A Dimension and Tolerance Data Model for Concurrent Design and Systems Integration

Shaw C. Feng
Mechanical Engineer[1]
Factory Automation Systems Division
National Institute of Standards and Technology
Gaithersburg, MD 20899

Yuhwei Yang
President
Product Data Integration Technologies, Inc.
Long Beach, CA 90806

ABSTRACT

Dimensions and tolerances are critical engineering design information for defining shape requirements of manufactured parts. As technologies for new design analysis and manufacturing planning are being developed, they must be seamlessly integrated into a computer aided product development environment. A data model is an effective technique to define the shareable semantics that are essential to the success of data communication in an integrated environment. This paper introduces a data model for dimensions and tolerances. The model is a component of an overall product data model. The dimension and tolerance data model has three major components: dimension schema, tolerance schema, and datum and shape aspect schema. These schemas specify data resources and structure for describing dimension and tolerance characteristics of products. Based on this model, descriptions of dimensions and tolerances of products can be communicated between tolerance-related software application systems.

KEY WORDS: STEP; computer aided design; data exchange; data model; dimension model; dimension representation; dimensioning and tolerancing; integration; geometric tolerance; product data; tolerance model; tolerance representation.

INTRODUCTION

Data modeling is an essential activity towards the development of a computer integrated design and manufacturing environment. A data model provides a semantic structure to describe data requirements for any enterprise activities. The International Organization for Standardization (ISO) Technical Committee 184 (TC184), Subcommittee 4 (SC4) has been developing a Standard for the Exchange of Product Model Data (STEP, ISO 10303). STEP will provide a product data model that describes the data required in design, manufacturing, and enterprise activities. This data model also provides the data required as a foundation for product performance characteristics simulation and analysis activities.

Dimensions and tolerances define aspects of the shape of a product and the allowable shape variations from the ideal shape of manufactured parts. The ideal shape of a product is usually

[1] This paper was written as part of U.S. Government official duty and is not subject to copyright.

captured by solid models generated in computer-aided design (CAD) systems or on engineering drawings. Current solid modelers have severe deficiencies in incorporating tolerancing data [1]. Vendor implementation of a dimension and tolerance (D&T) data model can enable D&T information to associate with solid models of part design and the exchange of D&T data in a heterogeneous computer systems environment.

As technology moves into an integrated design and manufacturing environment, specific needs are felt for a dimension and tolerance data model. This model must:

- Convey part functional requirements from design to manufacturing in a computer-compatible format;
- Define a universal (common) semantic and format for the representation and exchange of dimensioning and tolerancing information;
- Be an integral part of product data standardization;
- Associate dimension and tolerance data with CAD to replace drawings on paper; and
- Provide a computer-interpretable data format for using D&T data and communicating them among CAD, CAM, tolerance analysis, tolerance synthesis, assembly analysis, computer-aided process planning, dimensional inspection planning, and assembly planning.

To support manufacturing applications, integrated data models that capture product-, process-, and enterprise-related information can establish the required core data resources in a neutral format for the development of a common manufacturing database. This database ensures the integration of product life cycle manufacturing applications and systems. This paper describes a data model which was developed to provide a foundation for the development of STEP Part 47: *Industrial automation systems and integration - Product data representation and exchange - Integrated generic resources - Shape variation tolerances* [2].

APPLIED DATA MODELING METHODS

Data modeling methods provide fundamental rules and guidelines in the development of a data model using modeling tools. These methods should provide a full range of concepts of model building that will ensure a precise model with shareability, extensibility, and logical data structure [3]. The development process includes establishing the purpose and scope of the data model, identifying the sources of data, defining data requirements, creating a data model, and defining the criteria for the quality assessment of the model.

The role of dimension and tolerance data model in design and manufacturing

A data model used in design and manufacturing is a representation of rigorously defined concepts, relationships, rules, and operations needed in design and manufacturing planning. The concepts should capture the properties and characteristics of the design and manufacturing processes. Furthermore, the representation must be human understandable and computer interpretable. Dimension and tolerance information is a critical component in product design and manufacturing. In order to properly specify and associate dimension and tolerance data, a dimension and tolerance data model must capture and relate all the necessary elements defining the geometric form of a product shape.

The process of data modeling

Data model development usually starts from a need of modeling a process for improving the productivity of an enterprise. The model developers first create a set of data requirements through studying current, available processes and their data. Based on the requirements, the developers then abstract information and document the data abstraction, which is followed by the selection of a data modeling language and the creation of a draft data model. This draft data model should be validated. In this validation stage, the model is evaluated against a set of criteria to ensure model quality.

In the data collection process, the developers collect data through interviewing enterprise subject area experts and application system users. This data collection activity may include visits to manufacturing sites, reviews of existing related industrial standards, studies of related research results, and discussions to resolve technical issues. During the data abstraction process, developers select a modeling language and tools that provide a formal representation of the model and facilitate the documentation of the model. The abstraction process is to transform the data semantic generalization, specialization, classification, aggregation, and association into modeling constructs. For model validation, the process includes elements of validation and quality assessment of the model, such as setting criteria and running tests. Example criteria for a quality data model are structure shareability, built-in extensibility, application system independent logical data structure, and usage of proper abstraction. A suite of test cases that represent industry requirements and experiences should be selected to test the model.

A data model does not stand alone. The relationships between the data model and the processes where the data is used as input, output, or controls are necessary information for using the data model. Many other models, such as process models and behavior models, although closely related, are not within the scope of discussion of this paper.

A data modeling language - EXPRESS

A data modeling language specifies the formal syntactics and semantics for defining data elements and the relationships between the elements. Most data modeling languages that are available today are used to describe a static view of data semantics. At least three data modeling languages are available and well-known; they are IDEF1x [4], EXPRESS [6], and NIAM [7]. IDEF1x provides a method to document data used in processes such as design process and manufacturing process. IDEF0 [5] defines activities and the associated information which supports the processes. EXPRESS is an international standard data modeling language for modeling product data [8]. EXPRESS was developed as part of the STEP development effort. In addition to the language syntax itself, STEP has also established standard practice guidelines for using the EXPRESS modeling language. Since the dimension and tolerance data is part of the product data that defines aspects of a product in its life cycle applications, EXPRESS and the usage practices were chosen for the D&T data model.

STATE OF TOLERANCING STANDARDS AND REPRESENTATION METHODS

Current status of tolerancing-related standards

The purpose of specifying dimensions and tolerances associated with product shape is to define the size and geometric characteristics with their allowable variations as part of the design data to manufacture parts. Currently, dimensioning and tolerancing standards [9, 10] specify the semantics and symbols that are used to present dimension and tolerance data on engineering drawings. The drawback of specifying symbols on drawings is that computer-aided design and tolerance analysis systems can not directly interpret or make use of these symbols. Although the Dimensional Measuring Interface Standard (DMIS) [11] specifies formats for geometric tolerances according to the ANSI Y14.5M-1982 [10], the data formats are for dimensional inspection purposes only and do not support the relationships between dimensions and tolerances for other product life cycle applications, such as manufacturing or maintenance.

The ANSI Y14.5 dimensioning and tolerancing standard is based on the functional requirement of parts for interchangeability - i.e., the Taylor principle. For dimensioning parts and their features, the standard currently provides specification of symbols and describes the practice of dimensioning including linear dimensioning, angular dimensioning, curvilinear dimensioning, size dimensioning, and location dimensioning. For tolerancing, the standard specifies symbols and practices for tolerancing the characteristics of a feature; the characteristics include tolerances of form, orientation, location, runout, and profile. ISO geometric tolerancing standards are based on both the Taylor Principle and the Independency Principle [12]. When the Independency Principle is applied, the form, location, orientation, or profile tolerances are evaluated independently of the actual size of the feature that is being checked. Datum symbols and practices of specifying and establishing datums are also included in the current standards. Datums form the basis from which relative dimensions and tolerances are referenced.

These dimensioning and tolerancing standards provide basic principles that are commonly used in part design in industry. These standards were used as the foundation upon which a dimension and tolerance data model is developed.

Status of current computerized representation for dimensions and tolerances

To represent the dimensions and tolerances, two methods are available: offset geometry and explicit data structure. Offset geometry [13] forms tolerance zones containing nominal geometry which is represented by geometric models. Explicit data structures capture dimension and tolerance information and associate them with the geometric models as an internal representation [14, 15, 16, 17, 18].

Offset geometry, also referred to as variational geometry, consists of envelopes that contain the geometric model. The envelopes are in similar geometry of the shape of the part. They are generated by offsetting the shape of the part using an algorithm, which usually is along the surface normal. These envelopes are in a pair of surfaces that define the tolerance zone within which the shapes of manufacturing parts can vary. The pair of envelopes is stored in a computer format that is associated with the solid models. Each tolerance zone is a pair of envelopes. This offset geometry method is good for tolerance zones that have fixed positions. Tolerance zones that have floating positions, such as form tolerances and orientation tolerances, are not suitable to be defined by two envelopes.

Data structure approaches create a set of data structures, sometimes called objects, that represent various geometric tolerances and datums in the format for developing tolerance-related application software using structural or object-oriented programming languages. This set of data structures is based on the ANSI Y14.5 specification. The data structures provide a link between tolerances and geometric entities used in geometric modeling and a common interface between geometric modelers and tolerance analysis application software. This structure was not created for a complete product description. The specific relationships between the data structures and other product data are not addressed. Furthermore, different dimensioning methods and statistical tolerancing methods are not included in the data structures.

REQUIREMENTS OF A DIMENSION AND TOLERANCE DATA MODEL

The first step in data modeling is to define data requirements. The data must meet not only current industry needs but also future needs. Data requirements should come from industry. The current industrial practices, state of the art technology, state of the industrial practice, and new technology under development should all be considered. Sources of data requirements for a dimension and tolerance data model come from current industrial practice which represents the industry needs, existing standards in dimensioning and tolerancing (both national and international), and new technology needs, such as tolerance analysis (analyze assigned tolerances for validity), tolerance synthesis (to find the optimal tolerance type and value), manufacturing process planning, inspection planning, and assembly planning. In addition, the functional requirements of modeling dimension and tolerance data are to be addressed in several areas: feature and product relationships, dimensioning needs, and tolerancing needs.

