



Recommended Practices for Dimensions, Dimensional and Geometric Tolerances

December 6, 2006

Contacts:

David Briggs
Boeing Information Technology
P.O. Box 3707 M/S 2R-97
Seattle, WA 98124 USA
david.d.briggs@boeing.com

Tom Hendrix
Boeing Information Technology
P.O. Box 3707 M/S 2R-97
Seattle, WA 98124 USA
thomas.e.hendrix@boeing.com

Steve Yates
Theorem Solutions Ltd.
Tame House
Fradley Park
Staffordshire
United Kingdom WS13 8RZ
steve@theorem.co.uk

REVISION HISTORY

01 May 2006	Dave Briggs Tom Hendrix	Initial merge of GD&T and DT Practices
18 May 2006	Steve Yates	Completed Merging documents
10 Jul 2006	Steve Yates	Revised some diagrams and added better specification for line and rectangular target areas
15 Sept 2006	Steve Yates	Revision of Representation diagrams and first Presentation practices inserted.

COMMENTS AGAINST THIS VERSION

1 – Ability of GD&T Engineering systems to “offset” Datum Targets from the specified Datum. cf. Olympic podium. Is this supported in STEP?

Answer – Yes, via a Parallel_offset which is a Derived Shape Aspect.

2 – Application of DOCUMENT to a Part as Product for Standard definition. Does this need an additional Document as Product to be correct? cf. Ext Ref Practices.

Answer – May need re-jigging to use DOCUMENT_FILE instead to follow the unmanaged document structure in the PDM Schema usage guide

3 – Shape_def_rep is not allowed to point at Property_Definition. Can point directly to Shape_aspect. Cf. Valprops Rec.Practices. For future usage, consider Item_identified_rep_usage?

Answer – Giedrius Liutkus pointed out that the Description is at odds with the actual EXPRESS for this. The way it is done is correct to the EXPRESS.

4– No specification as to how to apply number of decimal places for Geometric tolerancing. Is this part of Presentation? Or does it form part of the Tolerance e.g. If the tolerance is 0.05 but to 2 decimal places, would the true band be 0.045 to 0.054 ?

5 – Difference between written Standard and Express Rules as to the number of allowable datums in a Common Datum. This needs resolving.

Table of Contents

1	Introduction.....	5
2	Scope.....	5
3	Fundamental Concepts.....	6
3.1	Dimension and Tolerance.....	6
3.2	Dimensions and dimensional tolerances	6
3.3	Geometric tolerances.....	7
3.4	Feature Entities and Attributes.....	7
3.5	Identifying Features.....	7
3.5.1	Data elements of the representation of GDT features and derived elements.....	8
3.6	Datum Systems.....	8
3.6.1	Datums.....	8
3.6.2	Datum Features	8
3.6.3	Datum Targets	9
4	Implementation Guidelines for Dimensional Tolerances.....	10
4.1	Associating Dimensions with Features or Geometry.....	10
4.1.1	Dimensional Location.....	10
4.2	Derived Shapes.....	11
4.2.1	Dimensional Size.....	12
4.2.2	Dimensional Location/Size with Path.....	13
4.3	Application of Values to Identified Tolerances	14
4.3.1	Nominal Value with a Value Limit.....	14
4.3.2	Nominal Value with Plus/Minus Bounds.....	15
4.3.3	Nominal Value with Value Range.....	16
4.3.4	Limits and Fits.....	17
4.3.5	Applying Number of Decimal Places.....	17
5	Implementation Guidelines For GD&T Representation.....	18
5.1	Defining the Dimensioning Standard	18
5.2	Associating tolerances with Features.....	19
5.3	Associating tolerances with Multiple Features.....	20
5.3.1	Associating tolerances with a Pattern of Features.....	21
5.4	Implementing DATUMS in a STEP File.....	22
5.5	Implementing DATUM TARGETS in a STEP File.....	23
5.5.1	DATUM TARGET types.....	25
5.6	Feature Control Frames.....	26
5.7	STEP Supported Tolerance Types.....	27
5.8	Implementing Feature Control Frames.....	28
5.8.1	Geometric Tolerance without Modification or Datums.....	28
5.8.2	Geometric Tolerance with Modification.....	29
5.8.3	Geometric Tolerance with Datums.....	30
5.8.4	Common or Multiple Datums.....	31
5.8.5	Composite Geometric Tolerances.....	32
5.9	Feature of Size.....	34
6	Presentation of Dimensional Tolerances.....	36
6.1	Relating the Presentation with the Representation.....	36
6.2	Presenting the Tolerance.....	37
6.3	Presenting the Associated Graphics.....	38

1 Introduction

This document describes the recommended practices for implementing dimensions, dimensional and geometric tolerances. This document incorporates the content of “Recommended Practices for Dimensions and Dimensional Tolerances” written by Markus Hauser, Mike Strub and Tom Hendrix, dated April 18, 2000 and “Recommended Practices Guide for Geometric Tolerances” written by David Briggs and Tom Hendrix dated March 14, 2003. These two documents were combined to ensure a consistent approach to dimensioning and tolerancing. This document conforms to the agreements reached by the Tolerance Harmonization team representing AP203, AP214, AP224, AP238 and AP240.

Tolerances treat the uncertainty with which the realized shape or measurements of a real manufactured object compare to their design ideals. If all parts could be manufactured perfectly as designed, there would be no need for tolerancing practices. However, it is certain that this cannot be done for finite cost in any but the most trivial cases. In the drawing world, tolerances are noted on the drawing per standard notations such as ANSI Y14.5 or ISO 1101.

This document is not intended as a primer on geometric tolerancing. The explanations included are only provided to relate common tolerancing techniques to the STEP entity structures. This is not a comprehensive coverage of any existing draughting standard but does provide a capability to exchange a variety of typical models. Future versions of this document will address additional capabilities, in particular the STEP Structures required to convey the Presentation of Geometric and Dimensional Tolerances, as well as the application of Tolerances at the Assembly level.

2 Scope

There are two main methods of tolerancing: dimensional tolerancing and geometric tolerancing.

Dimensional tolerancing is the less complex of the two methods of applying tolerances. It is also called “direct tolerancing of dimensions” because a tolerance can be specified only where a dimension is defined. Direct dimensioning and tolerancing address the acceptable range of values of an individual dimension of a manufactured object. Direct tolerancing amounts to generalizing the single value of a dimension to be a range.

Geometric tolerances are the more complex of these two types. A geometric tolerance specifies a geometric region, such as an area or a volume, in which the realized feature must lie in order to meet the design criteria. Geometric tolerancing separates the specification of tolerance from the dimensioning, thus allowing more flexibility and allowing more precise controls that relate more directly to the form, fit and function of the part. This document covers the recommended usage and implementation of geometric tolerances defined in Application Integrated Construct (AIC) 519.

This document covers the application of tolerances to boundary representation solid models. The application of tolerances to wireframe or other geometric models is not covered here.

3 Fundamental Concepts

A tolerance describes a constraint on the acceptable deviation of a manufactured object from the ideal design. Tolerances are applied to the geometric aspects or features of a part, such as edges, faces and holes.

The fundamental principles of geometric tolerances can be found in national and international standards such as ANSI Y14.5M-1994 or ISO 5459-1981.

There are several subtypes of the `geometric_tolerance` entity, which are not mutually exclusive. For example, tolerances that reference datums are of type `geometric_tolerance_with_datum_reference`. Tolerances that include a modifier such as maximum material condition are of type `modified_geometric_tolerance`. Many typical engineering tolerances combine these. In these cases, complex entities instances will occur in the Part 21 file.

3.1 Dimension and Tolerance

Dimension is a term for a specification of the value of a parameter of some aspect of the shape of a mechanical part or assembly. A dimension can be implied by the geometric model, or it can be explicitly modeled, which is what this guide covers. The term dimension can also refer to the numerical value itself; however in this document the term value is used. Tolerance is a general term for permitted variations in the shape of manufactured parts. Tolerance treats the how the realized form or measurements of a real manufactured object compare to their design ideals.

3.2 Dimensions and dimensional tolerances

The dimensions and dimensional tolerances addressed in this document are:

- directional dimensions
- location dimensions such as angular, curved, or linear distances
- size dimensions such as angular, thickness, or other
- the association of dimensions with geometry
- the representation of dimensional tolerances including:
 - plus-or-minus deviations
 - maxima, minima, and nominal dimensions
 - limits and fits
 - significant digits
- the association of dimensional tolerances with dimensions

3.3 Geometric tolerances

The geometric tolerances addressed in this document are:

- Angularity
- Circular runout
- Circularity/Roundness
- Coaxiality/Concentricity
- Cylindricity
- Flatness
- Parallelism
- Perpendicularity
- Position
- Profile of a line
- Profile of a surface
- Straightness
- Symmetry
- Total runout

Tolerance modifiers (Maximum and minimum material condition, regardless of feature size and projected tolerance zone) are also addressed.

3.4 Feature Entities and Attributes

Dimensions and dimensional tolerances are applied to aspects of the product shape. The product and product shape are modeled as in other geometry and PDM applications.

GDT Features in STEP are modeled as shape_aspects. The term “feature” in some dimensioning standards is reserved for definitional elements that lie in the surface of the part. In such cases, a term such as “derived element”, is used for derived geometry. In computing systems and in some dimensioning standards, either is called a feature, and these are distinguished as “integral” feature and “derived” feature. In this recommendation, all are modeled as shape aspect. For integral features, shape_aspects.product_definitional=.TRUE.” For derived elements it is shape_aspects.product_definitional=.FALSE.”

3.5 Identifying Features

The surface of a part can be partitioned into **features**, which to which dimensions are applied. Normally a feature boundary corresponds to a locus of discontinuities of surface curvature, as when a straight side encounters a corner fillet. For the purposes of GDT, every point on the surface is either in the interior of one feature or on the boundary of two or more features. When a finished part is measured, each point of the surface belongs to exactly one feature.

It is sometimes convenient to treat as a single feature a union of these natural geometric features. Unions can be disjoint, for example:

- a pattern of holes
- a surface that is “interrupted” by a slot.
- the two sides of a slot

Unions can be made of contiguous features for example:

- the all-around shape of an irregular hole.

Similarly it may be necessary to identify a restricted region of a feature, for example:

- Where a tighter tolerance is required
- to indicate how a finished piece is mounted on an inspection workbench.

3.5.1 Data elements of the representation of GDT features and derived elements.

It is recommended that when available the advanced_face be used for representing a feature, since its topology is well-defined.

Derived center elements such as points, curves, and surfaces may be unbounded and can be represented by geometry primitives. In GDT the derived elements are considered to be implicitly bounded where they intersect another feature of a part. Any geometric_representation_item or topological_representation_item could potentially be incorporated into a feature or derived element's representation.

3.6 Datum Systems

Some types of tolerances refer to one or more datums in order to represent the requirements on the shape. Datum systems are related datums that provide a reference system for describing requirements on the product shape.

3.6.1 Datums

A datum is a theoretically exact geometric reference, i.e. an exact point, line or plane, to which toleranced features are related. A datum is the origin from which the location or geometric characteristics of features of a part are established. A datum may be based on one or more datum features of a part.

3.6.2 Datum Features

Datum Features are tangible features of a part, for example a face that provides a reference system for measurements of the actual part. Datum Features must lie on the physical boundary of the shape. Consequentially, Datum Feature entities are related to topological entities that represent those boundaries in the solid model such as an advanced_face.

3.6.3 Datum Targets

A datum Target designates a specific point, line or area of contact on a part that is used in establishing a data reference frame (definition from ANSI Y14.5). It differs from a datum_feature in that it identifies a restricted region of a feature, i.e. a point, line or area of a surface rather than a topological feature. Typically, two or more datum target elements are used to define a datum

4 Implementation Guidelines for Dimensional Tolerances

4.1 Associating Dimensions with Features or Geometry

Dimensional tolerances are attached to the items of geometry or topology that the dimension applies to. This is done by means of a subtype of the Shape_aspect_relationship Entity, i.e. dimensional_location, or by means of a dimensional_size entity. The identification of the type of tolerance is done via a specified string in the “name” attribute of the Dimensional Location or Dimensional Size entities. Note that Angular Dimensions are an exception to this, being defined by the Angular Location and Angular Size entities. The mappings for the Dimensional Tolerance types to the required string are shown in Table 1 and Table 3.

4.1.1 Dimensional Location

For the Dimensional_Location entity, the items between which the dimension applies are defined by shape_aspects. This is shown in Illustration 1.

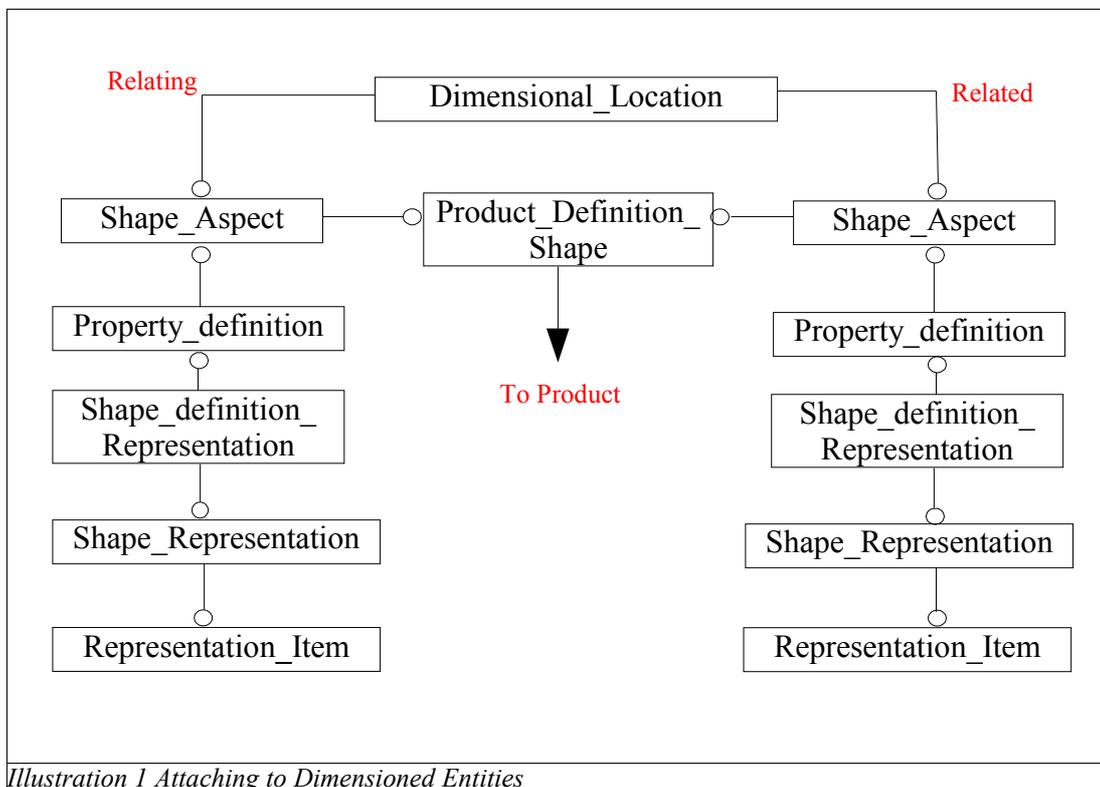


Illustration 1 Attaching to Dimensioned Entities

Note that the same structure is applied for the specialised subtypes of Dimensional Location, namely “Dimensional Location with Path” and “Directed Dimensional Location”. The former of these provides a “path” for the measurement to follow by means of a shape_aspect (see Illustration 4), the latter provides additional semantics to the “Related” and “Relating” attributes of the Dimensional Location, i.e. The measurement is to occur from the “relating” shape aspect to the “related” shape aspect.

