

---

---

**Cutting tool data representation and  
exchange —**

Part 1:  
**Overview, fundamental principles and  
general information model**

*Représentation et échange des données relatives aux outils  
coupants —*

*Partie 1: Vue d'ensemble, principes fondamentaux et modèle général  
d'informations*



**PDF disclaimer**

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

STANDARDSISO.COM : Click to view the full PDF of ISO 13399-1:2006

© ISO 2006

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office  
Case postale 56 • CH-1211 Geneva 20  
Tel. + 41 22 749 01 11  
Fax + 41 22 749 09 47  
E-mail [copyright@iso.org](mailto:copyright@iso.org)  
Web [www.iso.org](http://www.iso.org)

Published in Switzerland

# Contents

Page

Foreword.....	vi
Introduction .....	vii
<b>1</b> <b>Scope</b> .....	<b>1</b>
<b>2</b> <b>Normative references</b> .....	<b>1</b>
<b>3</b> <b>Terms and definitions</b> .....	<b>2</b>
<b>4</b> <b>Fundamental concepts and assumptions</b> .....	<b>3</b>
<b>5</b> <b>Information requirements</b> .....	<b>6</b>
<b>5.1</b> <b>Units of Functionality</b> .....	<b>6</b>
5.1.1 <b>classification</b> .....	<b>6</b>
5.1.2 <b>dimension_tolerance</b> .....	<b>6</b>
5.1.3 <b>effectivity</b> .....	<b>7</b>
5.1.4 <b>external_reference_mechanism</b> .....	<b>7</b>
5.1.5 <b>individual_definition_structure</b> .....	<b>8</b>
5.1.6 <b>individual_management_data</b> .....	<b>8</b>
5.1.7 <b>item_definition_structure</b> .....	<b>9</b>
5.1.8 <b>item_property</b> .....	<b>9</b>
5.1.9 <b>product_management_data</b> .....	<b>10</b>
<b>5.2</b> <b>Application objects</b> .....	<b>11</b>
5.2.1 <b>default_language_string</b> .....	<b>11</b>
5.2.2 <b>alias_select</b> .....	<b>11</b>
5.2.3 <b>assigned_document_select</b> .....	<b>12</b>
5.2.4 <b>classification_source_select</b> .....	<b>12</b>
5.2.5 <b>classified_element_select</b> .....	<b>12</b>
5.2.6 <b>documented_element_select</b> .....	<b>12</b>
5.2.7 <b>effective_element_select</b> .....	<b>13</b>
5.2.8 <b>general_organizational_data_select</b> .....	<b>13</b>
5.2.9 <b>item_characteristic_select</b> .....	<b>14</b>
5.2.10 <b>item_property_select</b> .....	<b>14</b>
5.2.11 <b>limitation_definition_select</b> .....	<b>14</b>
5.2.12 <b>person_organization_select</b> .....	<b>14</b>
5.2.13 <b>property_source_select</b> .....	<b>14</b>
5.2.14 <b>string_select</b> .....	<b>15</b>
5.2.15 <b>alias_identification</b> .....	<b>15</b>
5.2.16 <b>application_context</b> .....	<b>16</b>
5.2.17 <b>assembly_association</b> .....	<b>16</b>
5.2.18 <b>assembly_definition</b> .....	<b>17</b>
5.2.19 <b>cartesian_coordinate_space</b> .....	<b>17</b>
5.2.20 <b>cartesian_coordinate_space_2d</b> .....	<b>17</b>
5.2.21 <b>cartesian_coordinate_space_3d</b> .....	<b>18</b>
5.2.22 <b>classification_system</b> .....	<b>18</b>
5.2.23 <b>classification_association</b> .....	<b>18</b>
5.2.24 <b>classification_association_relationship</b> .....	<b>19</b>
5.2.25 <b>classification_attribute</b> .....	<b>20</b>
5.2.26 <b>coating</b> .....	<b>21</b>
5.2.27 <b>coupling</b> .....	<b>21</b>
5.2.28 <b>cutting_condition</b> .....	<b>21</b>
5.2.29 <b>cutting_data_association</b> .....	<b>22</b>
5.2.30 <b>date_time</b> .....	<b>22</b>
5.2.31 <b>digital_document</b> .....	<b>23</b>

5.2.32	digital_file.....	23
5.2.33	din4000_reference.....	23
5.2.34	document.....	23
5.2.35	document_assignment.....	24
5.2.36	document_content_property.....	25
5.2.37	document_creation_property.....	26
5.2.38	document_file.....	27
5.2.39	document_format_property.....	28
5.2.40	document_location_property.....	29
5.2.41	document_representation.....	30
5.2.42	document_size_property.....	31
5.2.43	document_type_property.....	31
5.2.44	document_version.....	32
5.2.45	document_version_relationship.....	33
5.2.46	duration.....	34
5.2.47	effectivity.....	34
5.2.48	effectivity_assignment.....	36
5.2.49	effectivity_relationship.....	36
5.2.50	external_file_id_and_location.....	37
5.2.51	external_geometric_model.....	38
5.2.52	external_library_reference.....	38
5.2.53	external_model.....	39
5.2.54	external_picture.....	39
5.2.55	general_classification_hierarchy.....	40
5.2.56	general_classification.....	40
5.2.57	geometric_model_relationship_with_transformation.....	41
5.2.58	grade.....	42
5.2.59	grade_relationship.....	42
5.2.60	item.....	43
5.2.61	item_characteristic_association.....	44
5.2.62	item_definition.....	44
5.2.63	item_definition_relationship.....	45
5.2.64	item_instance.....	45
5.2.65	item_structure_association.....	46
5.2.66	item_version.....	47
5.2.67	item_version_relationship.....	47
5.2.68	language.....	49
5.2.69	limits_and_fits.....	49
5.2.70	location.....	50
5.2.71	mated_item_relationship.....	50
5.2.72	material_designation.....	50
5.2.73	mating_association.....	50
5.2.74	mating_definition.....	51
5.2.75	multi_language_string.....	51
5.2.76	named_size.....	52
5.2.77	numerical_value.....	52
5.2.78	organization.....	53
5.2.79	person.....	54
5.2.80	person_in_organization.....	54
5.2.81	person_organization_assignment.....	55
5.2.82	physical_document.....	56
5.2.83	physical_file.....	56
5.2.84	physical_item.....	57
5.2.85	physical_item_location_association.....	57
5.2.86	physical_item_definition.....	57
5.2.87	physical_item_state_association.....	57
5.2.88	physical_item_structure_association.....	57
5.2.89	physical_item_version.....	58
5.2.90	plib_class_reference.....	58
5.2.91	plib_property_reference.....	58

5.2.92	plus_minus_bounds .....	59
5.2.93	property .....	59
5.2.94	property_relationship .....	61
5.2.95	property_value .....	61
5.2.96	property_value_association .....	62
5.2.97	property_value_representation .....	63
5.2.98	property_value_representation_relationship .....	64
5.2.99	quantified_instance .....	65
5.2.100	realized_item_association .....	65
5.2.101	rectangular_size .....	65
5.2.102	specific_item_classification .....	66
5.2.103	specific_item_classification_hierarchy .....	67
5.2.104	state .....	67
5.2.105	string_value .....	67
5.2.106	string_with_language .....	67
5.2.107	substrate .....	68
5.2.108	transformation .....	68
5.2.109	transformation_2d .....	68
5.2.110	transformation_3d .....	68
5.2.111	unit .....	69
5.2.112	value_limit .....	69
5.2.113	value_limitation .....	69
5.2.114	value_list .....	70
5.2.115	value_range .....	70
5.2.116	value_with_unit .....	71
5.2.117	workpiece_feature .....	71
<b>6</b>	<b>Application interpreted model .....</b>	<b>73</b>
6.1	Mapping specification .....	73
6.1.1	classification UoF .....	75
6.1.2	dimension_tolerance UoF .....	93
6.1.3	effectivity UoF .....	96
6.1.4	external_reference_mechanism UoF .....	107
6.1.5	individual_definition_structure UoF .....	140
6.1.6	individual_management_data UoF .....	142
6.1.7	item_definition_structure UoF .....	151
6.1.8	item_property UoF .....	160
6.1.9	product_management_data UoF .....	186
6.2	AIM EXPRESS short listing .....	226
6.2.1	cutting_tool_schema types .....	226
6.2.2	cutting_tool_schema entities .....	231
6.2.3	cutting_tool_schema rules .....	232
<b>Annex A</b>	<b>(normative) AIM longform .....</b>	<b>285</b>
<b>Annex B</b>	<b>(normative) AIM short names .....</b>	<b>372</b>
<b>Annex C</b>	<b>(normative) ARM EXPRESS listing .....</b>	<b>376</b>
<b>Annex D</b>	<b>(informative) ARM EXPRESS-G diagrams .....</b>	<b>390</b>
<b>Annex E</b>	<b>(informative) AIM EXPRESS-G diagrams .....</b>	<b>411</b>
<b>Bibliography</b>	<b>.....</b>	<b>443</b>

## Foreword

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

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

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

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

ISO 13399-1 was prepared by Technical Committee ISO/TC 29, *Small tools*.

ISO 13399 consists of the following parts, under the general title *Cutting tool data representation and exchange*:

- *Part 1: Overview, fundamental principles and general information model*
- *Part 2: Reference dictionary for the cutting items* [Technical Specification]
- *Part 100: Definitions, principles and methods for reference dictionaries* [Technical Specification]

The following parts are under preparation:

- *Part 3: Reference dictionary for the tool items* [Technical Specification]
- *Part 4: Reference dictionary of terms for adaptive items* [Technical Specification]
- *Part 50: Reference dictionary for reference systems and common concepts* [Technical Specification]
- *Part 60: Reference dictionary for connection systems* [Technical Specification]

Assembly items is to form the subject of a future Part 5.

## Introduction

ISO 13399 provides the means to achieve an electronic representation of cutting tool data by providing the information structure needed to describe various data about cutting tools and cutting tool assemblies. It is intended to facilitate the use, manipulation and exchange of cutting tool data within and between manufacturing, distribution, and usage.

A cutting tool with defined cutting edges is used on a machine tool to remove workpiece material through a shearing action at the cutting edge(s) of the tool. Cutting tool data are characteristics of the cutting tool and its use that must be known and evaluated in order to make manufacturing decisions and to perform manufacturing operations.

ISO 13399 includes the data representation of everything between the workpiece and the machine tool. Information about inserts, solid tools (e.g. solid drill and solid endmill), assembled tools (e.g. boring bars, indexable drills and indexable milling cutters), adaptors (e.g. milling arbor and chucks), components (e.g. shims, screws and clamps) or any combination of the above can be exchanged.

The cutting tool data described include, but are not limited to, geometrical and dimensional data, identification and designation data, miscellaneous and spare part data, cutting material data, and component connectivity.

The use of the tool information model established by ISO 13399 will provide increased productivity for the user in the same way as do the tools. The effective management of tool information will improve the management of the tools themselves. Use of the tool information model will enable the identification of the “right” tool in every operation — from tool purchase, through planning, set-up in machine-tools, maintenance and reuse of the tools — with short lead times and with high reliability and product quality. Tool users will benefit from improved support from the tool vendors who will be able to provide a standard information product to accompany the tool products. Computer interfaces for information exchange will be more efficient.

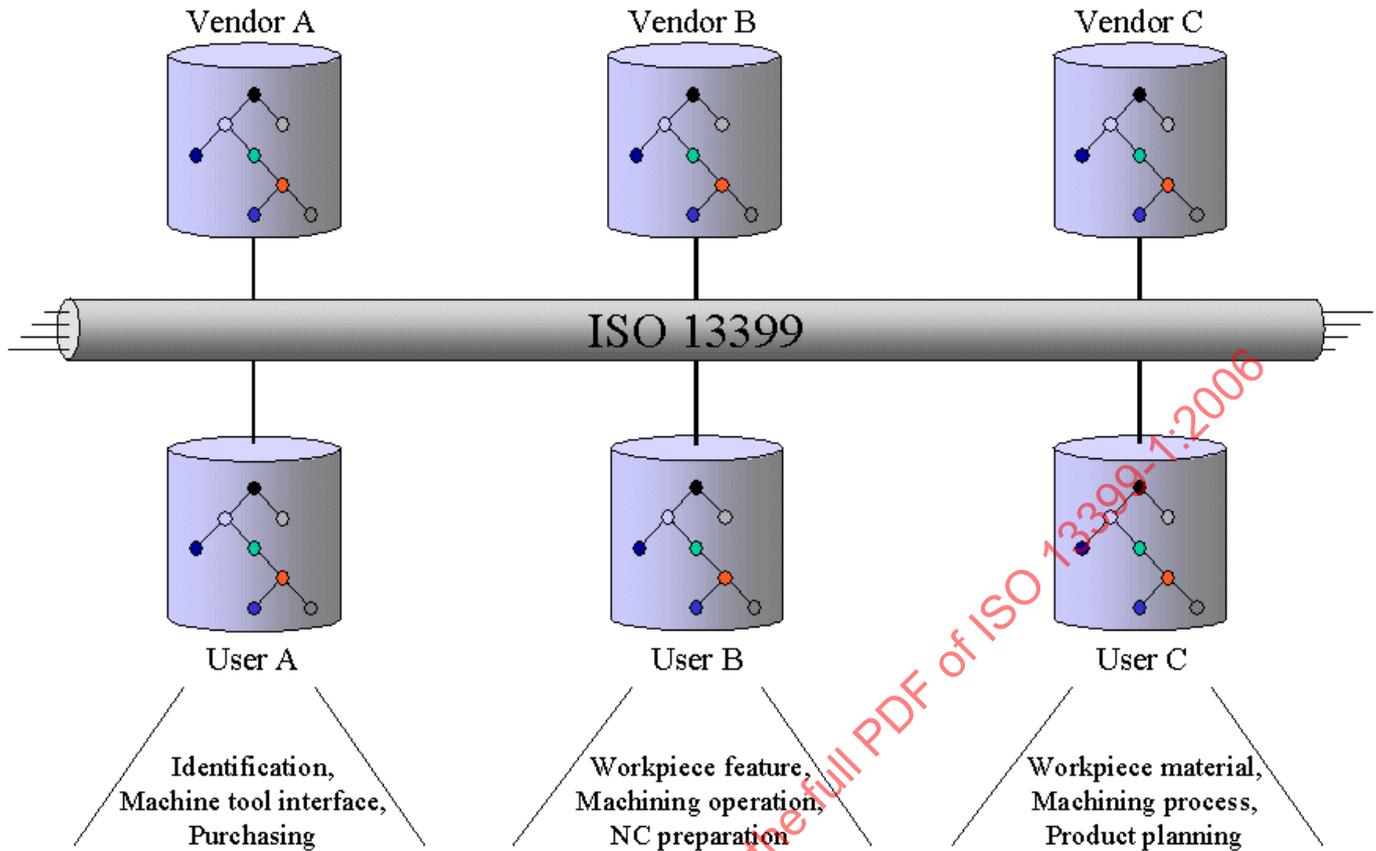
The representation of product data is defined by ISO 10303, which specifies the computer-interpretable representation of product information and the exchange of product data. The objective is to provide a neutral mechanism capable of describing products throughout their life cycle. This mechanism is suitable not only for neutral file exchange, but also as a basis for implementing and sharing product databases, and as a basis for archiving (ISO 10303-1).

This part of ISO 13399 uses the following resources according to ISO 10303:

- a) the EXPRESS language defined in ISO 10303-11;
- b) the file format for data exchange defined in ISO 10303-21 and ISO 10303-28;
- c) the integrated resources given in parts ISO 10303-40 to ISO 10303-56.

ISO 13399 is intended for use by manufacturers, tool vendors or producers, and developers of manufacturing software, among others. It provides a common structure for exchanging data about cutting tools (see Figure 1), and is intended to allow or improve several capabilities, including

- provision of a common set of definitions for use in describing cutting tools and cutting tool assemblies,
- the integration and sharing of cutting tool and assembly data between software applications,
- direct import of vendor cutting tool data into customer databases or applications, and
- a reduction in the level of effort required for manufacturers to maintain accurate and current cutting tool information from multiple sources and for multiple applications.



**Figure 1 — Role of ISO 13399 in communication of cutting tool data and potential user viewpoints**

While the use of ISO 13399 can facilitate the creation of an electronic catalogue, it does not set forth the rules or procedures for creating electronic catalogues. These are defined in ISO 13584.

The responsibility for providing different viewpoints lies outside the scope of ISO 13399. Viewpoints might be the material to be cut, the shape to be produced, the application for which the tool will be used or any other valid viewpoint.

**NOTE** Use of the modelled information for different purposes can be regarded as different viewpoints of the information. Although ISO 13399 does not provide the viewpoints, the information may be viewed from several alternative viewpoints (see Figure 1).

Different companies use different business models to determine their need for the communication of information about their products. For example, one cutting tool manufacturer could regrind its customers' tools while another could allow its customers to do the regrinding and provide the information to enable them to do so. Therefore, the two cutting tool manufacturers could have a different set of cutting tool properties to communicate using the information model and dictionaries provided by ISO 13399.

ISO 13399 defines only that information which could be communicated; it not specify what information must be communicated.

# Cutting tool data representation and exchange —

## Part 1: Overview, fundamental principles and general information model

### 1 Scope

This part of ISO 13399 covers the main categories of cutting tool data and the relationships between them. It provides a general information model of data representation and information exchange for these categories, as well as an overview of the principles of product data exchange used in ISO 13399 as a whole, a description of the other parts of ISO 13399 and a method for transferring cutting tool data.

The following is not covered by this part of ISO 13399:

- general data transfer and representations other than cutting tool data;
- data describing rules, guidelines and expert knowledge used to design and manufacture machined parts and cutting tools;
- data describing why a particular design or manufacturing decision was made.

Cutting tool data and exchange for specific items (cutting, tool, adaptive and assembly items) and the classification and definition of terms used in the description of cutting tools are defined in the other parts of ISO 13399, respectively, and referenced by them.

ISO 13399 establishes a means of communicating industrial data between different computer systems that is independent of any proprietary system.

### 2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO 639-2, *Codes for the representation of names of languages — Part 2: Alpha-3 code*

ISO 3002-1, *Basic quantities in cutting and grinding — Part 1: Geometry of the active part of cutting tools — General terms, reference systems, tool and working angles, chip breakers*

ISO 3002-3, *Basic quantities in cutting and grinding — Part 3: Geometric and kinematic quantities in cutting*

ISO 3166-1, *Codes for the representation of names of countries and their subdivisions — Part 1: Country codes*

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

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

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

ISO 10303-44, *Industrial automation systems and integration — Product data representation and exchange — Part 44: Integrated generic resource: Product structure configuration*

ISO 10303-45, *Industrial automation systems and integration — Product data representation and exchange — Part 45: Integrated generic resource: Materials*

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

ISO 10303-47, *Industrial automation systems and integration — Product data representation and exchange — Part 47: Integrated generic resource: Shape variation tolerances*

ISO/DIS 10303-56, *Industrial automation systems and integration — Product data representation and exchange — Part 56: Integrated generic resource: State*

ISO 10303-214, *Industrial automation systems and integration — Product data representation and exchange — Part 214: Application protocol: Core data for automotive mechanical design processes*

ISO 13584-26, *Industrial automation systems and integration — Parts library — Part 26: Logical resource: Information supplier identification*

ISO 13584-42, *Industrial automation systems and integration — Parts library — Part 42: Description methodology: Methodology for structuring part families*

### 3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO 13399-100 and the following apply.

#### 3.1 cutting tool

device or assembly of items for removing material from a workpiece through a shearing action at the defined cutting edge or edges of the device

NOTE A cutting tool could be the assembly of one or more adaptive items, a tool item and several cutting items on a tool item. See Figure 3.

#### 3.2 information model

formal model of a bounded set of facts, concepts, or instructions to meet a specified requirement

NOTE See Reference [3].

NOTE "An information model is a formal description of types of ideas, facts and processes which together form a model of a portion of interest of the real world and which provides an *explicit set of interpretation rules*. (If an information model is written in EXPRESS or any other computer-sensible representation, it has the additional quality of being computer processible.)"<sup>[9]</sup>.

## 4 Fundamental concepts and assumptions

ISO 13399 does not standardize cutting tools but the representation of information about cutting tools. Any standard dealing with information representation must be designed with the intent to standardize the *description* of the items that the standard deals with and not to standardize the items themselves. This method is also a step towards having standards that are not made obsolete by developments within the area where the standard applies <sup>[10]</sup>.

The intent of ISO 13399 is achieved by means of a computer-processable information model that defines the framework within which data values can be placed. The benefits of such an information model are that it can be interpreted by different computer applications and be generic enough to describe many different kinds of tools. The aim of ISO 13399 is to allow for the different tools and machining operations that can be expected in the future, while meeting the needs of current types of tool and their uses. It should also be possible to describe a tool independently of its use, since the combination of modern tools and machines may allow a tool to be used for many different kinds of cutting operation.

ISO 13399 specifies the representation of information on

- cutting tools,
- cutting items,
- tool items,
- adaptive items, and
- assembly items.

See Figure 2.

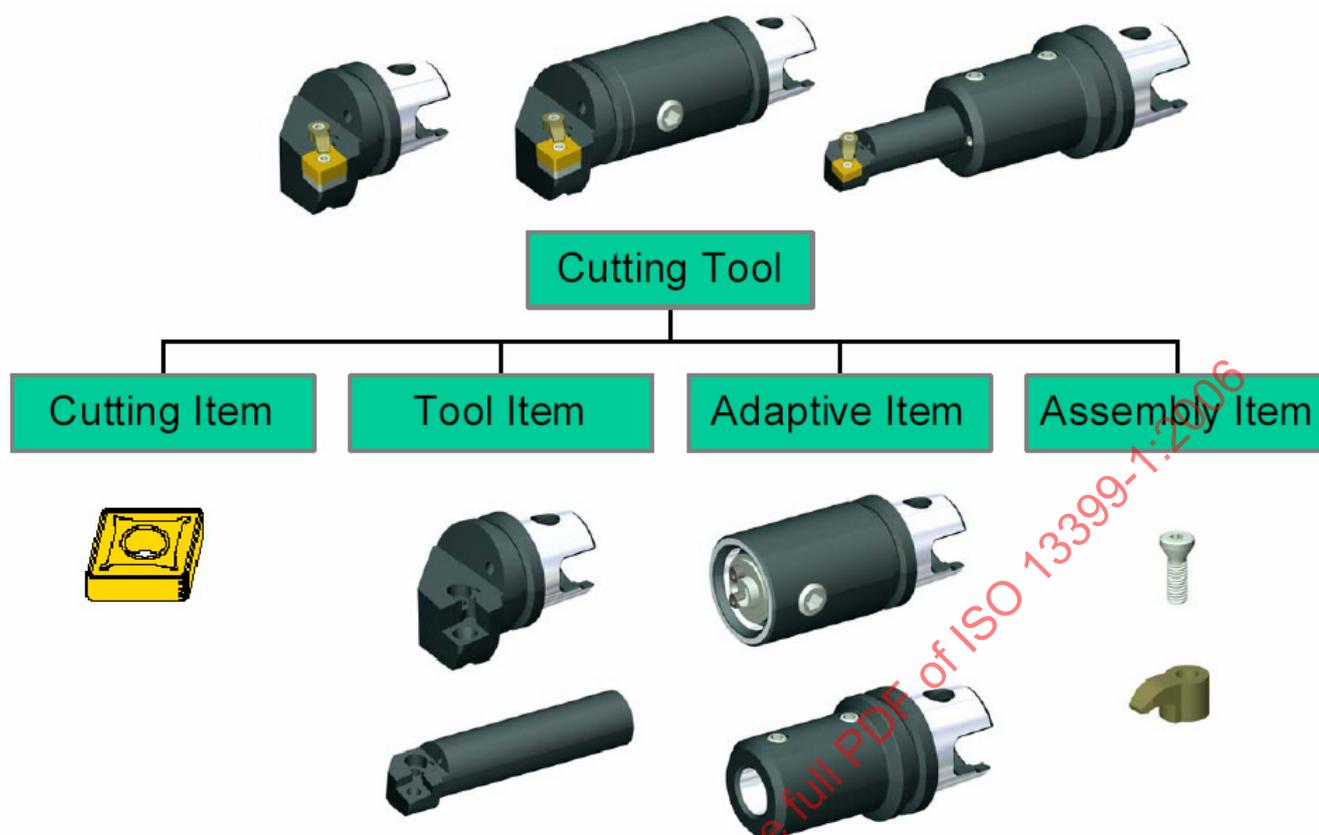


Figure 2 — Examples of the main groups

The definitions of these groups are given in their respective parts of ISO 13399. Different combinations of these groups are shown in Figure 3.

# Assemblies

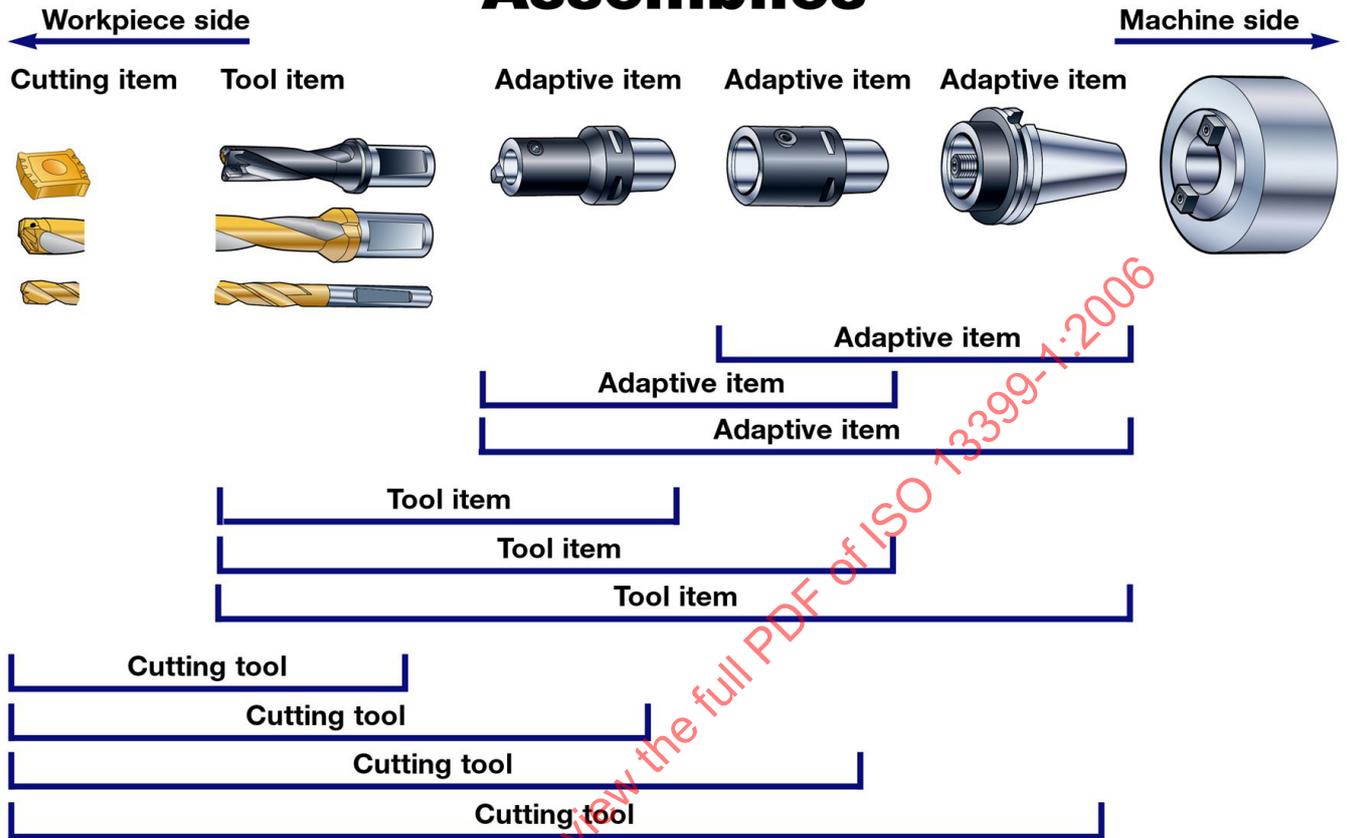


Figure 3 — Examples of different types of assemblies of items

The allowed relationships between these groups are shown in Table 1.

Table 1 — Matrix of Combinations

		Machine side		
		Tool item	Adaptive item	Assembly item
Workpiece side	Cutting item	X		X
	Tool item	X	X	X
	Adaptive item		X	X
	Assembly item	X	X	X

ISO 13399 does not deal with how to construct a cutting tool item, but provides an information representation for cutting tools. Tools may be classified according to any scheme, and viewed from many viewpoints.

## 5 Information requirements

### 5.1 Units of Functionality

#### 5.1.1 classification

This unit of functionality provides the capability to classify items into specific categories.

EXAMPLE Examples are the classification as raw materials, parts, or tools, or as safety or in-process items.

Additionally items may be classified by their attributes.

EXAMPLE Examples for categories of attribute based classification are screws, bolts, nuts, shafts, or brackets.

The following application objects are used by the classification UoF:

- classification\_association
- classification\_association\_relationship
- classification\_attribute
- classification\_system
- din4000\_reference
- external\_library\_reference
- general\_classification
- general\_classification\_hierarchy
- plib\_class\_reference
- plib\_property\_reference
- specific\_item\_classification
- specific\_item\_classification\_hierarchy

#### 5.1.2 dimension\_tolerance

This unit of functionality specifies the representation of geometric dimensions and of tolerances limiting geometric dimensions. Tolerances for geometric dimensions are characterized by plus-minus-tolerances, limits and fits, dimension ranges, or dimension limits and describe the allowable deviation range.

The following application objects are used by the dimension\_tolerance UoF:

- limits\_and\_fits
- plus\_minus\_bounds
- value\_limitation

### 5.1.3 effectivity

This unit of functionality provides the capability to represent information concerning the validity of data. Implicit propagation of data specifying validity is not available.

The validity of data can be expressed by effectivities that specify time ranges within which data may be used.

The following application objects are used by the effectivity UoF:

- duration
- effectivity
- effectivity\_assignment
- effectivity\_relationship

### 5.1.4 external\_reference\_mechanism

This unit of functionality provides a reference mechanism to specify external documents that are associated with objects defined in this part of ISO 13399. These documents may contain data that are not in scope of this part of ISO 13399. This includes conventional non-digital data and digital data in other computer interpretable formats.

**EXAMPLE** Technical drawings on paper, standards documents, or hand-written documents, are examples of non-digital data referenced by the external reference mechanism.

**EXAMPLE** CAD data in a system's native format, text documents in SGML, or NC-data in ISO 6983 are examples of digital data referenced by the external reference mechanism.

The following application objects are used by the external\_reference\_mechanism UoF:

- cartesian\_coordinate\_space
- cartesian\_coordinate\_space\_2d
- cartesian\_coordinate\_space\_3d
- digital\_document
- digital\_file
- document
- document\_assignment
- document\_content\_property
- document\_creation\_property
- document\_file
- document\_format\_property
- document\_location\_property
- document\_representation

- document\_size\_property
- document\_type\_property
- document\_version
- document\_version\_relationship
- external\_file\_id\_and\_location
- external\_geometric\_model
- external\_model
- external\_picture
- geometric\_model\_relationship\_with\_transformation
- named\_size
- physical\_document
- physical\_file
- rectangular\_size
- transformation
- transformation\_2d
- transformation\_3d

#### 5.1.5 individual\_definition\_structure

The following application objects are used by the individual\_definition\_structure UoF:

- physical\_item\_definition
- physical\_item\_structure\_association

#### 5.1.6 individual\_management\_data

The following application objects are used by the individual\_management\_data UoF:

- item\_version\_relationship
- location
- physical\_item
- physical\_item\_definition
- physical\_item\_location\_association
- physical\_item\_state\_association
- physical\_item\_structure\_association

- physical\_item\_version
- realized\_item\_association
- state

### 5.1.7 item\_definition\_structure

This unit of functionality provides the capability to represent relationships between items to build up various kinds of structures.

The structures are defined for item definitions, for item instances that are particular occurrences of item definitions or between item definitions and item instances.

Among these structures is a hierarchical assembly structure to represent the relationships between constituents and the assembly.

The following application objects are used by the item\_definition\_structure UoF:

- assembly\_association
- assembly\_definition
- item\_definition\_relationship
- item\_instance
- item\_structure\_association
- mated\_item\_relationship
- mating\_association
- mating\_definition
- quantified\_instance

### 5.1.8 item\_property

This unit of functionality specifies various properties for a part, such as material, recyclability, mass, and general properties.

The information about recyclability is required to describe whether and how material or component parts may be reused. The costs, as planned or calculated for a part, as well as a quality measure property to keep track of the results of some processes, e.g., prototype building, may be specified. The mass is required to express the weight of a part. Shape related properties that are rotation and translation invariant such as the centre of mass and the moments of inertia may also be specified.

The general properties are required to define property data which are not covered by one of the other UoFs, and, due to their variety, cannot be specified all by explicit application objects.

**EXAMPLE** Noise development of an engine, maintenance intervals of parts, and heat or noise absorption of insulating parts between the engine and passenger compartment are examples for general properties.

The following application objects are used by the item\_property UoF:

- coating

- coupling
- cutting\_condition
- cutting\_data\_association
- grade
- grade\_relationship
- item\_characteristic\_association
- item\_property\_association
- material\_designation
- property
- property\_relationship
- property\_value\_association
- property\_value\_representation
- property\_value\_representation\_relationship
- substrate
- workpiece\_feature

#### 5.1.9 product\_management\_data

This unit of functionality provides the capability to represent product management information. It includes information about items that are either raw materials, parts, or tools, about versions and views of items, and about persons and dates.

Additionally, this unit of functionality provides the capability to represent descriptive information about objects in different languages and to assign alias identifiers to objects.

