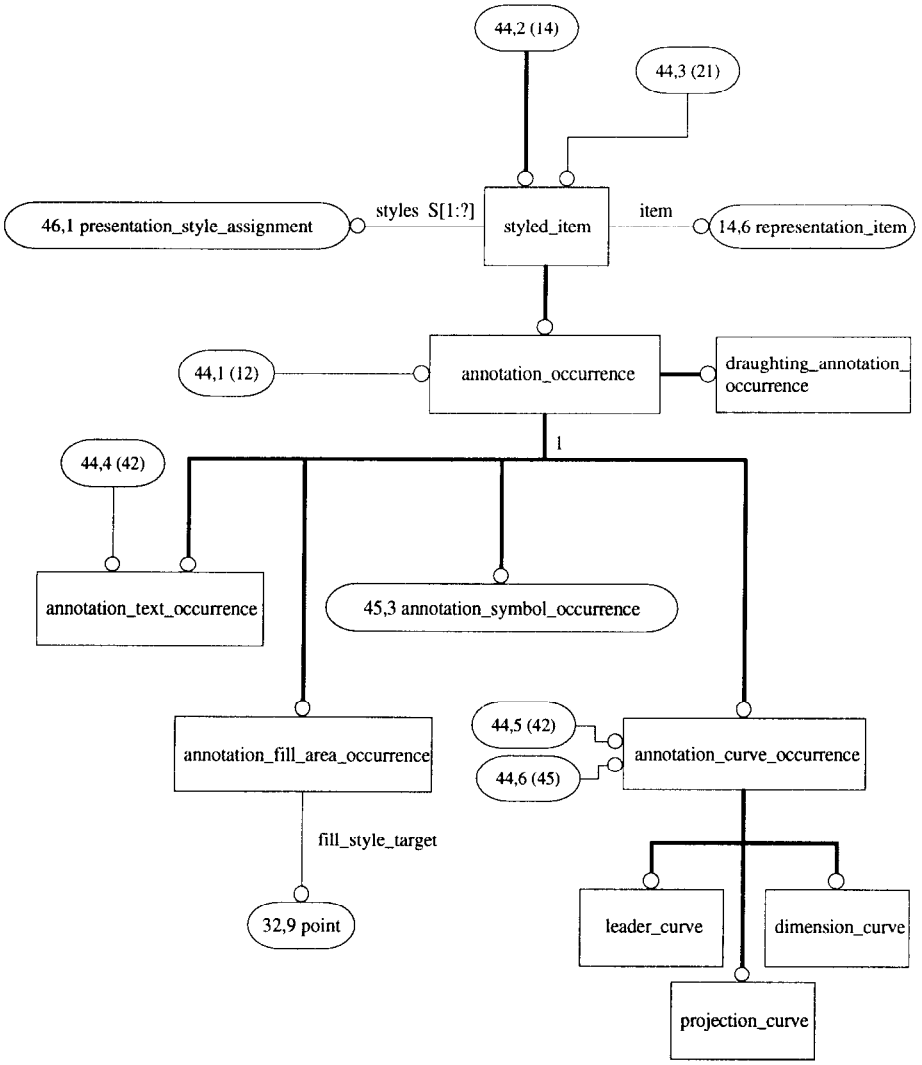

Figure H.44 – AIM EXPRESS-G diagram 44 of 52

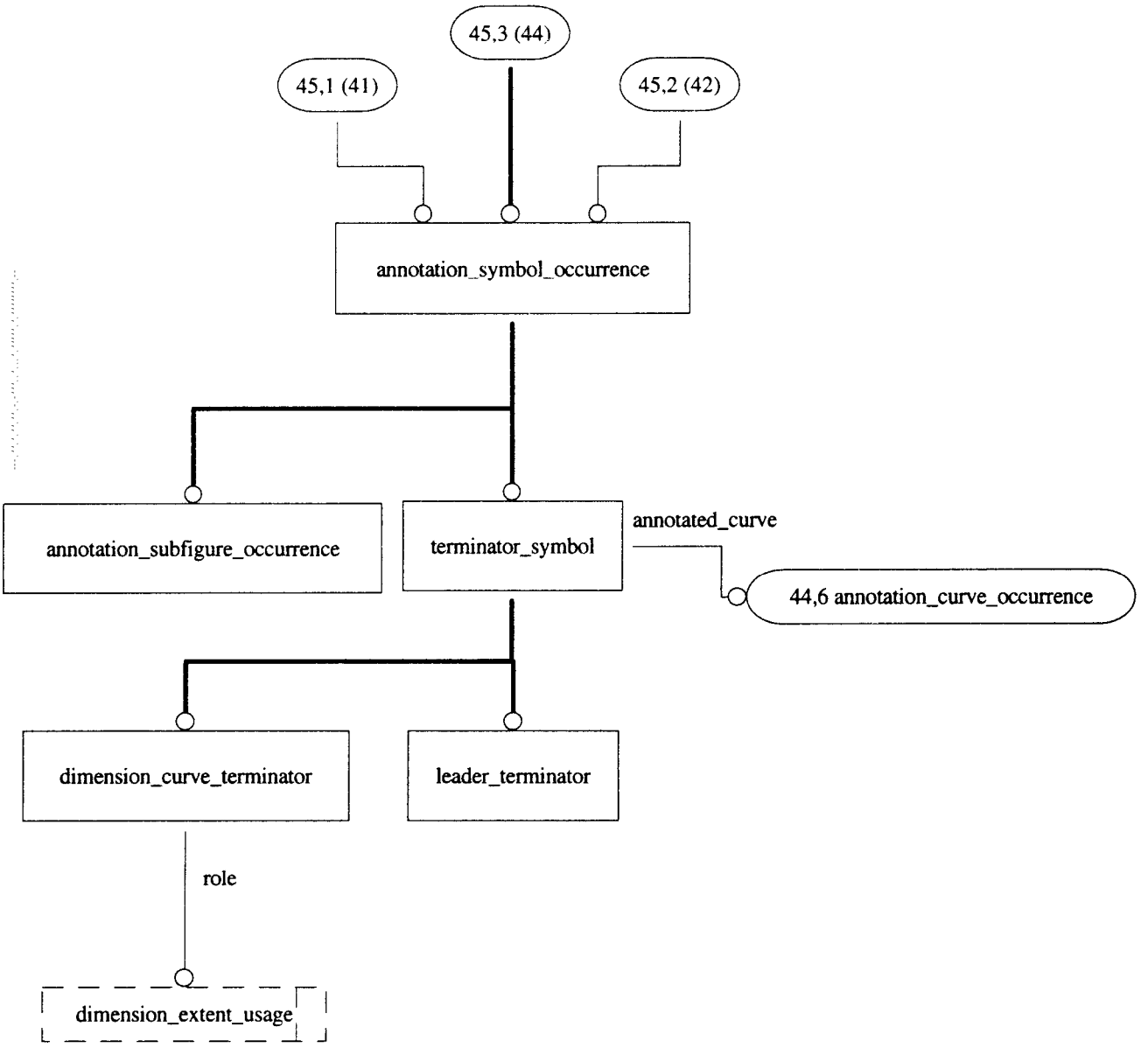

Figure H.45 – AIM EXPRESS-G diagram 45 of 52

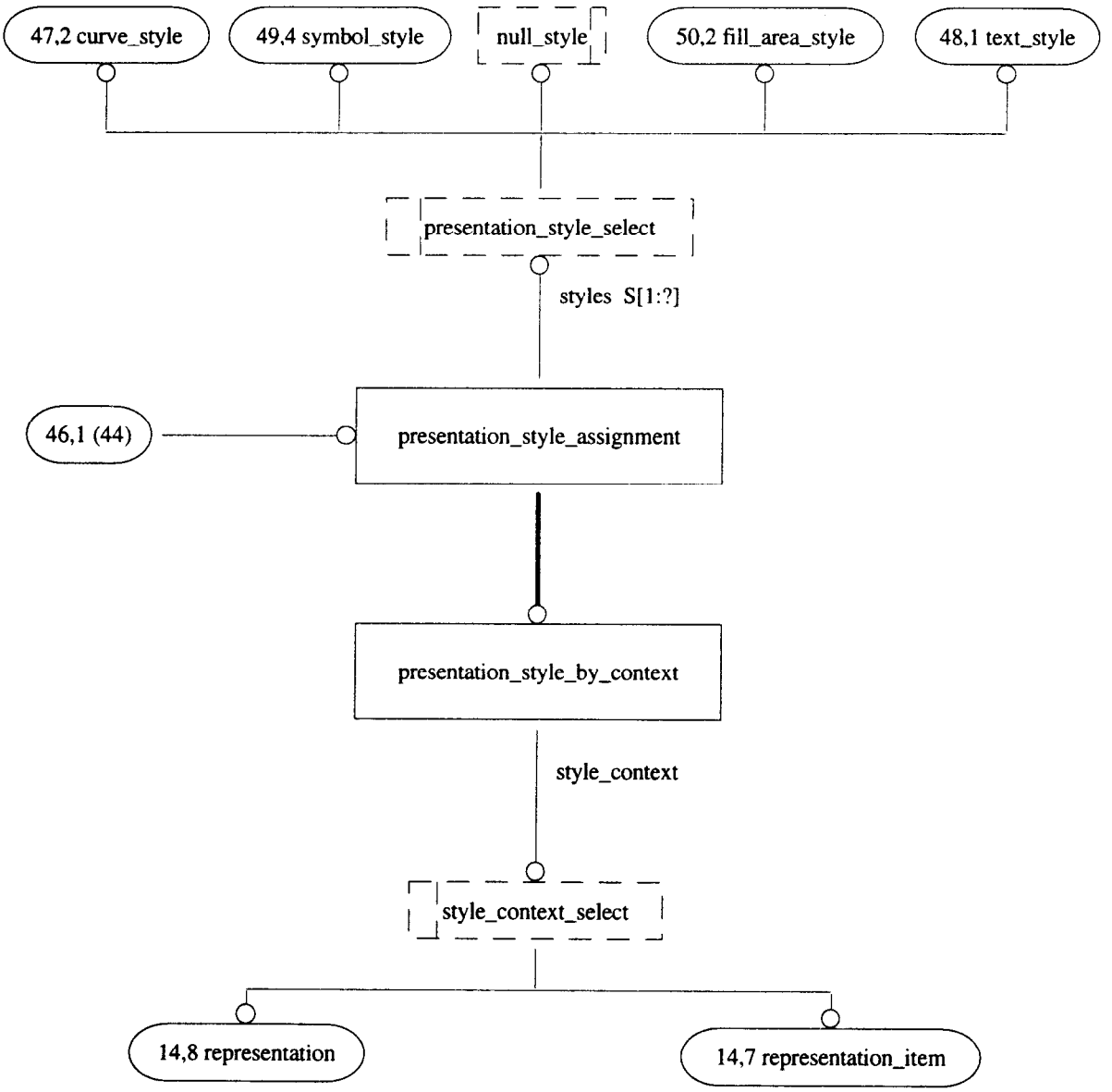

Figure H.46 – AIM EXPRESS-G diagram 46 of 52

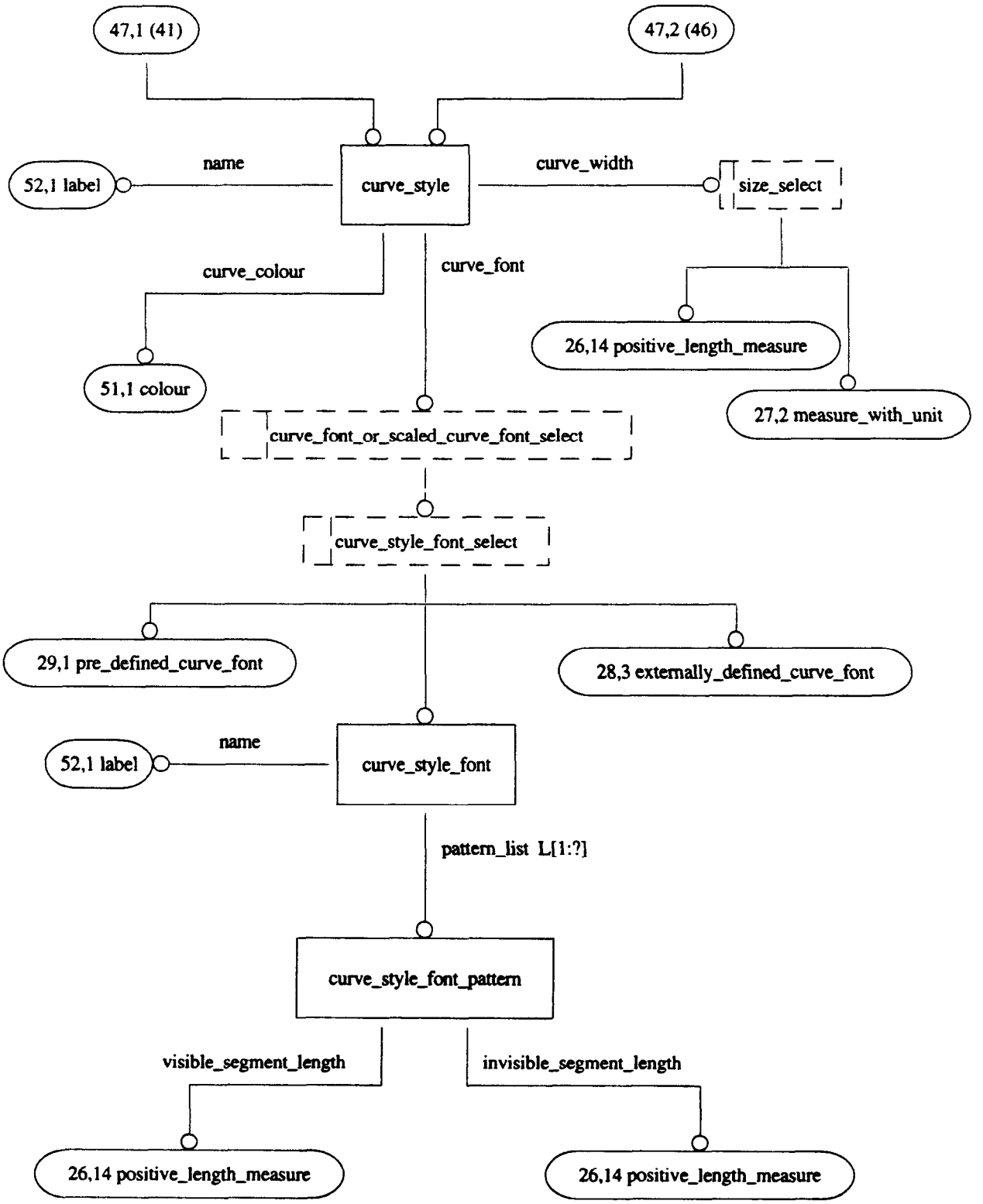