Dimension information requirement and abstraction

The most generic dimension consists of a magnitude and one or more dimensioned shape aspects. Shape aspect is defined in [20] as a physical portion of a product shape. For modeling dimensions and tolerances, the shape aspect and the feature as used in ANSI Y14.5 are equivalent and used interchangeably in this paper. A dimension defines a geometric characteristic of a feature, such as the diameter of a cylindrical hole, radius of an arc, the cone angle, or the width of a slot. A dimension may also be defined as the geometric relationship between two features, such as the distance between two parallel planes or the angle between two hole axes. A dimension can have a tolerance which is a plus-minus tolerance. Statistical tolerance can be applied to tolerance representation to further specify the dimensional distribution of the dimensions for manufactured parts. Table 1 summarizes the information requirements of a dimension.

Table 1 Information Abstraction of Dimensions

<table>
<tr><th colspan="4">Dimension Data Elements</th></tr>
<tr><th>Abstraction Level</th><th colspan="2">Element</th><th>Component</th></tr>
<tr><td rowspan="3">Level 0</td><td colspan="2">dimension value</td><td>magnitude, unit</td></tr>
<tr><td rowspan="2">dimensioned feature</td><td>one feature</td><td>shape aspect</td></tr>
<tr><td>two features</td><td>shape aspects</td></tr>
<tr><td rowspan="2">Level 1</td><td colspan="2">measuring path</td><td>linear, curvilinear, angular</td></tr>
<tr><td colspan="2">plus-minus tolerance</td><td>tolerance value</td></tr>
<tr><td>Level 2</td><td colspan="2">limit and fit</td><td>grade, deviation, fitting type</td></tr>
<tr><td>Level 3</td><td colspan="2">statistical tolerance</td><td>types of distribution function</td></tr>
</table>

In this table, higher level information is based upon the information specified on lower levels. Most generic dimensioning data are on Level 0. Dimension value has a magnitude and unit that are used to define the length of dimension. Dimension applies to either a single feature or the relationship between two features.

Measuring path and plus-minus tolerance data elements are classified in Level 1. Measuring path defines how the dimension is measured, e.g., measured linearly, along an arc, or angularly. Plus-minus tolerance applies to a dimension for specifying allowable variation. If no plus-minus tolerance is specified, the dimension is a nominal dimension which is a theoretically exact dimension.

Limit and fit is specified on Level 2. ISO 286 [19] specifies a standard tolerance, which includes fit grades and deviations. Fit type is for specifying a hole or shaft in a data representation without graphical display.

Level 3 includes statistical tolerances used in process control. Statistical tolerance defines the type of statistical distribution functions, such as normal distribution, binomial distribution, and Poisson distribution. Statistical tolerance applies to plus-minus tolerance.

Tolerance information requirement and abstraction

The most generic information for geometric tolerance data includes tolerance magnitude, toleranced feature, tolerance type, and tolerance zone (floating, floating with an orientation, or fixed relative to the datum reference frame). Specific geometric tolerances are form, orientation, location, runout, and profile. Form tolerances include straightness, flatness, roundness (circularity), and cylindricity. Orientation tolerances include parallelism, perpendicularity, and angularity. Location tolerances include position tolerance, concentricity, coaxiality, and symmetry. Runout tolerances include circular runout and total runout. Profile tolerances include profile of a line and profile of a surface. Detailed specifications for geometric tolerancing are in ISO 1101 [9].

An information structure for tolerances is shown in Table 2, which specifies levels of abstraction of tolerance information.

Table 2 Information Abstraction of Geometric Tolerance

Geometric Tolerance Data Elements		
Abstraction Level	Element	Component
Level 0	tolerance value	magnitude, unit
	tolerance type	a list of text strings
	toleranced feature	shape aspect
	tolerance zone form	list of all the possible forms of tolerance zone used in geometric tolerancing
Level 1	material conditions	MMC, LMC, RFS
	datum reference	a set of datums
	unit tolerance zone	magnitude, unit
	zone orientation	angular relation of measurement device to the datum
Level 2	datum	shape aspect
	datum feature	shape aspect
	datum target	shape aspect
Level 3	composite tolerance	two geometric tolerances

Similar to Table 1, higher level tolerance information is based on the information specified on lower levels. Level 0 includes the most fundamental elements for a geometric tolerance:

tolerance value, tolerance type, the toleranced feature, and the tolerance zone form. A feature has specific functional characteristics. For examples, a hole, a face, a slot, etc. are features on the part, and an axis, representing the position of a hole, is a feature in the space that is not on the part. The elements on this level are necessary for defining any geometric tolerance.

On Level 1, elements considered are material condition, datum reference, unit tolerance zone, and zone orientation. Material conditions include maximum material condition (MMC), least material condition (LMC), and regardless of feature size (RFS). These conditions only apply to tolerance features that have size characteristics and the variation of size affects tolerances. A hole or shaft, a uniform slot or key way, a wedge, and a cone are considered size features. Datum reference includes a set of datums that form a reference frame to which the geometric tolerance and its toleranced feature reference. Because form tolerances do not reference to a datum, the datum reference is included in Level 1, instead of Level 0. Unit tolerance zone is used by straightness and flatness tolerances. For a relatively large feature, straightness and flatness tolerances can be applied to the feature in a piece-wise fashion. A straightness tolerance is applied to a theoretically straight feature; similarly, a flatness tolerance is applied to a theoretically flat feature. Tolerance zone forms found in geometric tolerancing are cylindrical, conical, two-parallel-line, two-concentric-arc, two-coaxial-cylinder, conical, two-parallel-curve, and two-parallel-curved-surface. The zone forms identify the geometric shape of a tolerance zone. Zone orientation is the orientation of a measuring device, such as a dial indicator, in measuring a runout tolerance of a feature with respect to the axis of rotation. Zone orientation is only used by a runout tolerance that is defined by device orientations which are illustrated in ISO 1101.

Level 2 elements are datum-related information. A datum is established from datum features. A datum is a theoretically exact geometry from where a dimension or tolerance references. A datum is a shape aspect. A datum feature is also a shape aspect and a portion of a part surface from which a datum is established. A datum target is a kind of a datum feature that provides a target on the boundary of a part shape, such as a point, line, or area, to establish a datum.

Composite tolerance is on Level 3 because it uses two geometric tolerances to constrain a set of features in a pattern. The specification of composite tolerance is in ANSI Y14.5M-1982 Section 5. Higher-level elements are dependent on lower-level elements. This abstraction table provides a foundation for developing the tolerance schema.

DIMENSION AND TOLERANCE DATA MODEL

This data model was developed to provide input to the STEP Part 47 development effort. The organization of the data model consists of three groups, each one specified as an EXPRESS schema: Dimension Schema, Tolerance Schema, and Datum and Shape Aspect Schema. The model is in EXPRESS, so chosen because it is an international standard data modeling language. The graphical representation of the model is in EXPRESS-G, which is an extension of EXPRESS for graphical representation. Many entities of this model are dependent upon entities defined in ISO 10303 Part 41 [20] and Part 43 [21]. The text between "*)" and "(*" is the definitions of the model constructs, which is a part of the model.

Dimension schema

The graphical representation of the schema in EXPRESS-G is shown in Figure 1.

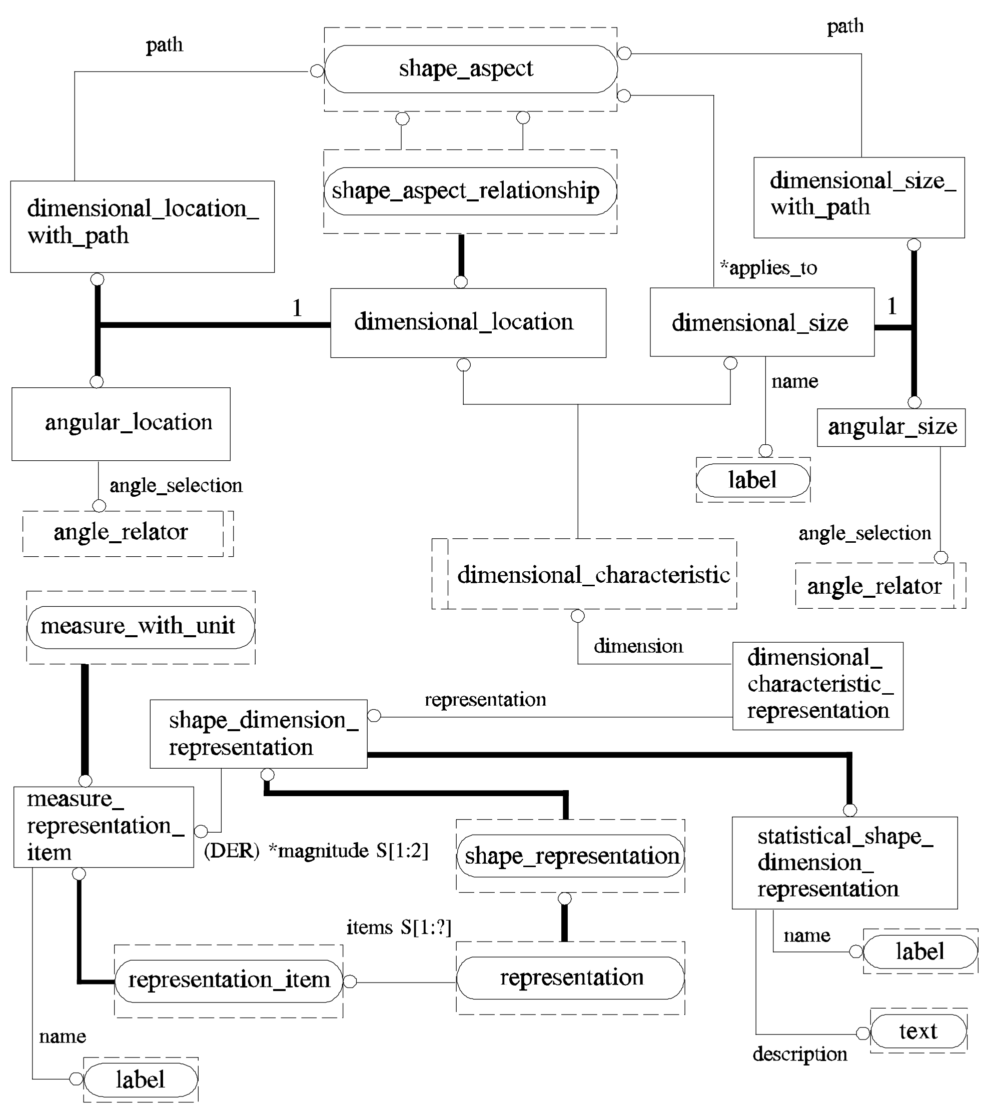

Figure 1 Graphical representation of Dimension Schema

Types of this schema are defined as follows.

```
*)
TYPE angle_relator = ENUMERATION OF
  (small,
   equal,
   large);
END_TYPE;
(*
```

An *angle_relator* type is an identification of an angle. This angle is one of a set of possible angles created by the intersection of two shape aspects. The item *small* indicates the angle at the point of intersection with the smaller absolute numerical measure. The item *equal* indicates any angle at the point of intersection since the numerical measures of the angles are all equal. The item *large* indicates the angle at the point of intersection with the larger absolute numerical measure.

```
*)
TYPE dimensional_characteristic = SELECT (dimensional_location,
                                          dimensional_size);
END_TYPE;
(*
```

A *dimensional_characteristic* is the selection of the dimension type to which a tolerance or explicit measure value applies.