The following application objects are used by the product\_management\_data UoF:

- alias\_identification
- application\_context
- date\_time
- item
- item\_definition
- item\_version
- item\_version\_relationship
- language

- multi\_language\_string
- numerical\_value
- organization
- person
- person\_in\_organization
- person\_organization\_assignment
- property\_value
- specific\_item\_classification
- specific\_item\_classification\_hierarchy
- string\_value
- string\_with\_language
- unit
- value\_limit
- value\_list
- value\_range
- value\_with\_unit

## 5.2 Application objects

SCHEMA ISO13399\_1;

### 5.2.1 default\_language\_string

EXPRESS specification:

```
TYPE default_language_string = STRING;
END_TYPE;
```

### 5.2.2 alias\_select

EXPRESS specification:

```
TYPE alias_select = SELECT
  (item,
   item_version,
   item_definition,
   property,
   grade,
   classification_attribute,
   classification_system,
   document,
   document_representation,
```

```
document_version,  
general_classification,  
item_instance,  
organization,  
document_type_property);  
END_TYPE;
```

### 5.2.3 assigned\_document\_select

#### EXPRESS specification:

```
TYPE assigned_document_select = SELECT  
  (document_version,  
   document);  
END_TYPE;
```

### 5.2.4 classification\_source\_select

#### EXPRESS specification:

```
TYPE classification_source_select = SELECT  
  (plib_class_reference,  
   document,  
   external_library_reference);  
END_TYPE;
```

### 5.2.5 classified\_element\_select

#### EXPRESS specification:

```
TYPE classified_element_select = SELECT  
  (item_version,  
   item,  
   item_definition,  
   document,  
   property,  
   property_value_association,  
   item_instance,  
   document_file,  
   document_version,  
   document_representation,  
   material_designation);  
END_TYPE;
```

### 5.2.6 documented\_element\_select

#### EXPRESS specification:

```
TYPE documented_element_select = SELECT  
  (item,  
   item_version,  
   classification_attribute,  
   classification_system,  
   item_definition,  
   general_classification,  
   item_structure_association,  
   item_instance,
```

```

organization,
person,
property,
specific_item_classification,
material_designation,
mated_item_relationship,
physical_item_structure_association);
END_TYPE;

```

### 5.2.7 effective\_element\_select

#### EXPRESS specification:

```

TYPE effective_element_select = SELECT
(document,
classification_system,
document_file,
document_representation,
document_version,
item,
item_version,
item_instance,
property,
property_value_association,
material_designation,
item_structure_association,
realized_item_association,
physical_item_state_association,
physical_item_location_association,
physical_item_structure_association);
END_TYPE;

```

### 5.2.8 general\_organizational\_data\_select

#### EXPRESS specification:

```

TYPE general_organizational_data_select = SELECT
(classification_association,
classification_system,
item_definition,
document,
document_file,
document_representation,
document_version,
general_classification,
item,
item_structure_association,
item_instance,
item_version,
item_version_relationship,
person_in_organization,
property,
property_value_association,
material_designation,
physical_item_structure_association);
END_TYPE;

```

### 5.2.9 item\_characteristic\_select

EXPRESS specification:

```
TYPE item_characteristic_select = SELECT
  (cutting_condition,
   coupling,
   workpiece_feature,
   material_designation,
   cutting_data_association,
   grade);
END_TYPE;
```

### 5.2.10 item\_property\_select

EXPRESS specification:

```
TYPE item_property_select = SELECT
  (item_definition,
   item_characteristic_select,
   document_file,
   document_representation,
   item_structure_association,
   item_instance,
   mated_item_relationship,
   physical_item_structure_association);
END_TYPE;
```

### 5.2.11 limitation\_definition\_select

EXPRESS specification:

```
TYPE limitation_definition_select = SELECT
  (limits_and_fits,
   plus_minus_bounds);
END_TYPE;
```

### 5.2.12 person\_organization\_select

EXPRESS specification:

```
TYPE person_organization_select = SELECT
  (organization,
   person_in_organization);
END_TYPE;
```

### 5.2.13 property\_source\_select

EXPRESS specification:

```
TYPE property_source_select = SELECT
  (plib_property_reference,
   external_library_reference,
   document);
END_TYPE;
```

### 5.2.14 string\_select

#### EXPRESS specification:

```
TYPE string_select = SELECT
    (default_language_string,
     multi_language_string);
END_TYPE;
```

### 5.2.15 alias\_identification

An `alias_identification` is a mechanism to associate an object with an additional identifier that is used to identify the object of interest in a different context, either in another organization, or in some other context. The scope of the `alias_identification` shall be specified either by the attribute 'alias\_scope' or by the attribute 'description'.

EXAMPLE A document (e.g., a book) may have a unique document id (ISBN) and an `alias_identification` as inventory number in the context of the inventory of a company.

#### EXPRESS specification:

```
ENTITY alias_identification;
    alias_id          : STRING;
    alias_scope       : OPTIONAL organization;
    alias_version_id  : OPTIONAL STRING;
    description       : OPTIONAL string_select;
    is_applied_to     : alias_select;
END_ENTITY;
```

#### 5.2.15.1 alias\_id

The `alias_id` specifies the identifier used in the context specified by the `alias_scope`, or by the description.

#### 5.2.15.2 alias\_scope

The `alias_scope` specifies the organization in which the `alias_identification` is valid.

The `alias_scope` need not be specified for a particular `alias_identification`.

#### 5.2.15.3 alias\_version\_id

The `alias_version_id` specifies the version of the object as known in the context of the `alias_identification`.

NOTE An `alias_version_id` may be applied even if the object of interest does not have versions.

The `alias_version_id` need not be specified for a particular `alias_identification`.

#### 5.2.15.4 description

The description specifies the type of the `alias_identification`.

EXAMPLE The description may be, e.g., 'inventory number'.

The description need not be specified for a particular `alias_identification`.

If present, there shall be exactly one object that defines the description for an `alias_identification`.

### 5.2.15.5 is\_applied\_to

The is\_applied\_to specifies the object that has an alias\_identification.

### 5.2.16 application\_context

An application\_context is a shared universe of discourse.

#### EXPRESS specification:

```
ENTITY application_context;  
  application_domain : STRING;  
  description : OPTIONAL string_select;  
END_ENTITY;
```

#### 5.2.16.1 application\_domain

The application\_domain is the identification of the applications for which an object may be relevant.

Where applicable the following values shall be used:

'machining':	The object may be relevant for machining;
'manufacturing simulation':	The object may be relevant for manufacturing simulation.
'nc preparation':	The object may be relevant for nc preparation.
'process planning':	The object may be relevant for process planning.
'product planning':	The object may be relevant for product planning.
'purchasing':	The object might be relevant for purchasing;

#### 5.2.16.2 description

The description specifies additional information about the application\_context.

The description need not be specified for a particular application\_context.

If present, there shall be exactly one object that defines the description for a application\_context.

### 5.2.17 assembly\_association

An assembly\_association is the relation between an assembly\_definition and an item\_instance representing a constituent of the assembly. The assembly\_definition and the item\_definition that serves as 'definition' of the item\_instance shall share at least one context.

NOTE The constituent may also be an assembly.

#### EXPRESS specification:

```
ENTITY assembly_association  
  SUBTYPE OF(item_structure_association);  
  SELF\item_structure_association.relatng : assembly_definition;  
END_ENTITY;
```

#### 5.2.17.1 relating

The relating specifies the assembly\_definition that has subordinate constituents.

### 5.2.18 assembly\_definition

An assembly\_definition is a definition of an item\_version that contains other subordinate objects.

An assembly\_definition is a type of item\_definition.

#### EXPRESS specification:

```
ENTITY assembly_definition
  SUBTYPE OF (item_definition);
  assembly_type : OPTIONAL STRING;
END_ENTITY;
```

#### 5.2.18.1 assembly\_type

The assembly\_type specifies the kind of the assembly\_definition.

EXAMPLE 'functional assembly', 'manufacturing assembly', and 'design assembly' are examples of an assembly\_type.

### 5.2.19 cartesian\_coordinate\_space

A cartesian\_coordinate\_space is a coordinate space in which geometric and annotation elements may be defined. It is either two-dimensional or three-dimensional. An origin for coordinate values is implicitly defined. The units applicable to the coordinate values of elements defined in the cartesian\_coordinate\_space are specified.

NOTE Cartesian transformation objects are used to specify the relationships between coordinate spaces.

Each cartesian\_coordinate\_space is a cartesian\_coordinate\_space\_3d or a cartesian\_coordinate\_space\_2d.

#### EXPRESS specification:

```
ENTITY cartesian_coordinate_space
  ABSTRACT SUPERTYPE OF (ONEOF (cartesian_coordinate_space_2d,
  cartesian_coordinate_space_3d));
  unit_of_values : OPTIONAL SET [2:?] OF unit;
END_ENTITY;
```

#### 5.2.19.1 unit\_of\_values

The unit\_of\_values specifies the various units in which any values are expressed. In the case where geometric elements are defined in the cartesian\_coordinate\_space, there shall be at least two units specified, the length unit and the plane angle unit. The same length unit is applied to each coordinate direction. Only one unit of a kind shall be specified.

NOTE If elements with different units are required, they have to be separated into different models with their own cartesian\_coordinate\_space.

EXAMPLE A length measure unit measured in inches and an angle measure unit measured in degrees are examples for two members of the attribute unit\_of\_values assigned to the same cartesian\_coordinate\_space.

The unit\_of\_values need not be specified for a particular cartesian\_coordinate\_space.

#### 5.2.20 cartesian\_coordinate\_space\_2d

A cartesian\_coordinate\_space\_2d is a two-dimensional coordinate space. Any two-dimensional geometric and annotation element shall be defined in a cartesian\_coordinate\_space\_2d.

A cartesian\_coordinate\_space\_2d is a type of cartesian\_coordinate\_space.

EXPRESS specification:

```
ENTITY cartesian_coordinate_space_2d
  SUBTYPE OF (cartesian_coordinate_space);
END_ENTITY;
```

**5.2.21 cartesian\_coordinate\_space\_3d**

A cartesian\_coordinate\_space\_3d is a three-dimensional coordinate space. Any three-dimensional geometric data shall be defined in a cartesian\_coordinate\_space\_3d.

A cartesian\_coordinate\_space\_3d is a type of cartesian\_coordinate\_space.

EXPRESS specification:

```
ENTITY cartesian_coordinate_space_3d
  SUBTYPE OF (cartesian_coordinate_space);
END_ENTITY;
```

**5.2.22 classification\_system**

A classification\_system is the scheme used to define the categorization of an item.

EXAMPLE A dictionary collecting the conventional definitions of technical properties is an example of classification\_system.

EXPRESS specification:

```
ENTITY classification_system;
  description : OPTIONAL string_select;
  id : STRING;
INVERSE
  allowed_classification : SET OF general_classification FOR
  used_classification_system;
END_ENTITY;
```

**5.2.22.1 description**

The description specifies additional information about the classification\_system.

The description need not be specified for a particular classification\_system.

If present, there shall be exactly one object that defines the description for a classification\_system.

**5.2.22.2 id**

The id specifies the identifier of the classification\_system.

**5.2.23 classification\_association**

A classification\_association associates a general\_classification with an object.

EXPRESS specification:

```
ENTITY classification_association;
```

```

associated_classification : general_classification;
classified_element       : classified_element_select;
defintional             : OPTIONAL BOOLEAN;
role                    : OPTIONAL STRING;
END_ENTITY;

```

#### 5.2.23.1 associated\_classification

The associated\_classification specifies the general\_classification object that provides classification information.

#### 5.2.23.2 classified\_element

The classified\_element specifies the object that is classified.

There shall be exactly one object that defines the classified\_item for a classification\_association.

#### 5.2.23.3 defintional

The defintional specifies whether a general\_classification serves as definition. A value of 'true' indicates that the general\_classification is defintional. The 'associated\_classification' does not take precedence over the descriptions of the 'classified\_element' made using property\_value objects.

The attribute defintional shall only be used, if the classified\_element is of one of the types material\_designation, item, item\_version, or item\_definition.

The defintional need not be specified for a particular classification\_association.

#### 5.2.23.4 role

The role specifies the relationship between the general\_classification and the associated item.

The role need not be specified for a particular classification\_association.

#### 5.2.24 classification\_association\_relationship

A classification\_association\_relationship is a relationship between two classification\_association objects.

#### EXPRESS specification:

```

ENTITY classification_association_relationship;
  related          : classification_association;
  relating         : classification_association;
  relationship_type : STRING;
END_ENTITY;

```

#### 5.2.24.1 related

Specifies the second general\_classification that is part of the classification\_relationship

#### 5.2.24.2 relating

Specifies the first general\_classification that is part of the classification\_relationship

#### 5.2.24.3 relationship\_type

Specifies the type of relationship represented by the classification\_relationship

Where applicable the following values shall be used:

'combination': specifies that the object being classified, by the general\_classification objects that are a part of the classification\_relationship, is classified as being a combination of the general\_classification objects.

'alternative': specifies that the object being classified can be considered as being either one or the other of the participating general\_classifications.

### 5.2.25 classification\_attribute

A classification\_attribute is a characteristic used to classify an object associated with the corresponding general\_classification. The definition attribute of each 'allowed\_value' shall refer to the property identified within 'attribute\_definition'.

#### EXPRESS specification:

```
ENTITY classification_attribute;  
  allowed_value          : SET OF property_value_representation;  
  associated_classification : general_classification;  
  attribute_definition   : property;  
  description            : OPTIONAL string_select;  
  id                     : STRING;  
  name                   : OPTIONAL string_select;  
END_ENTITY;
```

#### 5.2.25.1 allowed\_value

The allowed\_value specifies the set of property\_value\_representation objects that represent characteristic values of the classification\_attribute.

#### 5.2.25.2 associated\_classification

The associated\_classification specifies the general\_classification the classification\_attribute is a characteristic of.

#### 5.2.25.3 attribute\_definition

The attribute\_definition specifies the property that characterizes the allowed values.

NOTE The specification of compound characteristics can be realized by using property\_relationship with 'relation\_type' set to 'peer' or 'decomposition'.

#### 5.2.25.4 description

The description specifies additional information about the classification\_attribute.

The description need not be specified for a particular classification\_attribute.

If present, there shall be exactly one object that defines the description for a classification\_attribute.

#### 5.2.25.5 id

The id specifies the identifier of the classification\_attribute that shall be unique within the scope of the associated general\_classification.

EXAMPLE 'a' or 'b' for length or width attributes or 'r' for radius attributes are examples for identifiers of classification\_attribute objects. The meaning of such ids is usually specified in external sources.

**5.2.25.6 name**

The name specifies the word or group of words by which the classification\_attribute is referred to.

EXAMPLE 'length', 'width', or 'radius' are examples for the name of a classification\_attribute.

The name need not be specified for a particular classification\_attribute.

If present, there shall be exactly one object that defines the name for a classification\_attribute.

**5.2.26 coating**EXPRESS specification:

```
ENTITY coating;
    coating_name      : STRING;
    coating_process   : STRING;
END_ENTITY;
```

**5.2.27 coupling**EXPRESS specification:

```
ENTITY coupling;
    coupling_type     : STRING;
    pieces            : OPTIONAL numerical_value;
    side              : STRING;
    size              : numerical_value;
    style             : STRING;
END_ENTITY;
```

**5.2.27.1 coupling\_type**

Mount style may be one of:

Code	Description
0	Other clamping systems, not listed below
1	Without fixing hole [ISO 1832:1991]
2	With cylindrical fixing hole [ISO 1832:1991]
3	With partly cylindrical fixing hole, countersink on one side only [ISO 1832:1991]
4	With partly cylindrical fixing hole, countersinks on both sides [ISO 1832:1991]
5	With notch clamping
6	With convex prismatic cross section
7	With concave prismatic cross section

**5.2.27.2 size**

The size of the coupling.

EXAMPLE diameter of a through hole.

**5.2.28 cutting\_condition**

Conditions relevant to the cutting process, e.g. finishing, roughing, interrupted cut, smooth cut. There may be more than one cutting\_condition relevant for any item.

EXPRESS specification:

```
ENTITY cutting_condition;  
    condition_name      :  STRING;  
END_ENTITY;
```

**5.2.29 cutting\_data\_association**

Cutting data is information about the cutting conditions and performance of the cutting tool.

NOTE Certain cutting data only appear through the combination of different items, machine tools and workpieces.

EXPRESS specification:

```
ENTITY cutting_data_association;  
    associated_material  :  material_designation;  
    depth_of_cut        :  value_with_unit;  
    feed                :  value_with_unit;  
    speed               :  value_with_unit;  
END_ENTITY;
```

**5.2.29.1 associated\_item**

Specifies one or more items associated with the cutting data.

**5.2.29.2 associated\_material**

Specifies the workpiece\_material associated with the cutting data.

**5.2.29.3 depth\_of\_cut**

Specifies the perpendicular distance between the original and final surfaces of the workpiece.

**5.2.29.4 feed**

Specifies the feed according to ISO 3002-3.

**5.2.29.5 speed**

Specifies the cutting speed according to ISO 3002-1

**5.2.30 date\_time**

A date\_time is the specification of a date and an optional time of day.

EXPRESS specification:

```
ENTITY date_time;  
    date :  STRING;  
    time :  OPTIONAL STRING;  
END_ENTITY;
```

**5.2.30.1 date**

The date specifies the calendar time, defined according to the Gregorian calendar, conveying information about the year, the month, and the day in no specific order. The representation of a date shall be complete, i.e., millenium, century, and year-within-century data shall be included.

**5.2.30.2 time**

The time specifies a moment of occurrence measured by hour, minute, and second.

The time need not be specified for a particular date\_time.

**5.2.31 digital\_document**

A digital\_document is a piece of product data that is archived in a digital format.

A digital\_document is a type of document\_representation.

EXPRESS specification:

```
ENTITY digital_document
  SUBTYPE OF(document_representation);
  file : SET OF digital_file;
END_ENTITY;
```

**5.2.31.1 file**

The file specifies a computer interpretable realization of the digital\_document.

**5.2.32 digital\_file**

A digital\_file contains computer interpretable data.

A digital\_file is a type of document\_file.

EXPRESS specification:

```
ENTITY digital_file
  SUBTYPE OF(document_file);
  INVERSE
    associated_model_space : SET OF external_model FOR is_defined_as;
END_ENTITY;
```

**5.2.33 din4000\_reference**

DIN 4000 reference is the information about the comparability of a property between the ISO 13399 standard and the DIN 4000 standard For the reference description the properties DIN 4000 - Part No and characteristics code no is required.

EXPRESS specification:

```
ENTITY din4000_reference
  SUBTYPE OF(external_library_reference);
  characteristics_code_no : STRING;
  part_no : STRING;
END_ENTITY;
```

**5.2.34 document**

A document is a logical document that serves as the identifier for a container for some product data.

**EXAMPLE** A document can either be any kind of drawing, solid model, or visible information about the item that is geometrically described with the properties of the ISO 13399.

EXPRESS specification:

```
ENTITY document;  
  description      : OPTIONAL string_select;  
  document_id     : STRING;  
  name            : string_select;  
INVERSE  
  associated_version : SET [1:?] OF document_version FOR  
  associated_document;  
END_ENTITY;
```

**5.2.34.1 description**

The description specifies additional information about the document.

The description need not be specified for a particular document.

If present, there shall be exactly one object that defines the description for a document.

**5.2.34.2 document\_id**

The document\_id specifies the identifier of the document.

**5.2.34.3 name**

The name specifies the word or group of words by which the document is referred to.

There shall be exactly one object that defines the name for a document.

**5.2.35 document\_assignment**

A document\_assignment is a mechanism to associate a document with an object, where the assigned document provides information about the object it is associated to.

EXPRESS specification:

```
ENTITY document_assignment;  
  assigned_document : assigned_document_select;  
  is_assigned_to   : documented_element_select;  
  role             : STRING;  
END_ENTITY;
```

**5.2.35.1 assigned\_document**

The assigned\_document specifies a document, or a document\_version that is used to provide information.

There shall be exactly one object that defines the assigned\_document for a document\_assignment.

**5.2.35.2 is\_assigned\_to**

The is\_assigned\_to specifies the object that information is provided for.

There shall be exactly one object that the document\_assignment is\_assigned\_to.

**5.2.35.3 role**

The role specifies the meaning of the document\_assignment.

Where applicable the following values shall be used:

'additional information':	The assigned document provides information that is relevant for the associated object, but is not a description of the associated object itself;
'behavior':	The assigned document specifies information about the behaviour of the associated object;
'description':	The assigned document provides textual information for the associated object itself;
'informative':	The assigned document may or may not be considered;
'mandatory':	The associated object shall conform to the content of the assigned document;
'mathematical description':	The assigned document specifies the associated object by providing the algorithmic specification of its behavior.

### 5.2.36 document\_content\_property

A document\_content\_property specifies characteristics precisising the content of a document\_file or of a document\_representation. At least one of the optional attributes shall be specified for each instance of this object.

In the case where a document\_content\_property is referred by a document\_representation, the characteristics apply to all individual document\_file objects, whereas in the case where it is referred by a document\_file, the characteristics apply on an individual basis.

#### EXPRESS specification:

```
ENTITY document_content_property;
  detail_level      : OPTIONAL STRING;
  geometry_type    : OPTIONAL STRING;
  languages        : SET OF language;
  real_world_scale : OPTIONAL numerical_value;
END_ENTITY;
```

#### 5.2.36.1 detail\_level

The detail\_level specifies the level of detail that the document\_file or the document\_representation provides.

Where applicable the following values shall be used:

'rough 3d shape':	3D shape model without edge rounds and fillets;
'rounded edges':	3D shape model with edge rounds and fillets.

The detail\_level need not be specified for a particular document\_content\_property.

#### 5.2.36.2 geometry\_type

The geometry\_type specifies the kind or kinds of geometry that an object contains.

Where applicable the following values shall be used:

'3D wireframe model':	The document contains a 3D shape model in wireframe representation;
'2D shape':	The document contains a 2D shape model or contours only;
'surface model':	The document contains a 3D shape model in surface representation;
'closed volume':	The document contains a 3D shape model in closed body topological surface representation;
'solid model':	The document contains a 3D shape model in advanced boundary representation;
'solid and surface model':	The document contains a 3D shape model in surface and advanced boundary representation;

- 'assembly': The document contains an assembly structure with reference to the assembled components and their transformation matrices;
- 'assembly with mating elements': The document contains an assembly structure including the mating components only, such as screws or rivets, with exact positioning information. This assembly representation is intended to be overlaid with the assembly structure for the main components;
- '2D drawing': The document contains a technical drawing without 3D shape representation;
- 'drawing derived from 3D data': The document contains a technical drawing that has been derived from a 3D shape model;
- 'drawing related to 3D data': The document contains a technical drawing that visualizes a 3D shape model and possibly establishes associative links to the 3D shape model.

The geometry\_type need not be specified for a particular document\_content\_property.

### 5.2.36.3 languages

The languages specifies which language or languages are used in the characterized objects.

EXAMPLE 'Japanese' and 'German' are examples for the fact that annotation on a drawing is provided in the language 'Japanese' or 'German' respectively.

### 5.2.36.4 real\_world\_scale

The real\_world\_scale specifies the scale that is used in the document\_file or in the document\_representation the document\_content\_property is referred by.

The real\_world\_scale need not be specified for a particular document\_content\_property.

### 5.2.37 document\_creation\_property

A document\_creation\_property specifies characteristics of document\_file or of document\_representation objects. It specifies the context of the creation of the object. At least one of the optional attributes shall be specified for each instance of this object.

In the case where a document\_creation\_property is referred by a document\_representation the characteristics apply to all individual document\_file objects, whereas in the case it is referred by a document\_file, the characteristics apply on an individual basis.

#### EXPRESS specification:

```
ENTITY document_creation_property;  
    creating_interface : OPTIONAL STRING;  
    creating_system   : STRING;  
    operating_system  : OPTIONAL STRING;  
END_ENTITY;
```

#### 5.2.37.1 creating\_interface

The creating\_interface specifies the computer application used to create the document\_file or document\_representation object.

EXAMPLE 'Postscript Printer Driver' is an example for a creating interface of a physical\_document.

The creating\_interface need not be specified for a particular document\_creation\_property.

### 5.2.37.2 creating\_system

The creating\_system specifies the computer application or the machine which is used to create the object that is characterized.

### 5.2.37.3 operating\_system

The operating\_system specifies the operating system that is used to execute the computer application that created the characterized object.

The operating\_system need not be specified for a particular document\_creation\_property.

### 5.2.38 document\_file

A document\_file is one of potentially more files on a computer system or in actual stacks of paper that make up a document\_representation.

#### EXPRESS specification:

```

ENTITY document_file
  ABSTRACT SUPERTYPE OF (ONEOF(digital_file, physical_file));
  content                : OPTIONAL document_content_property;
  creation               : OPTIONAL document_creation_property;
  document_file_type    : OPTIONAL document_type_property;
  external_id_and_location : SET OF external_file_id_and_location;
  file_format           : OPTIONAL document_format_property;
  file_id               : STRING;
  size                  : OPTIONAL document_size_property;
  version_id            : OPTIONAL STRING;
END_ENTITY;
```

#### 5.2.38.1 content

The content characterizes the content of the document\_file.

The content need not be specified for a particular document\_file.

#### 5.2.38.2 creation

The creation specifies further details of the context of the creation of the document\_file.

The creation need not be specified for a particular document\_file.

#### 5.2.38.3 document\_file\_type

The document\_file\_type specifies the format of the document\_file. It shall only be specified, if the document\_file does not participate in a document.

The document\_file\_type need not be specified for a particular document\_file.

#### 5.2.38.4 external\_id\_and\_location

The external\_id\_and\_location specifies alternatives of the identifier and location of the document\_file.

EXAMPLE A copy of a document may be found in another department with a different id.

#### 5.2.38.5 file\_format

The file\_format specifies the characteristics of the document\_file that specify the format of the object.

The file\_format need not be specified for a particular document\_file.

#### 5.2.38.6 file\_id

The file\_id specifies the identifier which is used to locate the file either on a computer system or in a repository of paper documents.

#### 5.2.38.7 size

The size specifies characteristics for the size of the document\_file.

The size need not be specified for a particular document\_file.

#### 5.2.38.8 version\_id

The version\_id specifies the identification of the version that distinguishes one document\_file object from other versions of document\_file objects with the same file\_id.

The version\_id need not be specified for a particular document\_file.

#### 5.2.39 document\_format\_property

A document\_format\_property specifies characteristics of a document\_file or of a document\_representation that specify the format of the object. At least one of the optional attributes shall be specified for each instance of this object.

In the case where a document\_format\_property is referred by a document\_representation, the characteristics apply to all individual document\_file objects, whereas in the case it is referred by a document\_file the characteristics apply on an individual basis.

##### EXPRESS specification:

```
ENTITY document_format_property;  
    character_code : OPTIONAL STRING;  
    data_format : OPTIONAL STRING;  
    size_format : OPTIONAL rectangular_size;  
END_ENTITY;
```

#### 5.2.39.1 character\_code

The character\_code specifies the character code that is used in the characterized object.

Where applicable the following values shall be used:

- 'binary': The document contains data in binary format;
- 'IEC 61286': The coded character set used to encode the document data according to IEC 61286;
- 'ISO 646': The coded character set used to encode the document data according to ISO 646;

NOTE The character set in ISO 646 is identical to the character set commonly known as ASCII

- 'ISO 3098-1': The coded character set used to encode the document data is according to ISO 3098-1;
- 'ISO 6937': The coded character set used to encode the document data is according to ISO/IEC 6937;
- 'ISO 8859-1': The coded character set used to encode the document data according to ISO 8859-1;

NOTE The character set in ISO 8859-1 is identical to the character set commonly known as LATIN-1.

'ISO 10646': The coded character set used to encode the document data according to ISO/IEC 10646.

The character\_code need not be specified for a particular document\_format\_property.

### 5.2.39.2 data\_format

The data\_format specifies the convention that was used to structure the information in the characterized object.

Where applicable the following values shall be used:

'DXF':	The document contains data in Drawing Exchange File format;
'IGES':	The document contains data in Initial Graphics Exchange Specification format;
'ISO 10303-203':	The document contains data in ISO 10303-203 format;
'ISO 10303-214':	The document contains data in ISO 10303-214 format;
'TIFF CCITT GR4':	The document contains data in TIFF CCITT GR4 format;
'VDAFS':	The document contains data in VDAFS format;
'VOXEL':	The document contains data in VOXEL format.

The data\_format need not be specified for a particular document\_format\_property.

### 5.2.39.3 size\_format

The size\_format specifies the dimensions of a physical presentation of the object the size\_format is provided for.

EXAMPLE 'ISO A0' is an example for the size\_format of a drawing that is stored digitally.

The size\_format need not be specified for a particular document\_format\_property.

### 5.2.40 document\_location\_property

A document\_location\_property specifies where a document\_file or a document\_representation can be found in a digital or physical data storage system.

In the case where a document\_location\_property is referred by a document\_representation, the characteristics apply to all individual objects, whereas in the case it is referred by a document\_file, the characteristics apply on an individual basis.

#### EXPRESS specification:

```
ENTITY document_location_property;
    location_name : STRING;
END_ENTITY;
```

#### 5.2.40.1 location\_name

The location\_name specifies the location, where the object that refers to the document\_location\_property, can be found. 'C:\mpbs\programs' and '/usr/local/bin' are examples for a location\_name.

NOTE Multiple paths may be specified for a single object, e.g., a database in the context of an electronic vault.

EXAMPLE The linking mechanism of many operating systems allows references to a single object from various places in the file system without copying its contents.

### 5.2.41 document\_representation

A document\_representation is one of potentially more alternative representations of a document.

EXAMPLE A logical document, which contains a shape model, may be represented in the different native formats of two 3D CAD systems.

Each document\_representation is a digital\_document or a physical\_document.

#### EXPRESS specification:

```
ENTITY document_representation
  ABSTRACT SUPERTYPE OF (ONEOF(digital_document, physical_document));
  associated_document_version : document_version;
  common_location            : SET OF document_location_property;
  content                    : OPTIONAL document_content_property;
  creation                   : OPTIONAL document_creation_property;
  description                 : OPTIONAL string_select;
  id                          : STRING;
  representation_format      : OPTIONAL document_format_property;
  size                        : OPTIONAL document_size_property;
END_ENTITY;
```

#### 5.2.41.1 associated\_document\_version

The associated\_document\_version specifies the version of the logical document that is being represented.

#### 5.2.41.2 common\_location

The common\_location specifies the location of a document\_representation, where all its constituents can be found.

NOTE Different common locations represent alternative representations of the same physical location.

#### 5.2.41.3 content

The content specifies characteristics of the content of the document\_representation.

The content need not be specified for a particular document\_representation.

#### 5.2.41.4 creation

The creation specifies further details of the creation of the document\_representation.

The creation need not be specified for a particular document\_representation.

#### 5.2.41.5 description

The description specifies additional information about the document\_representation.

The description need not be specified for a particular document\_representation.

If present, there shall be exactly one object that defines the description for a document\_representation.

#### 5.2.41.6 id

The id specifies the identifier of the document\_representation.

**5.2.41.7 representation\_format**

The `representation_format` specifies the format of the document represented by `document_representation`.

The `representation_format` need not be specified for a particular `document_representation`.

**5.2.41.8 size**

The `size` specifies the size of the represented document.

The `size` need not be specified for a particular `document_representation`.

**5.2.42 document\_size\_property**

A `document_size_property` specifies the size of a `document_file` or of a `document_representation` object. At least one of the optional attributes shall be specified for each instance of this object.

In the case where a `document_size_property` is referred by a `document_representation`, the size information is the sum of the sizes of all individual objects that are collected by this object, whereas in the case it is referred by a `document_file`, the size information is the one of the individual objects that is referenced.

EXPRESS specification:

```
ENTITY document_size_property;
  file_size : OPTIONAL value_with_unit;
  page_count : OPTIONAL value_with_unit;
END_ENTITY;
```

**5.2.42.1 file\_size**

The `file_size` specifies the `value_with_unit` that represents the size of a digitally stored document. The `file_size` shall only be applied in cases where the `document_size_property` is referred by a `digital_document` or a `document_file`.

EXAMPLE '15021 Bytes' and 'less than 500 Bytes' are examples for a `file_size`.

The `file_size` need not be specified for a particular `document_size_property`.

**5.2.42.2 page\_count**

The `page_count` specifies the number of pages of the application object the `document_size_property` is referred by. The `page_count` shall only be used in cases where the `document_size_property` is referred by a `physical_file` or a `physical_document`.

EXAMPLE '42 pages' and 'more than 1 page' are examples of a `page_count`.

The `page_count` need not be specified for a particular `document_size_property`.

**5.2.43 document\_type\_property**

A `document_type_property` specifies the kind of a `document_file`.

EXPRESS specification:

```
ENTITY document_type_property;
  document_type_name : STRING;
  used_classification_system : OPTIONAL classification_system;
END_ENTITY;
```

#### 5.2.43.1 document\_type\_name

The document\_type\_name specifies the word or the group of words that describe the kind of object the characteristics are provided for.

Where applicable the following values shall be used:

'geometry':	The document represents a shape model;
'NC data':	The document represents numerical control data;
'FE data':	The document represents finite element data;
'sample data':	The document represents measured data;
'process plan':	The document represents process planning data;
'check plan':	The document represents quality control planning data;
'drawing':	The document represents a technical drawing.

#### 5.2.43.2 used\_classification\_system

The used\_classification\_system specifies the classification\_system the document\_type\_name is defined in.

EXAMPLE A document\_type\_property may be used with 'specification type 1' and 'specification type 2' as values for the attribute 'document\_type\_name' due to a company wide definition. By inclusion of these types in a general\_classification with name 'specification' and description 'specification of company XY' these types of documents are better classified and the meaning of 'specification type 1' and 'specification type 2' can be further detailed in an attached document to the general\_classification or by a reference to a classification\_system for document types with that organization.

The used\_classification\_system need not be specified for a particular document\_type\_property.

#### 5.2.44 document\_version

A document\_version is a release of a document.

##### EXPRESS specification:

```
ENTITY document_version;  
    associated_document : document;  
    description         : OPTIONAL string_select;  
    id                  : STRING;  
END_ENTITY;
```

#### 5.2.44.1 associated\_document

The associated\_document specifies the logical document the document\_version is a version of.

#### 5.2.44.2 description

The description specifies additional information about the document\_version.

The description need not be specified for a particular document\_version.

If present, there shall be exactly one object that defines the description for a document\_version.

#### 5.2.44.3 id

The id specifies the identifier of the document\_version. The id shall be unique within the scope of the associated document.

### 5.2.45 document\_version\_relationship

A document\_version\_relationship is a relationship between two document\_version objects.

#### EXPRESS specification:

```
ENTITY document_version_relationship;
  description      : OPTIONAL string_select;
  related         : document_version;
  relating        : document_version;
  relation_type   : STRING;
END_ENTITY;
```

#### 5.2.45.1 description

The description specifies additional information about the document\_version\_relationship.

The description need not be specified for a particular document\_version\_relationship.

If present, there shall be exactly one object that defines the description for a document\_version\_relationship.

#### 5.2.45.2 related

The related specifies the second of the two objects related by the document\_version\_relationship.

NOTE The semantics of this attribute are defined by the attribute relation\_type.

#### 5.2.45.3 relating

The relating specifies the first of the two objects related by the document\_version\_relationship.

NOTE The semantics of this attribute are defined by the attribute relation\_type.

#### 5.2.45.4 relation\_type

The relation\_type specifies the meaning of the relationship.

Where applicable the following values shall be used:

'derivation': The application object defines a deriving relationship where the related document\_version is based on the relating document\_version which is an earlier version of the same or of a different document;  
 'hierarchy': The application object defines a hierarchical relationship where the related document\_version is a sub version of the relating document\_version;

EXAMPLE 'Revision 1.1' and 'revision 1.2' are examples of sub-versions of 'version 1'.

'sequence': The application object defines a version sequence where the relating document\_version is the preceding version and the related document\_version is the following version.

Each document shall be referred to by at most one document\_version\_relationship as related and a 'relation\_type' of 'sequence';

'supplied document': The application object defines a relationship between two document\_version objects representing the same object in different organizational contexts.

NOTE The different organizational contexts can be represented by different general organizational data associations with the related document\_version objects.

### 5.2.46 duration

A duration is the definition of a period of time.

#### EXPRESS specification:

```
ENTITY duration;  
  time      : STRING;  
  time_unit : STRING;  
END_ENTITY;
```

#### 5.2.46.1 time

The time specifies the extend of the duration.

#### 5.2.46.2 time\_unit

The time\_unit specifies the unit in which the time is specified.

### 5.2.47 effectivity

An effectivity is the identification of the valid use of an aspect of product data tracked by date or event.

NOTE An instance of effectivity may define a period of ineffectivity, i.e., of invalid use, if it is referred to by an instance of effectivity\_assignment with 'effectivity\_indication' set to 'false'.