Dimensions that map to a Dimensional Location are :

Dimensional Tolerance	Dimensional Location.Name
Angular Location Dimension	N/A
Curved Distance Dimension	'curved distance'
Linear Distance Dimension	'linear distance'

Table 1 Dimensional Location Types

4.2 Derived Shapes

The preceding figure (Illustration 1) shows the attachment of the Representation Items that form either end of the Dimension_Location to the Geometric or Topological entities at those points . In some cases, a Representation Item may not exist for that location in the model, e.g. The centre of a Hole , so in this case a “derived_shape_aspect” will need to be created to provide an anchor point.,. In these cases the additional Entities required are shown in Illustration 2.

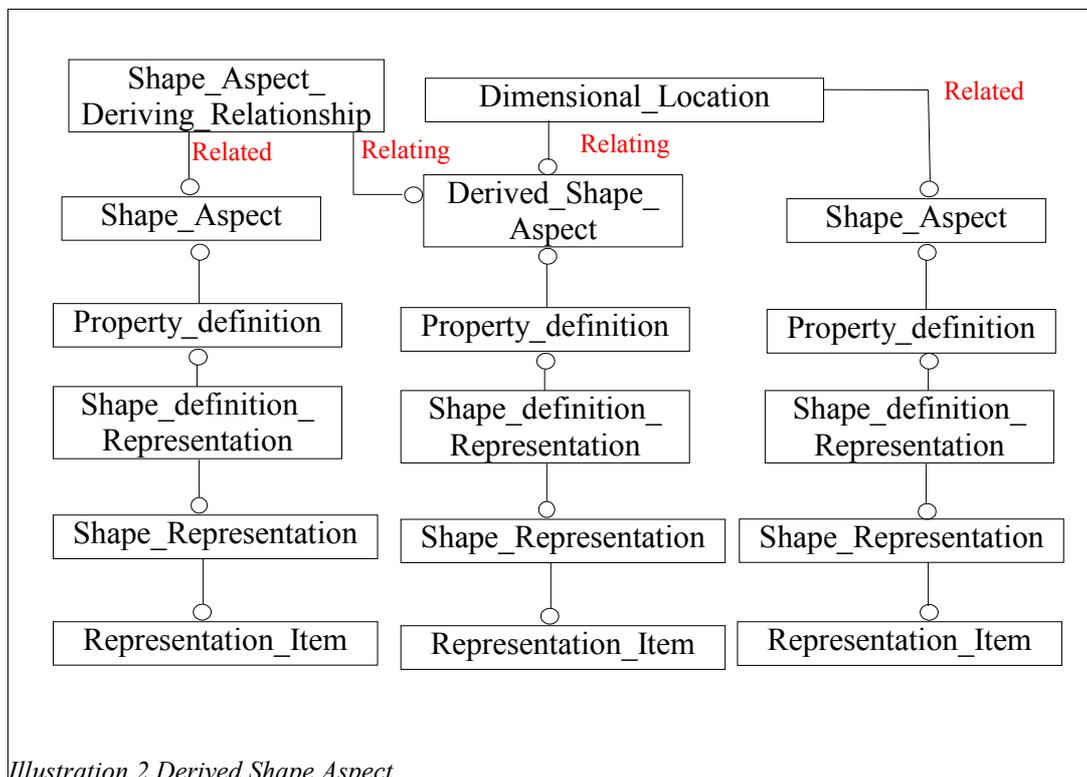


Illustration 2 Derived Shape Aspect

Under certain conditions, a specialised subtype of the Derived_Shape_Aspect should be used. These conditions are listed in Table 2. Under any other conditions, the plain Derived_Shape_Aspect entity should be used. It is recommended, but not obligatory, that a meaningful string for the type of derivation be entered in the “name” attribute of this entity.

Condition	STEP Entity Used
Apex of a Cone	APEX

Condition	STEP Entity Used
Centre of a Symmetrical Feature	CENTRE_OF_SYMMETRY
Geometric Alignment of two Features	GEOMETRIC_ALIGNMENT
Perpendicular to Feature	PERPENDICULAR_TO
Spatial Extension to Feature	EXTENSION
Tangential to Feature	TANGENT
Parallel Offset from Feature	PARALLEL_OFFSET

Table 2 Types of Derived Shape

4.2.1 Dimensional Size

The Dimensional Size Entity is used where the measurement only applies to one object, rather than being a measurement between two distinct geometric or topological feature. Note that this “one object” can, under certain circumstances, be a composite of several shape_aspects. This will be illustrated in a later section. Illustration 3 shows how a Dimensional Size Entity is attached to the Representation Objects that it is a dimension of.

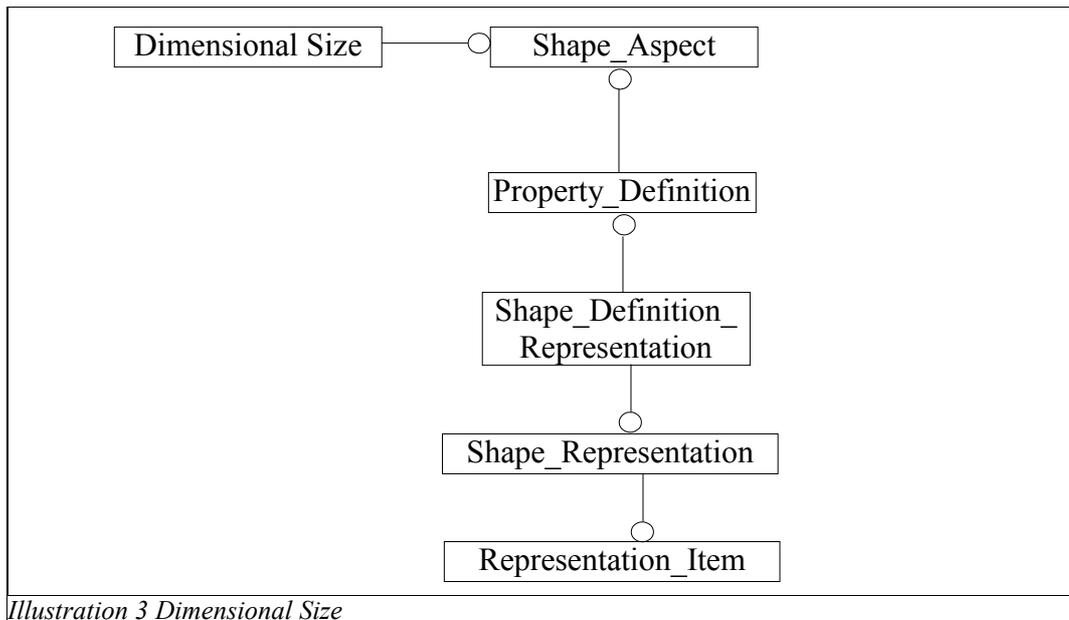


Illustration 3 Dimensional Size

Note that the same structure is applied for the specialised subtype of Dimensional Size, namely “Dimensional Size with Path”. This provides a “path” for the measurement to follow by means of a shape_aspect (see Illustration 4).

Dimensions that map to Dimensional Size are :

Dimensional Tolerance	Dimensional Size.Name
Angular Size Dimension	N/A
Curved Size Dimension	'curve length'
Diameter Size Dimension	'diameter'
Height Size Dimension	'height'
Length Size Dimension	'length'
Radial Size Dimension	'radius'
Thickness Size Dimension	'thickness size'
Width Size Dimension	'width'

Table 3 Dimensional Size Types

4.2.2 Dimensional Location/Size with Path

For some measurements, a path for the measurement needs to be defined, for example, when measuring the linear distance between two point on a curved surface. In this case, the measurement would need to follow the curve of the surface and not be the shortest straight line distance between the two points. In order to convey this information, this specialised subtype has an additional attribute, “path”, which points to a shape_aspect defining the path the measurement is to take. The instantiation of this is shown in Illustration 4.

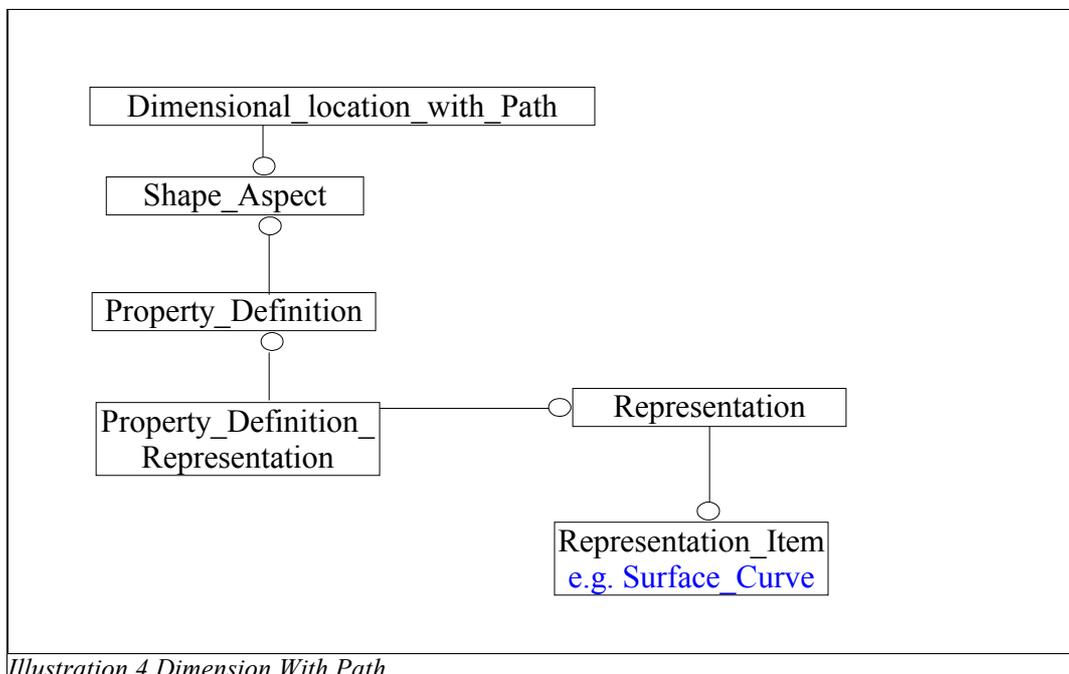


Illustration 4 Dimension With Path

4.3 Application of Values to Identified Tolerances

Once you have identified the tolerance and attached it to the relevant pieces of geometry or topology as shown in the previous section, you can assign the tolerance value. The types of value that can be applied are :

- nominal value with a value limit
- nominal value with plus/minus bounds
- nominal value with value range
- limits and fits

4.3.1 Nominal Value with a Value Limit

In this case, a nominal value is applied to the dimension, i.e. A set value for the measurement, without any tolerance bounds. However, the value can be limited as to whether it is a Maximum value or Minimum Value. There may be other limits allowed, and as the limit maps to a String in the Part21 Instantiation, there is no restriction on what value can be entered. For the present, I suggest that implementations limit this value to “MAX” or “MIN” as these are the only values which have set semantics.

Thus for a nominal value, the actual value applied to the Dimension is conveyed by the value_component attribute of the Measure Representation Item, and the Limit on this value by the name attribute in the associated Type Qualifier Entity. Illustration 5 shows the STEP entities used to instantiate this. Notes that the Dimensional Location referred to in the diagram could also be a Dimensional Size entity.

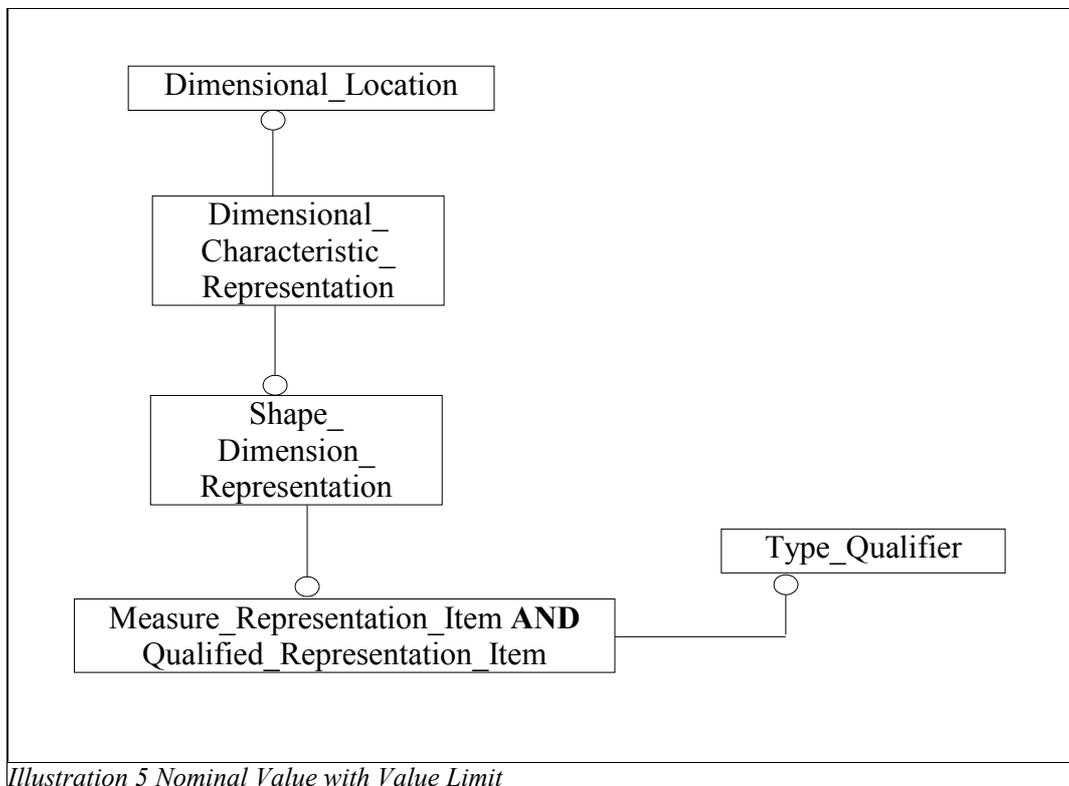
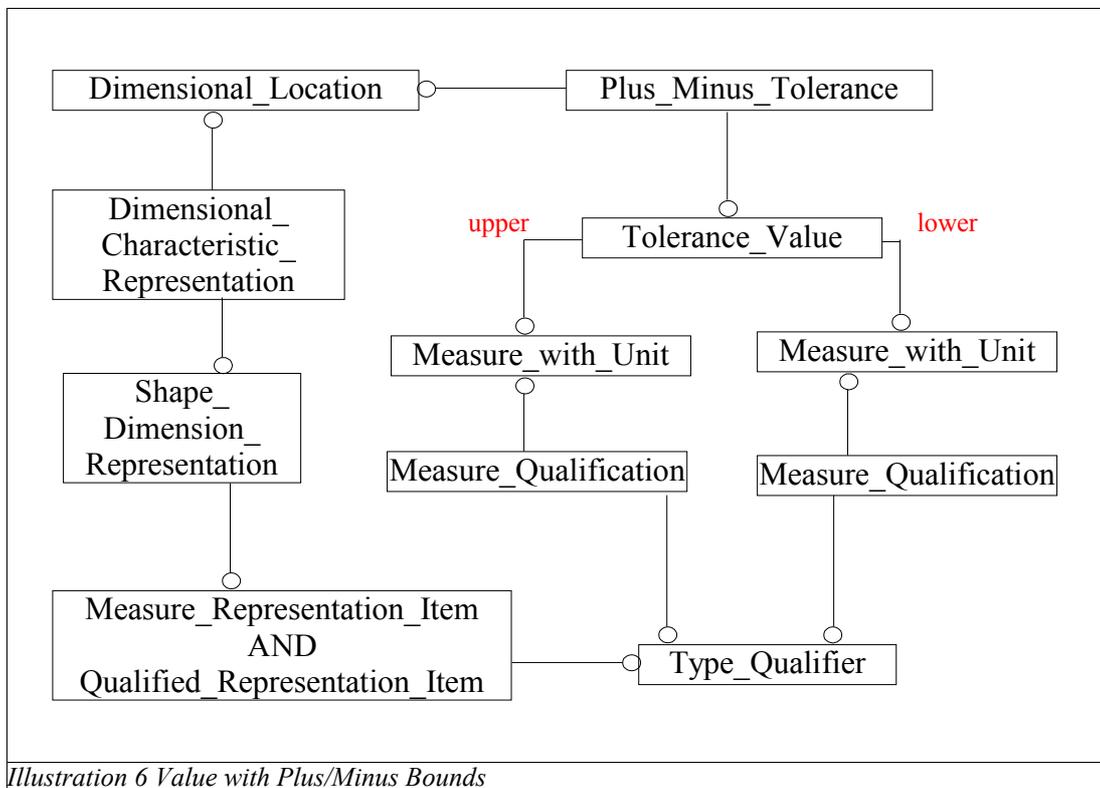