Figure H.47 – AIM EXPRESS-G diagram 47 of 52

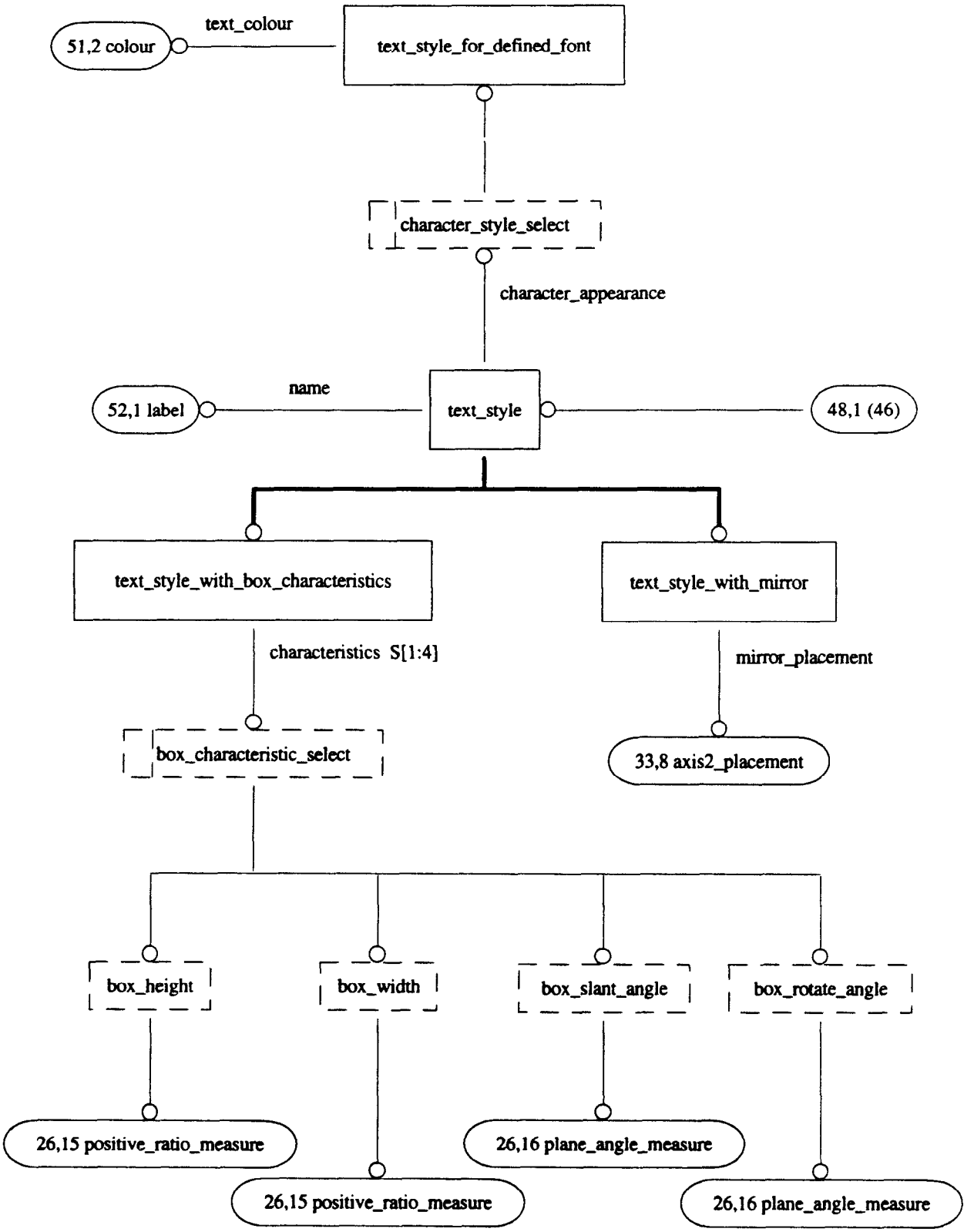

Figure H.48 – AIM EXPRESS-G diagram 48 of 52

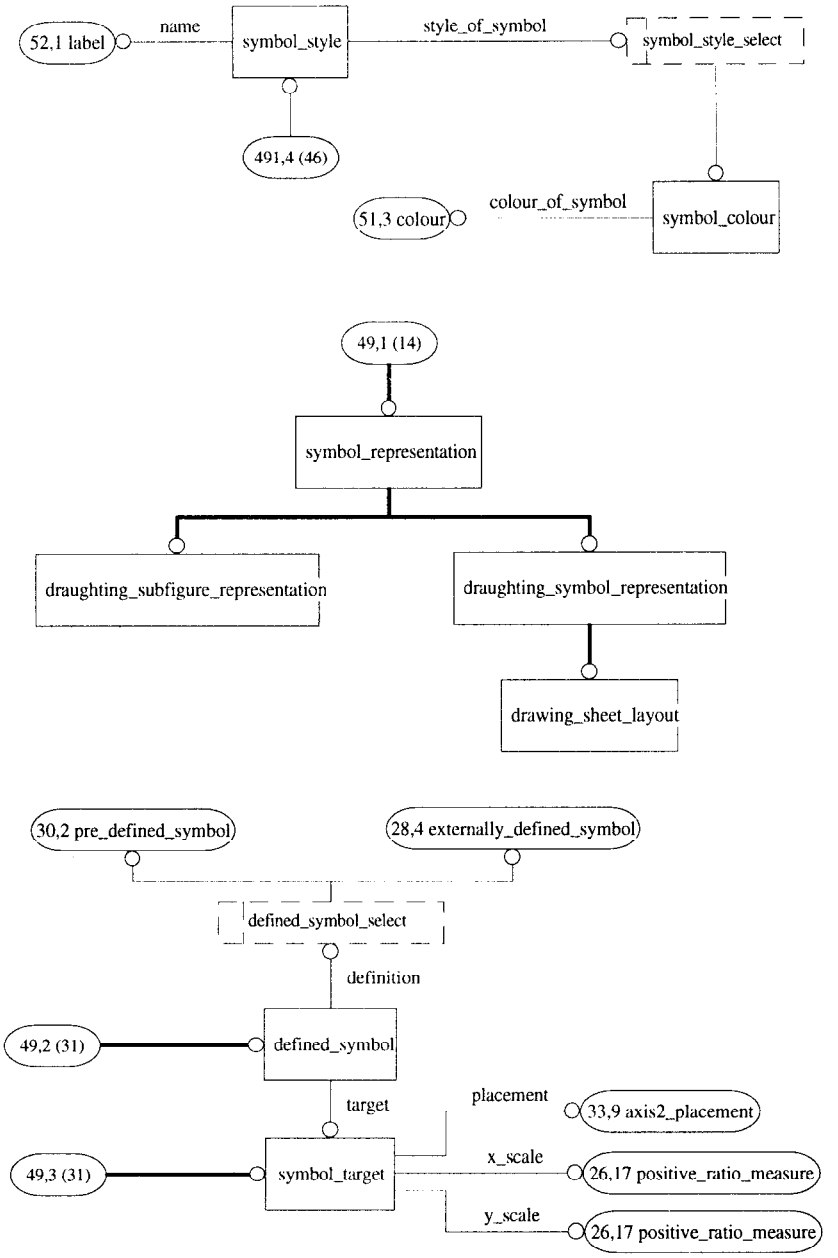

Figure H.49 – AIM EXPRESS-G diagram 49 of 52

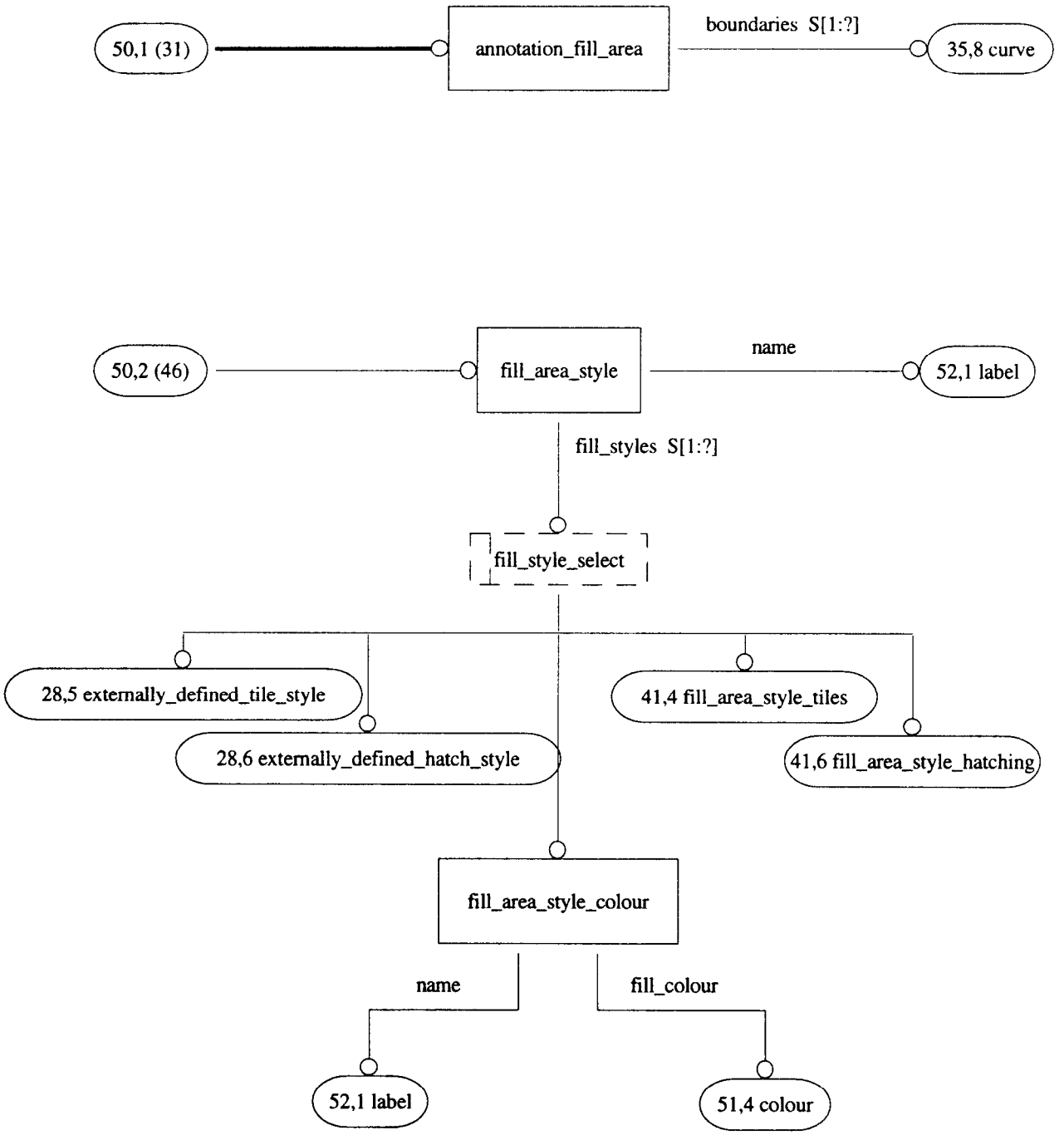

Figure H.50 – AIM EXPRESS-G diagram 50 of 52