Either 'start\_definition' or 'end\_definition' or both shall be specified for a particular effectivity. The only exception from this rule is an effectivity that 'inherits' its interval from another effectivity through an effectivity\_relationship with relation\_type 'inheritance'. If a start\_definition and no end\_definition is specified, the period is added to start\_definition in order to get the actual end of the effectivity or ineffectivity period.

NOTE If period is not specified, the effectivity or ineffectivity period is endless. If no start\_definition, but an end\_definition is specified, the period shall also be specified and is subtracted from end\_definition in order to get the actual start of the effectivity or ineffectivity period.

NOTE This standard does not allow to specify effectivity or ineffectivity intervals which are left-unbounded. If both start\_definition and end\_definition are specified, the period should not be specified because of the redundancy. If specified, it shall be equal to the measure of time between the start\_definition and the end\_definition of the effectivity or ineffectivity period.

#### EXPRESS specification:

```
ENTITY effectivity;  
  concerned_organization : SET OF organization;  
  description            : OPTIONAL string_select;  
  effectivity_context    : OPTIONAL STRING;  
  end_definition        : OPTIONAL date_time;  
  id                    : OPTIONAL STRING;  
  period                : OPTIONAL duration;  
  start_definition      : OPTIONAL date_time;  
  version_id            : OPTIONAL STRING;  
END_ENTITY;
```

#### 5.2.47.1 concerned\_organization

The concerned\_organization specifies the set of organization objects in which the effectivity is valid.

EXAMPLE The effectivity of the same item may be different in the various production sites of a manufacturer.

The case where the `concerned_organization` is an empty set means that the effectivity regards any organization that may consider the product data, the effectivity is applied to.

NOTE As effectivity is most of the time organization dependent, it is recommended always to populate the `concerned_organization`

#### 5.2.47.2 description

The description specifies additional information about the effectivity.

The description need not be specified for a particular effectivity.

If present, there shall be exactly one object that defines the description for an effectivity.

#### 5.2.47.3 effectivity\_context

The `effectivity_context` specifies the life cycle stage for which the effectivity is valid.

The `effectivity_context` need not be specified for a particular effectivity.

#### 5.2.47.4 end\_definition

The `end_definition` specifies the end of the period. The bound specified by the `end_definition` is excluded from the interval of effectivity.

The `end_definition` need not be specified for a particular effectivity.

If present, there shall be exactly one object that defines the `end_definition` for an effectivity.

#### 5.2.47.5 id

The `id` specifies the identifier of the effectivity.

The `id` need not be specified for a particular effectivity.

#### 5.2.47.6 period

The period specifies the period of time in which the effectivity is defined, either starting at the point in time specified by '`start_definition`' or ending at the point in time specified by '`end_definition`'. period shall be specified with a positive value.

The period need not be specified for a particular effectivity.

#### 5.2.47.7 start\_definition

The `start_definition` specifies the start of the period. The bound specified by the `start_definition` is included in the interval of effectivity.

NOTE The meaning of this attribute is further detailed by the attribute role of `effectivity_assignment`.

Either `start_definition` or `end_definition` shall be specified for a particular effectivity.

The `start_definition` need not be specified for a particular effectivity.

If present, there shall be exactly one object that defines the `start_definition` for an effectivity.

#### 5.2.47.8 version\_id

The version\_id specifies the identification of a particular version of the effectivity.

The version\_id need not be specified for a particular effectivity.

#### 5.2.48 effectivity\_assignment

An effectivity\_assignment associates an effectivity with the object whose effectivity is controlled by the associated effectivity. The association of an effectivity to product data does not imply any statement concerning the effectivity outside of the specified interval. The same applies in the absence of any assigned effectivity, i.e. no statement concerning the effectivity is implied.

##### EXPRESS specification:

```
ENTITY effectivity_assignment;  
  assigned_effectivity      : effectivity;  
  effective_element        : effective_element_select;  
  effectivity_indication   : BOOLEAN;  
  role                     : STRING;  
END_ENTITY;
```

#### 5.2.48.1 assigned\_effectivity

The assigned\_effectivity specifies the effectivity that is assigned.

#### 5.2.48.2 effective\_element

The effective\_element specifies the object that has an effectivity assigned to it.

There shall be exactly one object that defines the effective\_element for an effectivity\_assignment.

#### 5.2.48.3 effectivity\_indication

The effectivity\_indication specifies whether the assigned\_effectivity defines a period of effectivity (value equal 'TRUE') or a period of ineffectivity (value equal 'FALSE') for the effective\_element. In the first case, use of the effective\_element is or was valid during the considered period.

NOTE This does not imply that use of the effective\_element is mandatory.

In the second case, use of effective\_element is or was not valid during the considered period.

#### 5.2.48.4 role

The role specifies the relationship between the effectivity and the object that has an effectivity assigned to it.

Where applicable the following values shall be used:

'actual': The actual period during which the effectivity lasted;

'planned': The period associated with the effectivity defines a planned period of time during which the associated object is or was supposed to be effective;

'required': The associated object must be kept effective for this period.

#### 5.2.49 effectivity\_relationship

An effectivity\_relationship is a relationship between two effectivity objects.

NOTE Sometimes the effectivity is not dependent on particular dates but on the effectivity of other items. In this case the dates are not instantiated and there is an effectivity\_relationship to the reference effectivity.

EXPRESS specification:

```
ENTITY effectivity_relationship;
  description      : OPTIONAL string_select;
  related          : effectivity;
  relating         : effectivity;
  relation_type    : STRING;
END_ENTITY;
```

#### 5.2.49.1 description

The description specifies additional information about the effectivity\_relationship.

The description need not be specified for a particular effectivity\_relationship.

If present, there shall be exactly one object that defines the description for an effectivity\_relationship.

#### 5.2.49.2 related

The related specifies the second of the two effectivity objects related by the effectivity\_relationship.

NOTE The semantics of this attribute are defined by the attribute 'relation\_type'.

#### 5.2.49.3 relating

The relating specifies the first of the two effectivity objects related by the effectivity\_relationship.

NOTE The semantics of this attribute are defined by the attribute 'relation\_type'.

#### 5.2.49.4 relation\_type

The relation\_type specifies the meaning of the relationship.

Where applicable the following values shall be used:

'constraint': The time period between the start and end definition of the related effectivity shall be within the time period of the relating effectivity;

'inheritance': The related effectivity shall not have a 'start\_definition' and 'end\_definition' specified but inherits the effectivity dates from the relating effectivity.

#### 5.2.50 external\_file\_id\_and\_location

An external\_file\_id\_and\_location specifies the location of a file in an external storage system.

EXPRESS specification:

```
ENTITY external_file_id_and_location;
  external_id      : OPTIONAL STRING;
  location         : document_location_property;
END_ENTITY;
```

#### 5.2.50.1 external\_id

The external\_id specifies the identifier of a document in an external storage system.

EXAMPLE An example for external\_id is 'specification.txt'.

The external\_id need not be specified for a particular external\_file\_id\_and\_location.

### 5.2.50.2 location

The location specifies the location of the document\_file in the external storage system.

EXAMPLE Examples for the location are 'D:\project1\' and '/projects/project1/'.

EXAMPLE The combination of file id and its location taken from the previous examples are 'D:\project1\specification.txt' and '/projects/project1/specification.txt' respectively.

### 5.2.51 external\_geometric\_model

An external\_geometric\_model is the identification of a model that contains geometry in a 3D context only.

An external\_geometric\_model is a type of external\_model.

EXPRESS specification:

```
ENTITY external_geometric_model
  SUBTYPE OF(external_model);
  model_extent : OPTIONAL STRING;
END_ENTITY;
```

#### 5.2.51.1 model\_extent

The model\_extent specifies the radius of a sphere that contains all elements of the model and whose centre is at the origin of the cartesian\_coordinate\_space of the external\_geometric\_model. The model\_extent is specified using a length unit.

The model\_extent need not be specified for a particular external\_geometric\_model.

### 5.2.52 external\_library\_reference

An external\_library\_reference is a mechanism to refer to an entry in an external library other than ISO 13584.

EXPRESS specification:

```
ENTITY external_library_reference;
  description : OPTIONAL string_select;
  external_id : STRING;
  library_type : STRING;
END_ENTITY;
```

#### 5.2.52.1 description

The description specifies additional information about the external\_library\_reference.

The description need not be specified for a particular external\_library\_reference.

If present, there shall be exactly one object that defines the description for an external\_library\_reference.

#### 5.2.52.2 external\_id

The external\_id specifies the unique identifier of the referenced entry in the external library.

### 5.2.52.3 library\_type

The library\_type specifies the type of library that is used.

### 5.2.53 external\_model

An external\_model is the identification of a model that is described in a digital\_file and by the cartesian\_coordinate\_space that is needed to further process the externally described information.

Each external\_model is an external\_picture or an external\_geometric\_model.

#### EXPRESS specification:

```

ENTITY external_model
  ABSTRACT SUPERTYPE OF (ONEOF(external_picture,
    external_geometric_model));
  description      : OPTIONAL string_select;
  is_defined_as    : digital_file;
  is_defined_in    : cartesian_coordinate_space;
  model_id         : STRING;
END_ENTITY;

```

#### 5.2.53.1 description

The description specifies additional information about the external\_model.

The description need not be specified for a particular external\_model.

If present, there shall be exactly one object that defines the description for an external\_model.

#### 5.2.53.2 is\_defined\_as

The is\_defined\_as specifies the digital\_file that contains the externally defined geometry information.

#### 5.2.53.3 is\_defined\_in

The is\_defined\_in specifies the cartesian\_coordinate\_space that defines the context for the externally described geometry.

#### 5.2.53.4 model\_id

The model\_id specifies the identifier of the external\_model.

### 5.2.54 external\_picture

An external\_picture is the identification of a model that is described by a two dimensional image.

An external\_picture is a type of external\_model.

#### EXPRESS specification:

```

ENTITY external_picture
  SUBTYPE OF(external_model);
  SELF\external_model.is_defined_in : cartesian_coordinate_space_2d;
END_ENTITY;

```

#### 5.2.54.1 is\_defined\_in

The `is_defined_in` specifies the `cartesian_coordinate_space_2d` that defines the context for the externally defined geometry.

For picture formats having no explicit origin, e.g., raster format, the lower left corner is defined as the origin of the picture, the x-axis pointing to the right, the y-axis pointing to the top.

#### 5.2.55 general\_classification\_hierarchy

A `general_classification_hierarchy` defines a hierarchical relationship between two instances of `general_classification`.

EXAMPLE With `general_classification`, part families such as screws can be classified. Additionally subclasses can be built and linked to the super-category screws, such as machine screw or self-tapping screw.

NOTE The `general_classification_hierarchy` is used to build up hierarchical structures of `general_classification` objects.

##### EXPRESS specification:

```
ENTITY general_classification_hierarchy;  
  sub_classification : general_classification;  
  super_classification : general_classification;  
END_ENTITY;
```

#### 5.2.55.1 sub\_classification

The `sub_classification` specifies the lower level of `general_classification` in a `general_classification_hierarchy` that is included in the super class.

#### 5.2.55.2 super\_classification

The `super_classification` specifies the higher level of `general_classification` in a `general_classification_hierarchy` that includes the sub class.

#### 5.2.56 general\_classification

A `general_classification` is a classification of an object which characterizes all objects of the same kind; such a classification is independent from the application of the classified object.

EXAMPLE A fastener with subclasses, such as fastener with 2 or 3 fixture holes, a bracket with subclasses, e.g., with 90 or 100 degrees bending angle, or screws with subclasses such as metal screws or machine screws are examples for `general_classification`.

EXAMPLE For facilitating an effective part retrieval for slotted screws, several `general_classification` objects may be defined to classify screws.

##### EXPRESS specification:

```
ENTITY general_classification;  
  classification_source : OPTIONAL classification_source_select;  
  description : OPTIONAL string_select;  
  id : STRING;  
  used_classification_system : OPTIONAL classification_system;  
  version_id : OPTIONAL STRING;  
END_ENTITY;
```

**5.2.56.1 classification\_source**

The `classification_source` specifies the `external_library_reference` or the `plib_class_reference` that contains the specification of the `general_classification`.

The `classification_source` need not be specified for a particular `general_classification`.

If present, there shall be exactly one object that defines the `classification_source` for a `general_classification`.

**5.2.56.2 description**

The `description` specifies additional information about the `general_classification`.

EXAMPLE 'fastener with 2 fixture holes' or 'bracket with 90 degree bending angle' are examples for `description`.

The `description` need not be specified for a particular `general_classification`.

If present, there shall be exactly one object that defines the `description` for a `general_classification`.

**5.2.56.3 id**

The `id` specifies the identifier of the `general_classification`.

**5.2.56.4 used\_classification\_system**

The `used_classification_system` specifies the `classification_system` that contains the information about the definition of the classification and how to interpret the name of the `general_classification`.

The `used_classification_system` need not be specified for a particular `general_classification`.

**5.2.56.5 version\_id**

The `version_id` specifies the identification of a particular version of the `general_classification`.

The `version_id` need not be specified for a particular `general_classification`.

**5.2.57 geometric\_model\_relationship\_with\_transformation**

A `geometric_model_relationship_with_transformation` is a relationship between two model objects with the additional information about a geometric transformation. This transformation defines the location and orientation of the related model relative to the relating model.

EXPRESS specification:

```
ENTITY geometric_model_relationship_with_transformation;
  description      : OPTIONAL string_select;
  model_placement  : transformation;
  related          : external_model;
  relating         : external_model;
  relation_type    : STRING;
END_ENTITY;
```

**5.2.57.1 description**

The `description` specifies additional information about the `geometric_model_relationship_with_transformation`.

The `description` need not be specified for a particular `geometric_model_relationship_with_transformation`.

If present, there shall be exactly one object that defines the description for a `geometric_model_relationship_with_transformation`.

#### 5.2.57.2 `model_placement`

The `model_placement` specifies the geometric transformation that places and orients the related model relative to the relating model.

#### 5.2.57.3 `related`

The `related` specifies the second of the two model objects related by the `geometric_model_relationship_with_transformation`.

NOTE The semantics of this attribute are defined by the attribute `relation_type`.

There shall be exactly one object that the `geometric_model_relationship_with_transformation` is related to.

#### 5.2.57.4 `relating`

The `relating` specifies the first of the two model objects related by the `geometric_model_relationship_with_transformation`.

NOTE The semantics of this attribute are defined by the attribute `relation_type`.

There shall be exactly one object that the `geometric_model_relationship_with_transformation` is relating to.

#### 5.2.57.5 `relation_type`

The `relation_type` specifies the meaning of the relationship.

EXAMPLE The `relation_type` may be, e.g., 'basis for' if the relating `geometric_model` provides a detailed representation of the shape of an object and if the related `geometric_model` provides an idealized representation of the shape of the same object.

#### 5.2.58 `grade`

##### EXPRESS specification:

```
ENTITY grade;  
    coating                : OPTIONAL coating;  
    cutting_condition      : SET [1:?] OF cutting_condition;  
    identifier              : STRING;  
    standard_designation   : OPTIONAL STRING;  
    substrate               : substrate;  
    workpiece_material     : SET [1:?] OF material_designation;  
END_ENTITY;
```

#### 5.2.59 `grade_relationship`

##### EXPRESS specification:

```
ENTITY grade_relationship;  
    related                : grade;  
    relating               : grade;  
    relation_type          : STRING;  
END_ENTITY;
```

**5.2.59.1 related****5.2.59.2 relating****5.2.59.3 relation\_type****5.2.60 item**

An item is either a single object or a unit in a group of objects. It collects the information that is common to all versions of the object. An item shall always be classified as 'cutting item', 'tool item', 'adaptive item', 'assembly item', or 'accessory item' using a `specific_item_classification`. Additionally, if an `assembly_definition` exists for at least one version of the item, the item shall be classified as being an 'assembly' using `specific_item_classification`.

NOTE An item may be either a single piece part or an assembly of arbitrary complexity.

EXAMPLE In the context of cutting tools, an item may be the cutting tool as a whole, the assembly of the tool body, an insert, a shim, a screw, an adaptor, or a solid drill.

**EXPRESS specification:**

```
ENTITY item;
  description: OPTIONAL string_select;
  id : STRING;
  name : string_select;
INVERSE
  associated_version : SET [1:?] OF item_version FOR associated_item;
  item_classification : SET [1:?] OF specific_item_classification FOR
  associated_item;
END_ENTITY;
```

**5.2.60.1 description**

The description specifies additional information about the item.

The description need not be specified for a particular item.

If present, there shall be exactly one object that defines the description for an item.

**5.2.60.2 id**

The id specifies the identifier of the item. For the id, an owner shall be specified by a `person_organization_assignment` with role 'id owner'. The id shall be unique within the scope of the organization that is specified by the `person_organization_assignment` with the role 'id owner'.

NOTE The scope of uniqueness is usually dependent on the form of implementation; it may be a physical file or a data base.

**5.2.60.3 name**

The name specifies the word or group of words used to refer to the item.

There shall be exactly one object that defines the name for an item.

**5.2.60.4 associated\_version**

The `associated_item` specifies the item with which the `item_version` is associated.

### 5.2.61 item\_characteristic\_association

An item\_characteristic\_association associates a characteristic to an item\_definition.

```
ENTITY item_characteristic_association;  
    associated_characteristic : item_characteristic_select;  
    associated_item           : item_definition;  
END_ENTITY;
```

#### 5.2.61.1 associated\_characteristic

Specifies the associated characteristic that in turn characterizes the item\_definition.

#### 5.2.61.2 associated\_item

Specifies the associated item\_definition.

### 5.2.62 item\_definition

An item\_definition is a view of an item\_version. This view is relevant for the requirements of one or more application domains and collects product data of the item\_version.

NOTE The selection of data describing an item\_version may be different for assembly purposes, shipping purposes or analysis purposes.

Each item\_definition may be a mating\_definition, an assembly\_definition, or a physical\_item\_definition.

#### EXPRESS specification:

```
ENTITY item_definition  
    SUPERTYPE OF (ONEOF(assembly_definition, mating_definition,  
    physical_item_definition));  
    additional_context : SET OF application_context;  
    associated_item_version : item_version;  
    id : STRING;  
    initial_context : application_context;  
    name : OPTIONAL string_select;  
END_ENTITY;
```

#### 5.2.62.1 additional\_context

The additional\_context specifies the set of application\_context objects in which this view of the item\_version is also relevant. The additional\_context shall not contain the application\_context that is referenced as the 'initial\_context'.

#### 5.2.62.2 associated\_item\_version

The associated\_item\_version specifies the item\_version for which the item\_definition is a view.

#### 5.2.62.3 id

The id specifies the identifier of the item\_definition.

#### 5.2.62.4 initial\_context

The initial\_context specifies the application\_context in which this view of the item\_version has been designed primarily.

**5.2.62.5 name**

The name specifies the word or group of words used to refer to the item\_definition.

The name need not be specified for a particular item\_definition.

If present, there shall be exactly one object that defines the name for a item\_definition.

**5.2.63 item\_definition\_relationship**

An item\_definition\_relationship is a relationship between two item\_definition objects whose meaning is defined by the attribute 'relation\_type'.

EXPRESS specification:

```
ENTITY item_definition_relationship
  description      : OPTIONAL string_select;
  related          : item_definition;
  relating         : item_definition;
  relation_type   : STRING;
END_ENTITY;
```

**5.2.63.1 description**

The description specifies additional information about the item\_definition\_relationship.

The description need not be specified for a particular item\_definition\_relationship.

If present, there shall be exactly one object that defines the description for a item\_definition\_relationship.

**5.2.63.2 related**

The related specifies the second of the item\_definition objects that are part of the relationship.

**5.2.63.3 relating**

The relating specifies the first of the item\_definition objects that are part of the relationship.

**5.2.63.4 relation\_type**

The relation\_type specifies the meaning of the relationship.

**5.2.64 item\_instance**

An item\_instance is the occurrence of an object in a product structure that is defined by an item\_definition.

An item\_instance shall be used, at least once, in an assembly\_association.

**EXAMPLE** In the case of an indexable milling cutter, the item 'insert' is defined once. Its item\_definition carries all the information necessary to define the insert (e.g., its dimensions) independent of its usage. Additionally, there are 3 item\_instance objects for this insert since there are 3 equal inserts used in this milling cutter. Each of these instances may carry additional information such as placement or function.

EXPRESS specification:

```
ENTITY item_instance;
  definition      : item_definition;
```

```
description    : OPTIONAL string_select;  
id             : STRING;  
END_ENTITY;
```

#### 5.2.64.1 definition

The definition specifies the item\_definition that serves as a definition for this particular occurrence.

There shall be exactly one object that defines the definition for an item\_instance.

#### 5.2.64.2 description

The description specifies additional information about the item\_instance.

The description need not be specified for a particular item\_instance.

If present, there shall be exactly one object that defines the description for a item\_instance.

#### 5.2.64.3 id

The id specifies the identifier of the item\_instance.

The id for an item\_instance must only be unique within the context of the defining item.

#### 5.2.65 item\_structure\_association

An item\_structure\_association is a relationship between a item\_definition and an item\_instance.

#### EXPRESS specification:

```
ENTITY item_structure_association  
  ABSTRACT SUPERTYPE OF (ONEOF(mating_association,  
  assembly_association));  
  placement : OPTIONAL  
  geometric_model_relationship_with_transformation;  
  related   : item_instance;  
  relating  : item_definition;  
END_ENTITY;
```

#### 5.2.65.1 placement

The placement specifies the geometric\_model\_relationship\_with\_transformation that specify the transformation information which is used to locate and orient the constituent in the coordinate space of the assembly\_definition.

The placement need not be specified for a particular item\_structure\_relationship.

If present, there shall be exactly one object that defines the placement for an item\_structure\_relationship.

#### 5.2.65.2 related

The related specifies the item\_instance that is part of the item\_structure\_association.

NOTE The semantics of this attribute are defined by the specializations of item\_structure\_association.

**5.2.65.3 relating**

The relating specifies the item\_definition that is part of the item\_structure\_association.

NOTE The semantics of this attribute are defined by the specializations of item\_structure\_association.

**5.2.66 item\_version**

An item\_version is a version of an item and serves as the collector of the data characterizing a physically realizable object in various application contexts.

NOTE An item\_version may be produced, consumed, used to produce other item\_version objects, or offered to the market.

NOTE The collection of defining information may be incomplete, i.e., not all of the item\_definition objects needed to define an item\_version are associated with the item\_version.

NOTE The set of item\_version objects of an item represents the history of the item within a particular life cycle stage or over its complete life cycle.

NOTE An item\_version does not have to be referenced by any item\_definition.

NOTE In the case, where two item\_version objects are related to each other by an item\_version\_relationship with a relation\_type of value 'supplied part', the relating item\_version is the version carrying the necessary information, e.g., shape and properties, and the related is the version either supplied by or supplied for another company.

**EXPRESS specification:**

```
ENTITY item_version;
  associated_item : item;
  description    : OPTIONAL string_select;
  id             : STRING;
END_ENTITY;
```

**5.2.66.1 associated\_item**

The associated\_item specifies the item with which the item\_version is associated.

**5.2.66.2 description**

The description specifies additional information about the item\_version.

The description need not be specified for a particular item\_version.

If present, there shall be exactly one object that defines the description for an item\_version.

**5.2.66.3 id**

The id specifies the identifier of the item\_version. The id shall be unique within the scope of the associated item.

**5.2.67 item\_version\_relationship**

An item\_version\_relationship is a relationship between two item\_version objects.

**EXPRESS specification:**

```
ENTITY item_version_relationship;
```

```
description      : OPTIONAL string_select;  
related         : item_version;  
relating        : item_version;  
relation_type   : STRING;  
END_ENTITY;
```

#### 5.2.67.1 description

The description specifies additional information about the item\_version\_relationship.

The description need not be specified for a particular item\_version\_relationship.

If present, there shall be exactly one object that defines the description for a item\_version\_relationship.

#### 5.2.67.2 related

The related specifies the second of the two item\_version objects related by the item\_version\_relationship.

NOTE The semantics of this attribute are defined by the attribute relation\_type.

#### 5.2.67.3 relating

The relating specifies the first of the two item\_version objects related by the item\_version\_relationship.

NOTE The semantics of this attribute are defined by the attribute relation\_type.

#### 5.2.67.4 relation\_type

The relation\_type specifies the meaning of the relationship.

Where applicable the following values shall be used:

'derivation': The application object defines a deriving relationship where the related item\_version is based on the relating item\_version which is an earlier version of the same or of a different item;

'hierarchy': The application object defines a hierarchical relationship where the related item\_version is a subordinate version of the relating item\_version;

EXAMPLE 'Rev. 1.1' and 'rev. 1.2' are subordinates of 'version 1'.

'sequence': The application object defines a version sequence where the relating item\_version is the preceding version of the related item\_version that is the following version. For a given item\_version there shall be at most one item\_version\_relationship of this relation\_type referring to this item\_version as 'relating' and at most one item\_version\_relationship of this relation\_type referring as 'related';

'supplied item': The application object defines a relationship between two item\_version objects representing the same object in different organizational contexts.

NOTE The different organizational contexts can be represented by different general organizational data associations.

NOTE The concept of alias\_identification may also be used for item\_version. However, item\_version\_relationship shall be used if the related item\_version corresponds to an item at the supplier's site and if a distinction of additional information such as name or organizational information is needed.

EXAMPLE Usually, items that are provided by suppliers have at the supplier's site an id different from the one at the contractor's site: an item that is identified in a company by the identifier 'C425' and the version id 'V2.0' may correspond to the item that the supplier identifies as '2X45' with the version id 'V3.2'.

**5.2.68 language**

A language is a specification of the language in which information is given.

EXPRESS specification:

```
ENTITY language;
  country_code   : OPTIONAL STRING;
  language_code  : STRING;
END_ENTITY;
```

**5.2.68.1 country\_code**

The country\_code specifies the country, as addition to the language, according to ISO 3166-1.

EXAMPLE Possible values for country\_code are, e.g., 'GBR' for the United Kingdom or 'USA' for the United States of America.

**5.2.68.2 language\_code**

The language\_code specifies the language of the text information in the Alpha-3 code specified in ISO 639-2.

EXAMPLE Possible values for language\_code are, e.g., 'eng' for English, 'fra' for French, 'rus' for Russian, or 'deu' for German.

**5.2.69 limits\_and\_fits**

A limits\_and\_fits contains the necessary information to express a tolerance of the limits-and-fits system standardized by ISO 286.

EXPRESS specification:

```
ENTITY limits_and_fits;
  deviation      : STRING;
  fitting_type   : OPTIONAL STRING;
  grade          : STRING;
END_ENTITY;
```

**5.2.69.1 deviation**

The deviation specifies the enumeration of limits and fits class descriptors by characters.

NOTE The characters 'A' to 'ZC' for holes or 'a' to 'zc' for shafts may be used for deviation.

**5.2.69.2 fitting\_type**

The fitting\_type specifies whether the tolerance declaration applies to a shaft or to a hole.

The fitting\_type need not be specified for a particular limits\_and\_fits.

**5.2.69.3 grade**

The grade specifies the quality or the accuracy grade of a tolerance.

NOTE The grade is based on the international standard tolerance grades IT01 to IT18.

### 5.2.70 location

EXPRESS specification:

```
ENTITY location;  
  location_id      :  STRING;  
  location_name   :  OPTIONAL STRING;  
  location_type   :  STRING;  
END_ENTITY;
```

### 5.2.71 mated\_item\_relationship

A mated\_item\_relationship is a relationship between two mating\_association objects. This relationship specifies additional information about the mating of two particular items that go into a mating\_definition. The two mating\_association objects that are referenced by the mated\_item\_relationship shall refer to the same mating\_definition.

EXPRESS specification:

```
ENTITY mated_item_relationship;  
  mating_material :  SET OF quantified_instance;  
  related         :  mating_association;  
  relating        :  mating_association;  
END_ENTITY;
```

#### 5.2.71.1 mating\_material

The mating\_material specifies the set of quantified\_instance objects used as material for the mating.

#### 5.2.71.2 related

The related specifies the second of the two mating\_association objects related by the mated\_item\_relationship.

#### 5.2.71.3 relating

The relating specifies the first of the two mating\_association objects related by the mated\_item\_relationship.

### 5.2.72 material\_designation

A material\_designation is an identifier for the substance out of which an item is made.

EXPRESS specification:

```
ENTITY material_designation;  
  material_name   :  STRING;  
END_ENTITY;
```

#### 5.2.72.1 material\_name

The material\_name specifies the word or group of words by which the material\_designation is referred to.

### 5.2.73 mating\_association

A mated\_item\_association is a relationship between a mating\_definition and an item\_instance that is used within the mating\_definition.

A mating\_association is a type of item\_structure association.

EXPRESS specification:

```
ENTITY mating_association
  SUBTYPE OF(item_structure_association);
  SELF\item_structure_association.relatng    : mating_definition;
END_ENTITY;
```

### 5.2.73.1 relating

The relating specifies the mating\_definition that is the result of the mating operation.

### 5.2.74 mating\_definition

A mating\_definition is a view of an item\_version, defining the physical connection of two or more item\_instance objects. It includes technical information about the kind of connection. This information is independent from the hierarchical assembly structure.

A mating\_definition is a type of item\_definition.

EXPRESS specification:

```
ENTITY mating_definition
  SUBTYPE OF(item_definition);
  mating_type    : STRING;
INVERSE
  mated_items    : SET [2:?] OF mating_association FOR relating;
END_ENTITY;
```

### 5.2.74.1 mating\_type

The mating\_type specifies the kind of mating, i.e., how the items shall be mated together.

Where applicable the following values shall be used:

- 'bolted joint': Attachment of two or more items that are screwed together using screws, bolt nuts, and washers;
- 'brazing': Attachment of two or more items established by soldering or brazing;
- 'clasping': Attachment of two or more items established by using a clamp at one common outer border;
- 'doweling': Attachment of two or more items using a pin that is fit into each of each part by pressing;
- 'flanging': Attachment of two or more items by crimping a projected length of one part over the edge of the other part;
- 'press fit': Attachment of two or more items obtained by forces resulting from the selected size tolerances;
- 'welding': Attachment of two or more items by welding together with one weld bead.

### 5.2.75 multi\_language\_string

A multi\_language\_string represents text information, expressed in one or more languages, that is associated with objects.

EXPRESS specification:

```
ENTITY multi_language_string;
  additional_language_string : SET OF string_with_language;
  primary_language_string   : string_with_language;
END_ENTITY;
```

### 5.2.75.1 additional\_language\_string

The additional\_language\_dependent\_string specifies the string\_with\_language objects that represent the text information in a particular language.

### 5.2.75.2 primary\_language\_string

The primary\_language\_dependent\_string specifies the string\_with\_language that represents the text information in the original language.

### 5.2.76 named\_size

A named\_size is the definition of the size of a document\_file or of a document\_representation where the size is specified by a standardized identifier.

A named\_size is a type of rectangular\_size.

#### EXPRESS specification:

```
ENTITY named_size
  SUBTYPE OF(rectangular_size);
  referenced_standard : OPTIONAL classification_system;
  size                : STRING;
END_ENTITY;
```

#### 5.2.76.1 referenced\_standard

The referenced\_standard specifies a standard according to which the size is specified.

The referenced\_standard need not be specified for a particular named\_size.

#### 5.2.76.2 size

The size specifies the size of the object. If the size differs from the dimensions specified by the inherited 'width' and 'height' attributes the size is overridden.

### 5.2.77 numerical\_value

A numerical\_value is a quantity expressed with a numerical value and a unit.

A numerical\_value is a type of value\_with\_unit.

#### EXPRESS specification:

```
ENTITY numerical_value
  SUBTYPE OF(value_with_unit);
  value_component : STRING;
INVERSE
  limitation : SET [0:1] OF value_limitation FOR limited_value;
END_ENTITY;
```

#### 5.2.77.1 value\_component

The value\_component specifies the quantity of the numerical\_value.

**5.2.78 organization**

An organization is a group of people involved in a particular business process.

EXPRESS specification:

```

ENTITY organization;
  delivery_address : OPTIONAL STRING;
  id                : STRING;
  organization_name : STRING;
  organization_type : OPTIONAL STRING;
  postal_address   : OPTIONAL STRING;
  visitor_address  : OPTIONAL STRING;
END_ENTITY;

```

**5.2.78.1 delivery\_address**

The `delivery_address` specifies the address where goods are delivered.

The `delivery_address` need not be specified for a particular organization.

**5.2.78.2 id**

The `id` specifies the identifier of the organization.

**NOTE** The assignment of this attribute is usually controlled by a registration authority. The registration authority may be a public organization that assigns identifiers to corporations, or it may be the parent corporation that assigns component identifiers to its components.

**EXAMPLE** The `id` may be the code assigned to the organization for a stock market listing, or it may be a department number.

**5.2.78.3 organization\_name**

The `organization_name` specifies the word or group of words used to refer to the organization.

**5.2.78.4 organization\_type**

The `organization_type` specifies the type of the organization.

Where applicable the following values shall be used:

'company': The `organization_type` specifies that the organization is a company;

'department': The `organization_type` specifies that the organization is a department;

'plant': The `organization_type` specifies that the organization is a plant.

The `organization_type` need not be specified for a particular organization.

**5.2.78.5 postal\_address**

The `postal_address` specifies the address where letter mail is delivered.

The `postal_address` need not be specified for a particular organization.

**5.2.78.6 visitor\_address**

The `visitor_address` specifies the address where the organization receives visitors.

The visitor\_address need not be specified for a particular organization.

### 5.2.79 person

A person is an individual human being who has some relationship to product data. The person shall always be identified in the context of one or more organizations.

#### EXPRESS specification:

```
ENTITY person;
  person_name : STRING;
  preferred_business_address : OPTIONAL STRING;
INVERSE
  associated_organization : SET [1:?] OF person_in_organization FOR
  associated_person;
END_ENTITY;
```

#### 5.2.79.1 person\_name

The person\_name specifies the word or group of words used to refer to the person.

NOTE The person\_name includes the first, middle, and last names as well as titles, if applicable.

#### 5.2.79.2 preferred\_business\_address

The preferred\_business\_address specifies the location of the office of the person.

The preferred\_business\_address need not be specified for a particular person.

### 5.2.80 person\_in\_organization

A person\_in\_organization is the specification of a person in the context of an organization.

#### EXPRESS specification:

```
ENTITY person_in_organization;
  associated_organization : organization;
  associated_person : person;
  id : OPTIONAL STRING;
  location : OPTIONAL STRING;
  role : STRING;
END_ENTITY;
```

#### 5.2.80.1 associated\_organization

The associated\_organization specifies the organization with which the person is associated.

#### 5.2.80.2 associated\_person

The associated\_person specifies the person.

#### 5.2.80.3 id

The id specifies an identifier of the person. The identifier shall be unique within the scope of the 'associated\_organization'.

EXAMPLE The id may be a staff number or a user id in a computer system.

The id need not be specified for a particular person\_in\_organization.

#### 5.2.80.4 location

The location specifies the relevant address of the person\_in\_organization.

The location need not be specified for a particular person\_in\_organization.

#### 5.2.80.5 role

The role specifies the relationship between the person and the organization.

### 5.2.81 person\_organization\_assignment

A person\_organization\_assignment is an object that associates an organization or a person\_in\_organization with product data.

This assignment provides additional information for the associated object. The provision of such data through this assignment has an organizational character whereas some objects require the same kind of data mandatorily in order to be semantically complete. This assignment shall not be used to associate the corresponding organizational data with an object whose attributes are referencing the organizational data directly.