Entities of this schema are defined as follows.

```
*)
ENTITY measure_representation_item
  SUBTYPE OF (measure_with_unit,
              representation_item);
  name : label;
END_ENTITY;
(*
```

A *measure_representation_item* is a representation that describes an explicit dimension. This representation is defined by measure values and other representation. The attribute *name* is the identification of the use of an explicit measure. The entity *measure_with_unit* (defines all ISO units of measurement) is in Part 41. The entity *representation_item* (defines geometric representation context) is defined in Part 43.

```
*)
ENTITY dimensional_characteristic_representation;
```

```
  dimension      : dimensional_characteristic;
  representation : shape_dimension_representation;
END_ENTITY;
(*
```

A *dimensional_characteristic_representation* is an association of an implicit dimension with an explicit representation of either the dimension or the lower and upper limits of the dimension. The attribute *dimension* is the implicit dimension for which an explicit measure or representation is defined. The attribute *representation* is the explicit representation assigned to the implicit dimension.

```
*)
ENTITY shape_dimension_representation
  SUPERTYPE OF (statistical_shape_dimension_representation)
  SUBTYPE OF (shape_representation);
DERIVE
  magnitude :  SET [1:2] OF measure_representation_item
          := SELF\shape_representation.items;
WHERE
  WR1: SIZEOF (QUERY ( V <* magnitude | V.value_component > 0.0)) =
       SIZEOF (magnitude);
END_ENTITY;
(*
```

A *shape_dimension_representation* is a representation of either *dimensional_location* or *dimensional_size*. It is a representation that explicitly describes a dimension of a shape aspect with a value, range of values, or a value in combination with another *representation_item*. When two values or a value in combination with another *representation_item* are defined for a specific *shape_dimension_representation*, they represent the upper and lower limits of a limit dimension. The attribute *magnitude* is the measures or representations that describe the dimension. The local rule *WR1* means dimensions shall be represented by positive measure values. [NOTE: Entities shape_representation and representation are defined in Part 43]

```
*)
ENTITY statistical_shape_dimension_representation
  SUBTYPE OF (shape_dimension_representation);
  name        : label;
  description : text;
WHERE
  WR1: SIZEOF (SELF\shape_dimension_representation.magnitude) = 2;
END_ENTITY;
(*
```

A *statistical_shape_dimension_representation* is a statistical distribution of the variation of a dimension as specified for a limit dimension. The attribute *name* is the common name of a probability distribution function. The attribute *description* is the specification of the statistical distribution function applied to the dimension. The local rule *WR1* states that the *statistical_shape_dimension_representation* shall represent the upper and lower bounds of the dimension. Informally, the *statistical_shape_dimension_representation* shall be used only to represent limit dimensions. Both *label* and *text* are defined in Part 41.

```
*)
ENTITY dimensional_location
  SUPERTYPE OF (ONEOF (angular_location,
                    dimensional_location_with_path))
  SUBTYPE OF (shape_aspect_relationship);
END_ENTITY;
(*
```

A *dimensional_location* defines a spatial constraint between two shape aspects. It is described by a non-directed measure that is derived along an implicit measurement path.

```
*)
ENTITY angular_location
  SUBTYPE OF (dimensional_location);
  angle_selection : angle_relator;
END_ENTITY;
(*
```

An *angular_location* defines a spatial constraint between two shape aspects that intersect or would intersect if projected. It is a non-directed measure of the angle defined by the two shape aspects and their common intersection or projected intersection. The attribute *angle_selection* is the indication of the angular measure being specified.

```
*)
ENTITY dimensional_location_with_path
  SUBTYPE OF (dimensional_location);
  path : shape_aspect;
END_ENTITY;
(*
```

A *dimensional_location_with_path* defines a spatial constraint between two shape aspects along an explicit path. It is a non-directed measure derived along the explicit path that is defined between the shape aspects. The attribute *path* is the shape aspect defining the measurement path for the *dimensional_location*.

```
*)
ENTITY dimensional_size
  SUPERTYPE OF (ONEOF (angular_size,
                       dimensional_size_with_path));
  applies_to : shape_aspect;
  name       : label;
WHERE
  WR1: applies_to.product_definitional = TRUE;
END_ENTITY;
(*
```

A *dimensional_size* defines a spatial characteristic of a shape aspect that is represented by a single magnitude. This magnitude is independent of the location of the shape aspect on or within the product. The attribute *applies_to* is the shape aspect being dimensioned. The attribute *name* is the identification of the application use of the dimension. A formal propositions *WR1* constrains that *size_dimension* shall apply only to shape aspects that define the product. The entity *shape_aspect* is defined in Part 41.

```
*)
ENTITY angular_size
  SUBTYPE OF (dimensional_size);
  angle_selection : angle_relator;
END_ENTITY;
(*
```

An *angular_size* defines an angular spatial characteristic of a shape aspect. An angular size is represented by a single magnitude and is independent of the location of the shape aspect on or within the product. It is the non-directed measure of the angle formed by two boundaries of the shape aspect and their common or projected intersection. The attribute *angle_selection* is the indication of the angle to be measured.

```
*)
ENTITY dimensional_size_with_path
  SUBTYPE OF (dimensional_size);
  path : shape_aspect;
END_ENTITY;
(*
```

A *dimensional_size_with_path* defines a spatial characteristic a shape aspect. It is represented by a single magnitude and is independent of the location of the shape aspect on or within the product. It is the non-directed measure derived along an explicit curve path that is defined between the two boundaries of the shape aspect. The attribute *path* is the *shape_aspect* defining the measurement path along which the dimension is specified.

The dimension schema consists the above entities for capturing dimensioning-related data.

Tolerance schema

The graphical representation of the schema in EXPRESS-G is shown in Figures 2, 3, and 4.