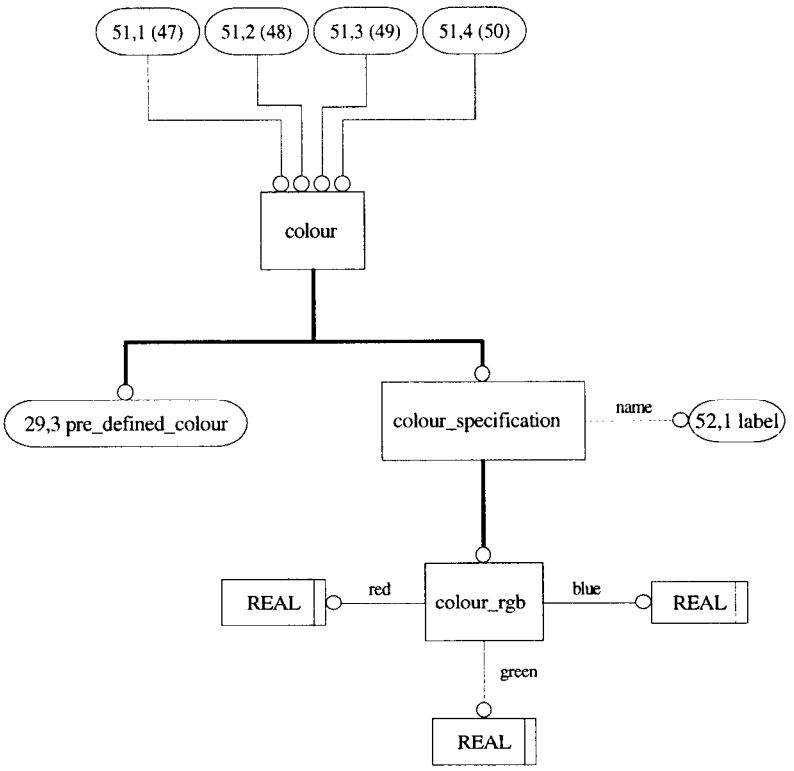

Figure H.51 – AIM EXPRESS-G diagram 51 of 52

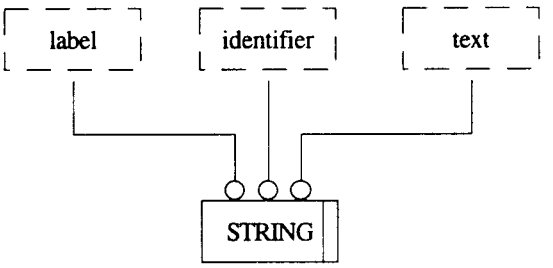

Figure H.52 – AIM EXPRESS-G diagram 52 of 52

Annex J
(informative)

AIM EXPRESS listing

This annex provides a listing of the table of short names and a listing of the *EXPRESS* specified in the AIM of