The person\_organization\_assignment shall not be used for the purpose of specifying a relationship between two person\_in\_organization objects, e.g., a succession in organizational responsibility.

#### EXPRESS specification:

```
ENTITY person_organization_assignment;
  assigned_person_organization : person_organization_select;
  description                 : OPTIONAL string_select;
  is_applied_to              : SET [1:?] OF
                              general_organizational_data_select;
  role                       : STRING;
END_ENTITY;
```

#### 5.2.81.1 assigned\_person\_organization

The assigned\_person\_organization specifies the concerned organization or the person\_in\_organization.

There shall be exactly one object that defines the assigned\_person\_organization for a person\_organization\_assignment.

#### 5.2.81.2 description

The description specifies additional information about the person\_organization\_assignment.

The description need not be specified for a particular person\_organization\_assignment.

If present, there shall be exactly one object that defines the description for a person\_organization\_assignment.

#### 5.2.81.3 is\_applied\_to

The is\_applied\_to specifies the object with which the person\_organization\_assignment is associated.

There shall be at least one object that the person\_organization\_assignment is\_applied\_to.

#### 5.2.81.4 role

The role specifies the responsibility of the assigned person or organization with respect to the object that it is applied to.

Where applicable the following values shall be used:

'author': The referenced object has been created by the assigned person or organization. The author holds the copyright;  
'classification officer': The assigned person or organization is formally responsible for the classification of the referenced object;  
'creator': The referenced object has been created by the assigned person or organization;  
'custodian': The assigned person or organization is responsible for the existence and integrity of the referenced object;  
'customer': The assigned person or organization acts as a purchaser or consumer of the referenced object;  
'design supplier': The assigned person or organization is the one who delivers the data describing the referenced object;

EXAMPLE The 'design supplier' may be a design by means of a 3D model or a drawing.

'editor': The assigned person or organization is responsible for making any changes to any attribute of the referenced object;  
'id owner': The assigned person or organization is the one responsible for the designation of an identifier;  
'location': The assigned organization is the place where the referenced object can be found or where it takes place;  
'owner': The assigned person or organization owns the referenced object, and has final say over its disposition and any changes to it;  
'supplier': The assigned person or organization is the one who delivers the actual (physical) object (e.g., a dealer);

#### 5.2.82 physical\_document

A physical\_document is a piece of product data that is archived in a non-digital form.

EXAMPLE Paper plots of technical drawings, micro fiche, or paper documents such as calculations or test reports are examples for a physical\_document.

A physical\_document is a type of document\_representation.

##### EXPRESS specification:

```
ENTITY physical_document
  SUBTYPE OF (document_representation);
  file : SET OF physical_file;
END_ENTITY;
```

#### 5.2.82.1 file

The file specifies the physical realization of the physical\_document.

#### 5.2.83 physical\_file

A physical\_file is the actual stack of paper consisting of one or more sheets, on which some product data is written, printed or plotted.

A physical\_file is a type of document\_file.

EXPRESS specification:

```

ENTITY physical_file
  SUBTYPE OF(document_file);
END_ENTITY;

```

**5.2.84 physical\_item**EXPRESS specification:

```

ENTITY physical_item
  SUBTYPE OF(item);
INVERSE
  SELF\item.associated_version : SET [1:?] OF physical_item version FOR
  associated_item;
END_ENTITY;

```

**5.2.85 physical\_item\_location\_association**EXPRESS specification:

```

ENTITY physical_item_location_association;
  located_item : physical_item_definition;
  location : location;
END_ENTITY;

```

**5.2.86 physical\_item\_definition**EXPRESS specification:

```

ENTITY physical_item_definition
  SUBTYPE OF(item_definition);
  SELF\item_definition.associated_item_version : physical_item_version;
END_ENTITY;

```

**5.2.87 physical\_item\_state\_association**EXPRESS specification:

```

ENTITY physical_item_state_association;
  associated_physical_item : physical_item_definition;
  associated_state : state;
  role : STRING;
END_ENTITY;

```

**5.2.88 physical\_item\_structure\_association**EXPRESS specification:

```

ENTITY physical_item_structure_association;
  related : physical_item_definition;
  relating : physical_item_definition;
END_ENTITY;

```

### 5.2.89 physical\_item\_version

EXPRESS specification:

```
ENTITY physical_item_version
  SUBTYPE OF (item_version);
  SELF\item_version.associated_item : physical_item;
END_ENTITY;
```

### 5.2.90 plib\_class\_reference

A plib\_class\_reference designates a class in a library compliant to ISO 13584 (Parts Library).

EXPRESS specification:

```
ENTITY plib_class_reference;
  code          : STRING;
  supplier_bsu  : STRING;
  version       : STRING;
END_ENTITY;
```

#### 5.2.90.1 code

The code specifies the class in the PLIB library. The format of this code is defined in ISO 13584-42.

#### 5.2.90.2 supplier\_bsu

The supplier\_bsu (basic semantic unit) specifies the supplier of the class in a PLIB library, in which the class is defined. The format of this specification is defined in ISO 13584-26.

#### 5.2.90.3 version

The version specifies the identification of a particular version of a class in a PLIB library. The format of this version is defined in ISO 13584-42.

### 5.2.91 plib\_property\_reference

A plib\_property\_reference designates a property in a library compliant to ISO 13584.

EXPRESS specification:

```
ENTITY plib_property_reference;
  code          : STRING;
  name_scope    : plib_class_reference;
  version       : STRING;
END_ENTITY;
```

#### 5.2.91.1 code

The code specifies the property in the PLIB library. The format of this code is defined in ISO 13584-42.

#### 5.2.91.2 name\_scope

The name\_scope specifies the plib\_class\_reference in which the property is visible.

### 5.2.91.3 version

The version specifies the identification of a particular version of a property in a PLIB library. The format of this version is defined in ISO 13584-42.

### 5.2.92 plus\_minus\_bounds

A plus\_minus\_bounds is the specification of the allowable deviation from a numerical value.

#### EXPRESS specification:

```
ENTITY plus_minus_bounds;
  lower_bound      : STRING;
  significant_digits : OPTIONAL STRING;
  upper_bound      : STRING;
  value_determination : OPTIONAL STRING;
END_ENTITY;
```

#### 5.2.92.1 lower\_bound

The lower\_bound specifies the value of the tolerance that shall be subtracted from the exact value to establish the minimum allowed value.

#### 5.2.92.2 significant\_digits

The significant\_digits specifies the number of decimal digits indicating the accuracy of the lower\_bound and upper\_bound values.

The significant\_digits need not be specified for a particular plus\_minus\_bounds.

#### 5.2.92.3 upper\_bound

The upper\_bound specifies the value of the tolerance that shall be added to the exact value to establish the maximum allowed value.

#### 5.2.92.4 value\_determination

The value\_determination specifies information on how the plus\_minus\_bounds shall be interpreted.

Where applicable the following values shall be used:

'calculated': The value has been calculated;  
 'designed': The value represents a value intended by the design;  
 'estimated': The value has been estimated;  
 'measured': The value has been measured;  
 'required': The value represents a requirement.

NOTE A plus\_minus\_bounds may be specified in the design stage of a product but it may also be documented as measured on a prototype.

The value\_determination need not be specified for a particular plus\_minus\_bounds.

### 5.2.93 property

A property is the definition of a particular quality.

NOTE A property may reflect physics or arbitrary user defined measurements.

EXPRESS specification:

```
ENTITY property;  
  allowed_unit      : SET OF unit;  
  description      : OPTIONAL string_select;  
  id                : STRING;  
  property_source  : OPTIONAL property_source_select;  
  property_type    : STRING;  
  version_id       : OPTIONAL STRING;  
END_ENTITY;
```

**5.2.93.1 allowed\_unit**

The allowed\_unit specifies the unit or set of units that are accepted.

EXAMPLE A company may accept a mass specified in kilograms or tons, but not in grams or pounds.

**5.2.93.2 description**

The description specifies additional information about the property.

The description need not be specified for a particular property.

If present, there shall be exactly one object that defines the description for a property.

**5.2.93.3 id**

The id specifies the identifier of the property.

**5.2.93.4 property\_source**

The property\_source specifies the external\_library\_reference or plib\_property\_reference object that defines this kind of property.

The property\_source need not be specified for a particular property.

If present, there shall be exactly one object that defines the property\_source for a property.

**5.2.93.5 property\_type**

The property\_type specifies the kind of property the property defines.

Where applicable the following values shall be used:

'cost': a property that specifies costs.  
'dimension': a property that describes a dimension of an object.  
'mass': a quantity of matter that an object consists of.  
'shape': a property that describes the shape of an object.  
'specification': a property that specifies a certain characteristic of an object.  
'usage': a property that describes the use of an object.

**5.2.93.6 version\_id**

The version\_id specifies the identification of a particular version of a property.

The version\_id need not be specified for a particular property.

### 5.2.94 property\_relationship

A property\_relationship is a relationship between two property objects.

EXAMPLE A property may reflect physics or arbitrary user defined measurements.

#### EXPRESS specification:

```
ENTITY property_relationship;
  description      : OPTIONAL string_select;
  related          : property;
  relating         : property;
  relation_type    : STRING;
END_ENTITY;
```

#### 5.2.94.1 description

The description specifies additional information about the property\_relationship.

The description need not be specified for a particular property\_relationship.

If present, there shall be exactly one object that defines the description for a property\_relationship.

#### 5.2.94.2 related

The related specifies the second of the two property objects related by the property\_relationship.

NOTE The semantics of this attribute are defined by the attribute relation\_type.

#### 5.2.94.3 relating

The relating specifies the first of the two property objects related by the property\_relationship.

NOTE The semantics of this attribute are defined by the attribute relation\_type.

#### 5.2.94.4 relation\_type

The relation\_type specifies the meaning of the relationship.

Where applicable the following values shall be used:

'decomposition': The property\_relationship defines a relationship where the related property is a member of a group of property objects that is established by the relating property;

'dependency': The related property is dependent upon the relating property;

'hierarchy': The application object defines a hierarchical relationship where the related property is on a lower level than the relating property;

'peer': The related property shall not be used without the relating property and vice versa;

'substitution': The property\_relationship defines a relationship where the related property replaces the relating property;

'value domain': The property\_relationship defines a relationship where the values assigned to the relating property shall be within the limits indicated by the values assigned to the related property.

### 5.2.95 property\_value

A property\_value is the numerical or textual value of a property.

Each property\_value is a value\_list, a string\_value, or a value\_with\_unit.

EXPRESS specification:

```
ENTITY property_value
  ABSTRACT SUPERTYPE OF (ONEOF(value_with_unit, string_value,
  value_list));
  value_name : STRING;
END_ENTITY;
```

**5.2.95.1 value\_name**

The value\_name specifies the word or group of words by which the property\_value is referred to.

EXAMPLE 'l1' or 'vol2' are examples for the value\_name of a property\_value.

**5.2.96 property\_value\_association**

A property\_value\_association is a mechanism to assign a property\_value\_representation to an object.

EXPRESS specification:

```
ENTITY property_value_association;
  definitional : OPTIONAL BOOLEAN;
  described_element : item_property_select;
  describing_property_value : property_value_representation;
  description : OPTIONAL string_select;
  validity_context : OPTIONAL organization;
END_ENTITY;
```

**5.2.96.1 definitional**

The definitional specifies whether the associated property\_value\_representation object may be used to distinguish the described\_element from others of the same kind. A value of 'true' indicates that the associated property\_value\_representation distinguishes it from others.

The definitional is only meaningful, if the described\_element is of type item\_definition.

The definitional need not be specified for a particular property\_value\_association.

**5.2.96.2 described\_element**

The described\_element specifies the object that is characterized by the property\_value.

There shall be exactly one object that defines the described\_element for an property\_value\_association.

**5.2.96.3 describing\_property\_value**

The describing\_property\_value specifies the value that is being assigned.

**5.2.96.4 description**

The description specifies additional information about the property\_value\_association.

The description need not be specified for a particular property\_value\_association.

If present, there shall be exactly one object that defines the description for a property\_value\_association.

**5.2.96.5 validity\_context**

The `validity_context` specifies the context in which a `property_value_association` is applicable.

EXAMPLE A property with `property_type` 'cost' is applicable in the context of supplier A and may be different if the item is supplied by a different supplier B.

The `validity_context` need not be specified for a particular `property_value_association`.

If present, there shall be exactly one object that defines the `validity_context` for a `property_value_association`.

**5.2.97 property\_value\_representation**

A `property_value_representation` is the representation of property.

EXPRESS specification:

```
ENTITY property_value_representation;
  definition          : property;
  global_unit        : OPTIONAL unit;
  qualifier          : OPTIONAL STRING;
  specified_value    : property_value;
  value_determination : OPTIONAL STRING;
END_ENTITY;
```

**5.2.97.1 definition**

The `definition` specifies the property that the `property_value_representation` characterizes.

**5.2.97.2 global\_unit**

The `global_unit` specifies a unit that is valid for all `property_value` that are referenced as 'specified\_value' by the `property_value_representation`.

The `global_unit` need not be specified for a particular `property_value_representation`.

**5.2.97.3 qualifier**

The `qualifier` specifies the kind of the `property_value_representation`.

The following values shall be used:

'nominal': The value is the nominal value;  
 'specified': The value is specified;  
 'typical': The value is a typical value.

The `qualifier` need not be specified for a particular `property_value_representation`.

**5.2.97.4 specified\_value**

The `specified_value` specifies the `property_value` that qualifies the `property_value_representation` by a `value_with_unit`, a `string_value`, or an arbitrary aggregate thereof.

**5.2.97.5 value\_determination**

The `value_determination` specifies information on how the `property_value_representation` shall be interpreted.

Where applicable the following values shall be used:

- 'calculated': The value has been calculated;
- 'designed': The value represents a value intended by the design;
- 'estimated': The value has been estimated;
- 'measured': The value has been measured;
- 'required': The value represents a requirement;
- 'set point': The value is used as the initialization value.

The value\_determination need not be specified for a particular property\_value\_representation.

### 5.2.98 property\_value\_representation\_relationship

A property\_value\_representation\_relationship is a relationship between two property\_value\_representation objects.

EXPRESS specification:

```
ENTITY property_value_representation_relationship;  
  description      : OPTIONAL string_select;  
  related          : property_value_representation;  
  relating         : property_value_representation;  
  relation_type    : STRING;  
END_ENTITY;
```

#### 5.2.98.1 description

The description specifies additional information about the property\_value\_representation\_relationship.

The description need not be specified for a particular property\_value\_representation\_relationship.

If present, there shall be exactly one object that defines the description for a property\_value\_representation\_relationship.

#### 5.2.98.2 related

The related specifies the second of the two objects related by the property\_value\_representation\_relationship.

NOTE The semantics of this attribute is defined by the attribute relation\_type.

#### 5.2.98.3 relating

The relating specifies the first of the two objects related by the property\_value\_representation\_relationship.

NOTE The semantics of this attribute is defined by the attribute relation\_type.

#### 5.2.98.4 relation\_type

The relation\_type specifies the meaning of the relationship.

EXAMPLE property\_value\_representation objects representing 'cycle time', 'feed time', and 'drill time' for a process may be related by a property\_value\_representation\_relationship with relation\_type 'dependency'.

Where applicable the following values shall be used:

- 'decomposition': The property\_value\_representation\_relationship specifies a relationship where the related property\_value\_representation is one of potentially more sub values of the relating property\_value\_representation;

'dependency': The `property_value_representation_relationship` defines a deriving relationship where the related `property_value_representation` is based on the relating `property_value_representation`;

'equivalence': The `property_value_representation_relationship` defines a relationship where the related `property_value_representation` represents the same matter of fact as by the relating `property_value_representation`;

EXAMPLE The distance of 1 km and 1000 m is the same. 1 km is equivalent to 1000 m, but they have different values and units.

'substitution': The `property_value_representation_relationship` defines a relationship where the related `property_value_representation` replaces the relating `property_value_representation`.

### 5.2.99 quantified\_instance

A `quantified_instance` is the identification of the quantified occurrence of an object that is defined as an `item_definition`.

#### EXPRESS specification:

```
ENTITY quantified_instance
  SUBTYPE OF(item_instance);
  quantity : numerical_value;
END_ENTITY;
```

#### 5.2.99.1 quantity

The `quantity` specifies a `numerical_value` specifying the quantity of occurrences.

### 5.2.100 realized\_item\_association

#### EXPRESS specification:

```
ENTITY realized_item_association;
  physical_item : physical_item;
  realized_item : item;
END_ENTITY;
```

### 5.2.101 rectangular\_size

A `rectangular_size` is the definition of the planar size of an object.

Each `rectangular_size` may be a `named_size`.

#### EXPRESS specification:

```
ENTITY rectangular_size;
  density : OPTIONAL value_with_unit;
  height : value_with_unit;
  width: value_with_unit;
END_ENTITY;
```

#### 5.2.101.1 density

The `density` specifies the resolution of the object if it is a raster picture.

The `density` need not be specified for a particular `rectangular_size`.

### 5.2.101.2 height

The height specifies the size of the object in vertical direction.

### 5.2.101.3 width

The width specifies the size of the object in horizontal direction.

### 5.2.102 specific\_item\_classification

A `specific_item_classification` is a classification of an item with respect to specific criteria. The specific criteria are covered in the 'classification\_name' attribute.

NOTE If an item requires classification by more than one criterion, several `specific_item_classification` objects are associated to the same item.

NOTE For the attribute 'classification\_name' a set of predefined values is specified hereafter. If values other than the proposed ones are used, they should be of general classifying nature. This kind of classification ought not to be used to classify names or identifiers of objects, e.g., in order to classify part families; for this purpose `general_classification` ought to be used.

#### EXPRESS specification:

```
ENTITY specific_item_classification;  
  associated_item      : SET [1:?] OF item;  
  classification_name : STRING;  
  description         : OPTIONAL string_select;  
END_ENTITY;
```

#### 5.2.102.1 associated\_item

The `associated_item` specifies the Item with which a particular `specific_item_classification` is associated.

#### 5.2.102.2 classification\_name

The `classification_name` provides high level classification information.

Where applicable the following values shall be used:

'accessory item': The item is of type accessory item.  
'adaptive item': The item is of type adaptive item.  
'assembly': This type of classification shall be used for any item that has an `assembly_definition` provided for at least one of its versions, i.e., it is decomposed further;  
'assembly item': The item is of type assembly item.  
'cutting item': The item is of type cutting item.  
'cutting tool': The item is a complete cutting tool.  
'detail': This type of classification shall be used for any item that has no `assembly_definition` provided for any of its versions, i.e., it is not further decomposed;  
'tool item': The item is of type tool item.

#### 5.2.102.3 description

The description specifies additional information about the `specific_item_classification`.

The description need not be specified for a particular `specific_item_classification`.

If present, there shall be exactly one object that defines the description for a `specific_item_classification`.

### 5.2.103 specific\_item\_classification\_hierarchy

A `specific_item_classification_hierarchy` is used to build up hierarchical structures of `specific_item_classification`.

#### EXPRESS specification:

```
ENTITY specific_item_classification_hierarchy;
  sub_classification : specific_item_classification;
  super_classification : specific_item_classification;
END_ENTITY;
```

#### 5.2.103.1 sub\_classification

The `sub_classification` specifies the lower level of `specific_item_classification` in a `specific_item_classification_hierarchy` that is included in the super class.

#### 5.2.103.2 super\_classification

The `super_classification` specifies the higher level of `specific_item_classification` in a `specific_item_classification_hierarchy` that includes the sub class.

### 5.2.104 state

#### EXPRESS specification:

```
ENTITY state;
  name : STRING;
END_ENTITY;
```

### 5.2.105 string\_value

A `string_value` represents a sequence of one or more alphanumeric characters.

A `string_value` is a type of `property_value`.

#### EXPRESS specification:

```
ENTITY string_value
  SUBTYPE OF (property_value);
  value_specification : string_select;
END_ENTITY;
```

#### 5.2.105.1 value\_specification

The description specifies additional information about the `string_value`.

There shall be exactly one object that defines the description for a `string_value`.

### 5.2.106 string\_with\_language

A `string_with_language` represents text information in a specific language together with an identification of the language used.

#### EXPRESS specification:

```
ENTITY string_with_language;
```

```
contents          :  STRING;
language_specification :  language;
INVERSE
  used_by :  SET [1:?] OF multi_language_string FOR
  primary_language_string;
END_ENTITY;
```

#### 5.2.106.1 contents

The contents is textual information stored in the language identified by the language attribute.

#### 5.2.106.2 language\_specification

The language\_specification specifies the language in which the contents is given.

#### 5.2.107 substrate

##### EXPRESS specification:

```
ENTITY substrate;
  name :  STRING;
END_ENTITY;
```

#### 5.2.108 transformation

A transformation is a geometric transformation composed of translation and rotation. Scaling is not included.

Each transformation is a transformation\_3d or a transformation\_2d.

##### EXPRESS specification:

```
ENTITY transformation
  ABSTRACT SUPERTYPE OF (ONEOF(transformation_2d, transformation_3d));
END_ENTITY;
```

#### 5.2.109 transformation\_2d

A transformation\_2d is the definition of a geometric transformation in 2D space.

A transformation\_2d is a type of transformation.

##### EXPRESS specification:

```
ENTITY transformation_2d
  SUBTYPE OF(transformation);
END_ENTITY;
```

#### 5.2.110 transformation\_3d

A transformation\_3d is the definition of a geometric transformation in 3D space.

A transformation\_3d is a type of transformation.

##### EXPRESS specification:

```
ENTITY transformation_3d
  SUBTYPE OF(transformation);
```

```
END_ENTITY;
```

### 5.2.111 unit

A unit is a quantity chosen as a standard in terms of which other quantities may be expressed.

The types of units supported by this standard are SI units as well as derived or conversion based units as defined in ISO 10303-41.

#### EXPRESS specification:

```
ENTITY unit;
  unit_name : STRING;
END_ENTITY;
```

#### 5.2.111.1 unit\_name

The unit\_name specifies the term representing the kind of unit.

EXAMPLE 'gram', 'litre', or 'volt' are examples for the unit\_name.

#### 5.2.112 value\_limit

A value\_limit is a qualified numerical value representing either the lower limit or the upper limit of a particular physical characteristic.

EXAMPLE '30.5 max' and '5 min' are examples for a value\_limit.

A value\_limit is a type of value\_with\_unit.

#### EXPRESS specification:

```
ENTITY value_limit
  SUBTYPE OF (value_with_unit);
  limit : STRING;
  limit_qualifier : STRING;
END_ENTITY;
```

#### 5.2.112.1 limit

The limit specifies the value of the limit.

#### 5.2.112.2 limit\_qualifier

The limit\_qualifier specifies the kind of limit.

The following values shall be used:

'maximum':	The specified limit is an upper limit;
'minimum':	The specified limit is a lower limit.

#### 5.2.113 value\_limitation

A value\_limitation is a mechanism to add information about limitations to a numerical\_value.

#### EXPRESS specification:

```
ENTITY value_limitation;  
  envelope      : OPTIONAL BOOLEAN;  
  is_defined_by : limitation_definition_select;  
  limited_value : numerical_value;  
END_ENTITY;
```

### 5.2.113.1 envelope

The envelope specifies whether a geometric constraint resulting from the limitation has to be fulfilled in itself or not. A value of 'true' indicates that the envelope of the perfect shape corresponding to the maximum material shall not be larger than the specified dimension and tolerance.

NOTE The concept of the envelope principle is defined in ISO 8015 [19].

NOTE A dimension where the envelope principle applies, appears on drawings with an 'E' enclosed in a circle following the dimension value.

The envelope need not be specified for a particular value\_limitation.

### 5.2.113.2 is\_defined\_by

The is\_defined\_by specifies the limitation that is associated with the numerical\_value.

There shall be exactly one object that the value\_limitation is\_defined\_by.

### 5.2.113.3 limited\_value

The limited\_value specifies the numerical\_value that is limited.

### 5.2.114 value\_list

A value\_list is an ordered collection of property\_value objects.

EXAMPLE A property may be composed of different property values such as 'mass', 'speed', and 'age' which are altogether necessary in a given context. The value\_list collects all of them in a given order, such that each is identifiable by its index in the list.

A value\_list is a type of property\_value.

#### EXPRESS specification:

```
ENTITY value_list  
  SUBTYPE OF (property_value);  
  values : LIST [1:?] OF property_value;  
END_ENTITY;
```

### 5.2.114.1 values

The values specifies the ordered collection of property\_value objects that together are provided as a property\_value.

### 5.2.115 value\_range

A value\_range is a pair of numerical values representing the range in which the value shall lie.

A value\_range is a type of value\_with\_unit.

EXPRESS specification:

```

ENTITY value_range
  SUBTYPE OF(value_with_unit);
  lower_limit:  STRING;
  upper_limit:  STRING;
END_ENTITY;

```

**5.2.115.1 lower\_limit**

The lower\_limit specifies the minimum acceptable value that is constrained by the value\_range.

**5.2.115.2 upper\_limit**

The upper\_limit specifies the maximum acceptable value that is constrained by the value\_range.

**5.2.116 value\_with\_unit**

A value\_with\_unit is either a single numerical measure, or a range of numerical measures with upper, lower, or upper and lower bounds.

A value\_with\_unit is a type of property\_value.

Each value\_with\_unit is a value\_range, a numerical\_value, or a value\_limit.

EXPRESS specification:

```

ENTITY value_with_unit
  ABSTRACT SUPERTYPE OF (ONEOF(value_limit, numerical_value,
  value_range))
  SUBTYPE OF(property_value);
  significant_digits : OPTIONAL STRING;
  unit_component    : OPTIONAL unit;
END_ENTITY;

```

**5.2.116.1.1 significant\_digits**

The significant\_digits specifies the number of decimal digits that are relevant for the use of the value\_with\_unit. If present, the numerical measure or range may be specified using more digits than the significant digits but shall not be specified using less digits.

The significant\_digits need not be specified for a particular value\_with\_unit.

**5.2.116.2 unit\_component**

The unit\_component specifies the unit in which the value\_with\_unit is expressed.

The unit\_component need not be specified for a particular value\_with\_unit.

**5.2.117 workpiece\_feature**

A workpiece\_feature is a characteristic of the workpiece to be machined by a cutting tool.

EXPRESS specification:

```

ENTITY workpiece_feature;
  description : OPTIONAL string_select;

```

```
id          : STRING;  
representation: SET [1:?] OF external_model;  
END_ENTITY;
```

#### 5.2.117.1 description

The description specifies additional information about the characteristic.

The description need not be specified for a particular characteristic.

If present, there shall be exactly one object that defines the description for a characteristic.

STANDARDSISO.COM : Click to view the full PDF of ISO 13399-1:2006

## 6 Application interpreted model

### 6.1 Mapping specification

In the following, "Application element" designates any entity data type defined in Clause 5, any of its explicit attributes and any subtype constraint. "AIM element" designates any entity data type defined in 6.2 or imported with a USE FROM statement, from another EXPRESS schema, any of its attributes and any subtype constraint defined in 6.2 or imported with a USE FROM statement.

This clause contains the mapping specification that defines how each UoF and application element of this part of ISO 13399 (see Clause 5) maps to one or more AIM elements (see 6.2).

The mapping for each application element is specified in a separate subclause below. The mapping specification of an attribute of an ARM entity is a subclause of the clause that contains the mapping specification of this entity. Each mapping specification subclause contains up to five elements.

**Title:** The clause title contains:

- the name of the considered ARM entity or subtype constraint, or
- the name of the considered ARM entity attribute when this attribute refers to a type that is not an entity data type or a SELECT type that contains or may contain entity data types, or
- a composite expression, <attribute name> to <referred type>, when this attribute refers to a type that is not an entity data type or a SELECT type that contains or may contain entity data types.

**AIM element:** This section contains, depending on the considered application element:

- the name of one or more AIM entity data types;
- the name of a AIM entity attribute, presented with the syntax <entity name>.<attribute name>, when the considered ARM attribute refers to a type that is not an entity data type or a SELECT type that contains or may contain entity data types;
- the term PATH, when the considered ARM entity attribute refers to an entity data type or to a SELECT type that contains or may contain entity data types;
- the term IDENTICAL MAPPING, when both application objects involved in an application assertion map to the same instance of a MIM entity data type;
- the syntax /SUPERTYPE(<supertype name>)/, when the considered ARM entity is mapped as its supertype;
- one or more constructs /SUBTYPE(<subtype name>)/, when the mapping of the considered ARM entity is the union of the mapping of its subtypes.

When the mapping of an application element involves more than one AIM element, each of these AIM elements is presented on a separate line in the mapping specification, enclosed between parentheses or brackets.

**Source:** This section contains:

- the ISO standard number and part number in which the AIM element is defined, for those AIM elements that are defined in a common resource document;
- the ISO standard number and part number of this part of ISO 13399, for those AIM elements that are defined in the AIM schema of this part.

This section is omitted when the keywords PATH or IDENTICAL MAPPING are used in the AIM element section.

**Rules:** This section contains the name of one or more global rules that apply to the population of the AIM entity data types listed in the AIM element section or in the reference path. When no rule applies, this section is omitted.

A reference to a global rule may be followed by a reference to the subclause in which the rule is defined.

**Constraint:** This section contains the name of one or more subtype constraints that apply to the population of the AIM entity data types listed in the AIM element section or in the reference path. When no subtype constraint applies, this section is omitted.

A reference to a subtype constraint may be followed by a reference to the subclause in which the subtype constraint is defined.

**Reference path:** This section contains:

- the reference path to its supertypes in the common resources, for each AIM element created within this part of ISO 13399;
- the specification of the relationships between AIM elements, when the mapping of an application element requires to relate instances of several AIM entity data types. In such a case, each line in the reference path documents the role of an AIM element relative to the referring AIM element or to the next referred AIM element.

For the expression of reference paths and of the constraints between AIM elements, the following notational conventions apply:

- [] enclosed section constrains multiple AIM elements or sections of the reference path are required to satisfy an information requirement;
- () enclosed section constrains multiple AIM elements or sections of the reference path are identified as alternatives within the mapping to satisfy an information requirement;
- { } enclosed section constrains the reference path to satisfy an information requirement;
- <> enclosed section constrains at one or more required reference path;
- || enclosed section constrains the supertype entity;
- > the attribute, whose name precedes the -> symbol, references the entity or select type whose name follows the -> symbol;
- <- the entity or select type, whose name precedes the <- symbol, is referenced by the entity attribute whose name follows the <- symbol;
- [i] the attribute, whose name precedes the [i] symbol, is an aggregate; any element of that aggregate is referred to;
- [n] the attribute, whose name precedes the [n] symbol, is an ordered aggregate; member n of that aggregate is referred to;

- => the entity, whose name precedes the => symbol, is a supertype of the entity whose name follows the => symbol;
- <= the entity, whose name precedes the <= symbol, is a subtype of the entity whose name follows the <= symbol;
- = the string, select, or enumeration type is constrained to a choice or value;
- \ the reference path expression continues on the next line;
- \* one or more instances of the relationship entity data type may be assembled in a relationship tree structure. The path between the relationship entity and the related entities, is enclosed with braces;
- the text following is a comment or introduces a clause reference;
- \*> the select or enumeration type, whose name precedes the \*> symbol, is extended into the select or enumeration type whose name follows the \*> symbol;
- <\* the select or enumeration type, whose name precedes the <\* symbol, is an extension of the select or enumeration type whose name follows the <\* symbol.

## 6.1.1 classification UoF

### 6.1.1.1 classification\_association

#1: If classification\_association.definitional is TRUE.

#2: If classification\_association.definitional is FALSE.

#3: If classification\_association.definitional is not instantiated.

AIM element: applied\_classification\_assignment

Source: ISO 10303-214

Rules: restrict\_applied\_classification\_assignment\_role

Reference path: applied\_classification\_assignment <=  
 classification\_assignment  
 {classification\_assignment.role ->  
 classification\_role  
 classification\_role.name  
 #1: (classification\_role.name = 'definitional')  
 #2: (classification\_role.name = 'non-definitional')  
 #3: (classification\_role.name = "")}

#### 6.1.1.1.1 definitional

#1: If classification\_association.definitional is TRUE.

#2: If classification\_association.definitional is FALSE.

#3: If classification\_association.definitional is not instantiated.