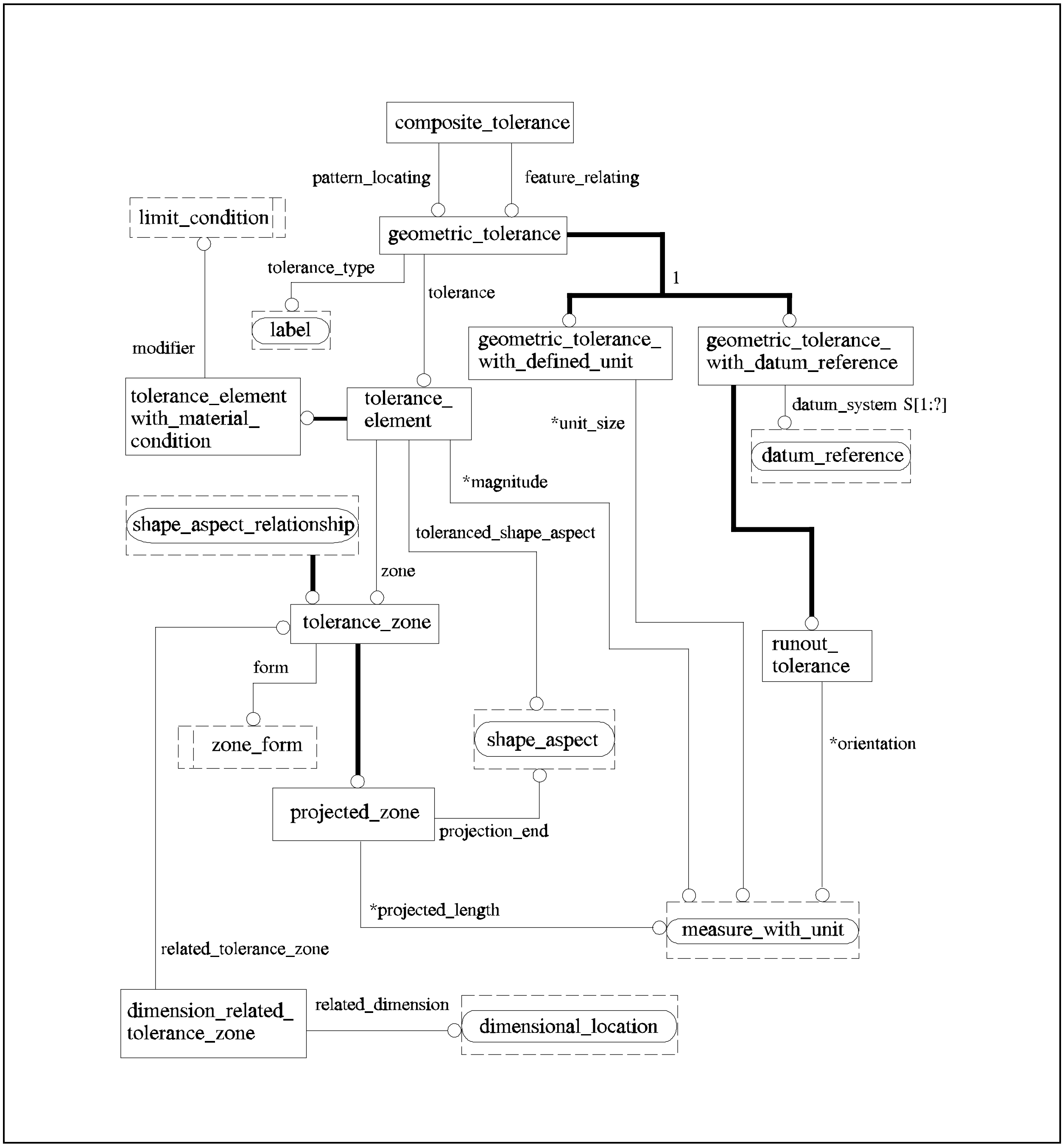

Figure 2 Graphical representation of Tolerance Schema - geometric tolerance

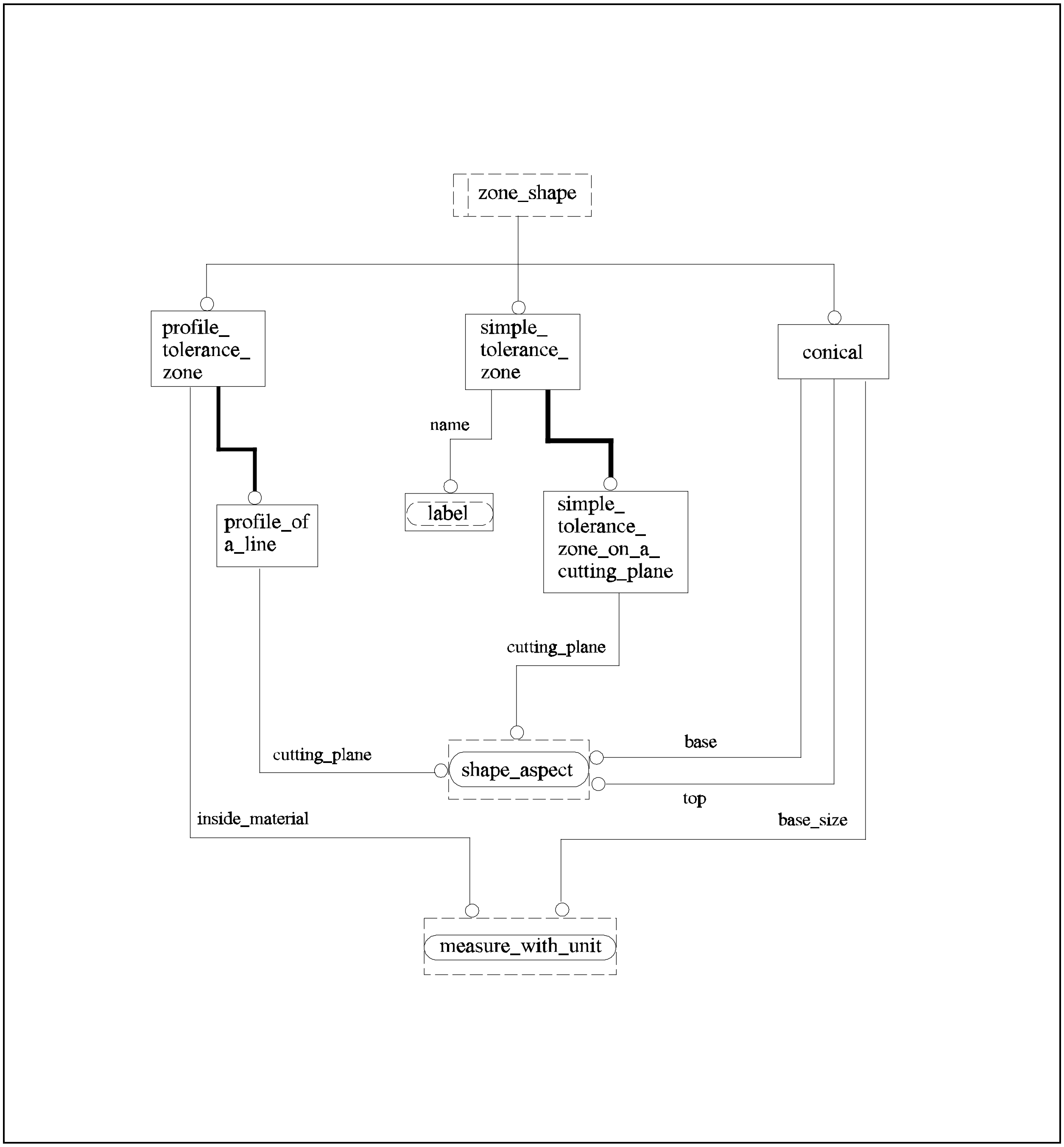

Figure 3 Graphical representation of Tolerance Schema - tolerance zone shape

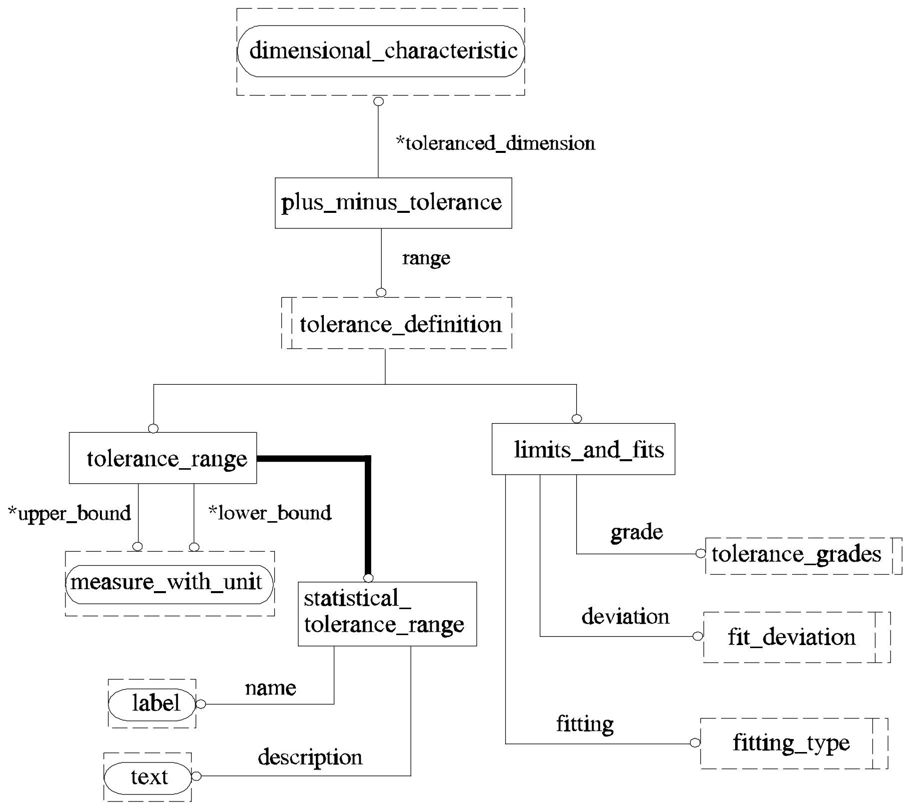

Figure 4 Graphical representation of Tolerance Schema - plus-minus tolerance

Types of this schema are defined as follows.

```
*)
TYPE limit_condition = ENUMERATION OF
  (maximum_material_condition,
```

```
   least_material_condition,
   regardless_of_feature_size);
END_TYPE;
(*
```

A *limit_condition* type indicates that a modification of a tolerance is permitted. This modification may allow for the addition of material or for the reduction of material to a product feature. The use of *limit_condition* applies only to product features that have size characteristics. Definitions of enumerated items *maximum_material_condition*, *least_material_condition*, and *regardless_of_feature_size* are same as defined in ANSI Y14.5 for modifying tolerances applied to a feature with size characteristic.

```
*)
TYPE tolerance_definition = SELECT
  (tolerance_range,
   limits_and_fits);
END_TYPE;
(*
```

The *tolerance_definition* type is the identification of the method used to generate a tolerance value in dimensioning parts.

```
*)
TYPE tolerance_grades = ENUMERATION OF
  (IT01, IT0, IT1, IT2, IT3, IT4, IT5, IT6, IT7, IT8, IT9, IT10,
   IT11, IT12, IT13, IT14, IT15, IT16, IT17, IT18);
END_TYPE;
(*
```

A *tolerance_grades* type is the designation of a standard tolerance as defined in 5.1.1 of ISO 286-1. Enumerated item definitions correspond to tolerance grades defined in 5.1.1 of ISO 286-1.

```
*)
TYPE fit_deviation = ENUMERATION OF
  (A, B, C, CD, D, E, EF, F, FG, G, H, J, JS, K, M, N,
   P, R, S, T, U, V, X, Y, Z, ZA, ZB, ZC);
END_TYPE;
*)
```

A *fit_deviation* type is the designation of a tolerance class as defined in 5.2.1 of ISO 286-1. The enumerated items represent *fit_deviation* magnitudes for both holes and shafts as defined in 5.2.1 of ISO 286-1.

```
*)
TYPE fitting_type = ENUMERATION OF
  (hole, shaft);
END_TYPE;
(*
```

A *fitting_type* is the designation that *fit_deviation* is applied to a hole or shaft. Enumerated item *hole* is meant that the *fit_deviation* is applied to a hole as defined in 5.1.2.1 of ISO 286-1. The item *shaft* is meant that the *fit_deviation* is applied to a shaft as defined in 5.1.2.1 of ISO 286-1.

```
*)
TYPE zone_shape = SELECT (profile_tolerance_zone,
                          simple_tolerance_zone,
                          conical);
END_TYPE;
(*
```

A *zone_shape* is the selection of the geometric form of the region in the spacial domain that a tolerance applies.

Entities of this schema are defined as follows.

```
*)
ENTITY geometric_tolerance
  SUPERTYPE OF ((ONEOF (geometric_tolerance_with_datum_reference,
                        geometric_tolerance_with_defined_unit));
  tolerance_type : label;
  tolerance      : tolerance_element;
END_ENTITY;
(*
```

A *geometric_tolerance* is the specification of the allowable range within which a geometrical property of a manufactured product may deviate from the intended exact shape. The attribute *tolerance_type* is an indication of the type of the tolerance. All the names of geometric tolerance specified in ISO 1101 are examples of tolerance type. The attribute *tolerance* is the specification of the size, the tolerance zone, and toleranced part features that the geometric tolerance applied.

```
*)
ENTITY composite_tolerance;
  pattern_locating  : geometric_tolerance;
  feature_relating  : geometric_tolerance;
WHERE
  WR1: pattern_locating.tolerance.magnitude.value_component >
       feature_relating.tolerance.magnitude.value_component;
```

```
END_ENTITY;
(*
```

A *composite_tolerance* is the specification of two geometric tolerances to a pattern, as a group, of part features. One tolerance is larger than the other, as stated in the formal proposition *WR1*. The larger tolerance controls individual feature in the group and the smaller tolerance controls the relationship among features. The attribute *pattern_locating* is the larger tolerance. The attribute *feature_relating* is the smaller tolerance. Examples of the usage of composite tolerance is specified in 5.4 ANSI Y14.5M-1982.

```
*)
ENTITY geometric_tolerance_with_datum_reference
  SUBTYPE OF (geometric_tolerance);
  datum_system : SET [1:?] OF datum_reference;
END_ENTITY;
(*
```