Illustration 5 Nominal Value with Value Limit

4.3.2 Nominal Value with Plus/Minus Bounds

If a tolerance is represented as a Nominal value with a set of plus and minus deviations or bounds to that tolerance, then the entities shown in Illustration 6 are used to instantiate the STEP File.



For this instantiation there are some points to note. The Type Qualifier “Name” attribute, used to denote MIN and MAX in Section 4.3.1, in this case takes the value “designed” in order to convey the fact that the Tolerance applied is the “as designed” tolerance. It can take the values: “required”, “calculated”, “measured” and “estimated” in addition to “designed”, but the semantics of these are unclear at present.

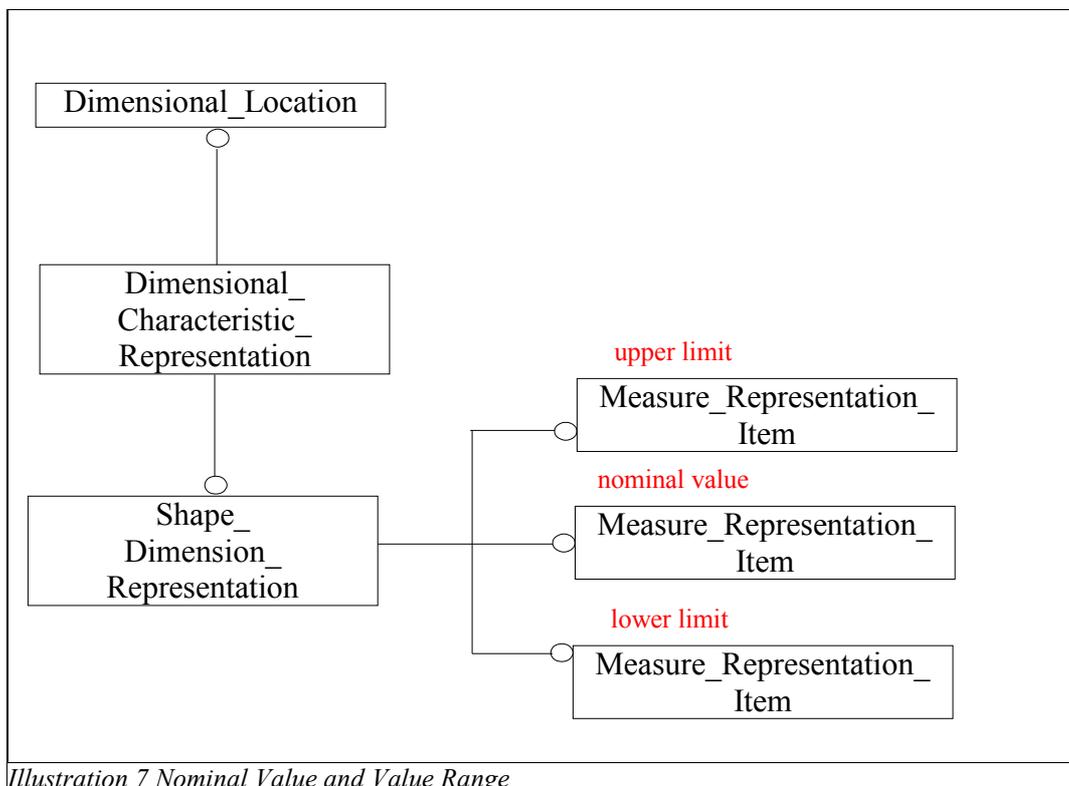
The Values of the Plus/Minus Tolerance are found in the Measure With Unit Entities referred to by the “upper” and “lower” attributes of the Tolerance Value Entity. Note that for clarity, even though it is given that the Tolerance Value is an upper or lower limit, the value should still be given as an offset from the Nominal. e.g. The Lower Limit should be a Negative number and the upper limit a Positive Number.

4.3.3 Nominal Value with Value Range

An alternative way of presenting, rather than representing the dimension, is as a range of values that is the extents of the tolerance applied. As originally mapped in the Standard, this would give two values, one for the upper limit and one for the lower limit. Upon implementation, however, it was found that the majority of CAD systems hold the tolerance value as a nominal, with a plus/minus deviation, and handle the presentation of this data separately, thus the lack of a nominal value through this method caused problems for re-importing the data, as this value was lost.

In order to alleviate this, a third value, representing the nominal value was added to the mapping.

The STEP instantiation of this is shown in Illustration 7.



To determine which of the Measure Representation Items denotes which value, they shall be identified by one of the following strings used for the “name” attribute : “nominal value”, “upper limit”, “lower limit”.

4.3.4 Limits and Fits

The implementation of a limits and fits tolerance is fairly easy. What is not immediately apparent is the meaning of the tolerance. The Limits and Fits designation is closely linked to the Tolerancing Standard applied for the file, as it references a table of standard tolerances maintained within that standard.

Illustration 8 specified the STEP entities used to carry this information in an exchange file.

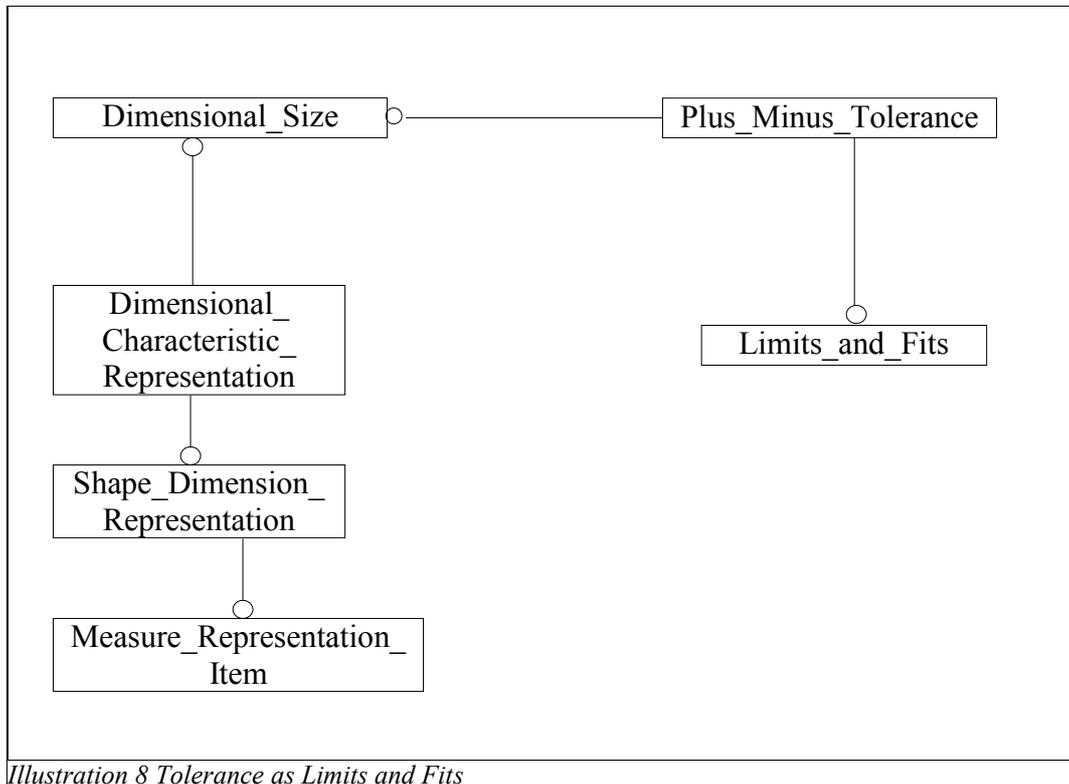


Illustration 8 Tolerance as Limits and Fits

4.3.5 Applying Number of Decimal Places

The displayed value of the Dimensional Tolerance can be truncated to a number of decimal places, by specifying a Precision Qualifier to the Representation Item which hold to tolerance value. This is done using a complex entity for the Measure_Representation_Item, containing the Qualified_Representation_Item entity, which in turn, points to a Precision_Qualifier, which specifies the number of decimal places by its precision_value attribute.

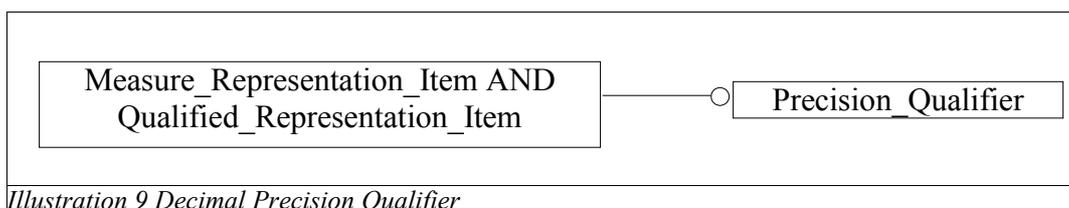


Illustration 9 Decimal Precision Qualifier

5 Implementation Guidelines For GD&T Representation

5.1 Defining the Dimensioning Standard

Different dimensioning standards define the visual representation and interpretation of the symbology of the tolerances. In order for the receiving system to create the correct visual representation of the tolerance, the dimensioning standard must be known. This information is captured as an `applied_document_reference` which applies the referenced document, i.e., the dimensioning standard, to the `product_definition` of the part. This is shown in Illustration 10.

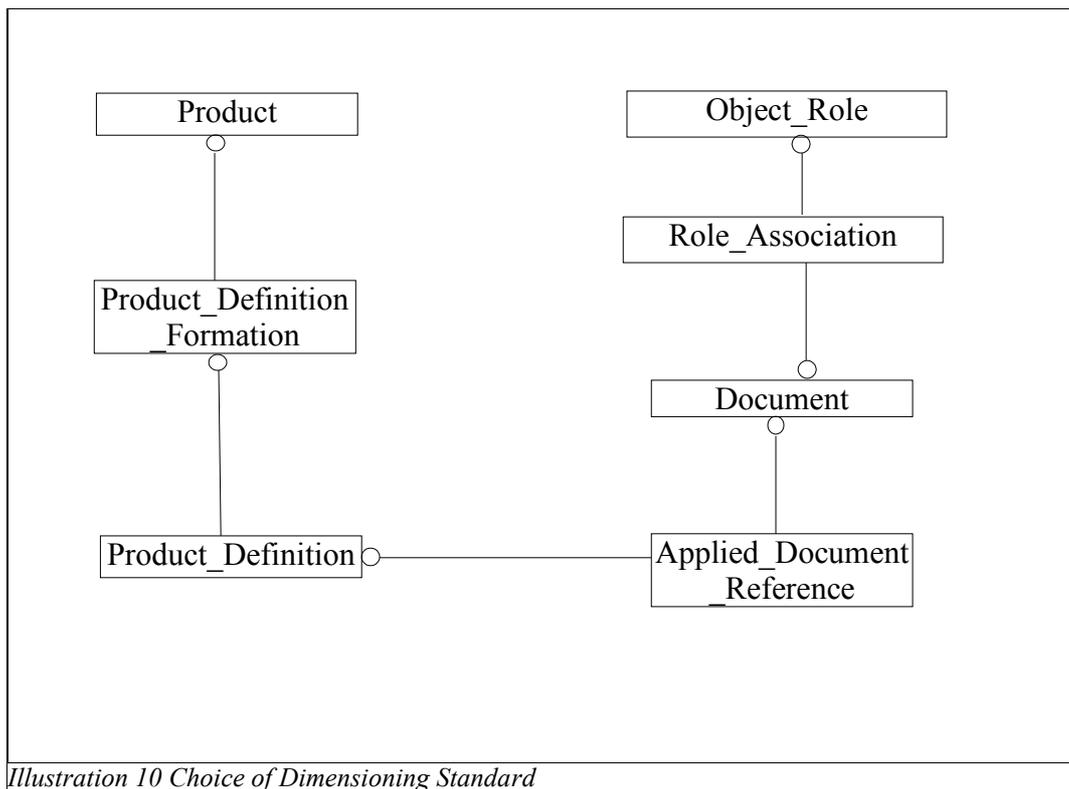


Illustration 10 Choice of Dimensioning Standard

The following standard strings may be used for the document identifier:

“ASME Y14.41-2003”
“ASME Y14.5M-1994”
“ISO 1101:2004”

Per the tolerancing harmonization project, the object role for the document has the string value of "dimensioning standard".