AIM element: classification\_role.name

Source: ISO 10303-41

Rules: restrict\_applied\_classification\_assignment\_role,  
dependent\_instantiable\_classification\_role

Reference path: applied\_classification\_assignment <=  
classification\_assignment  
classification\_assignment.role ->  
classification\_role  
classification\_role.name  
{#1: (classification\_role.name = 'definitional')  
#2: (classification\_role.name = 'non-definitional')  
#3: (classification\_role.name = "")}

#### 6.1.1.1.2 role

AIM element: classification\_role.description

Source: ISO 10303-41

Reference path: applied\_classification\_assignment <=  
classification\_assignment  
classification\_assignment.role ->  
classification\_role  
classification\_role.description

#### 6.1.1.1.3 classification\_association to general\_classification (as associated\_classification)

AIM element: PATH

Reference path: applied\_classification\_assignment <=  
classification\_assignment  
classification\_assignment.assigned\_class ->  
group =>  
class

#### 6.1.1.1.4 classification\_association to document (as classified\_element)

AIM element: PATH

Reference path: applied\_classification\_assignment  
applied\_classification\_assignment.items[i] ->  
classification\_item  
classification\_item = product  
product  
{product <-  
product\_related\_product\_category.products[i]  
product\_related\_product\_category <=  
product\_category  
product\_category.name = 'document'}

#### 6.1.1.1.5 classification\_association to document\_file (as classified\_element)

AIM element: PATH

Reference path: applied\_classification\_assignment  
 applied\_classification\_assignment.items[i] ->  
 classification\_item  
 classification\_item = document\_file  
 document\_file

#### 6.1.1.1.6 classification\_association to document\_representation (as classified\_element)

AIM element: PATH

Reference path: applied\_classification\_assignment  
 applied\_classification\_assignment.items[i] ->  
 classification\_item  
 classification\_item = product\_definition  
 product\_definition

#### 6.1.1.1.7 classification\_association to document\_version (as classified\_element)

AIM element: PATH

Reference path: applied\_classification\_assignment  
 applied\_classification\_assignment.items[i] ->  
 classification\_item = product\_definition\_formation  
 {product\_definition\_formation  
 product\_definition\_formation.of\_product ->  
 product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=  
 product\_category  
 product\_category.name = 'document'}

#### 6.1.1.1.8 classification\_association to item (as classified\_element)

AIM element: PATH

Reference path: applied\_classification\_assignment  
 applied\_classification\_assignment.items[i] ->  
 classification\_item  
 classification\_item = product  
 product

#### 6.1.1.1.9 classification\_association to item\_definition (as classified\_element)

AIM element: PATH

Reference path: applied\_classification\_assignment  
 applied\_classification\_assignment.items[i] ->  
 classification\_item  
 classification\_item = product\_definition  
 product\_definition

#### 6.1.1.1.10 classification\_association to item\_instance (as classified\_element)

AIM element: PATH

Reference path: applied\_classification\_assignment  
 applied\_classification\_assignment.items[i] ->  
 classification\_item

classification\_item = product\_definition\_relationship  
product\_definition\_relationship =>  
product\_definition\_usage =>  
assembly\_component\_usage

**6.1.1.1.11 classification\_association to item\_version (as classified\_element)**

AIM element: PATH

Reference path: applied\_classification\_assignment  
applied\_classification\_assignment.items[i] ->  
classification\_item  
classification\_item = product\_definition\_formation  
product\_definition\_formation

**6.1.1.1.12 classification\_association to material\_designation (as classified\_element)**

AIM element: PATH

Reference path: applied\_classification\_assignment  
applied\_classification\_assignment.items[i] ->  
classification\_item  
classification\_item = material\_designation  
material\_designation

**6.1.1.1.13 classification\_association to property (as classified\_element)**

AIM element: PATH

Reference path: applied\_classification\_assignment  
applied\_classification\_assignment.items[i] ->  
classification\_item  
classification\_item = general\_property  
general\_property

**6.1.1.1.14 classification\_association to property\_value\_association (as classified\_element)**

AIM element: PATH

Reference path: applied\_classification\_assignment  
applied\_classification\_assignment.items[i] ->  
classification\_item  
classification\_item = property\_definition  
property\_definition

**6.1.1.2 classification\_association\_relationship**

AIM element:

Source:

Rules:

Reference path:

**6.1.1.2.1 classification\_association\_relationship to classification\_association (as related)**

AIM element:

Source:

Rules:

Reference path:

#### **6.1.1.2.2 classification\_association\_relationship to classification\_association (as relating)**

AIM element:

Source:

Rules:

Reference path:

#### **6.1.1.2.3 relationship\_type**

AIM element:

Source:

Rules:

Reference path:

#### **6.1.1.3 classification\_attribute**

AIM element: property\_definition

Source: ISO 10303-41

##### **6.1.1.3.1 description**

AIM element: property\_definition.description

Source: ISO 10303-41

##### **6.1.1.3.2 id**

AIM element: property\_definition.id

Source: ISO 10303-41

##### **6.1.1.3.3 name**

AIM element: property\_definition.name

Source: ISO 10303-41

##### **6.1.1.3.4 classification\_attribute to property\_value\_representation (as allowed\_value)**

AIM element: PATH

Reference path: property\_definition  
 {derived\_property\_select = property\_definition  
 derived\_property\_select <-

```

general_property_association.derived_definition
general_property_association
general_property_association.base_definition ->
general_property}
represented_definition = property_definition
represented_definition <-
property_definition_representation.definition
property_definition_representation
property_definition_representation.used_representation ->
representation
    
```

**6.1.1.3.5 classification\_attribute to general\_classification (as associated\_classification)**

AIM element: PATH

Reference path: property\_definition  
 property\_definition.definition ->  
 characterized\_definition  
 characterized\_definition = characterized\_object  
 characterized\_object =>  
 characterized\_class

**6.1.1.3.6 classification\_attribute to property (as attribute\_definition)**

AIM element: PATH

Reference path: property\_definition  
 derived\_property\_select = property\_definition  
 derived\_property\_select <-  
 general\_property\_association.derived\_definition  
 general\_property\_association  
 {general\_property\_association.name = 'attribute definition'}  
 general\_property\_association.base\_definition ->  
 general\_property

**6.1.1.3.7 classification\_attribute to multi\_language\_string (as description)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([property\_definition.description]  
 [PATH])  
 #2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_property\_definition

Reference path: property\_definition  
 #1: (attribute\_language\_item = property\_definition  
 attribute\_language\_item <-  
 attribute\_language\_assignment.items[i]  
 attribute\_language\_assignment  
 {attribute\_language\_assignment <=  
 attribute\_classification\_assignment  
 attribute\_classification\_assignment.attribute\_name = 'description'})  
 #2: (multi\_language\_attribute\_item = property\_definition  
 multi\_language\_attribute\_item <-

```

multi_language_attribute_assignment.items[i]
multi_language_attribute_assignment
{multi_language_attribute_assignment <=
attribute_value_assignment
attribute_value_assignment.attribute_name = 'description'})

```

#### 6.1.1.3.8 classification\_attribute to multi\_language\_string (as name)

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string. [D:\Work\Documents\Standards\STEP\10303-214\html\\_with\\_change\\_marks\htmls\clause5aim\\_index\pxref.htm](D:\Work\Documents\Standards\STEP\10303-214\html_with_change_marks\htmls\clause5aim_index\pxref.htm)

AIM element: #1: ([property\_definition.name]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_property\_definition

Reference path: property\_definition  
#1: (attribute\_language\_item = property\_definition  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'name'})  
#2: (multi\_language\_attribute\_item = property\_definition  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'name'})

#### 6.1.1.4 classification\_system

AIM element: class\_system

Source: ISO 10303-214

Reference path: class\_system <=  
group

##### 6.1.1.4.1 description

AIM element: group.description

Source: ISO 10303-41

Reference path: class\_system <=  
group  
group.description

6.1.1.4.2 id

AIM element: group.name  
Source: ISO 10303-41  
Reference path: class\_system <=  
group  
group.name

6.1.1.4.3 classification\_system to multi\_language\_string (as description)

#1: For the primary\_language\_dependent\_string.  
#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([group.description]  
[PATH])  
#2: (PATH)  
Source: ISO 10303-41  
Rules: restrict\_multi\_language\_for\_group  
Reference path: class\_system <=  
group  
#1: (attribute\_language\_item = group  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#2: (multi\_language\_attribute\_item = group  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

6.1.1.5 din4000\_reference

#1: If din4000\_reference is referenced by a general\_classification as the classification\_source.  
#2: If din4000\_reference is referenced by a property as the property\_source.

AIM element: #1: (externally\_defined\_class)  
#2: (externally\_defined\_general\_property)  
Source: ISO 10303-214  
ISO 10303-214  
Reference path: #1: (externally\_defined\_class <=  
[class <=  
group]  
[externally\_defined\_item])  
#2: (externally\_defined\_general\_property <=

[general\_property]  
[externally\_defined\_item])

#### 6.1.1.5.1 characteristics\_code\_no

#1: If din4000\_reference is referenced by a general\_classification as the classification\_source.

#2: If din4000\_reference is referenced by a property as the property\_source.

AIM element: externally\_defined\_item.item\_id

Source: ISO 10303-41

Reference path: #1: (externally\_defined\_class <=)  
#2: (externally\_defined\_general\_property <=)  
externally\_defined\_item  
externally\_defined\_item.item\_id  
{externally\_defined\_item.item\_id ->  
source\_item  
source\_item = identifier  
identifier}

#### 6.1.1.5.2 part\_no

#1: If din4000\_reference is referenced by a general\_classification as the classification\_source.

#2: If din4000\_reference is referenced by a property as the property\_source.

AIM element: external\_source.source\_id

Source: ISO 10303-41

Reference path: #1: (externally\_defined\_class <=)  
#2: (externally\_defined\_general\_property <=)  
externally\_defined\_item  
externally\_defined\_item.source ->  
external\_source  
external\_source.source\_id  
{external\_source.source\_id ->  
source\_item  
source\_item = identifier  
identifier}

#### 6.1.1.6 external\_library\_reference

#1: If external\_library\_reference is referenced by a general\_classification as the classification\_source.

#2: If external\_library\_reference is referenced by a property as the property\_source.

AIM element: #1: (externally\_defined\_class)  
#2: (externally\_defined\_general\_property)

Source: ISO 10303-214  
ISO 10303-214

Reference path: #1: (externally\_defined\_class <=  
[class <=  
group]

[externally\_defined\_item])  
#2: (externally\_defined\_general\_property <=  
[general\_property]  
[externally\_defined\_item])

#### 6.1.1.6.1 description

#1: If external\_library\_reference is referenced by a general\_classification as the classification\_source.

#2: If external\_library\_reference is referenced by a property as the property\_source.

AIM element: external\_source.description

Source: ISO 10303-41

Reference path: #1: (externally\_defined\_class <=)  
#2: (externally\_defined\_general\_property <=)  
externally\_defined\_item  
externally\_defined\_item.source ->  
external\_source  
external\_source.description

#### 6.1.1.6.2 external\_id

#1: If external\_library\_reference is referenced by a general\_classification as the classification\_source.

#2: If external\_library\_reference is referenced by a property as the property\_source.

AIM element: externally\_defined\_item.item\_id

Source: ISO 10303-41

Reference path: #1: (externally\_defined\_class <=)  
#2: (externally\_defined\_general\_property <=)  
externally\_defined\_item  
externally\_defined\_item.item\_id  
{externally\_defined\_item.item\_id ->  
source\_item  
source\_item = identifier  
identifier}

#### 6.1.1.6.3 library\_type

#1: If external\_library\_reference is referenced by a general\_classification as the classification\_source.

#2: If external\_library\_reference is referenced by a property as the property\_source.

AIM element: external\_source.source\_id

Source: ISO 10303-41

Reference path: #1: (externally\_defined\_class <=)  
#2: (externally\_defined\_general\_property <=)  
externally\_defined\_item  
externally\_defined\_item.source ->  
external\_source  
external\_source.source\_id  
{external\_source.source\_id ->

```

source_item
source_item = identifier
identifier}

```

#### 6.1.1.6.4 external\_library\_reference to multi\_language\_string (as description)

#1: If external\_library\_reference is referenced by a general\_classification as the classification\_source.

#2: If external\_library\_reference is referenced by a property as the property\_source.

#3: For the primary\_language\_dependent\_string.

#4: For any additional\_language\_dependent\_string.

AIM element: #3: ([external\_source.description]  
[PATH])  
#4: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_external\_source

Reference path: #1: (externally\_defined\_class <=)  
#2: (externally\_defined\_general\_property <=)  
externally\_defined\_item  
externally\_defined\_item.source ->  
external\_source  
#3: (attribute\_language\_item = external\_source  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#4: (multi\_language\_attribute\_item = external\_source  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

#### 6.1.1.7 general\_classification

#1: If the general\_classification is not referenced by an classification\_attribute.

#2: If the general\_classification is referenced by an classification\_attribute.

AIM element: #1: (class)  
#2: (characterized\_class)

Source: ISO 10303-214  
ISO 10303-214

Reference path: #1: (class <=)  
#2: (characterized\_class <=  
class <=)  
group

**6.1.1.7.1 description**

AIM element: group.description  
Source: ISO 10303-41  
Reference path: class <=  
group  
group.description

**6.1.1.7.2 id**

AIM element: group.name  
Source: ISO 10303-41  
Reference path: class <=  
group  
group.name

**6.1.1.7.3 version\_id**

AIM element: identification\_assignment.assigned\_id  
Source: ISO 10303-41  
Rules: dependent\_instantiable\_identification\_role,  
restrict\_version\_assignment\_for\_class  
Reference path: class  
identification\_item = class  
identification\_item <-  
applied\_identification\_assignment.items[i]  
applied\_identification\_assignment <=  
identification\_assignment  
{identification\_assignment.role ->  
identification\_role  
identification\_role.name = 'version'}  
identification\_assignment.assigned\_id

**6.1.1.7.4 general\_classification to external\_library\_reference (as classification\_source)**

AIM element: IDENTICAL MAPPING  
Reference path: {class =>  
externally\_defined\_class}

**6.1.1.7.5 general\_classification to plib\_class\_reference (as classification\_source)**

AIM element: IDENTICAL MAPPING  
Reference path: {class =>  
externally\_defined\_class <=  
externally\_defined\_item  
externally\_defined\_item.source ->  
external\_source =>  
known\_source <=

```
pre_defined_item
pre_defined_item.name = 'ISO 13584 library'}
```

#### 6.1.1.7.6 general\_classification to multi\_language\_string (as description)

#1: If the general\_classification is not referenced by an classification\_attribute.

#2: If the general\_classification is referenced by an classification\_attribute.

#3: For the primary\_language\_dependent\_string.

#4: For any additional\_language\_dependent\_string.

AIM element: #3: ([group.description]  
[PATH])  
#4: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_group

Reference path: #2: (characterized\_class <=  
class <=)  
#1: (class <=)  
group  
#3: (attribute\_language\_item = group  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#4: (multi\_language\_attribute\_item = group  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

#### 6.1.1.7.7 general\_classification to classification\_system (as used\_classification\_system)

#1: If the general\_classification is not referenced by an classification\_attribute.

#2: If the general\_classification is referenced by an classification\_attribute.

AIM element: PATH

Rules: restrict\_applied\_classification\_assignment\_role,  
restrict\_class\_system\_assignment\_for\_class

Reference path: #2: (characterized\_class <=  
class)  
#1: (class)  
classification\_item = class  
classification\_item <-  
applied\_classification\_assignment.items[i]  
applied\_classification\_assignment <=  
classification\_assignment

```
{classification_assignment.role ->
classification_role
classification_role.name = 'class system membership'}
classification_assignment.assigned_class ->
group =>
class_system
```

#### 6.1.1.8 general\_classification\_hierarchy

AIM element: group\_relationship

Source: ISO 10303-41

Rules: restrict\_group\_relationship\_for\_general\_classification\_hierarchy

Reference path: {group\_relationship  
[group\_relationship.name = 'class hierarchy']  
[[group\_relationship.related\_group ->]  
[group\_relationship.relying\_group ->]  
group =>  
class]}

##### 6.1.1.8.1 general\_classification\_hierarchy to general\_classification (as sub\_classification)

AIM element: PATH

Reference path: group\_relationship  
group\_relationship.related\_group ->  
group =>  
class

##### 6.1.1.8.2 general\_classification\_hierarchy to general\_classification (as super\_classification)

AIM element: PATH

Reference path: group\_relationship  
group\_relationship.relying\_group ->  
group =>  
class

#### 6.1.1.9 plib\_class\_reference

AIM element: externally\_defined\_class

Source: ISO 10303-214

Rules: restrict\_name\_for\_known\_source

Reference path: externally\_defined\_class <=  
[class <=  
group]  
[externally\_defined\_item  
{externally\_defined\_item.source ->  
external\_source =>  
known\_source <=  
pre\_defined\_item  
pre\_defined\_item.name = 'ISO 13584 library'}]

**6.1.1.9.1 code**

AIM element: externally\_defined\_item.item\_id

Source: ISO 10303-41

Reference path: externally\_defined\_class <=  
 externally\_defined\_item  
 {externally\_defined\_item  
 externally\_defined\_item.source ->  
 external\_source =>  
 known\_source <=  
 pre\_defined\_item  
 pre\_defined\_item.name = 'ISO 13584 library'}  
 externally\_defined\_item.item\_id  
 {externally\_defined\_item.item\_id ->  
 source\_item  
 source\_item = identifier  
 identifier}

**6.1.1.9.2 supplier\_bsu**

AIM element: organization.id

Source: ISO 10303-41

Rules: dependent\_instantiable\_organization\_role,  
 externally\_defined\_class\_with\_known\_source\_requirement

Reference path: externally\_defined\_class <=  
 {externally\_defined\_class <=  
 externally\_defined\_item  
 externally\_defined\_item.source ->  
 external\_source =>  
 known\_source <=  
 pre\_defined\_item  
 pre\_defined\_item.name = 'ISO 13584 library'}  
 class  
 organization\_item = class  
 organization\_item <-  
 applied\_organization\_assignment.items[i]  
 applied\_organization\_assignment <=  
 organization\_assignment  
 {organization\_assignment.role ->  
 organization\_role  
 organization\_role.name = 'class supplier'}  
 organization\_assignment.assigned\_organization ->  
 organization  
 organization.id

**6.1.1.9.3 version**

AIM element: identification\_assignment.assigned\_id

Source: ISO 10303-41

Rules: plib\_class\_reference\_requires\_version

Reference path: externally\_defined\_class  
{externally\_defined\_class <=  
externally\_defined\_item  
externally\_defined\_item.source ->  
external\_source =>  
known\_source <=  
pre\_defined\_item  
pre\_defined\_item.name = 'ISO 13584 library'}  
external\_identification\_item = externally\_defined\_class  
external\_identification\_item <-  
applied\_external\_identification\_assignment.items[i]  
applied\_external\_identification\_assignment <=  
external\_identification\_assignment <=  
{external\_identification\_assignment  
external\_identification\_assignment.source ->  
external\_source =>  
known\_source <=  
pre\_defined\_item  
pre\_defined\_item.name = 'ISO 13584 library'}  
identification\_assignment  
{identification\_assignment.role ->  
identification\_role  
identification\_role.name = 'version'}  
identification\_assignment.assigned\_id

#### 6.1.1.10 plib\_property\_reference

AIM element: externally\_defined\_general\_property

Source: ISO 10303-214

Reference path: externally\_defined\_general\_property <=  
[general\_property]  
[externally\_defined\_item]

##### 6.1.1.10.1 code

AIM element: externally\_defined\_item.item\_id

Source: ISO 10303-41

Reference path: externally\_defined\_general\_property <=  
externally\_defined\_item  
externally\_defined\_item.item\_id  
{externally\_defined\_item.item\_id ->  
source\_item  
source\_item = identifier  
identifier}

##### 6.1.1.10.2 version

AIM element: identification\_assignment.assigned\_id

Source: ISO 10303-41

Rules: plib\_property\_reference\_requires\_version

Reference path: externally\_defined\_general\_property  
{externally\_defined\_general\_property <=

```

externally_defined_item
externally_defined_item.source ->
external_source =>
known_source <=
pre_defined_item
pre_defined_item.name = 'ISO 13584 library'}
external_identification_item = externally_defined_general_property
external_identification_item <-
applied_external_identification_assignment.items[i]
applied_external_identification_assignment <=
external_identification_assignment <=
{external_identification_assignment
external_identification_assignment.source ->
external_source =>
known_source <=
pre_defined_item
pre_defined_item.name = 'ISO 13584 library'}
identification_assignment
{identification_assignment.role ->
identification_role
identification_role.name = 'version'}
identification_assignment.assigned_id

```

#### 6.1.1.10.3 plib\_property\_reference to plib\_class\_reference (as name\_scope)

AIM element: PATH

Source: ISO 10303-41

Rules: plib\_property\_reference\_requires\_name\_scope,  
restrict\_externally\_defined\_item\_relationship

Reference path: externally\_defined\_general\_property <=  
externally\_defined\_item <-  
{externally\_defined\_item.source ->  
external\_source =>  
known\_source <=  
pre\_defined\_item  
pre\_defined\_item.name = 'ISO 13584 library'}  
externally\_defined\_item\_relationship.relateing\_item  
externally\_defined\_item\_relationship  
{externally\_defined\_item\_relationship.name = 'name scope'}  
externally\_defined\_item\_relationship.related\_item ->  
externally\_defined\_item =>  
{externally\_defined\_item.source ->  
external\_source =>  
known\_source <=  
pre\_defined\_item  
pre\_defined\_item.name = 'ISO 13584 library'}  
externally\_defined\_class

#### 6.1.1.11 specific\_item\_classification

AIM element: product\_related\_product\_category

Source: ISO 10303-41

**6.1.1.11.1 classification\_name**

AIM element: product\_category.name

Source: ISO 10303-41

Reference path: product\_related\_product\_category <=  
product\_category  
product\_category.name  
{(product\_category.name)  
(product\_category.name = 'accessory item')  
(product\_category.name = 'adaptive item')  
(product\_category.name = 'assembly')  
(product\_category.name = 'assembly item')  
(product\_category.name = 'cutting item')  
(product\_category.name = 'cutting tool')  
(product\_category.name = 'detail')  
(product\_category.name = 'tool item')}

**6.1.1.11.2 description**

AIM element: product\_category.description

Source: ISO 10303-41

Reference path: product\_related\_product\_category <=  
product\_category  
product\_category.description

**6.1.1.11.3 specific\_item\_classification to item (as associated\_item)**

AIM element: PATH

Reference path: product\_related\_product\_category  
product\_related\_product\_category.products[i] ->  
product

**6.1.1.11.4 specific\_item\_classification to multi\_language\_string (as description)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([product\_category.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product\_related\_product\_category

Reference path: product\_related\_product\_category  
#1: (attribute\_language\_item = product\_related\_product\_category  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment

```

attribute_classification_assignment.attribute_name = 'description'})
#2: (multi_language_attribute_item = product_related_product_category
multi_language_attribute_item <-
multi_language_attribute_assignment.items[i]
multi_language_attribute_assignment
{multi_language_attribute_assignment <=
attribute_value_assignment
attribute_value_assignment.attribute_name = 'description'})

```

#### 6.1.1.12 specific\_item\_classification\_hierarchy

AIM element: product\_category\_relationship

Source: ISO 10303-41

Reference path: {product\_category\_relationship  
product\_category\_relationship.name = 'hierarchy'}

##### 6.1.1.12.1 specific\_item\_classification\_hierarchy to specific\_item\_classification (as sub\_classification)

AIM element: PATH

Reference path: product\_category\_relationship  
product\_category\_relationship.sub\_category ->  
product\_category =>  
product\_related\_product\_category

##### 6.1.1.12.2 specific\_item\_classification\_hierarchy to specific\_item\_classification (as super\_classification)

AIM element: PATH

Reference path: product\_category\_relationship  
product\_category\_relationship.category ->  
product\_category =>  
product\_related\_product\_category

#### 6.1.2 dimension\_tolerance UoF

##### 6.1.2.1 limits\_and\_fits

AIM element: limits\_and\_fits

Source: ISO 10303-47

##### 6.1.2.1.1 deviation

AIM element: limits\_and\_fits.form\_variance

Source: ISO 10303-47

##### 6.1.2.1.2 fitting\_type

AIM element: limits\_and\_fits.zone\_variance

Source: ISO 10303-47

**6.1.2.1.3 grade**

AIM element: limits\_and\_fits.grade

Source: ISO 10303-47

**6.1.2.2 plus\_minus\_bounds**

AIM element: tolerance\_value

Source: ISO 10303-47

Rules: dependent\_instantiable\_tolerance\_value

**6.1.2.2.1 lower\_bound**

AIM element: tolerance\_value.lower\_bound

Source: ISO 10303-47

**6.1.2.2.2 significant\_digits**

AIM element: precision\_qualifier.precision\_value

Source: ISO 10303-45

Rules: dependent\_instantiable\_precision\_qualifier

Reference path: [tolerance\_value  
tolerance\_value.lower\_bound ->  
measure\_with\_unit <-]  
[tolerance\_value  
tolerance\_value.upper\_bound ->  
measure\_with\_unit <-]  
measure\_qualification.qualified\_measure  
measure\_qualification  
measure\_qualification.qualifiers[i] ->  
value\_qualifier  
value\_qualifier = precision\_qualifier  
precision\_qualifier  
precision\_qualifier.precision\_value

**6.1.2.2.3 upper\_bound**

AIM element: tolerance\_value.upper\_bound

Source: ISO 10303-47

**6.1.2.2.4 value\_determination**

AIM element: type\_qualifier.name

Source: ISO 10303-45

Rules: dependent\_instantiable\_type\_qualifier

Reference path: tolerance\_value  
[tolerance\_value.lower\_bound ->]

```
[tolerance_value.upper_bound->]
measure_with_unit <-
measure_qualification.qualified_measure
measure_qualification
measure_qualification.qualifiers[i] ->
value_qualifier
value_qualifier = type_qualifier
type_qualifier
type_qualifier.name
{(type_qualifier.name)
(type_qualifier.name = 'required')
(type_qualifier.name = 'designed')
(type_qualifier.name = 'calculated')
(type_qualifier.name = 'measured')
(type_qualifier.name = 'estimated')}
```

### 6.1.2.3 value\_limitation

AIM element: plus\_minus\_tolerance

Source: ISO 10303-47

#### 6.1.2.3.1 envelope

#1: If value\_limitation.envelope is TRUE.

#2: If value\_limitation.envelope is FALSE.

AIM element: #1: (applied\_classification\_assignment)  
#2: (IDENTICAL MAPPING)

Source: ISO 10303-214

Rules: dependent\_instantiable\_classification\_role

Reference path: #1: (plus\_minus\_tolerance  
classification\_item = plus\_minus\_tolerance  
classification\_item <-  
applied\_classification\_assignment.items[i]  
applied\_classification\_assignment <=  
classification\_assignment  
{[classification\_assignment.role ->  
classification\_role  
classification\_role.name = 'further tolerance requirement']  
[classification\_assignment.assigned\_class ->  
group  
{[group.name = 'E']  
[group.description = 'dimensioning principle']}]})  
#2: (plus\_minus\_tolerance)

#### 6.1.2.3.2 value\_limitation to limits\_and\_fits (as is\_defined\_by)

AIM element: PATH

Reference path: plus\_minus\_tolerance  
plus\_minus\_tolerance.range ->  
tolerance\_method\_definition  
tolerance\_method\_definition = limits\_and\_fits  
limits\_and\_fits

**6.1.2.3.3 value\_limitation to plus\_minus\_bounds (as is\_defined\_by)**

AIM element: PATH

Reference path: plus\_minus\_tolerance  
plus\_minus\_tolerance.range ->  
tolerance\_method\_definition  
tolerance\_method\_definition = tolerance\_value  
tolerance\_value

**6.1.2.3.4 value\_limitation to numerical\_value (as limited\_value)**

AIM element: PATH

Reference path: plus\_minus\_tolerance  
plus\_minus\_tolerance.toleranced\_dimension ->  
dimensional\_characteristic  
dimensional\_characteristic = dimensional\_size  
dimensional\_size  
dimensional\_characteristic = dimensional\_size  
dimensional\_characteristic <-  
dimensional\_characteristic\_representation.dimension  
dimensional\_characteristic\_representation  
dimensional\_characteristic\_representation.representation ->  
shape\_dimension\_representation <=  
shape\_representation <=  
representation  
representation.items[i] ->  
representation\_item =>  
measure\_representation\_item

**6.1.3 effectivity UoF**

**6.1.3.1 duration**

AIM element: time\_measure\_with\_unit

Source: ISO 10303-41

Rules: dependent\_instantiable\_measure\_with\_unit

**6.1.3.1.1 time**

AIM element: time\_measure

Source: ISO 10303-41

Reference path: time\_measure\_with\_unit <=  
measure\_with\_unit  
measure\_with\_unit.value\_component ->  
measure\_value  
measure\_value = time\_measure  
time\_measure

**6.1.3.1.2 time\_unit**

AIM element: time\_unit

Source: ISO 10303-41

Rules: dependent\_instantiable\_named\_unit

Reference path: time\_measure\_with\_unit <=  
 measure\_with\_unit  
 measure\_with\_unit.unit\_component ->  
 unit  
 unit = named\_unit  
 named\_unit =>  
 time\_unit

### 6.1.3.2 effectivity

#1: If at least one of the attributes start\_definition or end\_definition is specified.

#2: If neither one of the attributes start\_definition and end\_definition are specified.

#3: If a start\_definition is given and no period is specified.

#4: If no start\_definition is given or a period is specified.

AIM element: effectivity

Source: ISO 10303-41

Reference path: {#1: (effectivity =>  
 #3: (dated\_effectivity)  
 #4: (time\_interval\_based\_effectivity))  
 #2: (|effectivity| <-  
 effectivity\_relationship.related\_effectivity  
 effectivity\_relationship  
 effectivity\_relationship.name = 'inheritance')}

#### 6.1.3.2.1 description

AIM element: effectivity.description

Source: ISO 10303-41

#### 6.1.3.2.2 effectivity\_context

AIM element: effectivity.name

Source: ISO 10303-41

#### 6.1.3.2.3 id

AIM element: effectivity.id

Source: ISO 10303-41

#### 6.1.3.2.4 version\_id

AIM element: identification\_assignment.assigned\_id

Source: ISO 10303-41

Rules: dependent\_instantiable\_identification\_role,  
restrict\_version\_assignment\_for\_effectivity

Reference path: effectivity  
identification\_item = effectivity  
identification\_item <-  
applied\_identification\_assignment.items[i]  
applied\_identification\_assignment <=  
identification\_assignment  
{identification\_assignment.role ->  
identification\_role  
identification\_role.name = 'version'}  
identification\_assignment.assigned\_id

**6.1.3.2.5 effectivity to organization (as concerned\_organization)**

AIM element: PATH

Reference path: effectivity <-  
applied\_organization\_assignment.items[i]  
applied\_organization\_assignment <=  
organization\_assignment  
{organization\_assignment.role ->  
organization\_role  
organization\_role.name = 'concerned organization'}  
organization\_assignment.assigned\_organization =>  
organization

**6.1.3.2.6 effectivity to multi\_language\_string (as description)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([effectivity.description] ·  
[PATH])  
#2: (PATH)

Rules: restrict\_multi\_language\_for\_effectivity

Reference path: effectivity  
#1: (attribute\_language\_item = effectivity  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#2: (multi\_language\_attribute\_item = effectivity  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

**6.1.3.2.7 effectivity to date\_time (as end\_definition)**

#1: If a start\_definition is given and no period is specified.

#2: If no start\_definition is given or a period is specified.

#3: If start\_definition or end\_definition reference a date\_time with no time specified.

#4: If start\_definition or end\_definition reference a date\_time with a time specified.

AIM element: PATH

Reference path: effectivity =>  
 #1: (dated\_effectivity  
 dated\_effectivity.effectivity\_end\_date ->  
 date\_time\_or\_event\_occurrence  
 date\_time\_or\_event\_occurrence = date\_time\_select  
 #3: (date\_time\_select = date  
 date =>  
 calendar\_date)  
 #4: (date\_time\_select = date\_and\_time  
 date\_and\_time))  
 #2: (time\_interval\_based\_effectivity  
 time\_interval\_based\_effectivity.effectivity\_period ->  
 time\_interval =>  
 time\_interval\_with\_bounds  
 time\_interval\_with\_bounds.secondary\_bound ->  
 date\_time\_or\_event\_occurrence  
 date\_time\_or\_event\_occurrence = date\_time\_select  
 #3: (date\_time\_select = date  
 date =>  
 calendar\_date)  
 #4: (date\_time\_select = date\_and\_time  
 date\_and\_time))

#### 6.1.3.2.8 effectivity to duration (as period)

AIM element: PATH

Reference path: effectivity =>  
 time\_interval\_based\_effectivity  
 time\_interval\_based\_effectivity.effectivity\_period ->  
 time\_interval =>  
 time\_interval\_with\_bounds  
 time\_interval\_with\_bounds.duration ->  
 time\_measure\_with\_unit

#### 6.1.3.2.9 effectivity to date\_time (as start\_definition)

#1: If a start\_definition is given and no period is specified.

#2: If no start\_definition is given or a period is specified.

#3: If start\_definition or end\_definition reference a date\_time with no time specified.

#4: If start\_definition or end\_definition reference a date\_time with a time specified.

AIM element: PATH

Reference path: effectivity =>  
 #1: (dated\_effectivity  
 dated\_effectivity.effectivity\_start\_date ->  
 date\_time\_or\_event\_occurrence

```

date_time_or_event_occurrence = date_time_select
#3: (date_time_select = date
date =>
calendar_date)
#4: (date_time_select = date_and_time
date_and_time))
#2: (time_interval_based_effectivity
time_interval_based_effectivity.effectivity_period ->
time_interval =>
time_interval_with_bounds
time_interval_with_bounds.primary_bound ->
date_time_or_event_occurrence
date_time_or_event_occurrence = date_time_select
#3: (date_time_select = date
date =>
calendar_date)
#4: (date_time_select = date_and_time
date_and_time))

```

### 6.1.3.3 effectivity\_assignment

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: #1: (applied\_effectivity\_assignment)  
#2: (applied\_ineffectivity\_assignment)

Source: ISO 10303-214  
ISO 10303-214

Reference path: #1: (applied\_effectivity\_assignment <=  
effectivity\_assignment)  
#2: (applied\_ineffectivity\_assignment <=  
effectivity\_assignment)

#### 6.1.3.3.1 effectivity\_indication

AIM element: IDENTICAL MAPPING

#### 6.1.3.3.2 role

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: object\_role.name

Source: ISO 10303-41

Rules: dependent\_instantiable\_object\_role

Reference path: #1: (applied\_effectivity\_assignment <=  
#2: (applied\_ineffectivity\_assignment <=  
effectivity\_assignment  
effectivity\_assignment.role ->  
object\_role  
object\_role.name  
{(object\_role.name)

```
(object_role.name = 'actual')
(object_role.name = 'planned')
(object_role.name = 'required')}
```

#### 6.1.3.3.3 effectivity\_assignment to effectivity (as assigned\_effectivity)

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: PATH

Reference path: #1: (applied\_effectivity\_assignment <=)  
 #2: (applied\_ineffectivity\_assignment <=)  
 effectivity\_assignment  
 effectivity\_assignment.assigned\_effectivity ->  
 effectivity

#### 6.1.3.3.4 effectivity\_assignment to classification\_system (as effective\_element)

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: PATH

Reference path: #1: (applied\_effectivity\_assignment  
 applied\_effectivity\_assignment.items[i] ->)  
 #2: (applied\_ineffectivity\_assignment  
 applied\_ineffectivity\_assignment.items[i] ->)  
 effectivity\_item  
 effectivity\_item = class\_system  
 class\_system

#### 6.1.3.3.5 effectivity\_assignment to document (as effective\_element)

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: PATH

Reference path: #1: (applied\_effectivity\_assignment  
 applied\_effectivity\_assignment.items[i] ->)  
 #2: (applied\_ineffectivity\_assignment  
 applied\_ineffectivity\_assignment.items[i] ->)  
 effectivity\_item  
 effectivity\_item = product  
 product  
 {product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=  
 product\_category  
 product\_category.name = 'document'}

#### 6.1.3.3.6 effectivity\_assignment to document\_file (as effective\_element)

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: PATH

Reference path: #1: (applied\_effectivity\_assignment  
applied\_effectivity\_assignment.items[i] ->)  
#2: (applied\_ineffectivity\_assignment  
applied\_ineffectivity\_assignment.items[i] ->)  
effectivity\_item  
effectivity\_item = document\_file  
document\_file

#### 6.1.3.3.7 effectivity\_assignment to document\_representation (as effective\_element)

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: PATH

Reference path: #1: (applied\_effectivity\_assignment  
applied\_effectivity\_assignment.items[i] ->)  
#2: (applied\_ineffectivity\_assignment  
applied\_ineffectivity\_assignment.items[i] ->)  
effectivity\_item  
effectivity\_item = product\_definition  
product\_definition

#### 6.1.3.3.8 effectivity\_assignment to document\_version (as effective\_element)

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: PATH

Reference path: #1: (applied\_effectivity\_assignment  
applied\_effectivity\_assignment.items[i] ->)  
#2: (applied\_ineffectivity\_assignment  
applied\_ineffectivity\_assignment.items[i] ->)  
effectivity\_item  
effectivity\_item = product\_definition\_formation  
product\_definition\_formation  
{product\_definition\_formation.of\_product ->  
product <-  
product\_related\_product\_category.products[i]  
product\_related\_product\_category <=  
product\_category  
product\_category.name = 'document'}

#### 6.1.3.3.9 effectivity\_assignment to item (as effective\_element)

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: PATH

Reference path: #1: (applied\_effectivity\_assignment  
 applied\_effectivity\_assignment.items[i] ->)  
 #2: (applied\_ineffectivity\_assignment  
 applied\_ineffectivity\_assignment.items[i] ->)  
 effectivity\_item  
 effectivity\_item = product  
 product

#### 6.1.3.3.10 effectivity\_assignment to item\_instance (as effective\_element)

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: PATH

Reference path: #1: (applied\_effectivity\_assignment  
 applied\_effectivity\_assignment.items[i] ->)  
 #2: (applied\_ineffectivity\_assignment  
 applied\_ineffectivity\_assignment.items[i] ->)  
 effectivity\_item  
 effectivity\_item = product\_definition\_relationship  
 product\_definition\_relationship =>  
 product\_definition\_usage =>  
 assembly\_component\_usage

#### 6.1.3.3.11 effectivity\_assignment to item\_structure\_association (as effective\_element)

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: PATH

Reference path: #1: (applied\_effectivity\_assignment  
 applied\_effectivity\_assignment.items[i] ->)  
 #2: (applied\_ineffectivity\_assignment  
 applied\_ineffectivity\_assignment.items[i] ->)  
 effectivity\_item  
 effectivity\_item = product\_definition\_relationship  
 product\_definition\_relationship

#### 6.1.3.3.12 effectivity\_assignment to item\_version (as effective\_element)

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: PATH

Reference path: #1: (applied\_effectivity\_assignment  
 applied\_effectivity\_assignment.items[i] ->)  
 #2: (applied\_ineffectivity\_assignment  
 applied\_ineffectivity\_assignment.items[i] ->)  
 effectivity\_item  
 effectivity\_item = product\_definition\_formation  
 product\_definition\_formation

**6.1.3.3.13 effectivity\_assignment to material\_designation (as effective\_element)**

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: PATH

Reference path: #1: (applied\_effectivity\_assignment  
applied\_effectivity\_assignment.items[i] ->)  
#2: (applied\_ineffectivity\_assignment  
applied\_ineffectivity\_assignment.items[i] ->)  
effectivity\_item  
effectivity\_item = material\_designation  
material\_designation

**6.1.3.3.14 effectivity\_assignment to physical\_item\_location\_association (as effective\_element)**

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: PATH

Reference path: #1: (applied\_effectivity\_assignment  
applied\_effectivity\_assignment.items[i] ->)  
#2: (applied\_ineffectivity\_assignment  
applied\_ineffectivity\_assignment.items[i] ->)  
effectivity\_item  
effectivity\_item = applied\_location\_assignment  
applied\_location\_assignment

**6.1.3.3.15 effectivity\_assignment to physical\_item\_state\_association (as effective\_element)**

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: PATH

Reference path: #1: (applied\_effectivity\_assignment  
applied\_effectivity\_assignment.items[i] ->)  
#2: (applied\_ineffectivity\_assignment  
applied\_ineffectivity\_assignment.items[i] ->)  
effectivity\_item  
effectivity\_item = applied\_state\_type\_assignment  
applied\_state\_type\_assignment

**6.1.3.3.16 effectivity\_assignment to physical\_item\_structure\_association (as effective\_element)**

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: PATH

Reference path: #1: (applied\_effectivity\_assignment  
applied\_effectivity\_assignment.items[i] ->)

#2: (applied\_ineffectivity\_assignment  
 applied\_ineffectivity\_assignment.items[i] ->  
 effectivity\_item  
 effectivity\_item = product\_definition\_relationship  
 product\_definition\_relationship

#### 6.1.3.3.17 property

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: PATH

Reference path: #1: (applied\_effectivity\_assignment  
 applied\_effectivity\_assignment.items[i] ->  
 #2: (applied\_ineffectivity\_assignment  
 applied\_ineffectivity\_assignment.items[i] ->  
 effectivity\_item  
 effectivity\_item = general\_property  
 general\_property

#### 6.1.3.3.18 effectivity\_assignment to property\_value\_association (as effective\_element)

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: PATH

Reference path: #1: (applied\_effectivity\_assignment  
 applied\_effectivity\_assignment.items[i] ->  
 #2: (applied\_ineffectivity\_assignment  
 applied\_ineffectivity\_assignment.items[i] ->  
 effectivity\_item  
 effectivity\_item = property\_definition  
 property\_definition

#### 6.1.3.3.19 effectivity\_assignment to realized\_item\_association (as effective\_element)

#1: If effectivity\_assignment.effectivity\_indication is TRUE.