A *geometric_tolerance_with_datum_reference* is the specification of a geometrical tolerance that references one or more datums. The attribute *datum_system* is the datums that are the reference for the origin of a *geometric_tolerance*. The entity *datum_reference* is defined in the datum and shape aspect schema.

```
*)
ENTITY geometric_tolerance_with_defined_unit
  SUBTYPE OF (geometric_tolerance);
  unit_size : measure_with_unit;
WHERE
  WR1: unit_size.value_component > 0.0;
END_ENTITY;
(*
```

A *geometric_tolerance_with_defined_unit* is the specification of a tolerance using a per unit length or per square unit area as the basis for the tolerance range. The attribute *unit_size* is the length over which a straightness tolerance applies, or the length of a side of a square unit area to which a flatness tolerance applies. A formal proposition *WR1* applied to the attribute is that the length of a side of a square unit area shall be greater than 0.0.

```
*)
ENTITY tolerance_element;
  magnitude                 : measure_with_unit;
  toleranced_shape_aspect : shape_aspect;
  zone                      : tolerance_zone;
WHERE
  WR1:   magnitude.value_component >= 0.0;
```

```
END_ENTITY;
(*
```

Attribute definitions are that *magnitude* is the size of a tolerance, *toleranced_shape_aspect* is the *shape_aspect* to which the tolerance applies, and *zone* is the shape of a tolerance zone. A formal proposition *WR1* is that the magnitude of the tolerance shall be greater than or equal to 0.0.

```
*)
ENTITY tolerance_element_with_material_condition
  SUBTYPE OF (tolerance_element);
  modifier : limit_condition;
END_ENTITY;
(*
```

A *tolerance_element_with_material_condition* is a *tolerance_element* with a limit condition. As specified in 4.2 of ISO 2692 [22], this limit condition modifies the tolerance value applied to the toleranced shape aspect. The attribute modifier is a *limit_condition* that modifies a geometrical tolerance. An informal proposition applies to the use of this entity that the *toleranced_shape_aspect* should be a shape aspect defining a product feature that has size characteristics.

```
*)
ENTITY dimension_related_tolerance_zone;
  related_dimension : dimensional_location;
  related_tolerance_zone   : tolerance_zone;
END_ENTITY;
(*
```

A *dimension_related_tolerance_zone* is the specification of a tolerance zone that defines the allowable variation of a *dimensional_location*. The attribute *related_dimension* is the location dimension to which the tolerance applies. The attribute *related_tolerance_zone* is the tolerance zone element that is being applied to a dimension.

```
*)
ENTITY tolerance_zone;
  zone : zone_shape;
END_ENTITY;
(*
```

A *tolerance_zone* is a description of the shape of the tolerance zone. The attribute *zone* is an indication of the geometric form of a tolerance zone.

```
*)
ENTITY projected_zone
```

```
  SUBTYPE OF (tolerance_zone);
  projection_end    : shape_aspect;
  projected_length : measure_with_unit;
WHERE
  WR1: projected_length.value_component > 0.0;
END_ENTITY;
(*
```

A *projected_zone* is a *tolerance_zone* that is projected from a feature of a product. The projection is external to the feature and made from one of the ends of the feature for a specified length. A projected tolerance zone is defined in clause 4 of ISO 10578 [23]. The attribute *projection_end* is the *shape_aspect* from which the projected tolerance zone originates. The attribute *projected_length* is the length of a projected tolerance zone. A formal proposition *WR1* states that the value of the projected length shall be greater than 0.0.

```
*)
ENTITY runout_tolerance
  SUBTYPE OF (geometric_tolerance_with_datum_reference);
  orientation  : measure_with_unit;
WHERE
  WR1 : 0.0 <= orientation.value_component <= PI/2;
END_ENTITY;
(*
```

A *runout_tolerance* is a tolerance_zone_element that is defined by the orientation of the measurement of the toleranced *shape_aspect* to its axis of rotation which is indicated by a datum axis. The attribute *orientation* is the representation of the orientation of the measurement of a toleranced *shape_aspect* to the datum axis. A formal proposition *WR1* states that the measurement orientation shall be less than or equal to 90 degrees.

```
*)
ENTITY profile_tolerance_zone;
  inside_material : measure_with_unit;
END_ENTITY;
(*
```

A *profile_tolerance_zone* is a region bounded by two surfaces of normal offset from a nominal surface of a part feature. The profile tolerance is defined in 6.5 ANSI Y14.5M-1982. The attribute *inside_material* is the size of the portion of a tolerance that is inside the material. The attribute magnitude in the Entity *tolerance_element*, as described previsouly, is the size of the portion of a tolerance that is outside the material. If the magnitudes of both attributes are equal, then is tolerance zone is evenly distributed around the nominal part feature surface. If inside_material size or tolerance_element.magnitude size is zero, then the tolerance zone is unilateral, i.e., the tolerance zone is either in the material or out of the material.

```
*)
ENTITY profile_of_a_line
  SUBTYPE OF (profile_tolerance_zone);
  cutting_plane : shape_aspect;
END_ENTITY;
(*
```

A *profile_of_a_line* is an area bounded by two curves that are normal offset from a nominal curve of a part feature. The attribute *cutting_plane* is a theoretical plane that cuts through the part. The intersection of the cutting plane and the part should be a curve -- the profile of a line.

```
*)
ENTITY simple_tolerance_zone;
  name : label;
END_ENTITY;
(*
```

A *simple_zone_shape* is a region bounded by a simple geometric form such that the size of the region can be defined by *tolerance_element.magnitude*. The attribute *name* is an indication of what the zone shape is, e.g., two-parallel-plane zone, cylindrical zone, and two-coaxial-arc-surface zone.

```
*)
ENTITY simple_tolerance_zone_on_a_cutting_plane
  SUBTYPE OF (simple_tolerance_zone);
  cutting_plane : shape_aspect;
END_ENTITY;
(*
```

A *simple_zone_shape_on_a_cutting_plane* is an area bounded by a geometry such that the size of the area can be defined by the *tolerance_element.magnitude*. Examples of zone shapes are two-parallel-line zone and two-concentric-arc zone.

```
*)
ENTITY conical;
  base_size : measure_with_unit;
  top, base : shape_aspect;
END_ENTITY;
(*
```

A *conical* is a region bounded by a conical surface. The attribute *base_size* is the diameter of the base of a conical zone, larger circle. The attribute *tolerance_element.magnitude*, previously described, is the size of the top of a conical zone, smaller circle.

```
*)
ENTITY   plus_minus_tolerance;
  toleranced_dimension : dimensional_characteristic;
  range                : tolerance_definition;
UNIQUE
  UR1: toleranced_dimension;
END_ENTITY;
(*
```

The *plus_minus_tolerance* is the specification of the limits within which a dimension may vary. A *plus_minus_tolerance* may be either *tolerance_range* or *limits_and_fits*. Attribute *toleranced_dimension* is the dimension to which the plus-minus tolerance applies. Attribute *range* is the limiting values within which the toleranced shape aspects may vary. A formal proposition *UR1* states that there shall be only one plus-minus tolerance for a dimension.

```
*)
ENTITY tolerance_range
  SUPERTYPE OF (statistical_tolerance_range);
  upper_bound : measure_with_unit;
  lower_bound : measure_with_unit;
WHERE
  WR1: upper_bound.value_component > lower_bound.value_component;
  WR2: upper_bound.unit_component :=: lower_bound.unit_component;
END_ENTITY;
(*
```

The *tolerance_range* is the representation of plus-minus tolerances for a dimension. A *tolerance_range* has the numeric values added algebraically to the nominal dimension of a shape aspect. Each range is applied to the value of the dimension to determine the acceptable range of measured values. A *tolerance_range* may be either assigned by the user or specified according to ISO 286-1 and ISO 286-2. Attribute *upper_bound* is the value of the tolerance that is added to the dimension value to establish the maximum deviation from the boundary toleranced from the nominal dimension. Attribute *lower_bound* is the value of the tolerance that is added to the dimension value to establish the minimum deviation from the boundary toleranced from the nominal dimension. Formal proposition *WR1* states that the value of the *upper_bound* shall be greater than the value of the *lower_bound*. Formal proposition *WR2* states that the *upper_bound* and the *lower_bound* shall have the same unit.

```
*)
ENTITY statistical_tolerance_range
  SUBTYPE OF (tolerance_range);
  name        : label;
  description : text;
END_ENTITY;
```

(*

A *statistical_tolerance_range* defines the type of the probability distribution used for specifying the allowed dimensional variation. Attribute *name* is the type representation of a statistical distribution. Attribute *description* is the description that defines the statistical distribution, e.g., "Gaussian" is the name of the type of distribution, and a specified standard deviation is in the description.

*)

```
ENTITY limits_and_fits;
  grade      : tolerance_grades;
  fitting    : fitting_type;
  deviation : fit_deviation;
END_ENTITY;
```

(*

A *limits_and_fits* is a standard fit system for specifying the tolerances associated with the assembly of a hole or shaft. See ISO 286-1 and 286-2. Attribute *grade* is the size of the tolerance zone required for the desired fit. Attribute *fitting* is the specification of whether the toleranced shape aspect is a shaft fit or a hole fit. Attribute *deviation* is an indicator of the position of a tolerance zone for a fit tolerance.

Datum and shape aspect schema

The graphical representation of the schema in EXPRESS-G is shown in Figures 5 and 6.