Part21 Example

```
#218=OBJECT_ROLE('#218','dimensioning standard');  
#219=DOCUMENT_TYPE('Standards Document');  
#220=DOCUMENT('ASME Y14.5M-1994','Geometric Tolerancing
```

```

Standard', 'Geometric Tolerancing Standard', #219);
#221=ROLE_ASSOCIATION(#218, #220);
#222=APPLIED_DOCUMENT_REFERENCE(#220, 'Standards
Body', (#167));

```

5.2 Associating tolerances with Features

In STEP, the tolerance entities are associated with a `shape_aspect` that identifies the tolerated feature. The feature is identified by a `shape_aspect` which has a representation. In the case of a solid boundary representation model, the feature of the part is represented by one or more `topological_representation_items` such as `advanced_face` entities. For example, a through hole in a solid model might be represented by two semi-circular surfaces, each an `advanced_face` entity. These `topological_representation_items` are collected together by a `shape_representation` which is representation of the `shape_aspect` for the feature. This `shape_representation` shall share the same `geometric_representation_context` as the solid. See Illustration 11 for an example of how the tolerance entities are related to the shape elements of the tolerated feature.

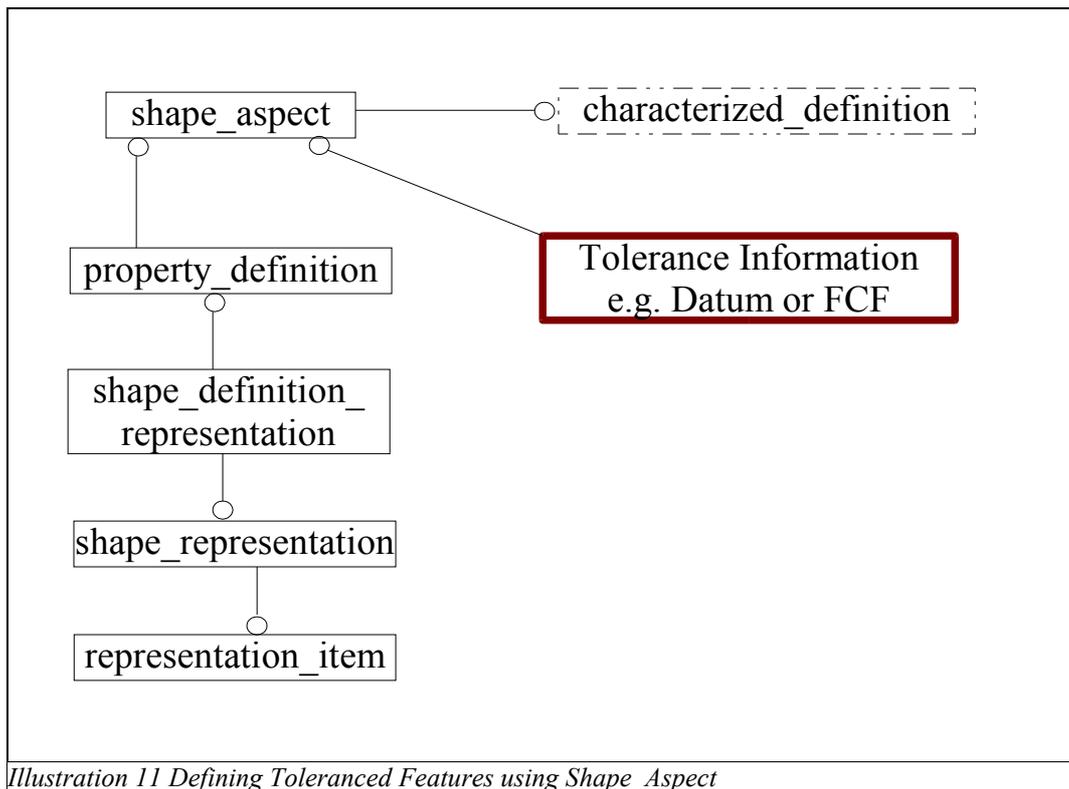


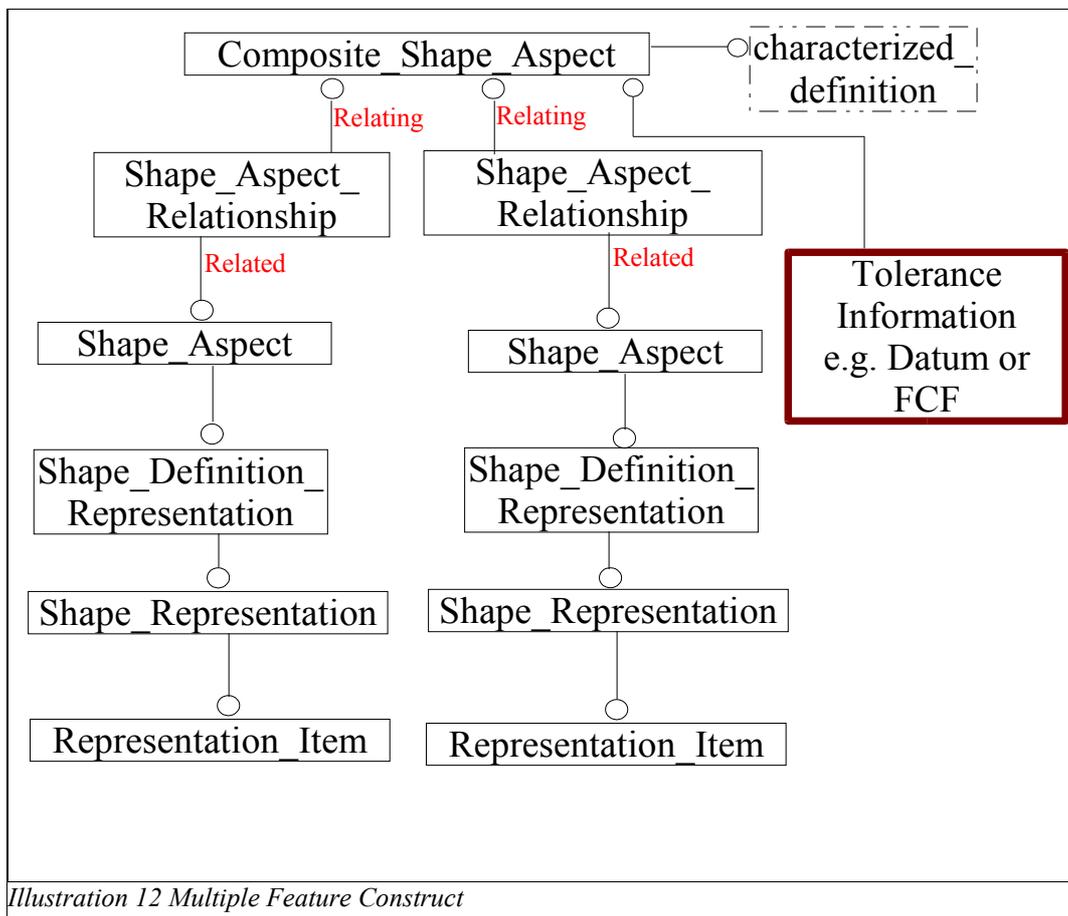
Illustration 11 Defining Toleranced Features using Shape Aspect

Part 21 Example

```
#187=ADVANCED_FACE('feature face',(#181),#186,.T.);
#252=PRODUCT_DEFINITION_SHAPE(",$,#17);
#315=SHAPE_ASPECT(",$,#252,.T.);
#316=PROPERTY_DEFINITION(",$,#315);
#317=SHAPE_REPRESENTATION(",(#187),#246);
#318=SHAPE_DEFINITION_REPRESENTATION(#316,#317);
#320=FLATNESS_TOLERANCE('#320','flatness tolerance',#319,#315);
```

5.3 Associating tolerances with Multiple Features

There are many cases where a Tolerance or Datum needs to be associated with more than one feature, for example, when a face based on the datum geometry has been split by a slot or other feature. In these cases, the associated features are combined under a Composite_Shape_Aspect. Illustration 12 shows the STEP Entities required to implement this functionality. In order to distinguish this case from the “Pattern of Features” case (see below), the name attribute of the composite_shape_aspect shall be set to the string “multiple elements”.



Part 21 Example

```
#10341=PRODUCT_DEFINITION_SHAPE(",$,#17);
#10396=SHAPE_ASPECT(",$,#10341,.T.);
#10397=PROPERTY_DEFINITION(",$,#10396);
#10398=SHAPE_REPRESENTATION(",(#5423),#10335);
#10399=SHAPE_DEFINITION_REPRESENTATION(#10397,#10398);
#10400=SHAPE_ASPECT(",$,#10341,.T.);
#10401=PROPERTY_DEFINITION(",$,#10400);
#10402=SHAPE_REPRESENTATION(",(#5411),#10335);
#10403=SHAPE_DEFINITION_REPRESENTATION(#10401,#10402);
#10404=COMPOSITE_SHAPE_ASPECT('multiple elements',$,#10341,.T.);
#10405=SHAPE_ASPECT_RELATIONSHIP(",$,#10404,#10396);
#10406=SHAPE_ASPECT_RELATIONSHIP(",$,#10404,#10400);
```

5.3.1 Associating tolerances with a Pattern of Features

One of the uses of the Multiple Feature construct is allowing the exchange of Tolerances applied to a pattern of features. In order to differentiate this from the normal case, the string “pattern of features” is used to populate the `composite_shape_aspect.name` attribute.
e.g. #299=COMPOSITE_SHAPE_ASPECT('pattern of features',\$,#246,.T.);

5.4 Implementing DATUMS in a STEP File

This section of the document deals with the implementation of Datums for geometric tolerancing in the STEP File. For a description of the meaning of Datums in the context of tolerancing see 3.6.

In STEP, a Datum is a specialisation of the Shape_Aspect entity. It needs to define the geometric portion of the datum; the “theoretically exact” geometric reference, as well as to attach to the feature of the model which represents the datum. These attachments are shown in Illustration 13 . Note that the Datum Identifier is stored as an Alphanumeric string in the DATUM.identification attribute.

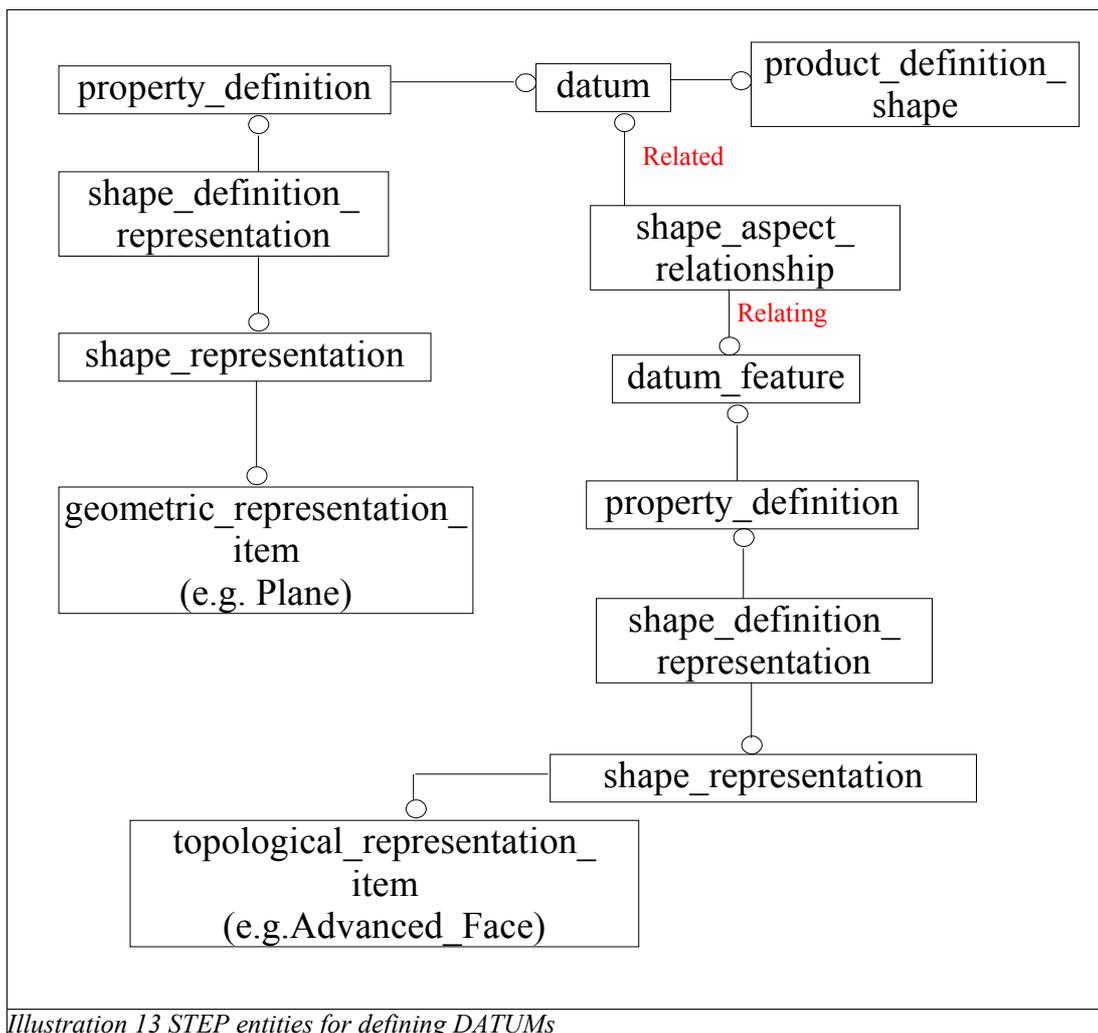


Illustration 13 STEP entities for defining DATUMS

Part21 Example

```
#180=PLANE("#,#179);
#181=ADVANCED_FACE("#,#175),#180,.T.);
#265=DATUM('#,$,#246,.F.,'B');
#266=SHAPE_REPRESENTATION("#,#180),#240);
#267=PROPERTY_DEFINITION('#,$,#265);
#268=SHAPE_DEFINITION_REPRESENTATION(#267,#266);
#269=DATUM_FEATURE("#,$,#246,.T.);
#270=SHAPE_ASPECT_RELATIONSHIP("#,$,#269,#265);
#271=PROPERTY_DEFINITION("#,$,#269);
#272=SHAPE_REPRESENTATION("#,#181),#240);
#273=SHAPE_DEFINITION_REPRESENTATION(#271,#272);
```

5.5 Implementing DATUM TARGETS in a STEP File

A Datum Target is a point, line or limited area of the part surface that is used in the construction of a Datum, when it is not practical to use an entire feature nor a substantial region. A Datum is constructed from one or more Datum Targets, and may be offset from the part surface. The implementation of Datum Targets is done by relating the Placed_Datum_Target_Feature to the Datum it is part of the definition of, by means of a Shape_Aspect_relationship. This is shown in Illustration 14.