#2: If effectivity\_assignment.effectivity\_indication is FALSE.

AIM element: PATH

Reference path: #1: (applied\_effectivity\_assignment  
 applied\_effectivity\_assignment.items[i] ->  
 #2: (applied\_ineffectivity\_assignment  
 applied\_ineffectivity\_assignment.items[i] ->  
 effectivity\_item  
 effectivity\_item = product\_definition\_relationship  
 product\_definition\_relationship

#### 6.1.3.4 effectivity\_relationship

AIM element: effectivity\_relationship

Source: ISO 10303-41

**6.1.3.4.1 description**

AIM element: effectivity\_relationship.description

Source: ISO 10303-41

**6.1.3.4.2 relation\_type**

AIM element: effectivity\_relationship.name

Source: ISO 10303-41

Reference path: {effectivity\_relationship  
(effectivity\_relationship.name)  
(effectivity\_relationship.name = 'constraint')  
(effectivity\_relationship.name = 'inheritance')}

**6.1.3.4.3 effectivity\_relationship to multi\_language\_string (as description)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([effectivity\_relationship.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_effectivity\_relationship

Reference path: effectivity\_relationship  
#1: (attribute\_language\_item = effectivity\_relationship  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#2: (multi\_language\_attribute\_item = effectivity\_relationship  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

**6.1.3.4.4 effectivity\_relationship to effectivity (as related)**

AIM element: PATH

Rules: restrict\_effectivity\_for\_effectivity\_relationship

Reference path: effectivity\_relationship  
effectivity\_relationship.related\_effectivity ->  
effectivity

**6.1.3.4.5 effectivity\_relationship to effectivity (as relating)**

AIM element: PATH

Rules: restrict\_effectivity\_for\_effectivity\_relationship

Reference path: effectivity\_relationship  
effectivity\_relationship.relating\_effectivity ->  
effectivity

**6.1.4 external\_reference\_mechanism UoF****6.1.4.1 cartesian\_coordinate\_space**

AIM element: geometric\_representation\_context

Source: ISO 10303-42

**6.1.4.1.1 cartesian\_coordinate\_space to unit (as unit\_of\_values)**

#1: If the unit has a commonly used descriptor.

#2: If the unit descriptor is constructed of more than one unit name.

AIM element: global\_unit\_assigned\_context.units

Source: ISO 10303-41

Reference path: geometric\_representation\_context <=  
representation\_context =>  
global\_unit\_assigned\_context  
global\_unit\_assigned\_context.units[i] ->  
unit  
#1: (unit = named\_unit  
named\_unit)  
#2: (unit = derived\_unit  
derived\_unit)

**6.1.4.2 cartesian\_coordinate\_space\_2d**

AIM element: geometric\_representation\_context

Source: ISO 10303-42

Reference path: {geometric\_representation\_context  
geometric\_representation\_context.coordinate\_space\_dimension = 2}

**6.1.4.3 cartesian\_coordinate\_space\_3d**

AIM element: geometric\_representation\_context

Source: ISO 10303-42

Reference path: {geometric\_representation\_context  
geometric\_representation\_context.coordinate\_space\_dimension = 3}

#### 6.1.4.4 digital\_document

#1: If element digital\_document.file is referencing no digital\_file.

#2: If element digital\_document.file is referencing one or more digital\_file.

AIM element: #1: (product\_definition)  
#2: (product\_definition\_with\_associated\_documents)

Source: ISO 10303-41  
ISO 10303-41

Reference path: {#1: (product\_definition)  
#2: (product\_definition\_with\_associated\_documents <=  
product\_definition)  
product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'digital document definition'}

##### 6.1.4.4.1 digital\_document to digital\_file (as file)

AIM element: PATH

Reference path: product\_definition =>  
{product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'digital document definition'}  
product\_definition\_with\_associated\_documents  
product\_definition\_with\_associated\_documents.documentation\_ids[i] ->  
document =>  
{document.representation\_types[i] ->  
document\_representation\_type  
document\_representation\_type.name = 'digital'}  
document\_file

##### 6.1.4.5 digital\_file

AIM element: document\_file

Source: ISO 10303-214

Reference path: document\_file <=  
[document  
{document.representation\_types[1] ->  
document\_representation\_type  
document\_representation\_type.name = 'digital'}]  
[characterized\_object]

##### 6.1.4.6 document

AIM element: product

Source: ISO 10303-41

Rules: product\_requires\_category,  
product\_requires\_version,  
restrict\_product\_category\_for\_product

Reference path: {product <-  
product\_related\_product\_category.products[i]  
product\_related\_product\_category <=  
product\_category  
product\_category.name = 'document'}

#### 6.1.4.6.1 description

AIM element: product.description

Source: ISO 10303-41

#### 6.1.4.6.2 document\_id

AIM element: product.id

Source: ISO 10303-41

#### 6.1.4.6.3 name

AIM element: product.name

Source: ISO 10303-41

#### 6.1.4.6.4 document to multi\_language\_string (as description)

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string. [D:\Work\Documents\Standards\STEP\10303-214\html\\_with\\_change\\_marks\htmls\clause5\aim\\_index\pxref.htm](D:\Work\Documents\Standards\STEP\10303-214\html_with_change_marks\htmls\clause5\aim_index\pxref.htm)

AIM element: #1: ([product.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product

Reference path: product  
#1: (attribute\_language\_item = product  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#2: (multi\_language\_attribute\_item = product  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

**6.1.4.6.5 document to multi\_language\_string (as name)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([product.name]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product

Reference path: product  
#1: (attribute\_language\_item = product  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'name'})  
#2: (multi\_language\_attribute\_item = product  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'name'})

**6.1.4.7 document\_assignment**

AIM element: applied\_document\_reference

Source: ISO 10303-214

Reference path: applied\_document\_reference <=  
document\_reference

**6.1.4.7.1 role**

AIM element: object\_role.name

Source: ISO 10303-41

Rules: dependent\_instantiable\_object\_role

Reference path: applied\_document\_reference <=  
document\_reference  
document\_reference.role ->  
object\_role  
object\_role.name  
{(object\_role.name)  
(object\_role.name = 'additional information')  
(object\_role.name = 'behavior')  
(object\_role.name = 'catalogue')  
(object\_role.name = 'description')  
(object\_role.name = 'informative')  
(object\_role.name = 'mandatory')}

```
(object_role.name = 'manual')
(object_role.name = 'mathematical description')
(object_role.name = 'specification')}
```

#### 6.1.4.7.2 document\_assignment to document (as assigned\_document)

AIM element: PATH

Reference path: applied\_document\_reference <=  
 document\_reference  
 document\_reference.assigned\_document ->  
 document <-  
 {document.kind ->  
 document\_type  
 document\_type.product\_data\_type = 'configuration controlled document'}  
 document\_product\_association.relying\_document  
 document\_product\_association  
 {document\_product\_association =>  
 document\_product\_equivalence}  
 document\_product\_association.related\_product ->  
 product\_or\_formation\_or\_definition  
 product\_or\_formation\_or\_definition = product  
 product  
 {product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=  
 product\_category  
 product\_category.name = 'document'}

#### 6.1.4.7.3 document\_assignment to document\_version (as assigned\_document)

AIM element: PATH

Reference path: applied\_document\_reference <=  
 document\_reference  
 document\_reference.assigned\_document ->  
 document <-  
 {document.kind ->  
 document\_type  
 document\_type.product\_data\_type = 'configuration controlled document version'}  
 document\_product\_association.relying\_document  
 document\_product\_association  
 {document\_product\_association =>  
 document\_product\_equivalence}  
 document\_product\_association.related\_product ->  
 product\_or\_formation\_or\_definition  
 product\_or\_formation\_or\_definition = product\_definition\_formation  
 product\_definition\_formation  
 {product\_definition\_formation.of\_product ->  
 product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=  
 product\_category  
 product\_category.name = 'document'}

#### 6.1.4.7.4 document\_assignment to classification\_attribute (as is\_assigned\_to)

AIM element: PATH

Reference path: applied\_document\_reference  
applied\_document\_reference.items[i] ->  
document\_reference\_item  
document\_reference\_item = property\_definition  
property\_definition

**6.1.4.7.5 document\_assignment to classification\_system (as is\_assigned\_to)**

AIM element: PATH

Reference path: applied\_document\_reference  
applied\_document\_reference.items[i] ->  
document\_reference\_item  
document\_reference\_item = class\_system  
class\_system

**6.1.4.7.6 document\_assignment to general\_classification (as is\_assigned\_to)**

AIM element: PATH

Reference path: applied\_document\_reference  
applied\_document\_reference.items[i] ->  
document\_reference\_item  
document\_reference\_item = class  
class

**6.1.4.7.7 document\_assignment to item (as is\_assigned\_to)**

AIM element: PATH

Reference path: applied\_document\_reference  
applied\_document\_reference.items[i] ->  
document\_reference\_item  
document\_reference\_item = product  
product

**6.1.4.7.8 document\_assignment to item\_definition (as is\_assigned\_to)**

AIM element: PATH

Reference path: applied\_document\_reference  
applied\_document\_reference.items[i] ->  
document\_reference\_item  
document\_reference\_item = product\_definition  
product\_definition  
{product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'part definition'}

**6.1.4.7.9 document\_assignment to item\_instance (as is\_assigned\_to)**

AIM element: PATH

Reference path: applied\_document\_reference  
applied\_document\_reference.items[i] ->  
document\_reference\_item  
document\_reference\_item = product\_definition\_relationship

product\_definition\_relationship =>  
 product\_definition\_usage =>  
 assembly\_component\_usage

#### 6.1.4.7.10 document\_assignment to item\_structure\_association (as is\_assigned\_to)

AIM element: PATH

Reference path: applied\_document\_reference  
 applied\_document\_reference.items[i] ->  
 document\_reference\_item  
 document\_reference\_item = product\_definition\_relationship  
 product\_definition\_relationship

#### 6.1.4.7.11 document\_assignment to item\_version (as is\_assigned\_to)

AIM element: PATH

Reference path: applied\_document\_reference  
 applied\_document\_reference.items[i] ->  
 document\_reference\_item  
 document\_reference\_item = product\_definition\_formation  
 product\_definition\_formation

#### 6.1.4.7.12 document\_assignment to mated\_item\_relationship (as is\_assigned\_to)

AIM element: PATH

Reference path: applied\_document\_reference  
 applied\_document\_reference.items[i] ->  
 document\_reference\_item  
 document\_reference\_item = product\_definition\_relationship  
 product\_definition\_relationship

#### 6.1.4.7.13 document\_assignment to material\_designation (as is\_assigned\_to)

AIM element: PATH

Reference path: applied\_document\_reference  
 applied\_document\_reference.items[i] ->  
 document\_reference\_item  
 document\_reference\_item = material\_designation  
 material\_designation

#### 6.1.4.7.14 document\_assignment to organization (as is\_assigned\_to)

AIM element: PATH

Reference path: applied\_document\_reference  
 applied\_document\_reference.items[i] ->  
 document\_reference\_item  
 document\_reference\_item = organization  
 organization

#### 6.1.4.7.15 document\_assignment to person (as is\_assigned\_to)

AIM element: PATH

Reference path: applied\_document\_reference  
applied\_document\_reference.items[i] ->  
document\_reference\_item  
document\_reference\_item = person  
person

**6.1.4.7.16 document\_assignment to physical\_item\_structure\_association (as is\_assigned\_to)**

AIM element: PATH

Reference path: applied\_document\_reference  
applied\_document\_reference.items[i] ->  
document\_reference\_item  
document\_reference\_item = product\_definition\_relationship  
product\_definition\_relationship  
{[product\_definition\_relationship.relying\_product\_definition ->  
product\_definition  
product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'physical occurrence']  
[product\_definition\_relationship.related\_product\_definition ->  
product\_definition  
product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'physical occurrence']]

**6.1.4.7.17 document\_assignment to property (as is\_assigned\_to)**

AIM element: PATH

Reference path: applied\_document\_reference  
applied\_document\_reference.items[i] ->  
document\_reference\_item  
document\_reference\_item = general\_property  
general\_property

**6.1.4.7.18 document\_assignment to specific\_item\_classification (as is\_assigned\_to)**

AIM element: PATH

Reference path: applied\_document\_reference  
applied\_document\_reference.items[i] ->  
document\_reference\_item  
document\_reference\_item = product\_related\_product\_category  
product\_related\_product\_category

**6.1.4.8 document\_content\_property**

AIM element: representation

Source: ISO 10303-43

Rules: restrict\_representation\_for\_document\_content\_property,  
restrict\_representation\_for\_document\_properties

Reference path: {representation <-  
 {[representation.name = 'document content']  
 [representation.context\_of\_items ->  
 representation\_context  
 representation\_context.context\_type = 'document parameters']}  
 property\_definition\_representation.used\_representation  
 property\_definition\_representation  
 property\_definition\_representation.definition ->  
 represented\_definition  
 represented\_definition = property\_definition  
 property\_definition  
 {property\_definition.name = 'document property'}}

#### 6.1.4.8.1 detail\_level

AIM element: descriptive\_representation\_item.description

Source: ISO 10303-45

Reference path: representation  
 representation.items[i] ->  
 representation\_item =>  
 {representation\_item.name = 'detail level'}  
 descriptive\_representation\_item  
 descriptive\_representation\_item.description  
 {(descriptive\_representation\_item.description)  
 (descriptive\_representation\_item.description = 'rough 3d shape')  
 (descriptive\_representation\_item.description = 'rounded edges')}

#### 6.1.4.8.2 geometry\_type

AIM element: descriptive\_representation\_item.description

Source: ISO 10303-45

Reference path: representation  
 representation.items[i] ->  
 representation\_item =>  
 {representation\_item.name = 'geometry type'}  
 descriptive\_representation\_item  
 descriptive\_representation\_item.description  
 {(descriptive\_representation\_item.description)  
 (descriptive\_representation\_item.description = '3D wireframe model')  
 (descriptive\_representation\_item.description = '2D shape')  
 (descriptive\_representation\_item.description = 'surface model')  
 (descriptive\_representation\_item.description = 'closed volume')  
 (descriptive\_representation\_item.description = 'solid model')  
 (descriptive\_representation\_item.description = 'solid and surface model')  
 (descriptive\_representation\_item.description = 'assembly')  
 (descriptive\_representation\_item.description = 'assembly with mating elements')  
 (descriptive\_representation\_item.description = '2D drawing')  
 (descriptive\_representation\_item.description = 'drawing derived from 3D data')  
 (descriptive\_representation\_item.description = 'drawing related to 3D data')}

#### 6.1.4.8.3 document\_content\_property to language (as languages)

AIM element: PATH

Reference path: representation  
 language\_item = representation  
 language\_item <-  
 language\_assignment.items[i]  
 language\_assignment <=  
 classification\_assignment  
 {classification\_assignment.role ->  
 classification\_role  
 classification\_role.name = 'language'}  
 classification\_assignment.assigned\_class ->  
 group =>  
 language

**6.1.4.8.4 document\_content\_property to numerical\_value (as real\_world\_scale)**

#1: If the unit is not assigned globally.

#2: If the unit is assigned globally.

AIM element: #1: (measure\_representation\_item)  
 #2: (value\_representation\_item)

Source: ISO 10303-45  
 ISO 10303-45

Reference path: representation  
 representation.items[i] ->  
 representation\_item =>  
 {representation\_item.name = 'real world scale'}  
 #1: (measure\_representation\_item  
 {measure\_representation\_item <=  
 measure\_with\_unit  
 [measure\_with\_unit.value\_component ->  
 measure\_value  
 measure\_value = ratio\_measure]  
 [measure\_with\_unit.unit\_component ->  
 unit  
 unit = named\_unit  
 named\_unit =>  
 ratio\_unit}}  
 #3: (value\_representation\_item)

**6.1.4.9 document\_creation\_property**

AIM element: representation

Source: ISO 10303-43

Rules: restrict\_representation\_for\_document\_creation\_property,  
 restrict\_representation\_for\_document\_properties

Reference path: {representation <-  
 {[representation.name = 'document creation']  
 [representation.context\_of\_items ->  
 representation\_context  
 representation\_context.context\_type = 'document parameters']}  
 property\_definition\_representation.used\_representation  
 property\_definition\_representation  
 property\_definition\_representation.definition ->

```

represented_definition
represented_definition = property_definition
property_definition
{property_definition.name = 'document property'}}

```

#### 6.1.4.9.1 creating\_interface

AIM element: descriptive\_representation\_item.description

Source: ISO 10303-45

Reference path: representation  
representation.items[i] ->  
representation\_item =>  
{representation\_item.name = 'creating interface'}  
descriptive\_representation\_item  
descriptive\_representation\_item.description

#### 6.1.4.9.2 creating\_system

AIM element: descriptive\_representation\_item.description

Source: ISO 10303-45

Reference path: representation  
representation.items[i] ->  
representation\_item =>  
{representation\_item.name = 'creating system'}  
descriptive\_representation\_item  
descriptive\_representation\_item.description

#### 6.1.4.9.3 operating\_system

AIM element: descriptive\_representation\_item.description

Source: ISO 10303-45

Reference path: representation  
representation.items[i] ->  
representation\_item =>  
{representation\_item.name = 'operating system'}  
descriptive\_representation\_item  
descriptive\_representation\_item.description

#### 6.1.4.10 document\_file

AIM element: document\_file

Source: ISO 10303-214

Rules: restrict\_properties\_of\_document\_file

Reference path: {document\_file <=  
[document  
{document.representation\_types[i] ->  
document\_representation\_type  
(document\_representation\_type.name = 'digital')}

```
(document_representation_type.name = 'physical'))]  
[characterized_object]]
```

#### 6.1.4.10.1 file\_id

AIM element: document.id

Source: ISO 10303-41

Reference path: document\_file <=  
document  
{document.representation\_types[i] ->  
document\_representation\_type  
(document\_representation\_type.name = 'digital')  
(document\_representation\_type.name = 'physical')}  
document.id

#### 6.1.4.10.2 version\_id

AIM element: identification\_assignment.assigned\_id

Source: ISO 10303-41

Rules: dependent\_instantiable\_identification\_role,  
restrict\_version\_assignment\_for\_document\_file

Reference path: document\_file  
identification\_item = document\_file  
identification\_item <=  
applied\_identification\_assignment.items[i]  
applied\_identification\_assignment <=  
identification\_assignment  
{identification\_assignment.role ->  
identification\_role  
identification\_role.name = 'version'}  
identification\_assignment.assigned\_id

#### 6.1.4.10.3 document\_file to document\_content\_property (as content)

AIM element: PATH

Reference path: document\_file <=  
characterized\_object  
characterized\_definition = characterized\_object  
characterized\_definition <=  
property\_definition.definition  
property\_definition  
{property\_definition.name = 'document property'}  
represented\_definition = property\_definition  
represented\_definition <=  
property\_definition\_representation.definition  
property\_definition\_representation  
property\_definition\_representation.used\_representation ->  
representation  
{representation.name = 'document content'}

**6.1.4.10.4 document\_file to document\_creation\_property (as creation)**

AIM element: PATH

Reference path: document\_file <=  
 characterized\_object  
 characterized\_definition = characterized\_object  
 characterized\_definition <=  
 property\_definition.definition  
 property\_definition  
 {property\_definition.name = 'document property'}  
 represented\_definition = property\_definition  
 represented\_definition <=  
 property\_definition\_representation.definition  
 property\_definition\_representation  
 property\_definition\_representation.used\_representation ->  
 representation  
 {representation.name = 'document creation'}

**6.1.4.10.5 document\_file to document\_type\_property (as document\_file\_type)**

AIM element: PATH

Reference path: document\_file <=  
 document  
 document.kind ->  
 document\_type

**6.1.4.10.6 document\_file to external\_file\_id\_and\_location (as external\_id\_and\_location)**

AIM element: PATH

Reference path: document\_file  
 external\_identification\_item = document\_file  
 external\_identification\_item <=  
 applied\_external\_identification\_assignment.items[i]  
 applied\_external\_identification\_assignment

**6.1.4.10.7 document\_file to document\_format\_property (as file\_format)**

AIM element: PATH

Reference path: document\_file <=  
 characterized\_object  
 characterized\_definition = characterized\_object  
 characterized\_definition <=  
 property\_definition.definition  
 property\_definition  
 {property\_definition.name = 'document property'}  
 represented\_definition = property\_definition  
 represented\_definition <=  
 property\_definition\_representation.definition  
 property\_definition\_representation  
 property\_definition\_representation.used\_representation ->  
 representation  
 {representation.name = 'document format'}

**6.1.4.10.8 document\_file to document\_size\_property (as size)**

AIM element: PATH

Reference path: document\_file <=  
characterized\_object  
characterized\_definition = characterized\_object  
characterized\_definition <=  
property\_definition.definition  
property\_definition  
{property\_definition.name = 'document property'}  
represented\_definition = property\_definition  
represented\_definition <=  
property\_definition\_representation.definition  
property\_definition\_representation  
property\_definition\_representation.used\_representation ->  
representation  
{representation.name = 'document size'}

**6.1.4.11 document\_format\_property**

AIM element: representation

Source: ISO 10303-43

Rules: restrict\_representation\_for\_document\_format\_property,  
restrict\_representation\_for\_document\_properties

Reference path: {representation <=  
{[representation.name = 'document format']  
[representation.context\_of\_items ->  
representation\_context  
representation\_context.context (type = 'document parameters')]}  
property\_definition\_representation.used\_representation  
property\_definition\_representation  
property\_definition\_representation.definition ->  
represented\_definition  
represented\_definition = property\_definition  
property\_definition  
{property\_definition.name = 'document property'}}

**6.1.4.11.1 character\_code**

AIM element: descriptive\_representation\_item.description

Source: ISO 10303-45

Reference path: representation  
representation.items[i] ->  
representation\_item =>  
{representation\_item.name = 'character code'}  
descriptive\_representation\_item  
descriptive\_representation\_item.description  
{(descriptive\_representation\_item.description)  
(descriptive\_representation\_item.description = 'binary')  
(descriptive\_representation\_item.description = 'IEC 61286')  
(descriptive\_representation\_item.description = 'ISO 646')  
(descriptive\_representation\_item.description = 'ISO 6937')}

```
(descriptive_representation_item.description = 'ISO 8859-1')
(descriptive_representation_item.description = 'ISO 10646')}
```

#### 6.1.4.11.2 data\_format

AIM element: descriptive\_representation\_item.description

Source: ISO 10303-45

Reference path: representation  
 representation.items[i] ->  
 representation\_item =>  
 {representation\_item.name = 'data format'}  
 descriptive\_representation\_item  
 descriptive\_representation\_item.description  
 {(descriptive\_representation\_item.description)  
 (descriptive\_representation\_item.description = 'DXF')  
 (descriptive\_representation\_item.description = 'IGES')  
 (descriptive\_representation\_item.description = 'ISO 10303-203')  
 (descriptive\_representation\_item.description = 'ISO 10303-214')  
 (descriptive\_representation\_item.description = 'TIFF CCITT GR4')  
 (descriptive\_representation\_item.description = 'VDAPS')  
 (descriptive\_representation\_item.description = 'VOXEL')}

#### 6.1.4.11.3 document\_format\_property to rectangular\_size (as size\_format)

AIM element: PATH

Reference path: representation  
 representation.items[i] ->  
 representation\_item =>  
 {representation\_item.name = 'size format'}  
 geometric\_representation\_item =>  
 planar\_extent

#### 6.1.4.12 document\_location\_property

AIM element: external\_source

Source: ISO 10303-41

##### 6.1.4.12.1 location\_name

AIM element: external\_source.source\_id

Source: ISO 10303-41

#### 6.1.4.13 document\_representation

AIM element: product\_definition

Source: ISO 10303-41

Rules: restrict\_properties\_of\_document\_representation

Reference path: {product\_definition  
 product\_definition.frame\_of\_reference ->  
 product\_definition\_context <=

```
application_context_element  
(application_context_element.name = 'digital document definition')  
(application_context_element.name = 'physical document definition')
```

**6.1.4.13.1 description**

AIM element: product\_definition.description

Source: ISO 10303-41

**6.1.4.13.2 id**

AIM element: product\_definition.id

Source: ISO 10303-41

**6.1.4.13.3 document\_representation to document\_version (as associated\_document\_version)**

AIM element: PATH

```
Reference path: product_definition  
{product_definition.frame_of_reference ->  
product_definition_context <=  
application_context_element  
(application_context_element.name = 'digital document definition')  
(application_context_element.name = 'physical document definition')}}  
product_definition.formation ->  
product_definition_formation  
{product_definition_formation.of_product ->  
product <-  
product_related_product_category.products[  
product_related_product_category <=  
product_category  
product_category.name = 'document'}
```

**6.1.4.13.4 document\_representation to document\_location\_property (as common\_location)**

AIM element: PATH

```
Reference path: product_definition  
{product_definition.frame_of_reference ->  
product_definition_context <=  
application_context_element  
(application_context_element.name = 'digital document definition')  
(application_context_element.name = 'physical document definition')}}  
external_identification_item = product_definition  
external_identification_item <-  
applied_external_identification_assignment.items[  
applied_external_identification_assignment <=  
external_identification_assignment  
{external_identification_assignment <=  
identification_assignment  
{identification_assignment  
identification_assignment.assigned_id = ""}  
identification_assignment.role ->  
identification_role  
identification_role.name = 'common location'}
```

```
external_identification_assignment.source ->
external_source
```

#### 6.1.4.13.5 document\_representation to document\_content\_property (as content)

AIM element: PATH

```
Reference path: product_definition
{product_definition.frame_of_reference ->
product_definition_context <=
application_context_element
(application_context_element.name = 'digital document definition')
(application_context_element.name = 'physical document definition')}
characterized_definition = product_definition
characterized_definition <-
property_definition.definition
property_definition
{property_definition.name = 'document property'}
represented_definition = property_definition
represented_definition <-
property_definition_representation.definition
property_definition_representation
property_definition_representation.used_representation ->
representation
{representation.name = 'document content'}
```

#### 6.1.4.13.6 document\_representation to document\_creation\_property (as creation)

AIM element: PATH

```
Reference path: product_definition
{product_definition.frame_of_reference ->
product_definition_context <=
application_context_element
(application_context_element.name = 'digital document definition')
(application_context_element.name = 'physical document definition')}
characterized_definition = product_definition
characterized_definition <-
property_definition.definition
property_definition
{property_definition.name = 'document property'}
represented_definition = property_definition
represented_definition <-
property_definition_representation.definition
property_definition_representation
property_definition_representation.used_representation ->
representation
{representation.name = 'document creation'}
```

#### 6.1.4.13.7 document\_representation to multi\_language\_string (as description)

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([product\_definition.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product\_definition

Reference path: product\_definition  
#1: (attribute\_language\_item = product\_definition  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#2: (multi\_language\_attribute\_item = product\_definition  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

**6.1.4.13.8 document\_representation to document\_format\_property (as representation\_format)**

AIM element: PATH

Reference path: product\_definition  
{product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
(application\_context\_element.name = 'digital document definition')  
(application\_context\_element.name = 'physical document definition')}  
characterized\_definition = product\_definition  
characterized\_definition <-  
property\_definition.definition  
property\_definition  
{property\_definition.name = 'document property'}  
represented\_definition = property\_definition  
represented\_definition <-  
property\_definition\_representation.definition  
property\_definition\_representation  
property\_definition\_representation.used\_representation ->  
representation  
{representation.name = 'document format'}

**6.1.4.13.9 document\_representation to document\_size\_property (as size)**

AIM element: PATH

Reference path: product\_definition  
{product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
(application\_context\_element.name = 'digital document definition')  
(application\_context\_element.name = 'physical document definition')}  
characterized\_definition = product\_definition  
characterized\_definition <-  
property\_definition.definition  
property\_definition  
{property\_definition.name = 'document property'}  
represented\_definition = property\_definition

```

represented_definition <-
property_definition_representation.definition
property_definition_representation
property_definition_representation.used_representation ->
representation
{representation.name = 'document size'}

```

#### 6.1.4.14 document\_size\_property

AIM element: representation

Source: ISO 10303-43

Rules: restrict\_representation\_for\_document\_properties,  
restrict\_representation\_for\_document\_size\_property

Reference path: {representation <-  
[[representation.name = 'document size']  
[representation.context\_of\_items ->  
representation\_context  
representation\_context.context\_type = 'document parameters']}  
property\_definition\_representation.used\_representation  
property\_definition\_representation  
property\_definition\_representation.definition ->  
represented\_definition  
represented\_definition = property\_definition  
property\_definition  
{property\_definition.name = 'document property'}}

##### 6.1.4.14.1 document\_size\_property to value\_with\_unit (as file\_size)

#1: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is not assigned globally.

#2: If the value\_with\_unit is a value\_range.

#3: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is specified globally.

AIM element: PATH

Reference path: representation  
representation.items[i] ->  
representation\_item =>  
{representation\_item.name = 'file size'}  
#1: (measure\_representation\_item)  
#2: (compound\_representation\_item =>  
value\_range)  
#3: (value\_representation\_item)

##### 6.1.4.14.2 document\_size\_property to value\_with\_unit (as page\_count)

#1: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is not assigned globally.

#2: If the value\_with\_unit is a value\_range.

#3: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is specified globally.