this part of ISO 10303. No text or annotation is included. This annex is provided only in computer-interpretable form.

NOTE – The information provided on this diskette is informative; the normative text is that contained in the body of this part of ISO 10303.

Annex K
(informative)

Bibliography

1. ISO 129 Technical drawings – Dimensioning – General principles, definitions, methods of execution and special indications.

2. ISO STD HDBK 12 Technical drawings, 1982.

3. IEC/ISO Directives: Part 3 "Drafting and presentation of International Standards", 1989.

4. ISO TC184/SC4/EC "Supplementary directives for the drafting and presentation of ISO 10303", June 1993.

5. ISO TC184/SC4/WG4 N32 "Guidelines for the Development and Approval of STEP Application Protocols", version 1.0, 8 January 1992.

6. IDEF0 (ICAM Definition Language 0): "Federal Information Processing Standards Publication XXX, Integration Definition for Function Modelling (IDEF0)", FIPS PUB XXX, National Institute of Standards and Technology, Draft (September 1992).

7. IDEF1X (ICAM Definition Language 1 Extended): "Federal Information Processing Standards Publication XXX, Integration Definition for Information Modelling (IDEF 1X)", FIPS PUB XXX, National Institute of Standards and Technology, Draft (September 1992).

8. ANSI Y14.2M-1979 American National Standard, Line Conventions and Lettering.

Index

ICS 25.040.40

Descriptors: automation, automation engineering, computer applications, industrial products, data, data representation, data exchange, technical drawings.

Price based on 476 pages