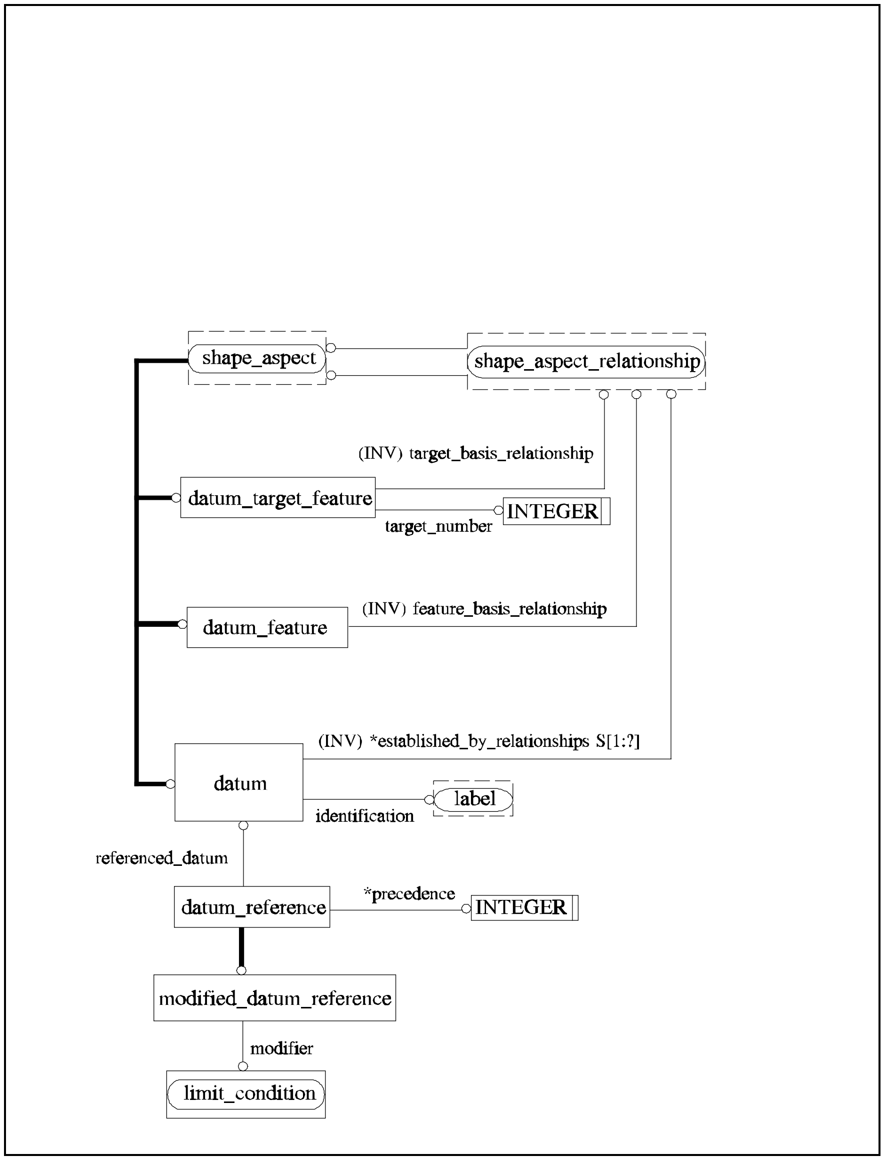

Figure 5 Graphical representation of Datum and Shape Aspect Schema - datum

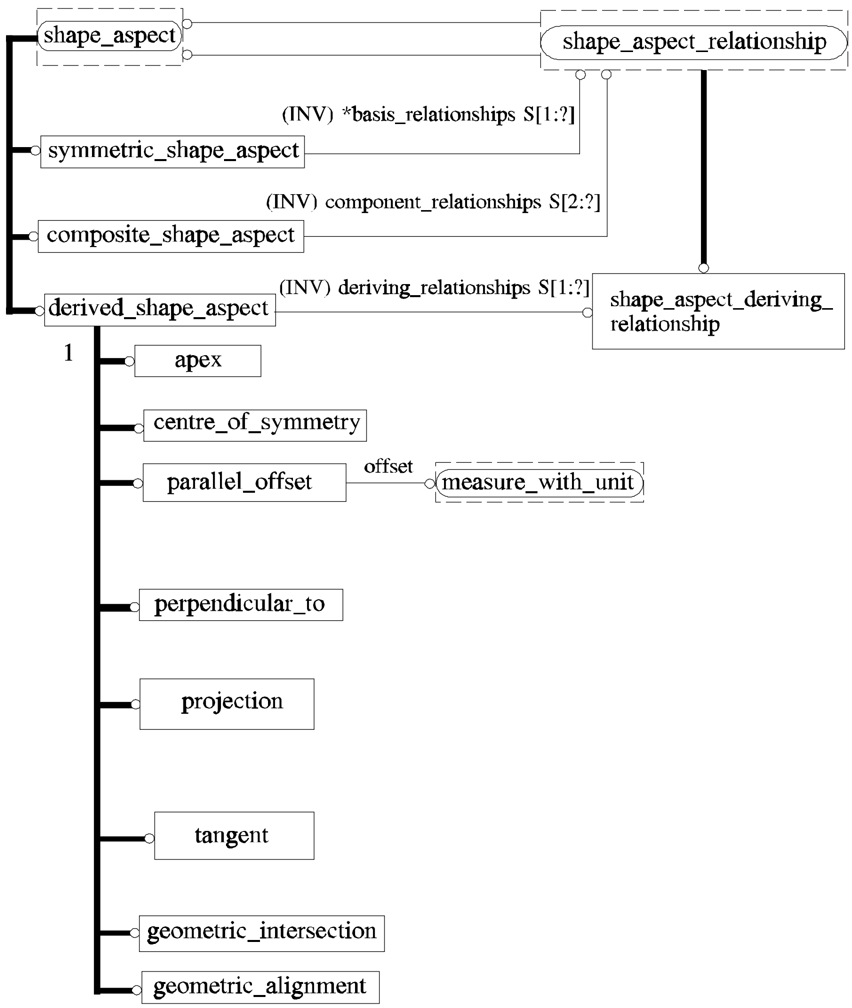

Figure 6 Graphical representation of Datum and Shape Aspect Schema - shape aspect

Entities of this schema are defined as follows.

```
*)
ENTITY datum
  SUBTYPE OF (shape_aspect);
```

```
  identification              : label;
DERIVE
  product_definitional          : LOGICAL := FALSE;
INVERSE
  established_by_relationships : SET [1:?] OF shape_aspect_relationship
                                 FOR relating_shape_aspect;
WHERE
  WR1: SIZEOF (QUERY (X<*SELF.established_by_relationships |
       SIZEOF (TYPEOF(X)*['SHAPE_ASPECT_DEFINITION_SCHEMA.DATUM_FEATURE',
       'SHAPE_ASPECT_DEFINITION_SCHEMA.DATUM_TARGET_FEATURE']) <> 1))=0;
  WR2: SIZEOF (QUERY (X<*SELF.established_by_relationships |
       SIZEOF (TYPEOF(X)*['SHAPE_ASPECT_DEFINITION_SCHEMA.DATUM_FEATURE',
       'SHAPE_ASPECT_DEFINITION_SCHEMA.DATUM_TARGET_FEATURE']) <>
       TYPEOF(SELF.established_by_relationships[1])*
       ['SHAPE_ASPECT_DEFINITION_SCHEMA.DATUM_FEATURE',
       'SHAPE_ASPECT_DEFINITION_SCHEMA.DATUM_TARGET_FEATURE'])) = 0;
END_ENTITY;
(*
```

A *datum* is a shape aspect from which dimensions and tolerances are established. This shape aspect may but need not coincide with the shape boundary defining the product. A datum is established by a datum feature, a set of datum target features, or a group of product features. Attribute *identification* is the designation for the datum. Attribute *product_definitional* is an indicator that the datum is not on the physical boundary of the shape that defines the product. Attribute *established_by_relationships* is the datum feature, the set of datum targets, or the group of derived shape aspects that produce the datum. Formal proposition *WR1* states that a *datum* shall not exist without the *datum_feature*s or *datum_target_feature*s that establish it. Attribute *WR2* states that a *datum* shall be established by exclusively either *datum_feature*s or *datum_target_feature*s.

```
*)
ENTITY datum_feature
  SUBTYPE OF (shape_aspect);
DERIVE
  product_definitional : LOGICAL := TRUE;
INVERSE
  feature_basis_relationship : shape_aspect_relationship
                               FOR relating_shape_aspect;
END_ENTITY;
(*
```

A *datum_feature* is an identified shape aspect on the boundary of the product. One *datum_feature* may be used to establish a single datum. Attribute *product_definitional* is an indicator that the *datum_feature* is on the physical boundary of the shape that defines the product.

Attribute *feature_basis_relationship* is the datum that the *datum_feature* defines.

```
*)
ENTITY datum_target_feature
  SUBTYPE OF (shape_aspect);
  target_number              : INTEGER;
DERIVE
  product_definitional       : LOGICAL := TRUE;
INVERSE
  target_basis_relationship : shape_aspect_relationship FOR
                             relating_shape_aspect;
END_ENTITY;
(*
```

A *datum_target_feature* is a *shape_aspect* that indicates a datum target on the boundary of a product shape. The shape aspect may be a point, line, or an area. The *datum_target_feature* is defined in addition to the shape aspects that define the product shape. Attribute *target_number* is the identification of the target feature. Attribute *product_definitional* is the indicator that the *datum_target_feature* is on the physical boundary of the shape that defines the product. Attribute *target_basis_relationship* is the datum for which the *datum_target_feature* provides either a partial or complete definition.

```
*)
ENTITY datum_reference
  SUPERTYPE OF (modified_datum_reference);
  referenced_datum : datum;
  precedence       : INTEGER;
WHERE
  WR1: precedence > 0;
END_ENTITY;
(*
```

A *datum_reference* is the specification of a datum from which a geometrical tolerance applies. Each *datum_reference* has a precedence that defines the order in which the *datum* participates in a datum system. Attribute *referenced_datum* is the datum that participates in a geometrical tolerance of a product feature. Attribute *precedence* is the order of the datum used in the definition of a datum system, as described in 6.2.3 of ISO 5459 [24]. Formal proposition *WR1* is the value of *precedence* shall be greater than 0.

```
*)
ENTITY modified_datum_reference
  SUBTYPE OF (datum_reference);
  modifier : limit_condition;
END_ENTITY;
```

```
(*
```

A *modified_datum_reference* is a *datum_reference* where the referenced datum may vary within the specified limits of size. Attribute *modifier* is the *limit_condition* that is assigned to the *datum*.

```
*)
ENTITY symmetric_shape_aspect
  SUBTYPE OF (shape_aspect);
INVERSE
  basis_relationships : SET [1:?] OF shape_aspect_relationship
                        FOR relating_shape_aspect;
WHERE
  WR1: SIZEOF (QUERY (X<*SELF.basis_relationships | NOT
       'SHAPE_ASPECT_DEFINITION_SCHEMA.CENTRE_OF_SYMMETRY' IN TYPEOF
       (X.related_shape_aspect)))=0;
END_ENTITY;
(*
```

A *symmetric_shape_aspect* is an identified shape aspect of a product that has the property of symmetry. Or alternatively, it may be a shape aspect derived from a group of identified shape aspects of a product that together have the property of symmetry. Attribute *basis_relationships* is the identified product feature that is symmetric about centres of symmetry, e.g. point, axis, or median plane. Attribute *related_shape_aspect*s, see ISO/DIS 10303-41, are *centre_of_symmetry*s. Attribute *relating_shape_aspect* is the *symmetric_shape_aspect*. Formal proposition *WR1* states that a *symmetric_shape_aspect* shall have *centre_of_symmetry*s.

```
*)
ENTITY composite_shape_aspect
  SUBTYPE OF (shape_aspect);
INVERSE
  component_relationships : SET [2:?] OF shape_aspect_relationship
                            FOR relating_shape_aspect;
END_ENTITY;
(*
```