AIM element: PATH

Reference path: representation  
 representation.items[i] ->  
 representation\_item =>  
 {representation\_item.name = 'page count'}  
 #1: (measure\_representation\_item <=  
 measure\_with\_unit  
 {measure\_with\_unit.value\_component ->  
 measure\_value  
 measure\_value = count\_measure})  
 #2: (compound\_representation\_item  
 {compound\_representation\_item =>  
 value\_range}  
 compound\_representation\_item.item\_element ->  
 compound\_item\_definition  
 compound\_item\_definition = set\_representation\_item  
 set\_representation\_item  
 set\_representation\_item[i] ->  
 representation\_item =>  
 measure\_representation\_item <=  
 measure\_with\_unit  
 {measure\_with\_unit.value\_component ->  
 measure\_value  
 measure\_value = count\_measure})  
 #3: (value\_representation\_item)

**6.1.4.15 document\_type\_property**

AIM element: document\_type

Source: ISO 10303-41

**6.1.4.15.1 document\_type\_name**

AIM element: document\_type.product\_data\_type

Source: ISO 10303-41

Reference path: {document\_type.product\_data\_type  
 (document\_type.product\_data\_type)  
 (document\_type.product\_data\_type = 'geometry')  
 (document\_type.product\_data\_type = 'NC data')  
 (document\_type.product\_data\_type = 'FE data')  
 (document\_type.product\_data\_type = 'sample data')  
 (document\_type.product\_data\_type = 'process plan')  
 (document\_type.product\_data\_type = 'check plan')  
 (document\_type.product\_data\_type = 'drawing')}

**6.1.4.15.2 document\_type\_property to classification\_system (as used\_classification\_system)**

AIM element: PATH

Rules: dependent\_instantiable\_classification\_role

Reference path: document\_type  
 classification\_item = document\_type  
 classification\_item <-  
 applied\_classification\_assignment.items[i]  
 applied\_classification\_assignment <=  
 classification\_assignment

```

{classification_assignment.role ->
classification_role
classification_role.name = 'class system membership'}
classification_assignment.assigned_class ->
group =>
class_system

```

#### 6.1.4.16 document\_version

AIM element: product\_definition\_formation

Source: ISO 10303-41

Reference path: {product\_definition\_formation  
product\_definition\_formation.of\_product ->  
product <-  
product\_related\_product\_category.products[i]  
product\_related\_product\_category <=  
product\_category  
product\_category.name = 'document'}

##### 6.1.4.16.1 description

AIM element: product\_definition\_formation.description

Source: ISO 10303-41

##### 6.1.4.16.2 id

AIM element: product\_definition\_formation.id

Source: ISO 10303-41

##### 6.1.4.16.3 document\_version to document (as associated\_document)

AIM element: PATH

Reference path: product\_definition\_formation  
product\_definition\_formation.of\_product ->  
product  
{product <-  
product\_related\_product\_category.products[i]  
product\_related\_product\_category <=  
product\_category  
product\_category.name = 'document'}

##### 6.1.4.16.4 document\_version to multi\_language\_string (as description)

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([product\_definition\_formation.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product\_definition\_formation

Reference path: product\_definition\_formation  
#1: (attribute\_language\_item = product\_definition\_formation  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#2: (multi\_language\_attribute\_item = product\_definition\_formation  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

#### 6.1.4.17 document\_version\_relationship

AIM element: product\_definition\_formation\_relationship

Source: ISO 10303-41

##### 6.1.4.17.1 description

AIM element: product\_definition\_formation\_relationship.description

Source: ISO 10303-41

##### 6.1.4.17.2 relation\_type

AIM element: product\_definition\_formation\_relationship.name

Source: ISO 10303-41

Reference path: {product\_definition\_formation\_relationship  
(product\_definition\_formation\_relationship.name)  
(product\_definition\_formation\_relationship.name = 'derivation')  
(product\_definition\_formation\_relationship.name = 'hierarchy')  
(product\_definition\_formation\_relationship.name = 'sequence')  
(product\_definition\_formation\_relationship.name = 'supplied document')}

##### 6.1.4.17.3 document\_version\_relationship to multi\_language\_string (as description)

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([product\_definition\_formation\_relationship.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product\_definition\_formation\_relationship

Reference path: product\_definition\_formation\_relationship  
 #1: (attribute\_language\_item = product\_definition\_formation\_relationship  
 attribute\_language\_item <-  
 attribute\_language\_assignment.items[i]  
 attribute\_language\_assignment  
 {attribute\_language\_assignment <=  
 attribute\_classification\_assignment  
 attribute\_classification\_assignment.attribute\_name = 'description'})  
 #2: (multi\_language\_attribute\_item = product\_definition\_formation\_relationship  
 multi\_language\_attribute\_item <-  
 multi\_language\_attribute\_assignment.items[i]  
 multi\_language\_attribute\_assignment  
 {multi\_language\_attribute\_assignment <=  
 attribute\_value\_assignment  
 attribute\_value\_assignment.attribute\_name = 'description'})

#### 6.1.4.17.4 document\_version\_relationship to document\_version (as related)

AIM element: PATH

Reference path: product\_definition\_formation\_relationship  
 product\_definition\_formation\_relationship.related\_product\_definition\_formation ->  
 product\_definition\_formation  
 {product\_definition\_formation.of\_product ->  
 product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=  
 product\_category  
 product\_category.name = 'document'}

#### 6.1.4.17.5 document\_version\_relationship to document\_version (as relating)

AIM element: PATH

Reference path: product\_definition\_formation\_relationship  
 product\_definition\_formation\_relationship.relying\_product\_definition\_formation ->  
 product\_definition\_formation  
 {product\_definition\_formation.of\_product ->  
 product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=  
 product\_category  
 product\_category.name = 'document'}

#### 6.1.4.18 external\_file\_id\_and\_location

AIM element: applied\_external\_identification\_assignment

Source: ISO 10303-214

Reference path: applied\_external\_identification\_assignment <=  
 external\_identification\_assignment  
 {external\_identification\_assignment <=  
 identification\_assignment  
 identification\_assignment.role ->  
 identification\_role  
 identification\_role.name = 'external document id and location'}

**6.1.4.18.1 external\_id**

AIM element: identification\_assignment.assigned\_id

Source: ISO 10303-41

Reference path: applied\_external\_identification\_assignment <=  
external\_identification\_assignment <=  
identification\_assignment  
identification\_assignment.assigned\_id

**6.1.4.18.2 external\_file\_id\_and\_location to document\_location\_property (as location)**

AIM element: PATH

Reference path: applied\_external\_identification\_assignment <=  
external\_identification\_assignment  
external\_identification\_assignment.source ->  
external\_source

**6.1.4.19 external\_geometric\_model**

AIM element: shape\_representation

Source: ISO 10303-41

Reference path: {shape\_representation <=  
representation  
representation.context\_of\_items ->  
representation\_context =>  
{representation\_context.context\_type = 'external'}  
geometric\_representation\_context}

**6.1.4.19.1 model\_extent**

#1: If the unit is assigned globally.

#2: If the unit is not assigned globally.

AIM element: #1: (value\_representation\_item)  
#2: (measure\_representation\_item)

Source: ISO 10303-43  
ISO 10303-43

Rules: dependent\_instantiable\_measure\_with\_unit

Reference path: shape\_representation <=  
representation <-  
representation\_relationship.rep\_1  
representation\_relationship  
{representation\_relationship.name = 'model extent association'}  
representation\_relationship.rep\_2 ->  
representation  
{representation.name = 'model extent representation'}  
representation.items[i] ->  
representation\_item =>  
{representation\_item.name = 'model extent value'}

#1: (value\_representation\_item)  
 #2: (measure\_representation\_item)

#### 6.1.4.20 external\_model

AIM element: representation

Source: ISO 10303-43

Reference path: {representation  
 [representation.context\_of\_items ->  
 representation\_context  
 representation\_context.context\_type = 'external']  
 [representation.items[1] ->  
 (axis2\_placement\_2d)  
 (axis2\_placement\_3d)]}

##### 6.1.4.20.1 description

AIM element: representation.description

Source: ISO 10303-43

##### 6.1.4.20.2 model\_id

AIM element: representation.name

Source: ISO 10303-43

##### 6.1.4.20.3 external\_model to multi\_language\_string (as description)

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([representation.description]  
 [PATH])  
 #2: (PATH)

Source: ISO 10303-43

Rules: restrict\_multi\_language\_for\_representation

Reference path: representation  
 #1: (attribute\_language\_item = representation  
 attribute\_language\_item <-  
 attribute\_language\_assignment.items[i]  
 attribute\_language\_assignment  
 {attribute\_language\_assignment <=  
 attribute\_classification\_assignment  
 attribute\_classification\_assignment.attribute\_name = 'description'})  
 #2: (multi\_language\_attribute\_item = representation  
 multi\_language\_attribute\_item <-  
 multi\_language\_attribute\_assignment.items[i]  
 multi\_language\_attribute\_assignment  
 {multi\_language\_attribute\_assignment <=  
 attribute\_value\_assignment  
 attribute\_value\_assignment.attribute\_name = 'description'})

**6.1.4.20.4 external\_model to digital\_file (as is\_defined\_as)**

AIM element: PATH

Reference path: representation <-  
{representation.context\_of\_items ->  
representation\_context  
representation\_context.context\_type = 'external'}  
property\_definition\_representation.used\_representation  
property\_definition\_representation  
property\_definition\_representation.definition ->  
represented\_definition  
represented\_definition = property\_definition  
property\_definition  
{property\_definition.name = 'external definition'}  
property\_definition.definition ->  
characterized\_definition  
characterized\_definition = characterized\_object  
characterized\_object =>  
document\_file  
{document\_file <=  
document  
document.representation\_types[i] ->  
document\_representation\_type  
document\_representation\_type.name = 'digital'}

**6.1.4.20.5 external\_model to cartesian\_coordinate\_space (as is\_defined\_in)**

AIM element: PATH

Reference path: representation  
representation.context\_of\_items ->  
representation\_context =>  
{representation\_context.context\_type = 'external'}  
geometric\_representation\_context

**6.1.4.21 external\_picture**

AIM element: representation

Source: ISO 10303-43

Reference path: {representation  
[representation.context\_of\_items ->  
representation\_context =>  
{representation\_context.context\_type = 'external'}  
geometric\_representation\_context  
geometric\_representation\_context.coordinate\_space\_dimension = 2]  
[representation.items[1] ->  
axis2\_placement\_2d]}

**6.1.4.21.1 external\_picture to cartesian\_coordinate\_space\_2d (as is\_defined\_in)**

AIM element: PATH

Reference path: representation  
representation.context\_of\_items ->  
representation\_context =>  
{representation\_context.context\_type = 'external'}

```
geometric_representation_context
{geometric_representation_context.coordinate_space_dimension = 2}
```

#### 6.1.4.22 geometric\_model\_relationship\_with\_transformation

#1: If referred to by an assembly\_association.placement.

#2: If not referred to by an assembly\_association.placement.

AIM element: #1: ([shape\_representation\_relationship]  
[representation\_relationship\_with\_transformation])  
#2: (representation\_relationship\_with\_transformation)

Source: ISO 10303-41  
ISO 10303-43  
ISO 10303-43

##### 6.1.4.22.1 geometric\_model\_relationship\_with\_transformation to transformation (as model\_placement)

#1: If the transformation is implicitly defined by two items.

#2: For a two-dimensional transformation.

#3: For a three-dimensional transformation.

#4: If the item is a placement.

#5: If the item is a transformation operator.

#6: If the transformation is explicitly defined.

AIM element: PATH

Reference path: representation\_relationship\_with\_transformation  
representation\_relationship\_with\_transformation.transformation\_operator ->  
transformation  
#1: (transformation = item\_defined\_transformation  
item\_defined\_transformation  
{#2: ([item\_defined\_transformation.transform\_item\_1 ->  
representation\_item =>  
geometric\_representation\_item =>  
placement =>  
axis2\_placement\_2d]  
[item\_defined\_transformation.transform\_item\_2 ->  
representation\_item =>  
geometric\_representation\_item =>  
#4: (placement =>  
axis2\_placement\_2d)  
#5: (cartesian\_transformation\_operator =>  
{cartesian\_transformation\_operator.scl = 1.0}  
cartesian\_transformation\_operator\_2d}))  
#3: ([item\_defined\_transformation.transform\_item\_1 ->  
representation\_item =>  
geometric\_representation\_item =>  
placement =>  
axis2\_placement\_3d]  
[item\_defined\_transformation.transform\_item\_2 ->  
representation\_item =>

```

geometric_representation_item =>
#4: (placement =>
axis2_placement_3d)
#5: (cartesian_transformation_operator =>
{cartesian_transformation_operator.scl = 1.0}
cartesian_transformation_operator_3d))}
#6: (transformation = functionally_defined_transformation
functionally_defined_transformation
{functionally_defined_transformation =>
cartesian_transformation_operator =>
{cartesian_transformation_operator.scl = 1.0}
#2: (cartesian_transformation_operator_2d)
#3: (cartesian_transformation_operator_3d))}
    
```

#### 6.1.4.23 named\_size

#1: If no density is specified for the rectangular\_size.

#2: If a density is specified for the rectangular\_size.

AIM element: #1: ([planar\_extent]  
[descriptive\_representation\_item])  
#2: ([planar\_extent]  
[descriptive\_representation\_item]  
[measure\_representation\_item])

Source: ISO 10303-46  
ISO 10303-46  
ISO 10303-46  
ISO 10303-46  
ISO 10303-46

Reference path: {#1: ([planar\_extent <=  
geometric\_representation\_item <=]  
[descriptive\_representation\_item <=])  
#2: ([planar\_extent <=  
geometric\_representation\_item <=  
[descriptive\_representation\_item <=]  
[measure\_representation\_item <=])  
representation\_item  
representation\_item.name = 'size format'}

##### 6.1.4.23.1 size

#1: If no density is specified for the rectangular\_size.

#2: If a density is specified for the rectangular\_size.

AIM element: descriptive\_representation\_item.description

Source: ISO 10303-45

Reference path: {#1: ([planar\_extent <=  
geometric\_representation\_item <=]  
[descriptive\_representation\_item <=])  
#2: ([planar\_extent <=  
geometric\_representation\_item <=  
[descriptive\_representation\_item <=]  
[measure\_representation\_item <=])

```

representation_item
representation_item.name = 'size format'}

```

#### 6.1.4.23.2 named\_size to classification\_system (as referenced\_standard)

#1: If no density is specified for the rectangular\_size.

#2: If a density is specified for the rectangular\_size.

AIM element: PATH

Rules: restrict\_class\_system\_assignment\_for\_planar\_extent

Reference path: planar\_extent  
 {#1: ([planar\_extent <=  
 geometric\_representation\_item <=  
 [descriptive\_representation\_item <=])  
 #2: ([planar\_extent <=  
 geometric\_representation\_item <=  
 [descriptive\_representation\_item <=  
 [measure\_representation\_item <=])  
 representation\_item  
 representation\_item.name = 'size format'}  
 classification\_item = planar\_extent  
 classification\_item <-  
 applied\_classification\_assignment.items[i]  
 applied\_classification\_assignment <=  
 classification\_assignment  
 {classification\_assignment.role ->  
 classification\_role  
 classification\_role.name = 'class system membership'}  
 classification\_assignment.assigned\_class ->  
 group =>  
 class\_system

#### 6.1.4.24 physical\_document

AIM element: product\_definition

Source: ISO 10303-41

Reference path: {product\_definition  
 product\_definition.frame\_of\_reference ->  
 product\_definition\_context <=  
 application\_context\_element  
 application\_context\_element.name = 'physical document definition'}

#### 6.1.4.24.1 physical\_document to physical\_file (as file)

AIM element: PATH

Reference path: product\_definition =>  
 {product\_definition.frame\_of\_reference ->  
 product\_definition\_context <=  
 application\_context\_element  
 application\_context\_element.name = 'physical document definition'}  
 product\_definition\_with\_associated\_documents  
 product\_definition\_with\_associated\_documents.documentation\_ids[i] ->  
 document =>

```
{document.representation_types[i] ->
document_representation_type
document_representation_type.name = 'physical'}
document_file
```

#### 6.1.4.25 physical\_file

AIM element: document\_file

Source: ISO 10303-214

Reference path: document\_file <=  
[document  
{document.representation\_types[i] ->  
document\_representation\_type  
document\_representation\_type.name = 'physical'}]  
[characterized\_object]

#### 6.1.4.26 rectangular\_size

#1: If no density is specified for the rectangular\_size.

#2: If a density is specified for the rectangular\_size.

#3: If the rectangular\_size is a named\_size.

#4: If the rectangular\_size is not a named\_size.

AIM element: #1: (#3: ([planar\_extent]  
[descriptive\_representation\_item])  
#4: (planar\_extent))  
#2: (#3: ([planar\_extent]  
[measure\_representation\_item]  
[descriptive\_representation\_item])  
#4: ([planar\_extent]  
[measure\_representation\_item]))

Source: ISO 10303-46  
ISO 10303-46

Reference path: {#1: (#3: ([planar\_extent <=  
geometric\_representation\_item <=]  
[descriptive\_representation\_item <=])  
#4: (planar\_extent <=  
geometric\_representation\_item <=))  
#2: (#3: ([planar\_extent <=  
geometric\_representation\_item <=]  
[descriptive\_representation\_item <=]  
[measure\_representation\_item <=])  
#4: ([planar\_extent <=  
geometric\_representation\_item <=]  
[measure\_representation\_item <=]))

```

representation_item
representation_item.name = 'size format'}

```

#### 6.1.4.26.1 rectangular\_size to value\_with\_unit (as density)

#1: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is not assigned globally.

#2: If the value\_with\_unit is a value\_range.

#3: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is specified globally.

AIM element: #1: (measure\_representation\_item)  
 #2: (value\_range)  
 #3: (value\_representation\_item)

Source: ISO 10303-45  
 ISO 10303-45  
 ISO 10303-45

Reference path: #1: ({measure\_representation\_item <=  
 representation\_item  
 representation\_item.name = 'size format'})  
 #2: ({value\_range <=  
 compound\_representation\_item <=  
 representation\_item  
 representation\_item.name = 'size format'})  
 #3: ({value\_representation\_item <=  
 representation\_item  
 representation\_item.name = 'size format'})

#### 6.1.4.26.2 rectangular\_size to value\_with\_unit (as height)

#1: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is not assigned globally.

#2: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is specified globally.

#3: If the unit of the value range is not assigned globally.

#4: If the unit of the value range is assigned globally.

AIM element: PATH

Reference path: planar\_extent  
 planar\_extent.size\_in\_y ->  
 length\_measure  
 measure\_value = length\_measure  
 measure\_value <-  
 (measure\_with\_unit.value\_component  
 measure\_with\_unit  
 #1: (measure\_with\_unit =>  
 measure\_representation\_item)  
 #3: (measure\_with\_unit =>  
 measure\_representation\_item <=  
 representation\_item <-  
 set\_representation\_item[i]  
 set\_representation\_item  
 compound\_item\_definition = set\_representation\_item  
 compound\_item\_definition <-  
 compound\_representation\_item.item\_element

```

compound_representation_item =>
value_range))
#4: (value_representation_item.value_component
value_representation_item <=
representation_item <-
set_representation_item[i]
set_representation_item
compound_item_definition = set_representation_item
compound_item_definition <-
compound_representation_item.item_element
compound_representation_item =>
value_range)
#2: (value_representation_item.value_component
value_representation_item)

```

**6.1.4.26.3 rectangular\_size to value\_with\_unit (as width)**

- #1: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is not assigned globally.
- #2: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is specified globally.
- #3: If the unit of the value range is not assigned globally.
- #4: If the unit of the value range is assigned globally.

AIM element:     PATH

```

Reference path: planar_extent
planar_extent.size_in_x ->
length_measure
measure_value = length_measure
measure_value <-
(measure_with_unit.value_component
measure_with_unit
#1: (measure_with_unit =>
measure_representation_item)
#3: (measure_with_unit =>
measure_representation_item <=
representation_item <-
set_representation_item[i]
set_representation_item
compound_item_definition = set_representation_item
compound_item_definition <-
compound_representation_item.item_element
compound_representation_item =>
value_range))
#4: (value_representation_item.value_component
value_representation_item <=
representation_item <-
set_representation_item[i]
set_representation_item
compound_item_definition = set_representation_item
compound_item_definition <-
compound_representation_item.item_element
compound_representation_item =>
value_range)
#2: (value_representation_item.value_component
value_representation_item)

```



**6.1.4.27 transformation**

#1: If the transformation is used with a template instance.

#2: If the transformation is implicitly defined by two items.

#3: If the transformation is explicitly defined.

AIM element: #1: ([mapped\_item.mapping\_target]  
[representation\_map.mapping\_origin])  
#2: (item\_defined\_transformation)  
#3: (functionally\_defined\_transformation)

Source: ISO 10303-43  
ISO 10303-43  
ISO 10303-43  
ISO 10303-43

**6.1.4.28 transformation\_2d**

#2: If the transformation is implicitly defined by two items.

#3: If the transformation is explicitly defined.

#4: If the item is a placement.

#5: If the item is a transformation operator.

AIM element: #2: (item\_defined\_transformation)  
#3: (functionally\_defined\_transformation)

Source: ISO 10303-43  
ISO 10303-43

Reference path: {#2: (item\_defined\_transformation  
[item\_defined\_transformation.transform\_item\_1 ->  
representation\_item =>  
geometric\_representation\_item =>  
placement =>  
axis2\_placement\_2d]  
[item\_defined\_transformation.transform\_item\_2 ->  
representation\_item =>  
geometric\_representation\_item =>  
#4: (placement =>  
axis2\_placement\_2d)  
#5: (cartesian\_transformation\_operator =>  
cartesian\_transformation\_operator\_2d)])  
#3: (functionally\_defined\_transformation =>  
cartesian\_transformation\_operator =>  
cartesian\_transformation\_operator\_2d)}

**6.1.4.29 transformation\_3d**

#2: If the transformation is implicitly defined by two items.

#3: If the transformation is explicitly defined.

#4: If the item is a placement.

#5: If the item is a transformation operator.

AIM element: #2: (item\_defined\_transformation)  
#3: (functionally\_defined\_transformation)

Source: ISO 10303-43  
ISO 10303-43

Reference path: {#2: (item\_defined\_transformation  
[item\_defined\_transformation.transform\_item\_1 ->  
representation\_item =>  
geometric\_representation\_item =>  
placement =>  
axis2\_placement\_3d]  
[item\_defined\_transformation.transform\_item\_2 ->  
representation\_item =>  
geometric\_representation\_item =>  
#4: (placement =>  
axis2\_placement\_3d)  
#5: (cartesian\_transformation\_operator =>  
cartesian\_transformation\_operator\_3d))  
#3: (functionally\_defined\_transformation =>  
cartesian\_transformation\_operator =>  
cartesian\_transformation\_operator\_3d)}

### 6.1.5 individual\_definition\_structure UoF

#### 6.1.5.1 physical\_item\_definition

AIM element: product\_definition

Source: ISO 10303-41

Rules: restrict\_product\_definition\_context\_for\_product

Reference path: product\_definition  
{product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'physical occurrence'}

##### 6.1.5.1.1 id

AIM element: product\_definition.id

Source: ISO 10303-41

##### 6.1.5.1.2 name

AIM element: product\_definition.description

Source: ISO 10303-41

##### 6.1.5.1.3 physical\_item\_definition to application\_context (as additional\_context)

AIM element: PATH

Reference path: product\_definition <-  
 product\_definition\_context\_association.definition  
 product\_definition\_context\_association  
 {product\_definition\_context\_association.role  
 product\_definition\_context\_role.name = 'additional context'}  
 product\_definition\_context\_association.frame\_of\_reference ->  
 product\_definition\_context

#### 6.1.5.1.4 physical\_item\_definition to physical\_item\_version (as associated\_item\_version)

AIM element: PATH

Reference path: product\_definition  
 product\_definition.formation ->  
 product\_definition\_formation  
 {product\_definition\_formation.of\_product ->  
 product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=  
 product\_category  
 product\_category.name = 'physically realized product'}

#### 6.1.5.1.5 physical\_item\_definition to application\_context (as initial\_context)

AIM element: PATH

Reference path: product\_definition  
 product\_definition.frame\_of\_reference ->  
 product\_definition\_context

#### 6.1.5.1.6 physical\_item\_definition to multi\_language\_string (as name)

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([product\_definition.description]  
 [PATH])  
 #2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product\_definition

Reference path: product\_definition  
 #1: (attribute\_language\_item = product\_definition  
 attribute\_language\_item <-  
 attribute\_language\_assignment.items[i]  
 attribute\_language\_assignment  
 {attribute\_language\_assignment <=  
 attribute\_classification\_assignment  
 attribute\_classification\_assignment.attribute\_name = 'description'})  
 #2: (multi\_language\_attribute\_item = product\_definition  
 multi\_language\_attribute\_item <-  
 multi\_language\_attribute\_assignment.items[i]  
 multi\_language\_attribute\_assignment  
 {multi\_language\_attribute\_assignment <=  
 attribute\_value\_assignment  
 attribute\_value\_assignment.attribute\_name = 'description'})

### 6.1.5.2 physical\_item\_structure\_association

AIM element: assembly\_component\_usage

Source: ISO 10303-44

Reference path: {assembly\_component\_usage <=  
product\_definition\_usage <=  
product\_definition\_relationship  
product\_definition\_relationship.name = 'physical occurrence usage'}

#### 6.1.5.2.1 physical\_item\_structure\_association to physical\_item\_definition (as related)

AIM element: PATH

Rules: restrict\_product\_definitions\_for\_product\_definition\_relationship

Reference path: assembly\_component\_usage <=  
product\_definition\_usage <=  
product\_definition\_relationship  
{product\_definition\_relationship.name = 'physical occurrence usage'}  
product\_definition\_relationship.related\_product\_definition ->  
product\_definition  
{product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'physical occurrence'}

#### 6.1.5.2.2 physical\_item\_structure\_association to physical\_item\_definition (as relating)

AIM element: PATH

Rules: restrict\_product\_definitions\_for\_product\_definition\_relationship

Reference path: assembly\_component\_usage <=  
product\_definition\_usage <=  
product\_definition\_relationship  
{product\_definition\_relationship.name = 'physical occurrence usage'}  
product\_definition\_relationship.relying\_product\_definition ->  
product\_definition  
{product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'physical occurrence'}

### 6.1.6 individual\_management\_data UoF

#### 6.1.6.1 item\_version\_relationship

AIM element: product\_definition\_formation\_relationship

Source: ISO 10303-41

##### 6.1.6.1.1 description

AIM element: product\_definition\_formation\_relationship.description

Source: ISO 10303-41

**6.1.6.1.2 relation\_type**

AIM element: product\_definition\_formation\_relationship.name

Source: ISO 10303-41

Reference path: product\_definition\_formation\_relationship  
 {product\_definition\_formation\_relationship.name  
 product\_definition\_formation\_relationship.name = 'sequence'}

**6.1.6.1.3 item\_version\_relationship to multi\_language\_string (as description)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string. [D:\Work\Documents\Standards\STEP\10303-214\html\\_with\\_change\\_marks\htmls\clause5\aim\\_index\pxref.htm](D:\Work\Documents\Standards\STEP\10303-214\html_with_change_marks\htmls\clause5\aim_index\pxref.htm)

AIM element: #1: ([product\_definition\_formation\_relationship.description]  
 [PATH])  
 #2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product\_definition\_formation\_relationship

Reference path: product\_definition\_formation\_relationship  
 #1: (attribute\_language\_item = product\_definition\_formation\_relationship  
 attribute\_language\_item <-  
 attribute\_language\_assignment.items[i]  
 attribute\_language\_assignment  
 {attribute\_language\_assignment <=  
 attribute\_classification\_assignment  
 attribute\_classification\_assignment.attribute\_name = 'description'})  
 #2: (multi\_language\_attribute\_item = product\_definition\_formation\_relationship  
 multi\_language\_attribute\_item <-  
 multi\_language\_attribute\_assignment.items[i]  
 multi\_language\_attribute\_assignment  
 {multi\_language\_attribute\_assignment <=  
 attribute\_value\_assignment  
 attribute\_value\_assignment.attribute\_name = 'description'})

**6.1.6.1.4 item\_version\_relationship to physical\_item\_version (as related)**

AIM element: PATH

Reference path: product\_definition\_formation\_relationship  
 product\_definition\_formation\_relationship.related\_product\_definition\_formation ->  
 product\_definition\_formation

**6.1.6.1.5 item\_version\_relationship to physical\_item\_version (as relating)**

AIM element: PATH

Reference path: product\_definition\_formation\_relationship  
 product\_definition\_formation\_relationship.relatng\_product\_definition\_formation ->  
 product\_definition\_formation

**6.1.6.2 location**

AIM element: location

Source: ISO 10303-41

**6.1.6.2.1 location\_id**

AIM element: location.id

Source: ISO 10303-41

**6.1.6.2.2 location\_name**

AIM element: location.name

Source: ISO 10303-41

**6.1.6.2.3 location\_type**

AIM element: location.description

Source: ISO 10303-41

**6.1.6.3 physical\_item**

AIM element: product

Source: ISO 10303-41

Rules: product\_requires\_category,  
product\_requires\_id\_owner,  
product\_requires\_version,  
restrict\_product\_category\_for\_product

Reference path: {product <-  
product\_related\_product\_category.products[i]  
product\_related\_product\_category <=  
product\_category  
product\_category.name  
product\_category.name = 'physically realized product'}

**6.1.6.3.1 description**

AIM element: product.description

Source: ISO 10303-41

**6.1.6.3.2 id**

AIM element: product.id

Source: ISO 10303-41

**6.1.6.3.3 name**

AIM element: product.name

Source: ISO 10303-41

#### 6.1.6.3.4 physical\_item to multi\_language\_string (as description)

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([product.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product

Reference path: product  
#1: (attribute\_language\_item = product  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#2: (multi\_language\_attribute\_item = product  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

#### 6.1.6.3.5 physical\_item to multi\_language\_string (as name)

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([product.name]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product

Reference path: product  
#1: (attribute\_language\_item = product  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'name'})  
#2: (multi\_language\_attribute\_item = product  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=

```
attribute_value_assignment  
attribute_value_assignment.attribute_name = 'name'))
```

#### 6.1.6.4 physical\_item\_definition

AIM element: product\_definition

Source: ISO 10303-41

Rules: restrict\_product\_definition\_context\_for\_product

Reference path: product\_definition  
{product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'physical occurrence'}

##### 6.1.6.4.1 id

AIM element: product\_definition.id

Source: ISO 10303-41

##### 6.1.6.4.2 name

AIM element: product\_definition.description

Source: ISO 10303-41

##### 6.1.6.4.3 physical\_item\_definition to application\_context (as additional\_context)

AIM element: PATH

Reference path: product\_definition <-  
product\_definition\_context\_association.definition  
product\_definition\_context\_association  
{product\_definition\_context\_association.role  
product\_definition\_context\_role.name = 'additional context'}  
product\_definition\_context\_association.frame\_of\_reference ->  
product\_definition\_context

##### 6.1.6.4.4 physical\_item\_definition to physical\_item\_version (as associated\_item\_version)

AIM element: PATH

Reference path: product\_definition  
product\_definition.formation ->  
product\_definition\_formation  
{product\_definition\_formation.of\_product ->  
product <-  
product\_related\_product\_category.products[i]  
product\_related\_product\_category <=  
product\_category  
product\_category.name = 'physically realized product'}

**6.1.6.4.5 physical\_item\_definition to application\_context (as initial\_context)**

AIM element: PATH

Reference path: product\_definition  
product\_definition.frame\_of\_reference ->  
product\_definition\_context

**6.1.6.4.6 physical\_item\_definition to multi\_language\_string (as name)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([product\_definition.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product\_definition

Reference path: product\_definition  
#1: (attribute\_language\_item = product\_definition  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#2: (multi\_language\_attribute\_item = product\_definition  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

**6.1.6.5 physical\_item\_location\_association**

AIM element: applied\_location\_assignment

Source: ISO 13399-1

Reference path: applied\_location\_assignment <=  
location\_assignment

**6.1.6.5.1 role**

AIM element: location\_assignment.role

Source: ISO 10303-41

Reference path: applied\_location\_assignment <=  
location\_assignment  
location\_assignment.role

**6.1.6.5.2 physical\_item\_location\_assignment to physical\_item\_definition (as located\_item)**

AIM element: applied\_location\_assignment.located\_product

Source: ISO 13399-1

Rules: restrict\_product\_defintion\_context\_for\_applied\_location\_assignment

Reference path: applied\_location\_assignment  
applied\_location\_assignment.located\_product ->  
{product\_definition  
product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'physical occurrence'}

**6.1.6.5.3 physical\_item\_location\_assignment to location (as location)**

AIM element: PATH

Reference path: applied\_location\_assignment <=  
location\_assignment  
location\_assignment.location ->  
location

**6.1.6.6 physical\_item\_state\_association**

AIM element: applied\_state\_type\_assignment

Source: ISO 10303-56

**6.1.6.6.1 role**

AIM element: state\_type\_role.name

Source: ISO 10303-56

Reference path: applied\_state\_type\_assignment  
applied\_state\_type\_assignment.role ->  
state\_type\_role  
state\_type\_role.name

**6.1.6.6.2 physical\_item\_state\_association to physical\_item\_definition (as associated\_physical\_item)**

AIM element: PATH

Rules: restrict\_product\_defintion\_context\_for\_applied\_state\_type\_assignment

Reference path: applied\_state\_type\_assignment  
applied\_state\_type\_assignment.item\_set[i] ->  
state\_type\_of\_item \*>  
applied\_state\_type\_of\_item  
applied\_state\_type\_of\_item = product\_definition  
{product\_definition  
product\_definition.frame\_of\_reference ->  
product\_definition\_context <=

application\_context\_element  
 application\_context\_element.name = 'physical occurrence'}

#### 6.1.6.6.3 physical\_item\_state\_association to state (as associated\_state)

AIM element: applied\_state\_assignment.assigned\_state\_type

Source: ISO 10303-56

#### 6.1.6.7 physical\_item\_version

AIM element: product\_definition\_formation

Source: ISO 10303-41

##### 6.1.6.7.1 description

AIM element: product\_definition\_formation.description

Source: ISO 10303-41

##### 6.1.6.7.2 id

AIM element: product\_definition\_formation.id

Source: ISO 10303-41

##### 6.1.6.7.3 physical\_item\_version to item (as associated\_item)

AIM element: PATH

Reference path: product\_definition\_formation  
 product\_definition\_formation.of\_product ->  
 product  
 {product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=  
 product\_category  
 product\_category.name  
 product\_category.name = 'physically realized product'}

##### 6.1.6.7.4 physical\_item\_version to multi\_language\_string (as description)

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([product\_definition\_formation.description]  
 [PATH])  
 #2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product\_definition\_formation

Reference path: product\_definition\_formation  
 #1: (attribute\_language\_item = product\_definition\_formation

```

attribute_language_item <-
attribute_language_assignment.items[i]
attribute_language_assignment
{attribute_language_assignment <=
attribute_classification_assignment
attribute_classification_assignment.attribute_name = 'description'})
#2: (multi_language_attribute_item = product_definition_formation
multi_language_attribute_item <-
multi_language_attribute_assignment.items[i]
multi_language_attribute_assignment
{multi_language_attribute_assignment <=
attribute_value_assignment
attribute_value_assignment.attribute_name = 'description'})

```

### 6.1.6.8 realized\_item\_association

AIM element: product\_definition\_relationship

Source: ISO 10303-41

Rules: restrict\_product\_definitions\_for\_product\_definition\_relationship

Reference path: product\_definition\_relationship  
{product\_definition\_relationship.name = 'physical realization' }

#### 6.1.6.8.1 realized\_item\_association to physical\_item (as physical\_item)

AIM element: PATH

Reference path: product\_definition\_relationship  
{product\_definition\_relationship.related\_product\_definition ->  
product\_definition  
product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'physical occurrence'}

#### 6.1.6.8.2 realized\_item\_association to item (as realized\_item)

AIM element: PATH

Reference path: product\_definition\_relationship  
{product\_definition\_relationship.relying\_product\_definition ->  
product\_definition  
product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'part definition' }

### 6.1.6.9 state

AIM element: state\_type

Source: ISO 10303-56

#### 6.1.6.9.1 name

AIM element: IDENTICAL MAPPING

## 6.1.7 item\_definition\_structure UoF

### 6.1.7.1 assembly\_association

AIM element: assembly\_component\_usage

Source: ISO 10303-44

Rules: restrict\_product\_definitions\_for\_product\_definition\_relationship

#### 6.1.7.1.1 assembly\_association to geometric\_model\_relationship\_with\_transformation (as placement)

AIM element: PATH

Reference path: assembly\_component\_usage <=  
product\_definition\_usage <=  
product\_definition\_relationship  
[characterized\_product\_definition = product\_definition\_relationship  
characterized\_definition = characterized\_product\_definition  
characterized\_definition <=  
property\_definition.definition  
property\_definition =>  
product\_definition\_shape <=  
context\_dependent\_shape\_representation.represented\_product\_relation  
context\_dependent\_shape\_representation  
context\_dependent\_shape\_representation.representation\_relation ->  
shape\_representation\_relationship <=]  
[product\_definition\_relationship.relatng\_product\_definition ->  
product\_definition  
characterized\_product\_definition = product\_definition  
characterized\_definition = characterized\_product\_definition  
characterized\_definition <=  
property\_definition.definition  
property\_definition .  
{property\_definition =>  
product\_definition\_shape}  
represented\_definition = property\_definition  
represented\_definition <=  
property\_definition\_representation.definition  
property\_definition\_representation  
{property\_definition\_representation =>  
shape\_definition\_representation}  
property\_definition\_representation.used\_representation ->  
representation <=  
{representation =>  
shape\_representation}  
representation\_relationship.rep\_2]  
[product\_definition\_relationship.related\_product\_definition ->  
product\_definition  
characterized\_product\_definition = product\_definition  
characterized\_definition = characterized\_product\_definition  
characterized\_definition <=  
property\_definition.definition  
property\_definition  
{property\_definition =>  
product\_definition\_shape}  
represented\_definition = property\_definition  
represented\_definition <=

```

property_definition_representation.definition
property_definition_representation
{property_definition_representation =>
shape_definition_representation}
property_definition_representation.used_representation ->
representation <-
{representation =>
shape_representation}
representation_relationship.rep_1]
representation_relationship =>
representation_relationship_with_transformation
    
```

**6.1.7.1.2 assembly\_association to assembly\_definition (as relating)**

AIM element: PATH

```

Reference path: assembly_component_usage <=
product_definition_usage <=
product_definition_relationship
product_definition_relationship.relating_product_definition ->
product_definition
{product_definition <-
product_definition_context_association.definition
product_definition_context_association
[product_definition_context_association.frame_of_reference ->
product_definition_context <=
application_context_element
application_context_element.name = 'assembly definition']
[product_definition_context_association.role ->
product_definition_context_role
product_definition_context_role.name = 'part definition type']}
    
```

**6.1.7.2 assembly\_definition**

AIM element: product\_definition

Source: ISO 10303-41

Rules: dependent\_instantiable\_product\_definition\_context\_role

```

Reference path: {product_definition <-
product_definition_context_association.definition
product_definition_context_association
[product_definition_context_association.frame_of_reference ->
product_definition_context <=
application_context_element
application_context_element.name = 'assembly definition']
[product_definition_context_association.role ->
product_definition_context_role
product_definition_context_role.name = 'part definition type']}
    
```

**6.1.7.2.1 assembly\_type**

AIM element: product\_definition.name

Source: ISO 10303-41

**6.1.7.3 item\_definition\_relationship**

AIM element: product\_definition\_relationship

Source: ISO 10303-41

**6.1.7.3.1 item\_definition\_relationship to multi\_language\_string (as description)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([product\_definition\_relationship.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product\_definition\_relationship

Reference path: product\_definition\_relationship  
#1: (attribute\_language\_item = product\_definition\_relationship  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#2: (multi\_language\_attribute\_item = product\_definition\_relationship  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

**6.1.7.3.2 item\_definition\_relationship to item\_definition (as related)**

AIM element: PATH

Reference path: product\_definition\_relationship  
product\_definition\_relationship.related\_product\_definition ->  
product\_definition  
{product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'part definition'}

**6.1.7.3.3 item\_definition\_relationship to item\_definition (as relating)**

AIM element: PATH

Reference path: product\_definition\_relationship  
product\_definition\_relationship.relatating\_product\_definition ->  
product\_definition  
{product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'part definition'}

**6.1.7.3.4 relation\_type**

AIM element: product\_definition\_relationship.name

Source: ISO 10303-41

**6.1.7.4 item\_instance**

AIM element: assembly\_component\_usage

Source: ISO 10303-44

Rules: restrict\_part\_occurrence,  
restrict\_product\_definition\_context\_for\_product

**6.1.7.4.1 description**

AIM element: product\_definition\_relationship.description

Source: ISO 10303-41

Reference path: assembly\_component\_usage <=  
product\_definition\_usage <=  
product\_definition\_relationship  
product\_definition\_relationship.description

**6.1.7.4.2 id**

AIM element: product\_definition\_relationship.id

Source: ISO 10303-41

Reference path: assembly\_component\_usage <=  
product\_definition\_usage <=  
product\_definition\_relationship  
product\_definition\_relationship.id

**6.1.7.4.3 item\_instance to item\_definition (as definition)**

AIM element: PATH

Rules: restrict\_product\_definitions\_for\_product\_definition\_relationship

Reference path: assembly\_component\_usage <=  
product\_definition\_usage <=  
product\_definition\_relationship  
product\_definition\_relationship.related\_product\_definition ->  
product\_definition  
{product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'part definition'}

**6.1.7.4.4 item\_instance to multi\_language\_string (as description)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([product\_definition\_relationship.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product\_definition,  
restrict\_multi\_language\_for\_product\_definition\_relationship

Reference path: assembly\_component\_usage <=  
product\_definition\_usage <=  
product\_definition\_relationship  
#1: (attribute\_language\_item = product\_definition\_relationship  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#2: (multi\_language\_attribute\_item = product\_definition\_relationship  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

#### 6.1.7.5 item\_structure\_association

#1: If item\_structure\_association is an assembly\_association.