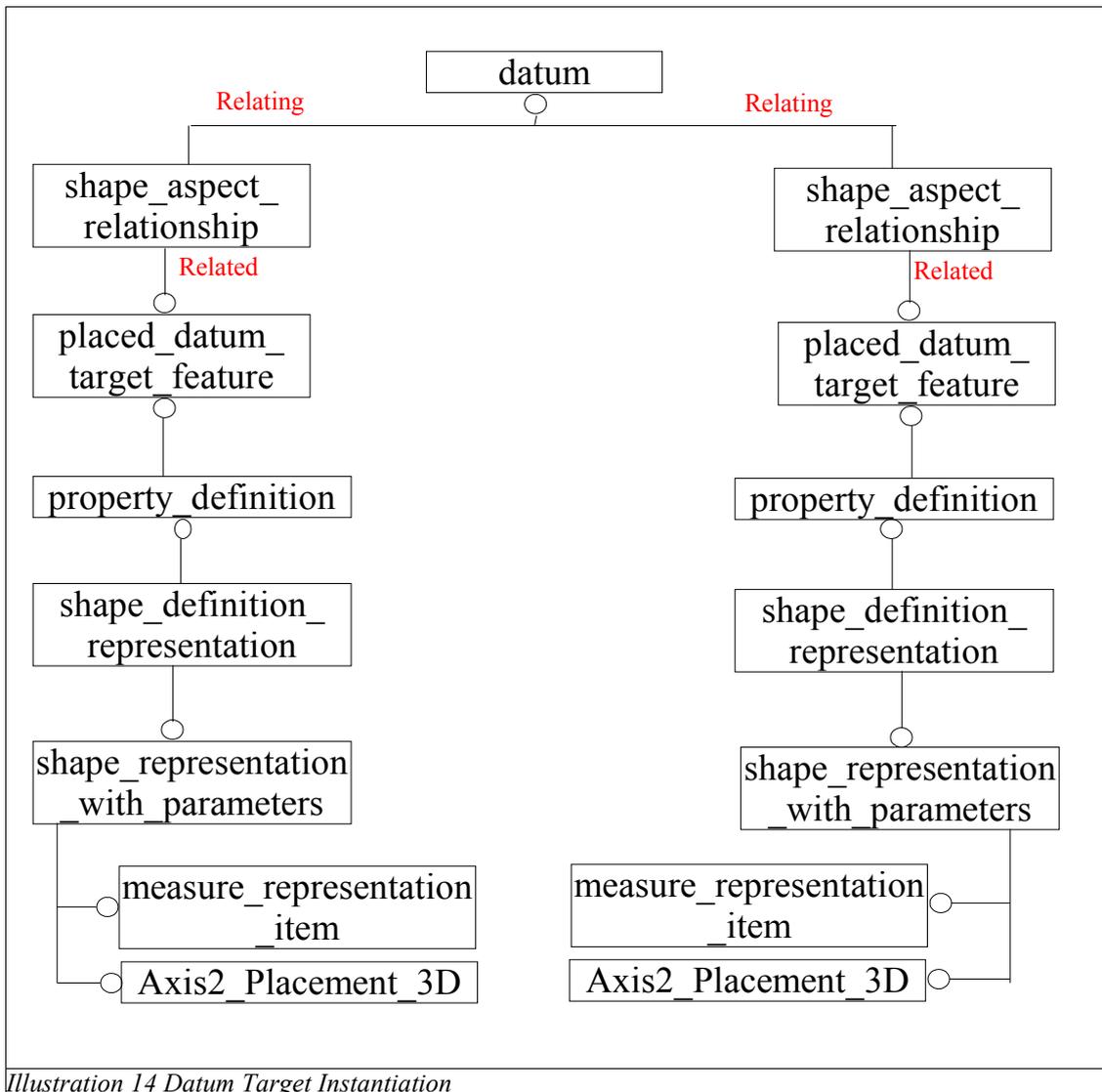


Illustration 14 Datum Target Instantiation

The identification of the Datum Target is stored in the `Placed_Datum_Target_Feature.identification` attribute. Note that for ease of use, it is intended that the full identification of the Datum Target is stored here, i.e. For Datum Target “A1”, store the string “A1”, not just the numeric portion of the target.

Part21 Example

```
#277=DATUM(",$,#252,.F.,'A');
#278=PLACED_DATUM_TARGET_FEATURE(",'circle',#252,.T.,'A1');
#279=SHAPE_ASPECT_RELATIONSHIP(",'datum target',#277,#278);
#280=PROPERTY_DEFINITION(",$,#278);
#281=CARTESIAN_POINT("(1.0,2.0,0.5));
#282=DIRECTION("(1.0,0.0,0.0));
#283=DIRECTION("(0.0,0.0,1.0));
#284=AXIS2_PLACEMENT_3D("#281,#283,#282);
#285=MEASURE_REPRESENTATION_ITEM('target diameter',
    LENGTH_MEASURE(0.75),#6);
#286=SHAPE_REPRESENTATION_WITH_PARAMETERS("#284,#285),#246);
```

```
#287=SHAPE_DEFINITION_REPRESENTATION(#280,#286);
#288=PLACED_DATUM_TARGET_FEATURE(", 'circle', #252, T., 'A2');
#289=SHAPE_ASPECT_RELATIONSHIP(", 'datum target', #277, #288);
#290=PROPERTY_DEFINITION(", $, #288);
#291=CARTESIAN_POINT(", (3.0, 2.0, 0.5));
#292=DIRECTION(", (1.0, 0.0, 0.0));
#293=DIRECTION(", (0.0, 0.0, 1.0));
#294=AXIS2_PLACEMENT_3D(", #291, #293, #292);
#295=MEASURE_REPRESENTATION_ITEM('target diameter',
    LENGTH_MEASURE(0.5), #6);
#296=SHAPE_REPRESENTATION_WITH_PARAMETERS(", (#294, #295), #246);
#297=SHAPE_DEFINITION_REPRESENTATION(#290, #296);
```

5.5.1 DATUM TARGET types

The particular geometric representation that forms the Datum Target is defined by the Shape Representation with Parameters, as well as by the Placed_Datum_Target_Feature.description field. The allowable values for this are :

- point
- line
- rectangle
- circle

Each of these requires particular representation items, complete with pre-defined strings, in the Shape_Representation_with_Parameters that conveys the target area. The table below defines the relevant entities and values required.

Target Type	Representation Entities
Point	A Placement entity (A2P3D) with a name of “orientation”
Line	A Placement entity (A2P3D) with a name of “orientation” Plus A Composite Entity of Measure_Representation_Item AND Length_Measure_With_Unit with a name of “target length” denoting the length along the “Z” axis of the Placement.
Rectangle	A Placement entity (A2P3D) with a name of “orientation” Plus Two Composite Entities of Measure_Representation_Item AND Length_Measure_With_Unit with names of “target length” and “target width”. The length is along the placement “X” axis and the width along the placement “Y” axis, with the placement itself positioned at the centre of the Rectangle.
Circle	A Placement entity (A2P3D) with a name of “orientation” Plus A Composite Entity of Measure_Representation_Item AND Length_Measure_With_Unit with a name of “target diameter”

5.6 Feature Control Frames

This section defines the instantiation requirements for the Feature Control Frames which define the Geometric Tolerance. A Feature Control Frame takes the form shown in Illustration 15 below, in order to define the type and value of the tolerance that has been applied to the feature.

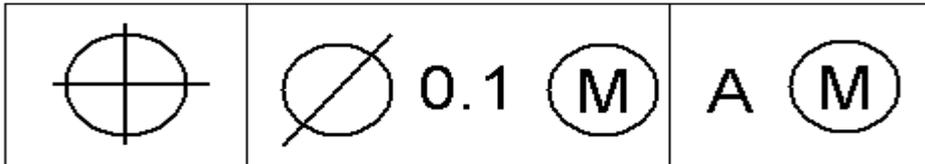


Illustration 15 Feature Control Frame

Reading from Left to right, Illustration 15 shows:

In the first Box, the type of the tolerance, in this case, a Position Tolerance. See Table 4 for the full list of tolerance types supported.

The second box defines the tolerance value itself (0.1) along with any tolerance modifiers.

The third and subsequent boxes specify the datum references along with any datum modifiers.

5.7 STEP Supported Tolerance Types

The following table shows the Geometric Tolerance types supported by STEP. The table lists the Tolerance name, the STEP Entity which is used to represent it, and additionally any restrictions on the number of Datum References that apply in the STEP Schema to that tolerance type.

Tolerance type	STEP Entity	Datums
Angularity	ANGULARITY_TOLERANCE	1, 2 or 3
Circular Runout	CIRCULAR_RUNOUT_TOLERANCE	1 or 2
Circularity / Roundness	ROUNDNESS_TOLERANCE	None
Concentricity / Coaxiality	COAXIALITY_TOLERANCE	1 or 2
Cylindricity	CYLINDRICITY_TOLERANCE	None
Flatness	FLATNESS_TOLERANCE	None
Parallelism	PARALLELISM_TOLERANCE	1 or 2
Perpendicularity	PERPENDICULARITY_TOLERANCE	1, 2 or 3
Position	POSITION_TOLERANCE	None, 1, 2 or 3
Profile of a Line	LINE_PROFILE_TOLERANCE	None, 1, 2 or 3
Profile of a Surface	SURFACE_PROFILE_TOLERANCE	None, 1, 2 or 3
Straightness	STRAIGHTNESS_TOLERANCE	None
Symmetry	SYMMETRY_TOLERANCE	1, 2 or 3
Total Runout	TOTAL_RUNOUT_TOLERANCE	1 or 2

Table 4 Supported Tolerance Types

5.8 Implementing Feature Control Frames

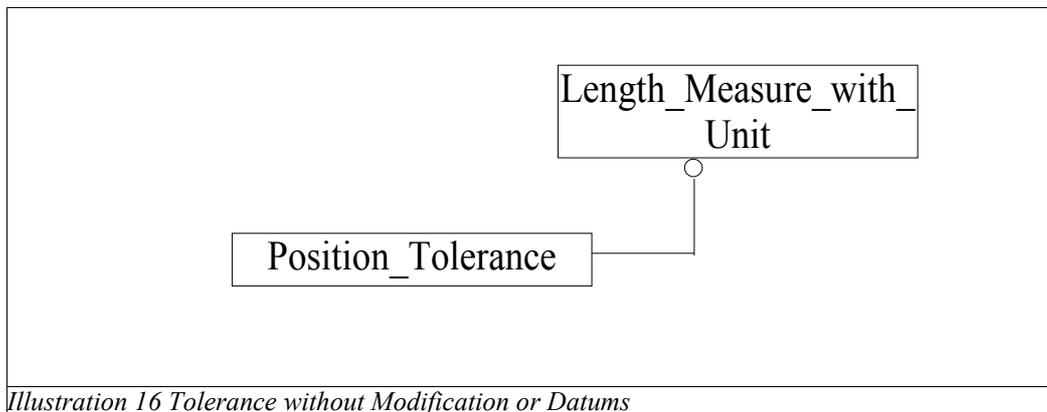
The STEP file implementation of the Geometric Tolerances makes use of the entities shown in the sections below, each section showing the entities required for the different cases, i.e. tolerance without modification or datums, tolerance with datums and modified tolerances. Note that the linkage to the Feature(s) being tolerated is covered in Illustration 11 and Illustration 12.

5.8.1 Geometric Tolerance without Modification or Datums

This case can only occur for the following Tolerance types :

- Roundness
- Cylindricity
- Flatness
- Position
- Profile of a Line
- Profile of a Surface
- Straightness

For this example, we will show a Position Tolerance. The Length_Measure_With_Unit entity is used to convey the Tolerance Value.



Part21 Example

```
#274=LENGTH_MEASURE_WITH_UNIT(LENGTH_MEASURE(0.1),#4);  
#277=POSITION_TOLERANCE('position tolerance',#274,#264);
```

5.8.2 Geometric Tolerance with Modification

A modified geometric tolerance is mapped to STEP as a complex entity, containing the geometric tolerance entity and a Modified_Geometric_Tolerance Entity. The modifier attribute of the Modified_Geometric_Tolerance Entity defines the type of modification applied to the tolerance, and is an enumeration with the following allowable values:

- .MAXIMUM_MATERIAL_CONDITION.
- .LEAST_MATERIAL_CONDITION.
- .REGARDLESS_OF_FEATURE_SIZE.

In the following example, Position Tolerance is used for illustration purposes:

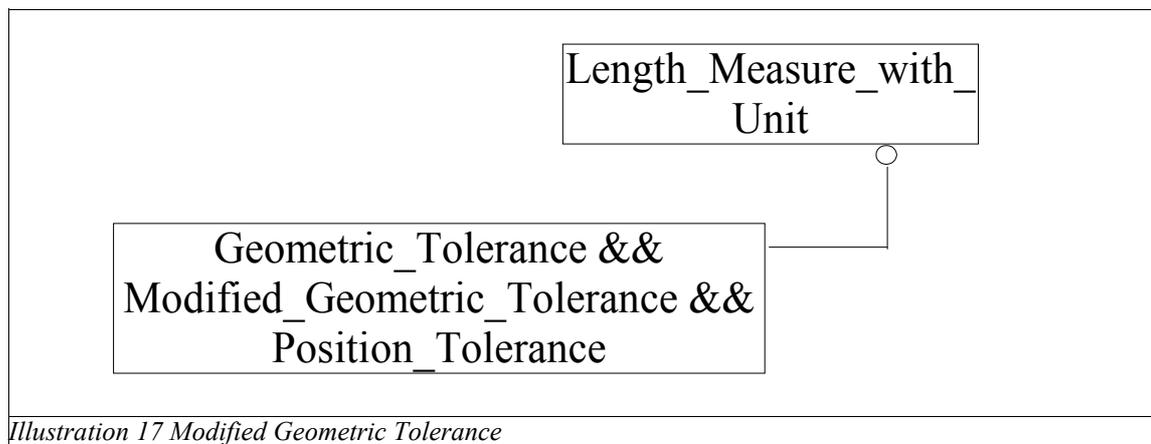


Illustration 17 Modified Geometric Tolerance

Part21 Example

```
#274=LENGTH_MEASURE_WITH_UNIT(LENGTH_MEASURE(0.1),#4);  
#277=(GEOMETRIC_TOLERANCE(",'position tolerance',#274,#264)  
MODIFIED_GEOMETRIC_TOLERANCE(.MAXIMUM_MATERIAL_CONDITION.)  
POSITION_TOLERANCE());
```

5.8.3 Geometric Tolerance with Datums

Most Geometric Tolerances require one or more Datums in order to specify from where the tolerance is measured. In the STEP File this is represented by a complex entity of Geometric_Tolerance and Geometric_Tolerance_With_Datum_Reference, as shown in Illustration 18.