A *composite_shape_aspect* is an identified group of shape aspects that are related to each other for a specific purpose. Attribute *component_relationships* is the set of member *shape_aspect*s that form the *composite_shape_aspect*. The *relating_shape_aspect* is the *composite_shape_aspect*, and the *related_shape_aspect*s are members of the set.

```
*)
ENTITY shape_aspect_deriving_relationship
  SUBTYPE OF (shape_aspect_relationship);
```

```
WHERE
  WR1: 'SHAPE_ASPECT_DEFINITION_SCHEMA.DERIVED_SHAPE_ASPECT' IN TYPEOF
       (SELF\SHAPE_ASPECT_RELATIONSHIP.RELATING_SHAPE_ASPECT);
END_ENTITY;
(*
```

A *shape_aspect_deriving_relationship* is a *shape_aspect_relationship* that defines the specific association between a *derived_shape_aspect* and other *shape_aspect*s. The *relating_shape_aspect* is the *derived_shape_aspect*. The *related_shape_aspect*s are the other *shape_aspect*s that are the basis for the *derived_shape_aspect*. Formal proposition *WR1* states that the *relating_shape_aspect* of *shape_aspect_deriving_relationship* shall be a *derived_shape_aspect*.

```
*)
ENTITY derived_shape_aspect
  SUPERTYPE OF (ONEOF (apex,
                       centre_of_symmetry,
                       geometric_alignment,
                       geometric_intersection,
                       parallel_offset,
                       perpendicular_to,
                       projection,
                       tangent))
  SUBTYPE OF (shape_aspect);
DERIVE
  product_definitional    : LOGICAL := FALSE;
INVERSE
  deriving_relationships : SET [1:?] OF shape_aspect_deriving_relationship
                           FOR relating_shape_aspect;
END_ENTITY;
(*
```

A *derived_shape_aspect* is a *shape_aspect* that is not on the physical boundary of the product shape. The existence of the *derived_shape_aspect* depends on and relates to other *shape_aspect*s. Attribute *product_definitional* is the indication that the *derived_shape_aspect* is not on the physical boundary of the product. Attribute *deriving_relationships* is the identification of *shape_aspect_deriving_relationship*s that define the *derived_shape_aspect*.

```
*)
ENTITY apex
  SUBTYPE OF (derived_shape_aspect);
END_ENTITY;
(*
```

An *apex* is a *derived_shape_aspect* that is the point created by the projection of a conical surface or the common point that is created by projections of two or more conical surfaces. This surface is established from the conical frustum. Attribute *SELF\derived_shape_aspect.deriving_relationships* is the projection of conical surfaces of frustums to the *apex*. The *relating_shape_aspect* is the *apex*. The *related_shape_aspect*s are *shape_aspect*s of the conical surfaces of frustums from which the *apex* is derived.

```
*)
ENTITY centre_of_symmetry
  SUBTYPE OF (derived_shape_aspect);
END_ENTITY;
(*
```

A *centre_of_symmetry* is a *derived_shape_aspect* that defines the geometric centre of one or more *shape_aspect*s. Attribute *SELF\derived_shape_aspect.deriving_relationships* is the *symmetric_shape_aspect*s that establish the *centre_of_symmetry*. The *relating_shape_aspect* is the *centre_of_symmetry*. The *related_shape_aspect*s are the symmetric *shape_aspect*s.

```
*)
ENTITY parallel_offset
  SUBTYPE OF (derived_shape_aspect);
  offset              : measure_with_unit;
WHERE
  WR1: SIZEOF (SELF\derived_shape_aspect.deriving_relationships)=2;
END_ENTITY;
(*
```

A *parallel_offset* is a *derived_shape_aspect* that is in a position parallel to and at a specified distance from a basis shape aspect. The *parallel_to* is positioned relative to the basis shape aspect such that the shortest linear path defined from the basis shape aspect to a third shape aspect intersects the *parallel_offset*. Attribute *offset* is the distance that the *parallel_offset* is from the *shape_aspect* to which the *parallel_offset* is parallel. Attribute *SELF\derived_shape_aspect.deriving_relationships* is a *shape_aspect* that is parallel to another *shape_aspect*. The *parallel_offset* has two *deriving_relationships*. The *relating_shape_aspect* is the *parallel_offset*. One *related_shape_aspect* is the basis that the *parallel_offset* is defined from. The other *related_shape_aspect* identifies the direction in which the *parallel_offset* is to be positioned. Formal proposition *WR1* states that there shall be two members in the set of *deriving_relationships*.

```
*)
ENTITY perpendicular_to
  SUBTYPE OF (derived_shape_aspect);
WHERE
  WR1: SIZEOF (SELF\derived_shape_aspect.deriving_relationships)=1;
END_ENTITY;
```

(*

A *perpendicular_to* is a *derived_shape_aspect* that is oriented perpendicular to another shape aspect. Attribute *SELF\derived_shape_aspect.deriving_relationships* is a *shape_aspect* that is perpendicular to other *shape_aspect*. The *relating_shape_aspect* is the *perpendicular_to*. The *related_shape_aspect* is the *shape_aspect* from which the *perpendicular_to* is derived. Formal propositions *WR1* states that there shall be one member in the set of the *deriving_relationships*.

*)
```
ENTITY projection
  SUBTYPE OF (derived_shape_aspect);
WHERE
  WR1: SIZEOF (SELF\derived_shape_aspect.deriving_relationships)=3;
END_ENTITY;
```
(*

A *projection* is a *derived_shape_aspect* that is the extension of a curve or surface from a shape aspect at a specified edge and along a specified path. Attribute *SELF\derived_shape_aspect.deriving_relationships* is a *shape_aspect* that is projected from other *shape_aspect*s. The *projection* has three *deriving_relationships*. The *relating_shape_aspect* is the *projection*. One *related_shape_aspect* is the basis for the projection. Another *relating_shape_aspect* is the edge of this basis *shape_aspect* from which the projection is made. The other *relating_shape_aspect* is the *shape_aspect* that defines the path of the projection. Formal proposition *WR1* states that there shall be three members in the set of *deriving_relationships*.

*)
```
ENTITY tangent
  SUBTYPE OF (derived_shape_aspect);
WHERE
  WR1: SIZEOF (SELF\derived_shape_aspect.deriving_relationships)=3;
END_ENTITY;
```
(*

A *tangent* is a *derived_shape_aspect* that touches a curve or surface *shape_aspect* at a single point or line. The normals of the *tangent* and the curve or surface shape aspect at the point of touching are coincident. The *tangent* is oriented such that it is parallel to another shape aspect. The *tangent* is positioned relative to the curve or surface shape aspect such that the shortest linear path defined from the centre of curvature at the point or line of touching to the other shape aspect intersects the *tangent*. Attribute *SELF\derived_shape_aspect.deriving_relationships* is the *shape_aspect* that is tangent to other *shape_aspect*. The *tangent* has three *deriving_relationships*. The *relating_shape_aspect* is the *tangent*. One *related_shape_aspect* is the *shape_aspect* that the *tangent* touches. Another *related_shape_aspect* is the *shape_aspect* that is parallel to the *tangent*. The other *related_shape_aspect* identifies the direction in which the *tangent* is to be positioned.

Formal propositions *WR1* states that there shall be three members in the set of *deriving_relationships*.

```
*)
ENTITY geometric_intersection
  SUBTYPE OF (derived_shape_aspect);
END_ENTITY;
(*
```

A *geometric_intersection* is a *derived_shape_aspect* that is the common intersection of two or more shape aspects. Attribute *SELF\derived_shape_aspect.deriving_relationships* is a common intersection among intersecting shape aspects. The *relating_shape_aspect* is the *geometric_intersection*. The *related_shape_aspect*s are the basis of the common intersection. An informal propositions states that the *shape_aspect*s that participate as *relating_shape_aspect*s shall intersect.

```
*)
ENTITY geometric_alignment
  SUBTYPE OF (derived_shape_aspect);
END_ENTITY;
(*
```

A *geometric_alignment* is a linear or planar *derived_shape_aspect* that is established to contain aligned shape aspects. In the case of planar *derived_shape_aspect*, it contains at least one linear shape aspect. Attribute *SELF\derived_shape_aspect.deriving_relationships* is a linear or planar *derived_shape_aspect* that contains two or more aligned shape aspects. The *relating_shape_aspect* is the *geometric_alignment*. The *related_shape_aspect*s are the aligned shape aspects.

EXAMPLES OF THE MODEL CAPTURING INFORMATION

Examples of dimensions and a tolerance that the model captures are given in this section. A part design is shown in Figure 7. The dimensions of the block are 110 mm by 120 mm by 250 mm. A 50 mm hole is located 50 mm from datum surface A and 60 mm from datum surface B. Two position tolerances which form a conical tolerance zone are applied to the hole.