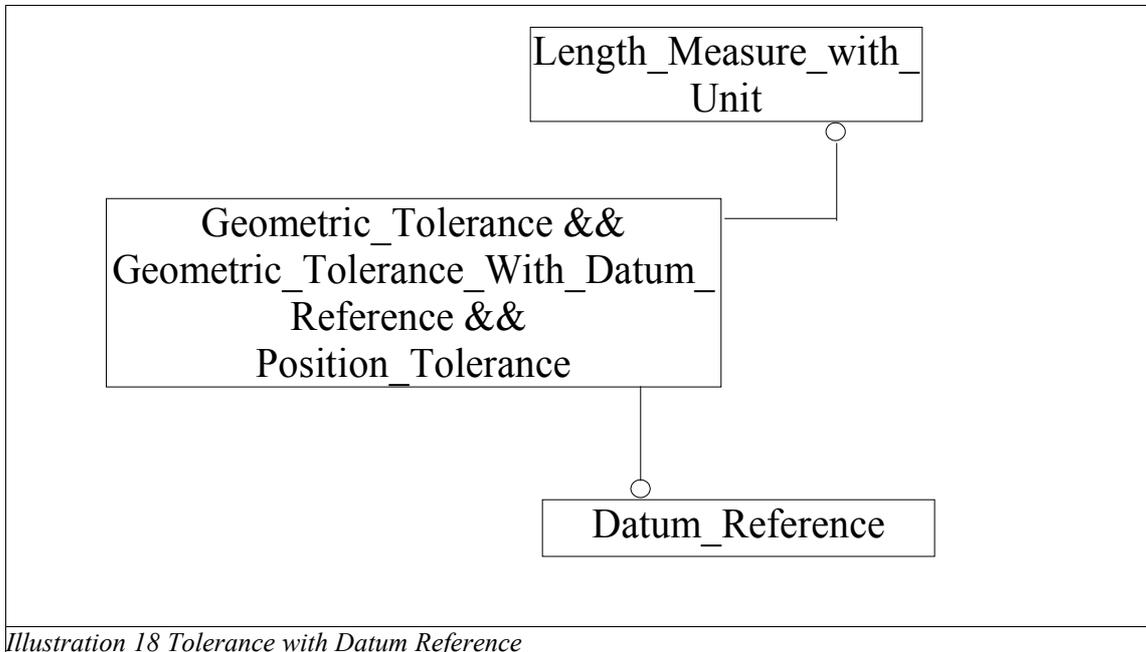


Illustration 18 Tolerance with Datum Reference

In the Feature Control Frame, the datums, if present, appear in a specific order, i.e. Primary, secondary and tertiary. The STEP implementation of the datum_system attribute is as a SET, which is an unordered list. The precedence of the Datum in the tolerance is therefore given by the precedence attribute in the Datum_Reference entity. It is important to note this, as there is no guarantee that the order in the Geometric_Tolerance_with_Datum entity is correct.

Note that if the Datum Reference is modified, as shown in Illustration 15, then the Datum_Reference entity is replaced with a subtype, Referenced_Modified_Datum.

Note that it is possible to have both datum references and modified tolerances. This is achieved in the STEP file by extending the complex entities to include both mechanisms.

Part21 Example

```
#274=LENGTH_MEASURE_WITH_UNIT(LENGTH_MEASURE(0.1),#4);
#275=DATUM_REFERENCE(1,#246);
#276=DATUM_REFERENCE(2,#255);
#277=(GEOMETRIC_TOLERANCE('position tolerance',#274,#264)
GEOMETRIC_TOLERANCE_WITH_DATUM_REFERENCE((#275,#276))
MODIFIED_GEOMETRIC_TOLERANCE(.MAXIMUM_MATERIAL_CONDITION.)
POSITION_TOLERANCE());
```

5.8.4 Common or Multiple Datums

It is possible for two Datums to have equal importance when specifying the Datum References. A Feature control frame with this condition is shown in Illustration 19.

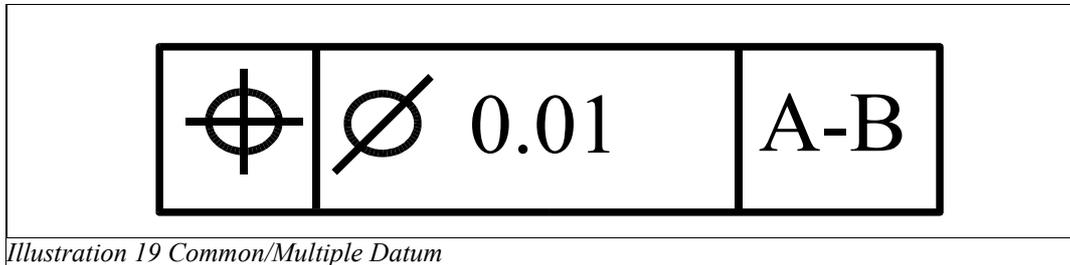


Illustration 19 Common/Multiple Datum

In the STEP File, this is implemented by using the Common_Datum entity, which is a subtype of both Datum and Composite_Shape_Aspect. This entity is used to collect the “equal weight” datums together and provide a target for the Datum_Reference SET. An instantiation of this is shown in Illustration 20.

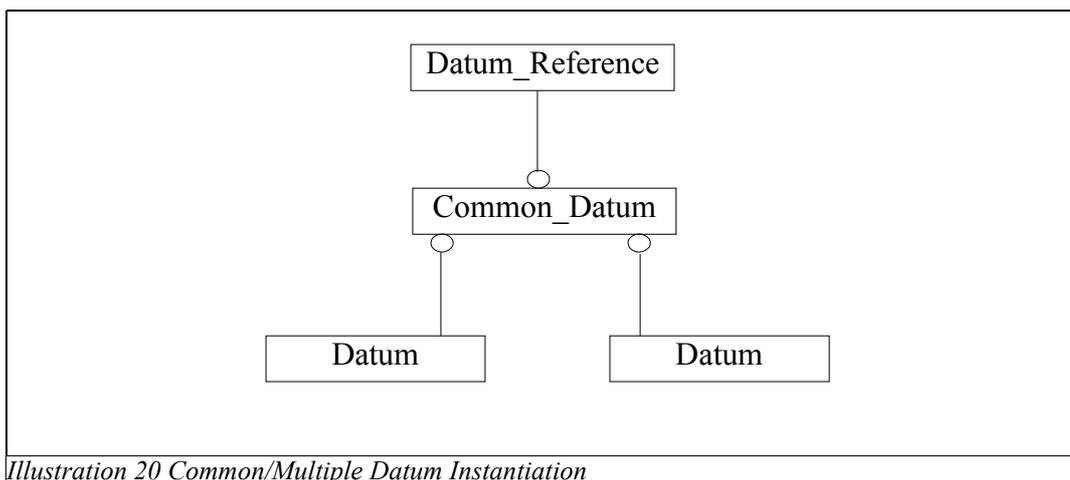


Illustration 20 Common/Multiple Datum Instantiation

As the Common_Datum entity is a subtype of DATUM, it requires a String for its identifier. Although the Identifier can be derived from the Datums that it is a multiple of, it is recommended that a string containing the identifiers from both of these, separated by a hyphen, is used, e.g. If the common datum is made up of “A” and “B”, then its identifier will be “A-B”.

5.8.5 Composite Geometric Tolerances

Some tolerances have multiple requirements as represented by a multiple frame tolerance control frame whose visual representation is shown in Illustration 21.

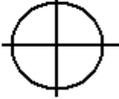
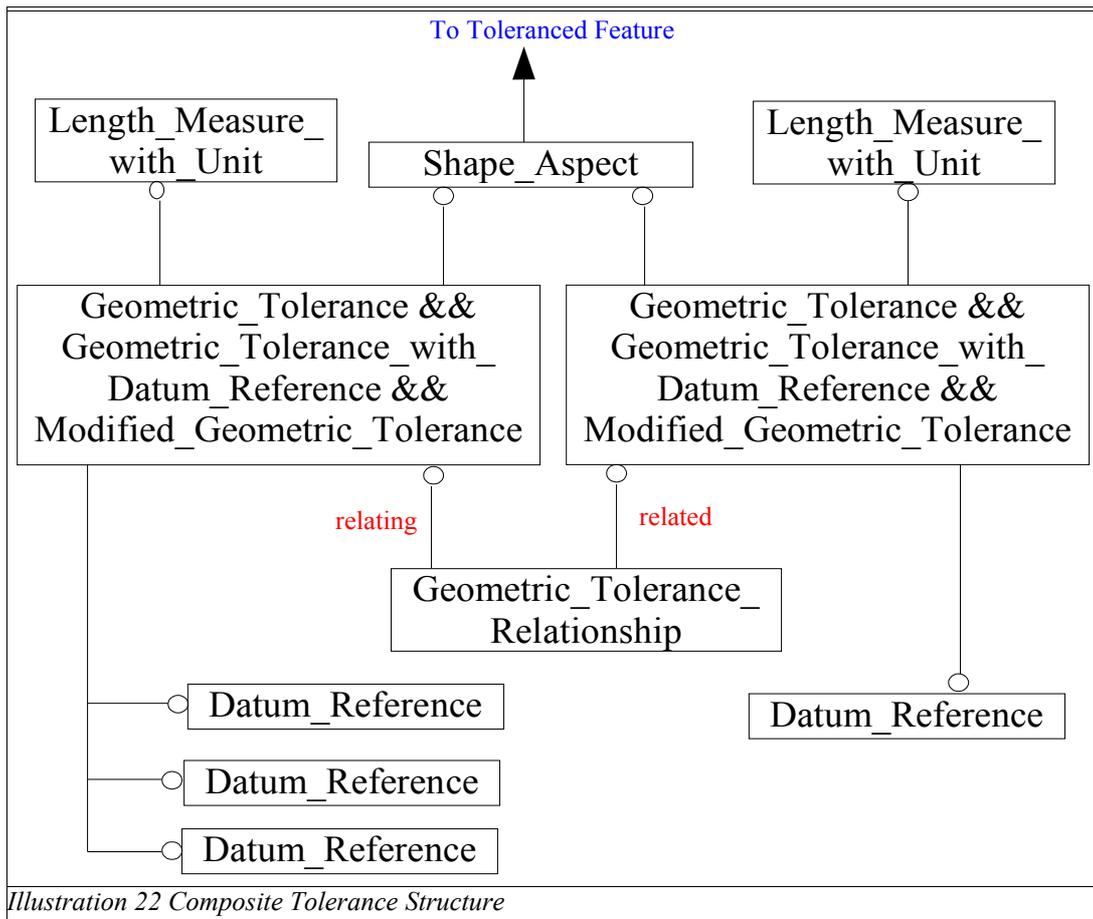
	 0.1 (M)	A	B	C
	 0.01 (M)	A		

Illustration 21 Composite Geometric Tolerance

This is implemented by creating the appropriate `geometric_tolerance` entity for each frame. The `geometric_tolerance` entities are then related via a `geometric_tolerance_relationship`. The description attribute of the `geometric_tolerance_relationship` entity shall contain the value “Composite Tolerance”. See Illustration 22 for the structure of the tolerance depicted in Illustration 21.

In the `geometric_tolerance_relationship` entity, the upper frame is the “relating” reference and the lower frame is the “related” reference. For tolerances with more than two frames, multiple `geometric_tolerance_relationship` entities are used to relate the frames together. For example, a tolerance with three frames would require two `geometric_tolerance_relationship` entities: one relating the top (“relating”) and middle frames (“related”), the second relating the middle (“relating”) and the bottom (“related”) frames. These relationships will allow the receiver to reconstruct the semantic relationships as well as the visual representation.



Part21 Example

```

#307=SHAPE_ASPECT(",$,#223,.T.);
#308=PROPERTY_DEFINITION(",$,#307);
#309=SHAPE_REPRESENTATION(",(#185),#217);
#310=SHAPE_DEFINITION_REPRESENTATION(#308,#309);
#311=LENGTH_MEASURE_WITH_UNIT(LENGTH_MEASURE(0.2),#4);
#312=DATUM_REFERENCE(1,#264);
#313=(GEOMETRIC_TOLERANCE(",'surface profile tolerance',#311,#307)
      GEOMETRIC_TOLERANCE_WITH_DATUM_REFERENCE((#312))
      SURFACE_PROFILE_TOLERANCE());
#314=LENGTH_MEASURE_WITH_UNIT(LENGTH_MEASURE(0.1),#4);
#315=SURFACE_PROFILE_TOLERANCE(",'surface profile tolerance',#314,
      #307);
#316=GEOMETRIC_TOLERANCE_RELATIONSHIP(",'composite tolerance',#313,#315);

```

5.9 Feature of Size

CAD systems can have a “Feature of Size” associated with the Geometric Tolerancing informations, especially on the position tolerance of a Hole or Pin. In this case, a mix of Geometric and Dimensional Tolerances are used to exchange this information, by mapping the “Feature of Size” as a Dimensional Size in the STEP File (See 4.2.1) applied to the same Shape Aspect as the Geometric Tolerance. An example of this is shown in Illustration 23.

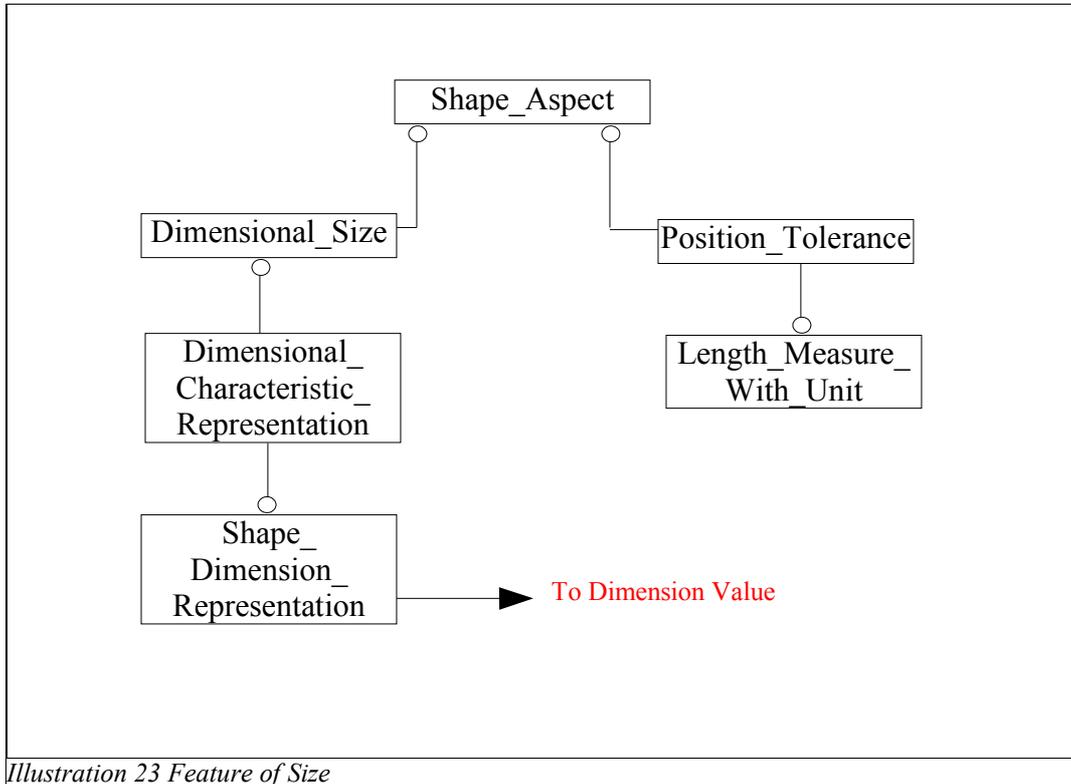


Illustration 23 Feature of Size

Part21 Example

```
#342=SHAPE_ASPECT(",$,#252,.T.);
#343=PROPERTY_DEFINITION(",$,#342);
#344=SHAPE_REPRESENTATION(",(#229),#246);
#345=SHAPE_DEFINITION_REPRESENTATION(#343,#344);
#346=(LENGTH_MEASURE_WITH_UNIT()
MEASURE_REPRESENTATION_ITEM()
MEASURE_WITH_UNIT(POSITIVE_PLANE_ANGLE_MEASURE(1.05),#6)
REPRESENTATION_ITEM('upper limit'));
#347=(LENGTH_MEASURE_WITH_UNIT()
MEASURE_REPRESENTATION_ITEM()
MEASURE_WITH_UNIT(POSITIVE_PLANE_ANGLE_MEASURE(0.95),#6)
REPRESENTATION_ITEM('lower limit'));
#348=(LENGTH_MEASURE_WITH_UNIT()
MEASURE_REPRESENTATION_ITEM()
MEASURE_WITH_UNIT(POSITIVE_PLANE_ANGLE_MEASURE(1.0),#6)
```

```
REPRESENTATION_ITEM('nominal value');  
#349=SHAPE_DIMENSION_REPRESENTATION(",(#346,#347,#348),#246);  
#350=DIMENSIONAL_SIZE(#342,'diameter');  
#351=DIMENSIONAL_CHARACTERISTIC_REPRESENTATION(#350,#349);  
#352=LENGTH_MEASURE_WITH_UNIT(LENGTH_MEASURE(0.08),#6);  
#353=ROUNDNESS_TOLERANCE(",'circularity tolerance',#352,#342);
```

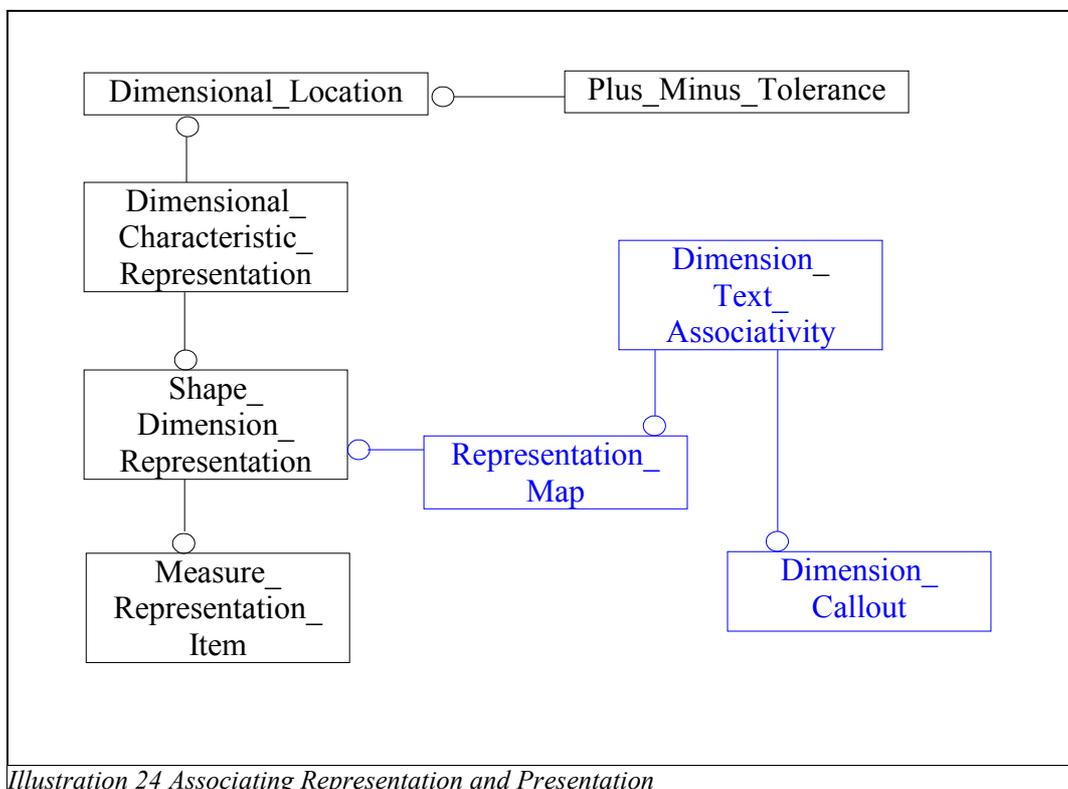
6 Presentation of Dimensional Tolerances

This section deals with the additional entities required to take a dimensional tolerance and associate it with a visual display of that tolerance in the 3D model workspace. Note that currently only the 3D model workspace is specified, mainly because of the lack of 2D implementations, but there should be very little additional work on top of this specification to associate the display of the tolerance with a specific 2D view of the model, once the 2D viewing mechanism has been implemented.

The information and mapping in this section has been drawn from a number of documents, notably the “Recommended Practices for model viewing, basic drawing structure and dimensions”, dated 5th October 1999 and co-authored by Rogerio Barra, Markus Hauser, Bettina Neuhof and Linus Polikaitis. This Document is available at the Cax-IF website (www.cax-if.org or www.cax-if.de). One Major difference is the removal of the Structured_Dimension_Callout entity from these practices. This was done as there did not appear to be any semantic value added by this entity to the exchange.

6.1 Relating the Presentation with the Representation

The Representation of the Dimensional Tolerance is covered in Section 4. The full description of Presentation will be covered in a later section, however, for the purposes of this section, it is enough to know that the associated Presentation information for the Representation is collected by a Complex Instantiation of one of the subtypes of the Dimension_Callout Entity, as shown in Illustration 24. For this example, a Linear Dimension with a Plus-Minus tolerance is used.



The Dimension_Text_Associativity entity is a subtype of Mapped_Item, hence the links to Representation_Map and Dimension_Callout, and also a subtype of Text_Literal, being able to hold the textual representation along with placement information and alignment. For presentation in a 3D context, the placement should be an Axis2_Placement_3D, with the text positioned in the X-Y plane, as for 3D associative text. Note that any of the subtypes of Text_Literal are required, e.g. Text_Literal_with_Blanking_Box, then this entity will have to be instantiated as a complex entity.

The Dimension_Callout will be a complex instantiation of two subtypes of Draughting_Callout, the particular subtype of draughting_callout which defines the type, such as linear_dimension, and Draughting_Elements. The dimension curve and witness lines are also defined in this Entity (See Section 6.3).

6.2 Presenting the Tolerance

The tolerance value is presented using the STEP entities shown in Illustration 25, attached to the Dimension_Callout defined in Illustration 24.

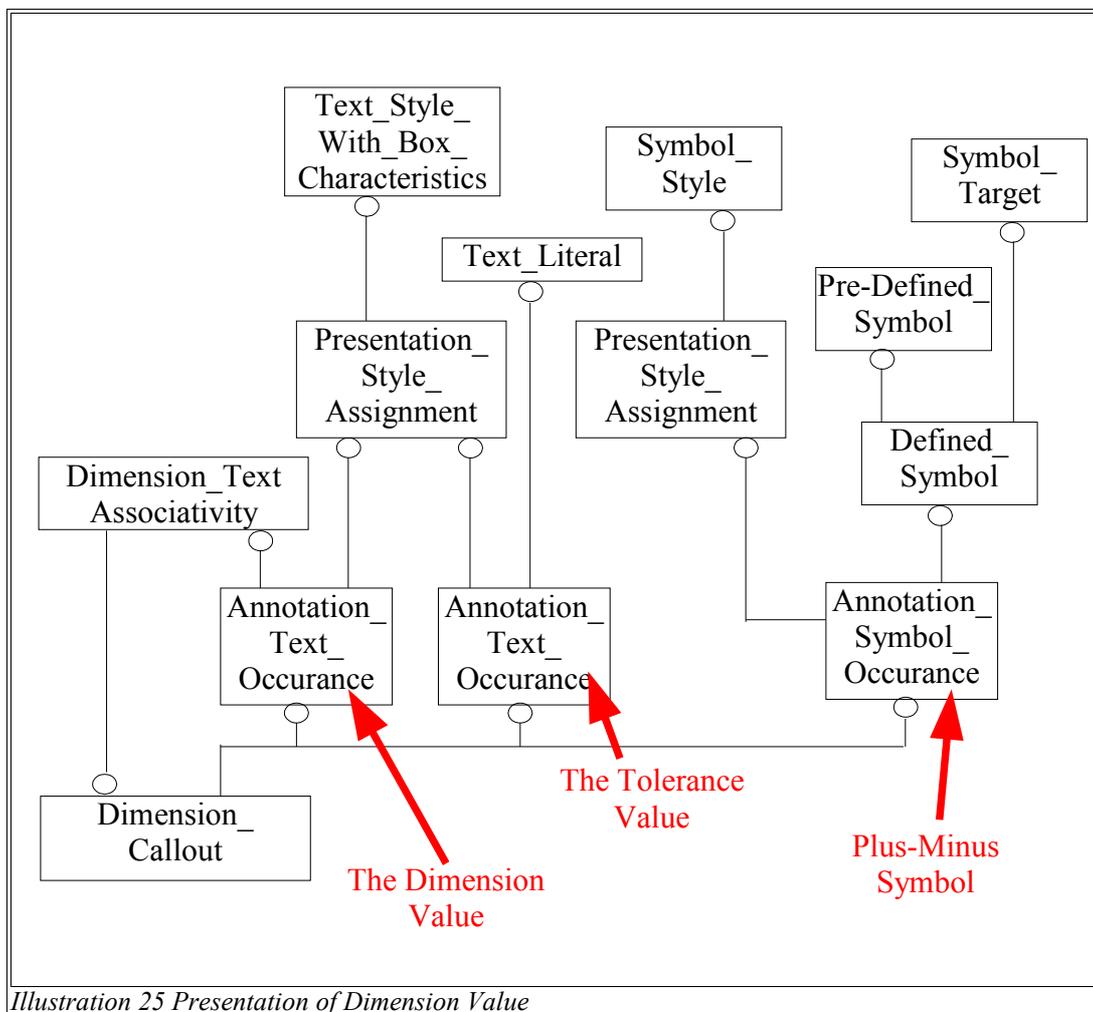


Illustration 25 Presentation of Dimension Value

The Dimension_Callout entity gathers together the components of the Dimension value

string, i.e the value, the tolerance and the symbol separating them. The linkage of the value back to the representation of the Dimension is done by attaching to the “Text_Literal” part of the Dimension_Text_Associativity entity. There is no equivalent linkage for the Tolerance value, although it is possible to navigate through the entities to pick up the true value. In fact, the entire Value and Tolerance string can be derived from the Representation, without the need for the textual presentation entities. However, if the receiving system does not support Dimensional tolerances, then at least it can make use of the textual information to present it on the screen to the user.

6.3 Presenting the Associated Graphics

The other part of visualising the Dimension is to apply the dimension and witness lines for the presentation. These do not exist in the Representation of the dimension, and are physically placed in the View or 3D Model space, without any linkage to the defining geometry of the part by specialisations of the Draughting_Callout entity.

An instantiation diagram of this associated geometry is shown in Illustration 26. Note that in this illustration, the Linear_Dimension Entity is the same entity as the draughting_callout in Illustration 25.

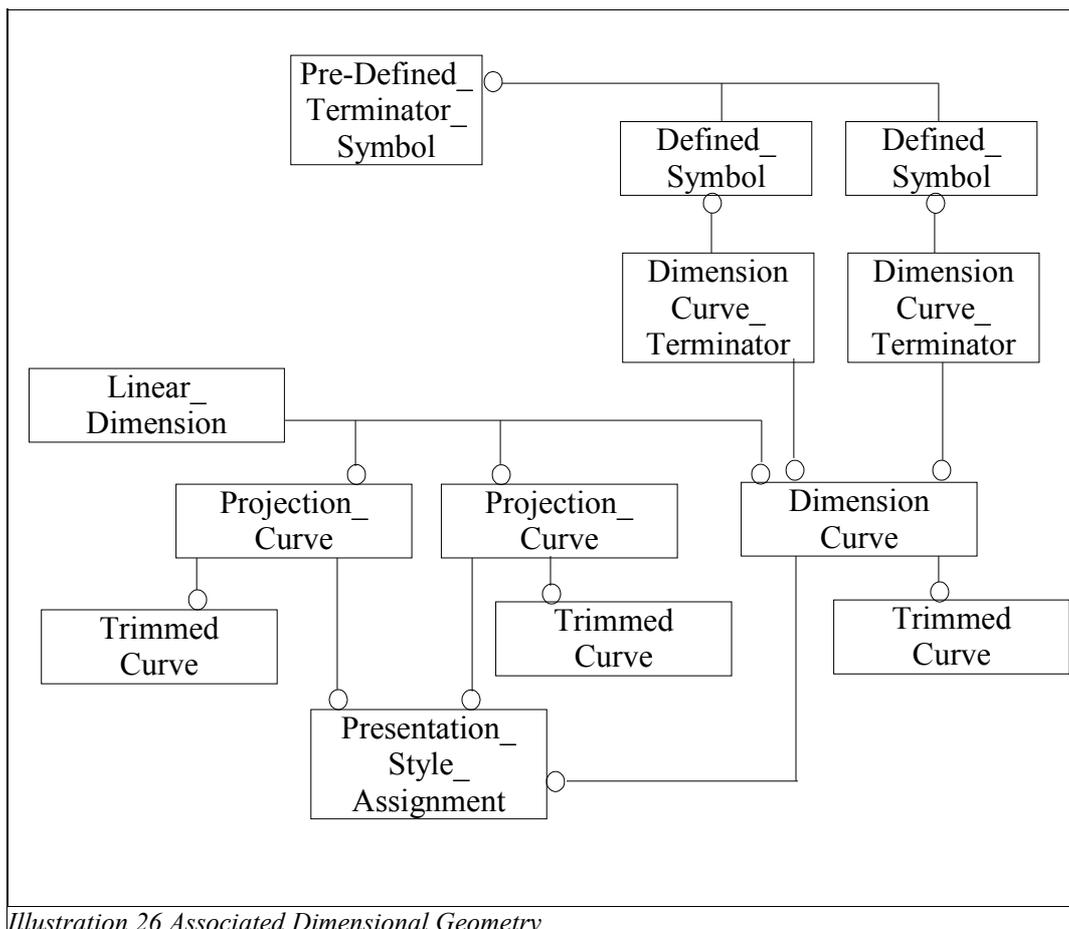


Illustration 26 Associated Dimensional Geometry

7 Presentation of Geometric Tolerances

This sections deals with the presentation of Geometric Tolerances, and their attachment to the relevant “represented” GD&T information. This is very much a first pass as defining this information and is heavily based upon the exchange of 3D Associative Text, as defined in “Recommended Practices for 3D Associative Text – Release 4 – January 13th 2000”, available as a download from www.cax-if.de or www.cax-if.org .

It is envisaged that both this document and the 3D Associative Text Recommended Practices will evolve as further implementations of both come into being. The intent is to provide this as a basis for further improvement, in such a way that the structures and methods defined form the basis for those improvements and are to be left as is, such that any files following these procedures are not rendered obsolete or unconfornant in the future.

I would like to acknowledge the considerable input made to this section of Lothar Klein and Giedrius Liutkus of LKSoft GmbH. Who have both put in a great deal of effort to lay these foundations and ensure future interoperability.