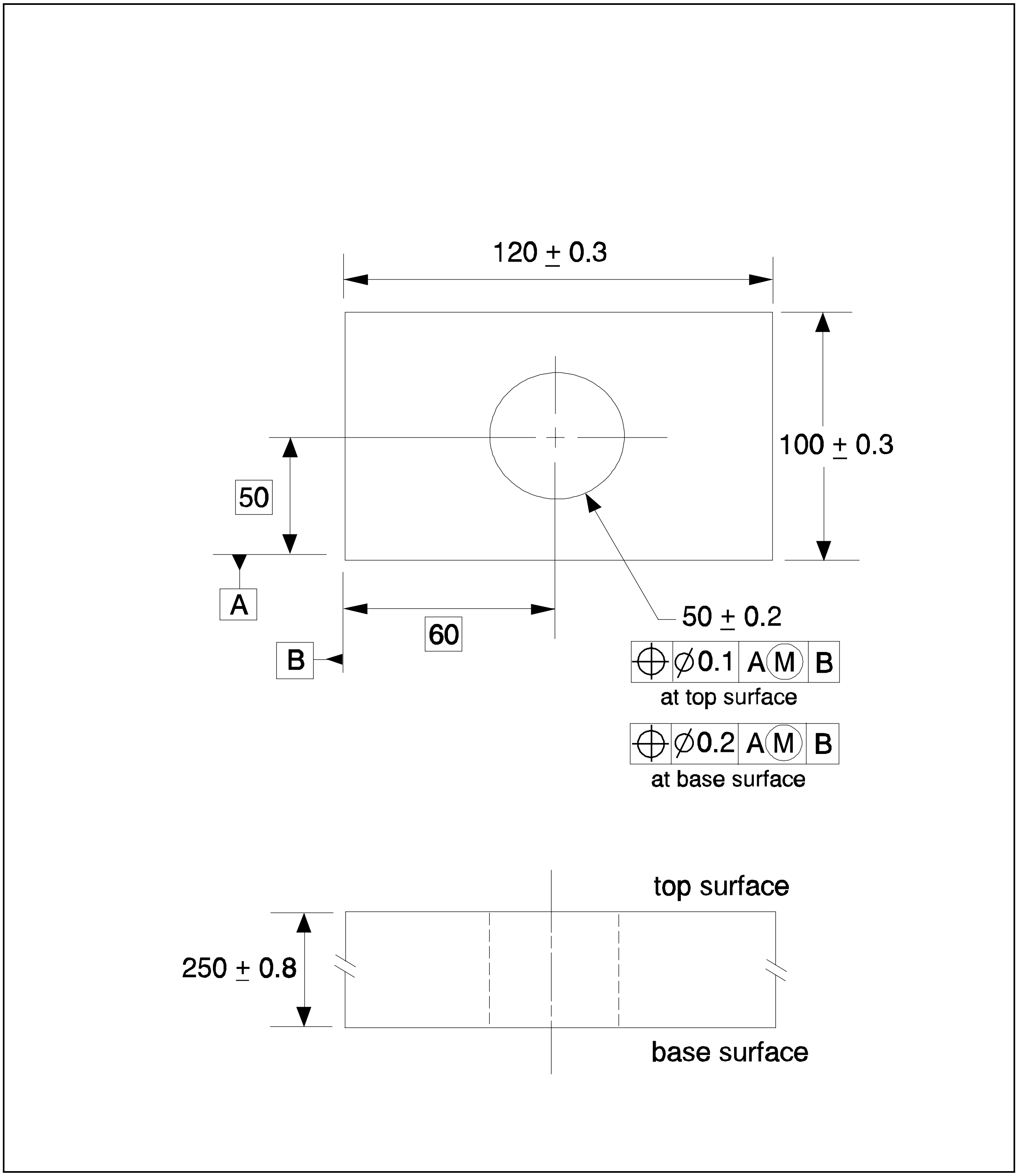

Figure 7 A part design

The position tolerance, the size dimensions, and a location dimension captured by the model are shown in the following text.

The data about the position tolerance is captured as follows:
geometric_tolerance_with_datum_reference (entity used)
    tolerance type (attribute) : "position tolerance"
    tolerance (tolerance_element)
        magnitude
            value_component : 0.1 (on the top surface)
            unit_component : mm
        toleranced_shape_aspect : hole [note: the 4mm hole]
        zone
            form (conical)
                top : top_surface
                base : base_surface
                base_size
                    value_component : 0.2
                    unit_component : mm
    datum_system (a set of 2 *datum_references*)
        datum_reference [note: the first element]
            precedence : primary
            referenced_datum
                Identification : A
                established_by_relationship.relating_shape_aspect : datum surface A
        datum_reference [note: the second element]
            precedence : secondary
            referenced_datum
                Identification : B
                established_by_relationship.relating_shape_aspect : datum surface B

The data about the size dimension is captured as follows:
dimensional_characteristic_representation (entity)
    dimension (dimensional_size)
        applies_to : hole
        name : diameter of the hole
    representation
        magnitude (set of one)
            value_component : 50
            unit_component : mm

The data about a location dimension, distance between the hole axis and datum surface B is captured as follows:
dimensional_characteristic_representation (entity)
    dimension (dimensional_location)
        relating_shape_aspect : axis of the hole
        related_shape_aspect : datum B
    representation

magnitude (set of one)
    value_component : 60
    unit_component : mm

CONCLUSION

To achieve computer integrated design and manufacturing, it is necessary to have datum, shape aspect, dimensioning, and tolerancing information shared and exchanged among various computer aided design and manufacturing applications along with the product shape geometry and topology. Typically, applications such as tolerance analysis, tolerance synthesis, assembly analysis, process planning, and dimensional inspection planning require support from a sharable engineering and manufacturing database to achieve integration. The data model represented in this paper is capable of capturing dimensioning and tolerancing data specified in ANSI and ISO geometric tolerancing standards and, therefore, the dimensioning and tolerancing information represented by conventional engineering drawings. This data model is also designed as part of an integrated model structure, the STEP integrated resources (Parts 41 through 49), that describe characteristics of products, including shape geometry. The model meets functional requirements for supporting current dimensioning and tolerancing practice and is expandable for future needs.

There are still unsolved research issues in D&T data modeling due primarily to the state of standards and practices: ambiguities in standards, misinterpretation of standards, and inconsistencies in use. For example, a more effective means to represent statistical tolerances is desired. However, statistical tolerancing is not well understood.

The future direction in achieving the goal of electronic commerce for manufacturing enterprises relies heavily upon a computer integrated design and manufacturing environment. STEP provides an integrated data structure that can be used for a database design that facilitates the communication of product data required in a computer integrated design and manufacturing environment. Following the development of this data model, the challenge is to develop application software systems that are based on STEP and the D&T data model to integrate D&T information as part of a product model. These integrated software application systems will advance the state of computer-aided design and manufacturing technology integration.

Acknowledgement

The authors gratefully acknowledge Jeane Ford, Ted Hopp, Mary Mitchell, Jesse Crusey, Bill Burkett, and Martin Holland for their supports during developing the model and members of ISO/TC 184/SC 4/WG 3/P 3 (STEP Part 47) for their technical assistance and many helpful discussions.

REFERENCES

1 Voelcker, H.B., "Modeling in the Design Process", Design and Analysis of Integrated Manufacturing Systems, W. Dale Compton editor, National Academy Press, Washington, D.C., 1988. pp. 167-199.

2 ISO/CD 10303-47, "Product Data Representation and Exchange - Part 47: Shape Variation Tolerances," ISO/TC 184/SC 4 N222, 1993, available from the National Institute of Standards and Technology, Gaithersburg, Maryland.

3 Brodie, M.L., "On the Development of Data Models," On Conceptual Modelling, M.L. Brodie, J. Mylopoulos, and J.W. Schmidt editors, Springer-Verlag, New York, 1984. pp. 19-47.

4 Integrated Computer-Aided Manufacturing (ICAM) Architecture Part II, Volume V - Information Modeling Manual (IDEF1), Material Laboratory, U.S. Air Force Wright Aeronautical Laboratories, Dayton, OH, June 1981.

5 Integrated Computer-Aided Manufacturing (ICAM) Architecture Part II, Volume IV - Function Modeling Manual (IDEF0), Material Laboratory, U.S. Air Force Wright Aeronautical Laboratories, Dayton, OH, June 1981.

6 ISO/DIS 10303-11, "Product Data Representation and Exchange - Part 11: The EXPRESS Language Reference Manual," International Organization for Standardization, Geneva, Switzerland, 1993.

7 Nijssen, G.M. and Halpin, T.A., Conceptual Schema and Relational Database Design, Prentice Hall, New York, 1989.

8 ISO/DIS 10303-1 "Product Data Representation and Exchange - Part 1: Overview and Fundamental Principles," International Organization for Standardization, Geneva, Switzerland, 1993.

9 ISO 1101, "Technical drawings - Geometrical tolerancing - Tolerancing of form, orientation, location and run-out - Generalities, definitions, symbols, indications on drawings," International Organization for Standardization, Geneva, Switzerland, 1983.

10 ANSI Y14.5, "Dimensioning and tolerancing," ANSI Y14.5M - 1982. The American Society of Mechanical Engineering, New York, N.Y., 1982.

11 DMIS, "Dimensional Measuring Interface Standard," ANSI/CAM-I 101, version 2.1, Computer Aided Manufacturing - International Inc., Arlington, TX, 1990.

12 ISO 8015, "Technical drawings - Fundamental tolerancing principle," International Organization for Standardization, Geneva, Switzerland, 1983.

13 Requicha, A.A.G. and Chan, S.C. "Representation of geometric features, tolerances, and attributes in solid modelers based on constructive solid geometry," IEEE Journal of Robotics and Automation, Vol. RA-2, No. 3, 1986. pp. 156-166.

14 Johnson, R.H., "Dimensioning and Tolerancing," CAM-I report R-84-GM-02.2, Computer Aided Manufacturing - International, Inc., Arlington, TX, 1985.

15 Roy, U. and Liu, C.R., "Feature-based representational schema of a solid modeler for providing dimensioning and tolerancing information", Robotics and Computer-Integrated Manufacturing, Vol. 4, No. 3/4, 1988. pp. 335-351.

16 Shah, J.J. and Miller, D.W., "A Structure for Supporting Geometric Tolerances in Product Definition Systems for CIM", Manufacturing Review, Vol. 3, No. 1, 1990. pp. 23-31.

17 Wang, N. and Ozsoy, T.M., "A Schema to Represent Features, Dimensions, and Tolerances in Geometric Modeling", Journal of Manufacturing Systems, Vol. 10, No. 3, 1991. pp. 233-240.

18 Guilford, J. and Turner, J., "Representational primitives for geometric tolerancing", Computer-Aided Design, Vol. 25, No. 9, 1993. pp. 577-586.

19 ISO 286-1, "Bases of tolerances, deviations and fits," International Organization for Standardization, Geneva, Switzerland, 1988.

20 ISO/DIS 10303-41, "Product Data Representation and Exchange - Part 41: Integrated generic resources: Fundamentals of product description and support," International Organization for Standardization, Geneva, Switzerland, 1993.

21 ISO/DIS 10303-43, "Product Data Representation and Exchange - Part 43: Integrated generic resources: Representation structure," International Organization for Standardization, Geneva, Switzerland, 1993.

22 ISO 2692, "Technical drawings - Geometrical tolerancing - Maximum material principle," International Organization for Standardization, Geneva, Switzerland, 1988.

23 ISO 10578, "Technical drawings - Tolerancing of orientation and location - Projected tolerance zone," International Organization for Standardization, Geneva, Switzerland, 1992.

24 ISO 5459, "Technical drawings - Geometrical Tolerancing - Position tolerancing," International Organization for Standardization, Geneva, Switzerland, 1981.