7.1 Geometric Tolerances as 3D Text

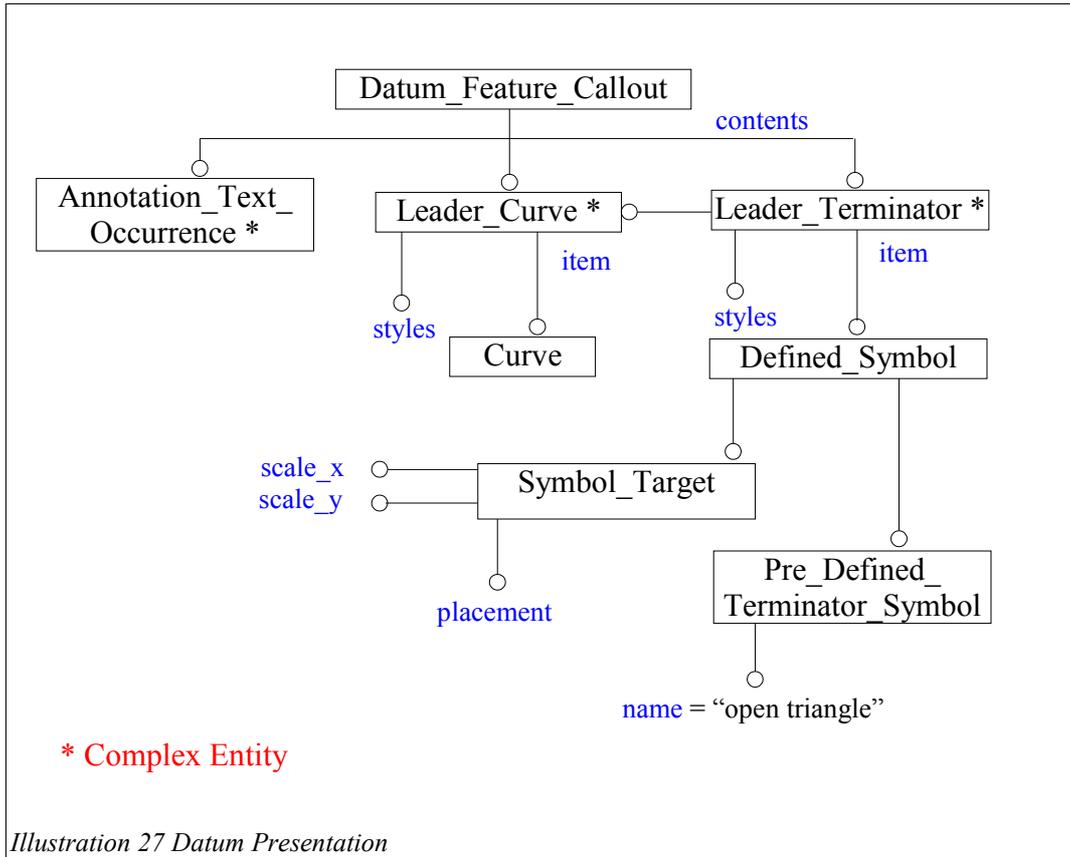
The method for exchanging the Presentation of Geometric Tolerance Data is to leverage the 3D Associative text capability developed in STEP and tested by the Cax-IF. The plus points of this approach is that either the Representation or the Presentation or both can be processed by a post-processor, depending on the capabilities of the receiving system. This means for fully intelligent systems, such as UGNX and CATIAV5, both the represented and presented GD&T can be exchanged, whereas for viewing systems where the representation is not stored, such as JT, the data intent can still be exchanged.

In order to simplify the STEP files containing this data, some changes and enhancement to existing STEP Modules will have to be implemented. This document will assume that these changes are already in place, but will draw the attention of the reader to them at the appropriate points.

7.1.1 DATUM Labels

The Presentation of Datum labels is quite straightforward, using the methods defined for 3D Associative text. Illustration 27 shows the STEP entities involved in the instantiation of a Datum label.

Note that this is the first instance of change to the existing standard, as the `pre_defined_terminator_symbol` entity does not at present contain the concept of a “datum triangle” and will require this to be made available. The alternative is to define this symbol as the set of curves which define it's shape, which would require more STEP entities to be instantiated.



In xxxx, the Leader_Curve entity is a complex instantiation of :

- annotation_curve_occurrence
- annotation_occurrence
- draughting_annotation_occurrence
- leader_curve
- representation_item
- geometric_representation_item
- styled_item

This gives it the ability to style the leader curve as well as a link to the geometric curve defining the leader in 3D space. The EXPRESS definition of annotation_curve_occurrence specifies that the styled_item should be a curve. This would allow the leader line to be one of the unbounded types of curve, which is meaningless. For implementations of this functionality, it is required that the underlying geometric representation of the leader be a subtype of the Bounded_Curve entity. It is also suggested that this restriction be implemented in the standards, perhaps as an additional restriction to the draughting_annotation_occurrence entity in AIC 504, as this already contains restrictions for Text and Symbols.

The Leader_Terminator entity is also a complex instantiation, in this case of :

- annotation_occurrence
- annotation_symbol_occurrence
- draughting_annotation_occurrence
- geometric_representation_item
- leader_terminator
- representation_item
- styled_item
- terminator_symbol

For the terminator symbol, it's position in the model is defined by the placement attribute of the symbol_target entity. What is currently unclear is how to define the size of the terminator symbol, as there is no mechanism to define a box size for the symbol as there is with text. In order to define this, which is a requirement for CAD systems, I suggest that the height and width of the symbol be stored in the scale_x and scale_y attributes respectively, defined in the unit system of the view on the model (e.g. as a ratio measure applied to “one” unit).

The annotation_text_occurrence entity is a complex entity, made up of :

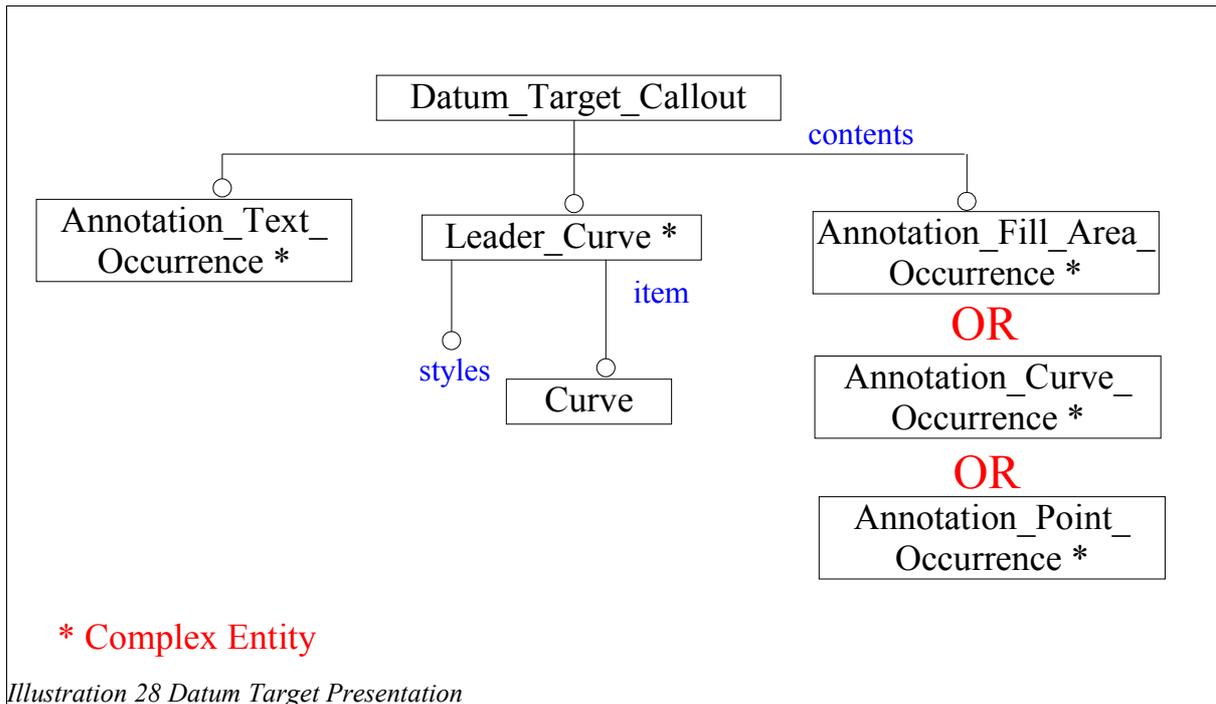
- annotation_occurrence
- annotation_text_occurrence
- draughting_annotation_occurrence
- geometric_representation_item
- representation_item
- styled_item

Where the item to be styled is the composite_text or text_literal.

One major problem from the viewpoint of GD&T presentation is the inability to simply define the box presented as a boundary around the Datum letter that is the text. One way of doing this is to define the text_literal as the text_literal_with_delineation subtype. Part 46 states that the AP can define the delineation types, and I propose that we use this for GD&T. In the case of a Datum, I would suggest a delineation of “rectangular frame”. This approach is in keeping with the approach taken within the CAD systems.

7.1.2 DATUM Targets

Datum Targets are presented in a similar way to Datum Features. The additional elements of the Datum Target above and beyond the Datum Feature are the Target Symbol (i.e. The circular boundary around the target name and value) and a representation of the target area. The instantiation for the Datum Target is shown in Illustration 28.



Depending upon the type of target area. The Datum_Callout Target will contain either an Annotation_Point_Occurrence (for a point target), an Annotation_Curve_Occurrence (for a line target) or an Annotation_Fill_Area (for a rectangle or circle target area).

Similarly to the Datum Feature, the Annotation_Occurrence entities are instantiated as complex entities. Also the Text_Literal or Composite_Text styled by the Annotation_Curve_Occurrence may have associated curves, defining the Datum Target Symbol itself. Some CAD systems do not have access to these curves, and thus must make the decision to draw the Target symbol based on the Callout type.

7.1.3 Feature Control Frames

7.2 Associating GD&T Presentation and Representation

The link between the Presented GD&T and the intelligent Representation is crucial. This gives the systems capable of handling the intelligent GD&T the information required both to maintain the intelligence, and to place the presentation of that data on the screen for the user to see.

The method proposed for instantiating this relationship is the same method as used in 3D annotation to associate the callout with a connection point on the geometry (i.e. The associated geometry for the text). This method defines a shape_aspect entity to contain the

Representation shape_aspect subtype is Placed_datum_target_Feature, rather than Datum.

For the associativity between the Geometric Tolerance Feature Control Frame, and the 3D Annotation set defining the presentation of this, the structure is slightly different. This is due to Geometric_Tolerance not being a subtype of Shape_aspect, and so not being a valid participant in the shape_aspect associativity relationship. However, the Geometric Tolerance does need to have a shape aspect associated with it in the Representation, in order to define the Feature being toleranced, and so for the Representation to Presentation associativity, we shall use this entity. This leads to an instantiation as shown in Illustration 30.

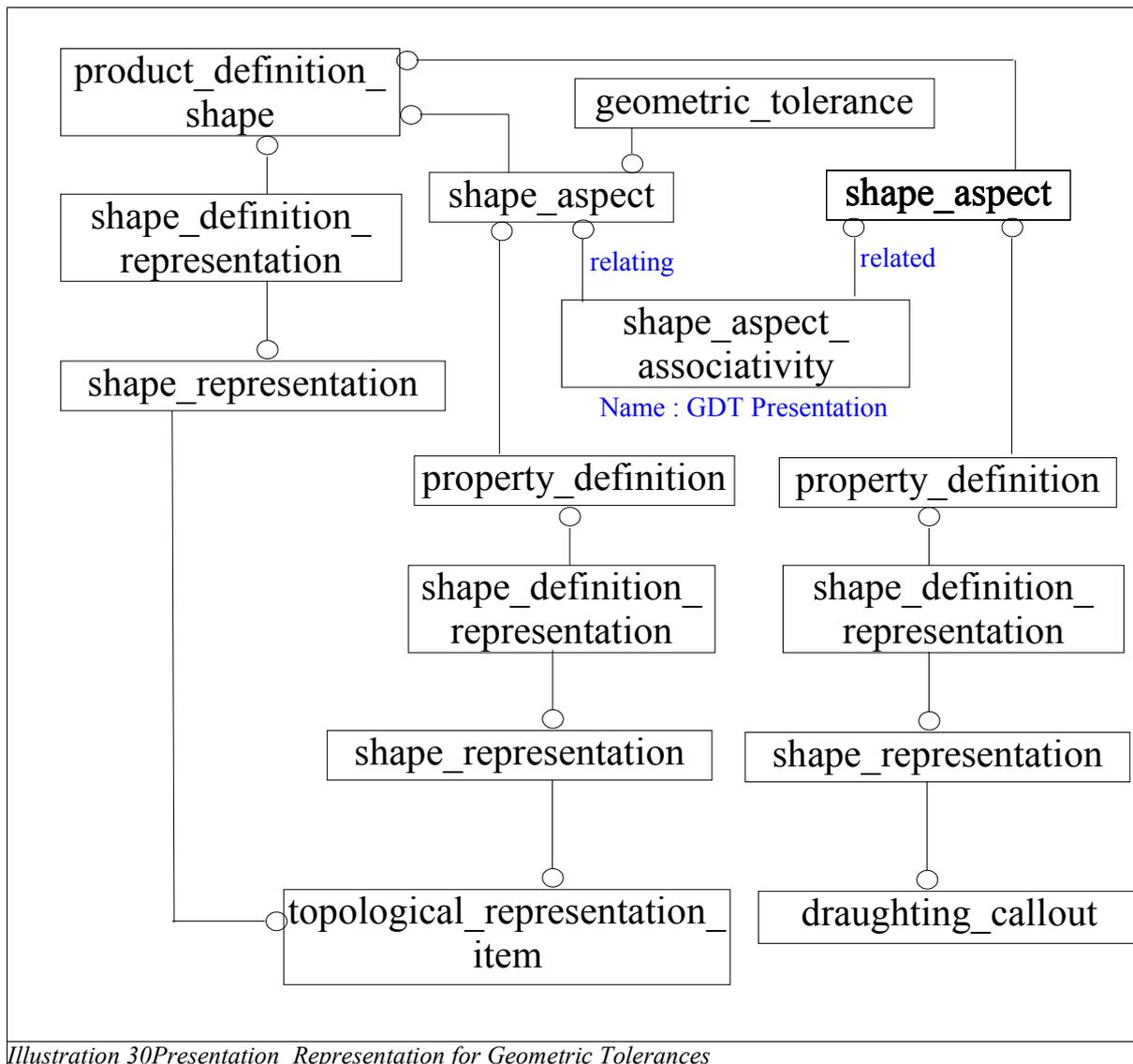


Illustration 30 Presentation Representation for Geometric Tolerances

8 Conclusions

In this Document, the practices for Representation of GD&T and the Representation and Presentation of Dimensional Tolerances is complete and implementable.

The Presentation of GD&T is based on 3D Annotation, which will require further work, not just for GD&T, but also in order to be able to exchange annotations in general. This work will need to determine a good way of encoding the GD&T presentation, as well as working up practices for different 3D views on the model and associating the relevant annotation with those views.

The resolution of these challenges is currently being undertaken by the Cax-IF, and it is envisaged that the sandbox implementations that are done to prove out the proposed methods, will eventually become full blown production capability.