AIM element: product\_definition\_relationship

Source: ISO 10303-41

Reference path: #1: {product\_definition\_relationship =>  
product\_definition\_usage =>  
assembly\_component\_usage}

##### 6.1.7.5.1 item\_structure\_association to item\_instance (as related)

#1: If item\_structure\_association is an assembly\_association.

AIM element: #1: IDENTICAL MAPPING

Reference path: #1: {product\_definition\_relationship =>  
product\_definition\_usage =>  
assembly\_component\_usage}

##### 6.1.7.5.2 item\_structure\_association to item\_definition (as relating)

AIM element: PATH

Reference path: product\_definition\_relationship  
product\_definition\_relationship.relateing\_product\_definition ->  
product\_definition  
{product\_definition.frame\_of\_reference ->  
product\_definition\_context <=

```
application_context_element  
application_context_element.name = 'part definition'}
```

#### 6.1.7.6 mated\_item\_relationship

AIM element: product\_definition\_relationship

Source: ISO 10303-41

Reference path: {product\_definition\_relationship  
product\_definition\_relationship.name = 'mating material'}

##### 6.1.7.6.1 mated\_item\_relationship to quantified\_instance (as mating\_material)

AIM element: PATH

```
Reference path: product_definition_relationship  
product_definition_relationship.related_product_definition ->  
product_definition  
{[product_definition.frame_of_reference ->  
product_definition_context <=  
application_context_element  
application_context_element.name = 'part occurrence']  
[name_attribute_select = product_definition  
name_attribute_select <-  
name_attribute.named_item  
name_attribute  
name_attribute.attribute_value = 'quantified instance']}
```

##### 6.1.7.6.2 mated\_item\_relationship to mating\_association (as related)

AIM element: PATH

Rules: dependent\_instantiable\_product\_definition\_context\_role

```
Reference path: product_definition_relationship  
product_definition_relationship.related_product_definition ->  
product_definition  
{product_definition <-  
product_definition_context_association.definition  
product_definition_context_association  
[product_definition_context_association.frame_of_reference ->  
product_definition_context <=  
application_context_element  
application_context_element.name = 'mating definition']  
[product_definition_context_association.role ->  
product_definition_context_role  
product_definition_context_role.name = 'part definition type']}
```

##### 6.1.7.6.3 mated\_item\_relationship to mating\_association (as relating)

AIM element: PATH

Rules: dependent\_instantiable\_product\_definition\_context\_role

```
Reference path: product_definition_relationship  
product_definition_relationship.relatating_product_definition ->  
product_definition
```

```
{product_definition <-
product_definition_context_association.definition
product_definition_context_association
[product_definition_context_association.frame_of_reference ->
product_definition_context <=
application_context_element
application_context_element.name = 'mating definition']
[product_definition_context_association.role ->
product_definition_context_role
product_definition_context_role.name = 'part definition type']}
```

#### 6.1.7.7 mating\_association

AIM element: product\_definition\_relationship

Source: ISO 10303-41

Reference path: {product\_definition\_relationship  
[product\_definition\_relationship.name = 'mating membership']  
[product\_definition\_relationship.relying\_product\_definition ->  
product\_definition <-  
product\_definition\_context\_association.definition  
product\_definition\_context\_association  
[product\_definition\_context\_association.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'mating definition']  
[product\_definition\_context\_association.role ->  
product\_definition\_context\_role  
product\_definition\_context\_role.name = 'part definition type']]  
[product\_definition\_relationship.related\_product\_definition ->  
product\_definition  
product\_definition <-  
product\_definition\_context\_association.definition  
product\_definition\_context\_association  
[product\_definition\_context\_association.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'mating definition']  
[product\_definition\_context\_association.role ->  
product\_definition\_context\_role  
product\_definition\_context\_role.name = 'part definition type']}

#### 6.1.7.7.1 mating\_association to geometric\_model\_relationship\_with\_transformation (as placement)

AIM element: PATH

Rules: dependent\_instantiable\_product\_definition\_context\_role

Reference path: product\_definition\_relationship  
{product\_definition\_relationship.name = 'mating membership'}  
[product\_definition\_relationship.relying\_product\_definition ->  
product\_definition  
{product\_definition <-  
product\_definition\_context\_association.definition  
product\_definition\_context\_association  
[product\_definition\_context\_association.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element

```

application_context_element.name = 'mating definition']
[product_definition_context_association.role ->
product_definition_context_role
product_definition_context_role.name = 'part definition type']]
characterized_product_definition = product_definition
characterized_product_definition
characterized_definition = characterized_product_definition
characterized_definition <-
property_definition.definition
property_definition
{property_definition =>
product_definition_shape}
property_definition
represented_definition = property_definition
represented_definition <-
property_definition_representation.definition
property_definition_representation
{property_definition_representation =>
shape_definition_representation}
property_definition_representation.used_representation ->
representation <-
representation_relationship.rep_2]
[product_definition_relationship.related_product_definition ->
product_definition
{product_definition <-
product_definition_context_association.definition
product_definition_context_association
[product_definition_context_association.frame_of_reference ->
product_definition_context <=
application_context_element
application_context_element.name = 'mating definition']
[product_definition_context_association.role ->
product_definition_context_role
product_definition_context_role.name = 'part definition type']]
characterized_product_definition = product_definition
characterized_product_definition
characterized_definition = characterized_product_definition
characterized_definition <-
property_definition.definition
property_definition
{property_definition =>
product_definition_shape}
property_definition
represented_definition = property_definition
represented_definition <-
property_definition_representation.definition
property_definition_representation
{property_definition_representation =>
shape_definition_representation}
property_definition_representation.used_representation ->
representation <-
representation_relationship.rep_1]
representation_relationship =>
representation_relationship_with_transformation

```

**6.1.7.7.2 mating\_association to mating\_definition (as relating)**

AIM element: PATH

Rules: dependent\_instantiable\_product\_definition\_context\_role

Reference path: product\_definition\_relationship.relatering\_product\_definition ->  
 product\_definition  
 {product\_definition <-  
 product\_definition\_context\_association.definition  
 product\_definition\_context\_association  
 [product\_definition\_context\_association.frame\_of\_reference ->  
 product\_definition\_context <=  
 application\_context\_element  
 application\_context\_element.name = 'mating definition']  
 [product\_definition\_context\_association.role ->  
 product\_definition\_context\_role  
 product\_definition\_context\_role.name = 'part definition type']}

#### 6.1.7.8 mating\_definition

#1: If the mating\_definition is the topmost node of a mating tree.

#2: If the mating\_definition is a subnode of the mating tree.

AIM element: product\_definition

Source: ISO 10303-41

Reference path: #1: ({product\_definition <-  
 product\_definition\_context\_association.definition  
 product\_definition\_context\_association  
 [product\_definition\_context\_association.frame\_of\_reference ->  
 product\_definition\_context <=  
 application\_context\_element  
 application\_context\_element.name = 'mating definition']  
 [product\_definition\_context\_association.role ->  
 product\_definition\_context\_role  
 product\_definition\_context\_role.name = 'part definition type']})  
 #2: (product\_definition <-  
 {product\_definition <-  
 product\_definition\_context\_association.definition  
 product\_definition\_context\_association  
 [product\_definition\_context\_association.frame\_of\_reference ->  
 product\_definition\_context <=  
 application\_context\_element  
 application\_context\_element.name = 'mating definition']  
 [product\_definition\_context\_association.role ->  
 product\_definition\_context\_role  
 product\_definition\_context\_role.name = 'part definition type']  
 product\_definition\_relationship.related\_product\_definition  
 product\_definition\_relationship  
 {[product\_definition\_relationship.name = 'mating membership']  
 [product\_definition\_relationship \*]}  
 product\_definition\_relationship.relatering\_product\_definition ->  
 product\_definition  
 {product\_definition <-  
 product\_definition\_context\_association.definition  
 product\_definition\_context\_association  
 [product\_definition\_context\_association.frame\_of\_reference ->  
 product\_definition\_context <=  
 application\_context\_element  
 application\_context\_element.name = 'mating definition']  
 [product\_definition\_context\_association.role ->

```
product_definition_context_role
product_definition_context_role.name = 'part definition type']})
```

#### 6.1.7.8.1 mating\_type

AIM element: product\_definition.name

Source: ISO 10303-41

Reference path: product\_definition  
(product\_definition.name)  
(product\_definition.name = 'bolted joint')  
(product\_definition.name = 'brazing')  
(product\_definition.name = 'clasping')  
(product\_definition.name = 'doweling')  
(product\_definition.name = 'flanging')  
(product\_definition.name = 'press fit')  
(product\_definition.name = 'welding')}}

#### 6.1.7.9 quantified\_instance

AIM element: quantified\_assembly\_component\_usage

Source: ISO 10303-44

##### 6.1.7.9.1 quantified\_instance to numerical\_value (as quantity)

AIM element: PATH

Rules: restrict\_part\_occurrence

Reference path: quantified\_assembly\_component\_usage  
quantified\_assembly\_component\_usage.quantity ->  
measure\_with\_unit

#### 6.1.8 item\_property UoF

##### 6.1.8.1 coating

AIM element: product\_definition

Source: ISO 10303-41

Rules: coating\_requires\_product\_definition,  
restrict\_product\_definition\_context\_for\_product

Reference path: {product\_definition  
[product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'material definition']  
[product\_definition.formation ->  
product\_definition\_formation  
product\_definition\_formation.of\_product ->  
product <-  
product\_related\_product\_category.products[i]  
product\_related\_product\_category <=

```
product_category
product_category.name = 'coating']}]
```

#### 6.1.8.1.1 coating\_name

AIM element: PATH

Reference path: product\_definition <-  
material\_designation.definitions[i]  
material\_designation  
material\_designation.name

#### 6.1.8.1.2 coating\_system

AIM element: PATH

Reference path: product\_definition <-  
process\_product\_association.defined\_product  
process\_product\_association.process ->  
product\_definition\_process  
product\_definition\_process.identification

#### 6.1.8.2 coupling

AIM element: representation

Source: ISO 13030-43

Rules: restrict\_representation\_for\_coupling

Reference path: {representation <-  
{[representation.name = 'coupling']  
[representation.context\_of\_items ->  
representation\_context  
representation\_context.context\_type = 'coupling parameters']}  
property\_definition\_representation.used\_representation  
property\_definition\_representation  
property\_definition\_representation.definition ->  
represented\_definition  
represented\_definition = property\_definition  
property\_definition  
{property\_definition.name = 'coupling property'}}

#### 6.1.8.2.1 coupling\_type

AIM element: descriptive\_representation\_item.description

Source: ISO 10303-45

Reference path: representation  
representation.items[i] ->  
representation\_item =>  
{representation\_item.name = 'coupling type'}  
descriptive\_representation\_item  
descriptive\_representation\_item.description  
{(descriptive\_representation\_item.description)  
(descriptive\_representation\_item.description = '0')  
(descriptive\_representation\_item.description = '1')}

```
(descriptive_representation_item.description = '2')
(descriptive_representation_item.description = '3')
(descriptive_representation_item.description = '4')
(descriptive_representation_item.description = '5')
(descriptive_representation_item.description = '6')
(descriptive_representation_item.description = '7')}
```

**6.1.8.2.2 side**

AIM element: descriptive\_representation\_item.description

Source: ISO 10303-45

Reference path: representation  
 representation.items[i] ->  
 representation\_item =>  
 {representation\_item.name = 'side'}  
 descriptive\_representation\_item  
 descriptive\_representation\_item.description  
 {(descriptive\_representation\_item.description)  
 (descriptive\_representation\_item.description = 'machine side')  
 (descriptive\_representation\_item.description = 'workpiece side')}

**6.1.8.2.3 style**

AIM element: descriptive\_representation\_item.description

Source: ISO 10303-45

Reference path: representation  
 representation.items[i] ->  
 representation\_item =>  
 {representation\_item.name = 'style'}  
 descriptive\_representation\_item  
 descriptive\_representation\_item.description  
 {(descriptive\_representation\_item.description)  
 (descriptive\_representation\_item.description = "")  
 (descriptive\_representation\_item.description = "")}

**6.1.8.2.4 coupling to numerical\_value (as pieces)**

#1: If the unit is not assigned globally.

#2: If the unit is assigned globally.

AIM element: #1: (measure\_representation\_item)  
 #2: (value\_representation\_item)

Source: ISO 10303-45  
 ISO 10303-45

Reference path: representation  
 representation.items[i] ->  
 representation\_item =>  
 {representation\_item.name = 'coupling pieces'}  
 #1: (measure\_representation\_item  
 {measure\_representation\_item <=  
 measure\_with\_unit  
 [measure\_with\_unit.value\_component ->

```

measure_value
measure_value = ratio_measure]
[measure_with_unit.unit_component ->
unit
unit = named_unit
named_unit =>
ratio_unit]])
#3: (value_representation_item)

```

#### 6.1.8.2.5 coupling to numerical\_value (as size)

#1: If the unit is not assigned globally.

#2: If the unit is assigned globally.

AIM element: #1: (measure\_representation\_item)  
#2: (value\_representation\_item)

Source: ISO 10303-45  
ISO 10303-45

Reference path: representation  
representation.items[i] ->  
representation\_item =>  
{representation\_item.name = 'coupling size'}  
#1: (measure\_representation\_item  
{measure\_representation\_item <=  
measure\_with\_unit  
[measure\_with\_unit.value\_component ->  
measure\_value  
measure\_value = ratio\_measure]  
[measure\_with\_unit.unit\_component ->  
unit  
unit = named\_unit  
named\_unit =>  
ratio\_unit]])  
#3: (value\_representation\_item)

#### 6.1.8.3 cutting\_condition

AIM element: representation

Source: ISO 13030-43

Rules: restrict\_representation\_for\_cutting\_condition

Reference path: {representation <-  
property\_definition\_representation.used\_representation  
property\_definition\_representation  
property\_definition\_representation.definition ->  
represented\_definition  
represented\_definition = property\_definition  
property\_definition  
{property\_definition.name = 'cutting condition property'}  
[representation.context\_of\_items ->  
representation\_context  
representation\_context.context\_type = 'cutting condition parameters']  
[representation.name = 'cutting condition']}

**6.1.8.3.1 condition\_name**

AIM element: descriptive\_representation\_item.description

Source: ISO 10303-45

Reference path: representation  
representation.items[i] ->  
representation\_item =>  
{representation\_item.name = 'condition name'}  
descriptive\_representation\_item  
descriptive\_representation\_item.description  
{(descriptive\_representation\_item.description)  
(descriptive\_representation\_item.description = "")  
(descriptive\_representation\_item.description = "")}

**6.1.8.4 cutting\_data\_association**

AIM element: representation

Source: ISO 13030-43

Rules: restrict\_representation\_for\_cutting\_data\_association

Reference path: {representation <-  
property\_definition\_representation.used\_representation  
property\_definition\_representation  
property\_definition\_representation.definition ->  
represented\_definition  
represented\_definition = property\_definition  
property\_definition  
{property\_definition.name = 'cutting data association'}  
[representation.context\_of\_items ->  
representation\_context  
representation\_context.context\_type = 'cutting data parameters']  
[representation.name = 'cutting data association']}

**6.1.8.4.1 cutting\_data\_association to material\_designation (as associated\_material)**

AIM element: PATH

Reference path: representation <-  
property\_definition\_representation.used\_representation  
property\_definition\_representation  
{property\_definition\_representation.name = 'associated material'}  
property\_definition\_representation.definition ->  
represented\_definition  
represented\_definition = property\_definition  
property\_definition  
{property\_definition.name = 'associated material'}  
property\_definition.definition ->  
characterized\_definition  
characterized\_definition = product\_definition  
product\_definition <-  
{product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'workpiece definition'}  
material\_designation.definitions[i]

material\_designation  
 material\_designation.name

#### 6.1.8.4.2 cutting\_data\_association to value\_with\_unit (as depth\_of\_cut)

#1: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is not assigned globally.

#2: If the value\_with\_unit is a value\_range.

#3: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is specified globally.

AIM element: PATH

Reference path: representation  
 representation.items[i] ->  
 representation\_item =>  
 {representation\_item.name = 'depth of cut'}  
 #1: (measure\_representation\_item)  
 #2: (compound\_representation\_item =>  
 value\_range)  
 #3: (value\_representation\_item)

#### 6.1.8.4.3 cutting\_data\_association to value\_with\_unit (as feed)

#1: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is not assigned globally.

#2: If the value\_with\_unit is a value\_range.

#3: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is specified globally.

AIM element: PATH

Reference path: representation  
 representation.items[i] ->  
 representation\_item =>  
 {representation\_item.name = 'feed'}  
 #1: (measure\_representation\_item)  
 #2: (compound\_representation\_item =>  
 value\_range)  
 #3: (value\_representation\_item)

#### 6.1.8.4.4 cutting\_data\_association to value\_with\_unit (as speed)

#1: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is not assigned globally.

#2: If the value\_with\_unit is a value\_range.

#3: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is specified globally.

AIM element: PATH

Reference path: representation  
 representation.items[i] ->  
 representation\_item =>  
 {representation\_item.name = 'speed'}  
 #1: (measure\_representation\_item)  
 #2: (compound\_representation\_item =>

value\_range)  
#3: (value\_representation\_item)

#### 6.1.8.5 grade

AIM element: product\_definition

Source: ISO 10303-41

Rules: grade\_requires\_product\_definition,  
restrict\_product\_definition\_context\_for\_product

Reference path: {product\_definition  
[product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'material definition']  
[product\_definition.formation ->  
product\_definition\_formation  
product\_definition\_formation.of\_product ->  
product <-  
product\_related\_product\_category.products[i]  
product\_related\_product\_category <=  
product\_category  
product\_category.name = 'grade']}

#### 6.1.8.5.1 identifier

AIM element: product.id

Source: ISO 10303-41

Reference path: product\_definition  
product\_definition.formation ->  
product\_definition\_formation  
product\_definition\_formation.of\_product ->  
product  
product.id

#### 6.1.8.5.2 standard\_designation

AIM element: identification\_assignment.assigned\_id

Source: ISO 10303-41

Rules: dependent\_instantiable\_identification\_role

Reference path: product\_definition  
product\_definition.formation ->  
product\_definition\_formation  
product\_definition\_formation.of\_product ->  
product  
identification\_item = product  
identification\_item <-  
applied\_identification\_assignment.items[i]  
applied\_identification\_assignment <=  
identification\_assignment  
{identification\_assignment.role ->  
identification\_role

```

identification_role.name = 'standard designation'}
identification_assignment.assigned_id

```

#### 6.1.8.5.3 grade to coating (as coating)

AIM element: product\_definition\_relationship

Source: ISO 10303-41

Rules: restrict\_product\_definitions\_for\_product\_definition\_relationship

Reference path: {product\_definition\_relationship  
 [product\_definition\_relationship.name = 'coating']  
 [product\_definition\_relationship.related\_product ->  
 {product\_definition  
 product\_definition.formation ->  
 product\_definition\_formation  
 product\_definition\_formation.of\_product ->  
 product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=  
 product\_category  
 product\_category.name = 'coating'}}  
 [product\_definition\_relationship.relying\_product ->  
 {product\_definition  
 product\_definition.formation ->  
 product\_definition\_formation  
 product\_definition\_formation.of\_product ->  
 product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=  
 product\_category  
 product\_category.name = 'grade'}}}

#### 6.1.8.5.4 grade to cutting\_condition (as cutting\_condition)

AIM element: property\_definition

Source: ISO 10303-41

Reference path: {property\_definition  
 [represented\_definition = property\_definition  
 represented\_definition <-  
 property\_definition\_representation.definition  
 property\_definition\_representation  
 property\_definition\_representation.used\_representation ->  
 representation  
 {representation.name = 'cutting condition'}}  
 [property\_definition.definition ->  
 characterized\_definition  
 characterized\_definition = characterized\_product\_definition  
 characterized\_product\_definition = product\_definition  
 product\_definition  
 {product\_definition.formation ->  
 product\_definition\_formation  
 product\_definition\_formation.of\_product ->  
 product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=

```
product_category
product_category.name = 'grade']}]}
```

**6.1.8.5.5 grade to substrate (as substrate)**

AIM element: product\_definition\_relationship

Source: ISO 10303-41

Rules: restrict\_product\_definitions\_for\_product\_definition\_relationship

Reference path: {product\_definition\_relationship  
 [product\_definition\_relationship.name = 'substrate']  
 [product\_definition\_relationship.related\_product ->  
 {product\_definition  
 product\_definition.formation ->  
 product\_definition\_formation  
 product\_definition\_formation.of\_product ->  
 product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=  
 product\_category  
 product\_category.name = 'substrate'}]  
 [product\_definition\_relationship.relying\_product ->  
 {product\_definition  
 product\_definition.formation ->  
 product\_definition\_formation  
 product\_definition\_formation.of\_product ->  
 product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=  
 product\_category  
 product\_category.name = 'grade'}]}}

**6.1.8.5.6 grade to material\_designation (as workpiece\_material)**

AIM element: PATH

Reference path: {product\_definition\_relationship  
 [product\_definition\_relationship.name = 'workpiece material']  
 [product\_definition\_relationship.related\_product ->  
 {product\_definition <-  
 material\_designation.definitions[i]  
 material\_designation  
 material\_designation.name  
 [product\_definition.formation ->  
 product\_definition\_formation  
 product\_definition\_formation.of\_product ->  
 product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=  
 product\_category  
 product\_category.name = 'workpiece'}]  
 [product\_definition\_relationship.relying\_product ->  
 {product\_definition  
 product\_definition.formation ->  
 product\_definition\_formation  
 product\_definition\_formation.of\_product ->  
 product <-

```

product_related_product_category.products[i]
product_related_product_category <=
product_category
product_category.name = 'grade'}}

```

#### 6.1.8.6 grade\_relationship

AIM element: product\_definition\_relationship

Source: ISO 10303-41

Rules: restrict\_product\_definitions\_for\_product\_definition\_relationship

##### 6.1.8.6.1 relation\_type

AIM element: product\_definition\_relationship.name

Source: ISO 10303-41

##### 6.1.8.6.2 grade\_relationship to grade (as related)

AIM element: product\_definition\_relationship.related\_product\_definition

Source: ISO 10303-41

Reference path: {product\_definition\_relationship.related\_product\_definition ->  
 {product\_definition  
 [product\_definition.frame\_of\_reference ->  
 product\_definition\_context <=  
 application\_context\_element  
 application\_context\_element.name = 'material definition']  
 [product\_definition\_formation ->  
 product\_definition\_formation  
 product\_definition\_formation.of\_product ->  
 product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=  
 product\_category  
 product\_category.name = 'grade']]}

##### 6.1.8.6.3 grade\_relationship to grade (as relating)

AIM element: product\_definition\_relationship.relying\_product\_definition

Source: ISO 10303-41

Reference path: {product\_definition\_relationship.relying\_product\_definition ->  
 {product\_definition  
 [product\_definition.frame\_of\_reference ->  
 product\_definition\_context <=  
 application\_context\_element  
 application\_context\_element.name = 'material definition']  
 [product\_definition\_formation ->  
 product\_definition\_formation  
 product\_definition\_formation.of\_product ->  
 product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=

```
product_category
product_category.name = 'grade']}]}
```

#### 6.1.8.7 item\_characteristic\_association

##### 6.1.8.7.1 item\_characteristic\_association to coupling (as associated\_characteristic)

AIM element: property\_definition

```
Reference path: property_definition
{property_definition.name = 'coupling property' }
represented_definition = property_definition
represented_definition <-
property_definition_representation.definition
property_definition_representation
property_definition_representation.used_representation ->
representation
{representation.name = 'coupling'}
```

##### 6.1.8.7.2 item\_characteristic\_association to cutting\_condition (as associated\_characteristic)

AIM element: property\_definition

```
Reference path: property_definition
{property_definition.name = 'cutting condition property' }
represented_definition = property_definition
represented_definition <-
property_definition_representation.definition
property_definition_representation
property_definition_representation.used_representation ->
representation
{representation.name = 'cutting condition'}
```

##### 6.1.8.7.3 item\_characteristic\_association to cutting\_data\_association (as associated\_characteristic)

AIM element: property\_definition

```
Reference path: property_definition
{property_definition.name = 'cutting data association' }
represented_definition = property_definition
represented_definition <-
property_definition_representation.definition
property_definition_representation
property_definition_representation.used_representation ->
representation
{representation.name = 'cutting data association'}
```

##### 6.1.8.7.4 item\_characteristic\_association to grade (as associated\_characteristic)

AIM element: product\_definition\_relationship

Source: ISO 10303-41

```
Reference path: {product_definition_relationship
[product_definition_relationship.name = 'item characteristic association']
[product_definition_relationship.related ->
product_definition
{product_definition.frame_of_reference ->
```

```

product_definition_context <=
application_context_element
application_context_element.name = 'material definition'}}

```

#### 6.1.8.7.5 item\_characteristic\_association to material\_designation (as associated\_characteristic)

AIM element: material\_designation.definition

Source: ISO 10303-45

Reference path: material\_designation  
material\_designation.definition ->  
characterized\_definition = characterized\_product\_definition  
characterized\_product\_definition = product\_definition  
product\_definition  
{product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'part definition'}

#### 6.1.8.7.6 item\_characteristic\_association to workpiece\_feature (as associated\_characteristic)

AIM element: product\_definition\_relationship

Reference path: {product\_definition\_relationship  
[product\_definition\_relationship.name = 'item characteristic association']  
[product\_definition\_relationship.related ->  
product\_definition  
product\_definition.formation ->  
product\_definition\_formation  
product\_definition\_formation.of\_product ->  
product <-  
product\_related\_product\_category.products[i]  
product\_related\_product\_category <=  
product\_category  
product\_category.name = 'workpiece'}}

#### 6.1.8.7.7 item\_characteristic\_association to item\_definition (as associated\_item)

#1: If the associated\_characteristic is a coupling, cutting\_condition or a cutting\_data\_association

#2: If the associated\_characteristic is a grade or a workpiece\_feature

AIM element: #1: (property\_definition.definition)  
#2: (product\_definition\_relationship.relying)

Reference path: #1: (property\_definition.definition ->  
characterized\_definition  
characterized\_definition = product\_definition  
product\_definition  
{product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'part definition'})  
#2: (product\_definition\_relationship.relying ->  
product\_definition  
{product\_definition.frame\_of\_reference ->  
product\_definition\_context <=

```
application_context_element  
application_context_element.name = 'part definition'})
```

#### 6.1.8.8 material\_designation

AIM element: material\_designation

Source: ISO 10303-45

##### 6.1.8.8.1 material\_name

AIM element: material\_designation.name

Source: ISO 10303-45

#### 6.1.8.9 property

AIM element: general\_property

Source: ISO 10303-41

##### 6.1.8.9.1 description

AIM element: general\_property.description

Source: ISO 10303-41

##### 6.1.8.9.2 id

AIM element: general\_property.id

Source: ISO 10303-41

##### 6.1.8.9.3 property\_type

AIM element: general\_property.name

Source: ISO 10303-41

Reference path: {(general\_property.name)  
(general\_property.name = 'cost')  
(general\_property.name = 'dimension')  
(general\_property.name = 'mass')  
(general\_property.name = 'shape')  
(general\_property.name = 'specification')  
(general\_property.name = 'usage')}

##### 6.1.8.9.4 version\_id

AIM element: identification\_assignment.assigned\_id

Source: ISO 10303-41

Reference path: general\_property  
identification\_item = general\_property  
identification\_item <-  
applied\_identification\_assignment.items[i]

```

applied_identification_assignment <=
identification_assignment
{identification_assignment.role ->
identification_role
identification_role.name = 'version'}
identification_assignment.assigned_id

```

#### 6.1.8.9.5 property to unit (as allowed\_unit)

#1: If the unit has a commonly used descriptor.

#2: If the unit descriptor is constructed of more than one unit name.

AIM element: PATH

```

Reference path: general_property
represented_definition = general_property
represented_definition <-
property_definition_representation.definition
property_definition_representation
property_definition_representation.used_representation ->
representation
{[representation.name = 'allowed units']
[representation.items[1] ->
representation_item =>
descriptive_representation_item
descriptive_representation_item.description = 'allowed units']}
representation.context_of_items ->
representation_context =>
global_unit_assigned_context
global_unit_assigned_context.units[i] ->
unit
#1: (unit = named_unit
named_unit)
#2: (unit = derived_unit
derived_unit)

```

#### 6.1.8.9.6 property to multi\_language\_string (as description)

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([general\_property.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_general\_property

```

Reference path: general_property
#1: (attribute_language_item = general_property
attribute_language_item <-
attribute_language_assignment.items[i]
attribute_language_assignment
{attribute_language_assignment <=
attribute_classification_assignment
attribute_classification_assignment.attribute_name = 'description'})

```

```
#2: (multi_language_attribute_item = general_property
multi_language_attribute_item <-
multi_language_attribute_assignment.items[i]
multi_language_attribute_assignment
{multi_language_attribute_assignment <=
attribute_value_assignment
attribute_value_assignment.attribute_name = 'description'})
```

**6.1.8.9.7 property to external\_library\_reference (as property\_source)**

AIM element: IDENTICAL MAPPING

Reference path: {general\_property =>  
externally\_defined\_general\_property}

**6.1.8.9.8 property to plib\_property\_reference (as property\_source)**

AIM element: IDENTICAL MAPPING

Reference path: {general\_property =>  
externally\_defined\_general\_property}

**6.1.8.10 property\_relationship**

AIM element: general\_property\_relationship

Source: ISO 10303-41

**6.1.8.10.1 description**

AIM element: general\_property\_relationship.description

Source: ISO 10303-41

**6.1.8.10.2 relation\_type**

AIM element: general\_property\_relationship.name

Source: ISO 10303-41

Reference path: {(general\_property\_relationship.name)  
(general\_property\_relationship.name = 'dependency')  
(general\_property\_relationship.name = 'hierarchy')}

**6.1.8.10.3 property\_relationship to multi\_language\_string (as description)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([general\_property\_relationship.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_general\_property\_relationship

Reference path: `general_property_relationship`  
`#1: (attribute_language_item = general_property_relationship`  
`attribute_language_item <-`  
`attribute_language_assignment.items[i]`  
`attribute_language_assignment`  
`{attribute_language_assignment <=`  
`attribute_classification_assignment`  
`attribute_classification_assignment.attribute_name = 'description'})`  
`#2: (multi_language_attribute_item = general_property_relationship`  
`multi_language_attribute_item <-`  
`multi_language_attribute_assignment.items[i]`  
`multi_language_attribute_assignment`  
`{multi_language_attribute_assignment <=`  
`attribute_value_assignment`  
`attribute_value_assignment.attribute_name = 'description'})`

#### 6.1.8.10.4 property\_relationship to property (as related)

AIM element: PATH

Reference path: `general_property_relationship`  
`general_property_relationship.related_property ->`  
`general_property`

#### 6.1.8.10.5 property\_relationship to property (as relating)

AIM element: PATH

Reference path: `general_property_relationship`  
`general_property_relationship.relatating_property ->`  
`general_property`

#### 6.1.8.11 property\_value\_association

AIM element: `property_definition`

Source: ISO 10303-41

##### 6.1.8.11.1 definitional

#1: If `item_property_association.definitional` is TRUE.

#2: If `item_property_association.definitional` is FALSE.

#3: If `item_property_association.definitional` is not instantiated.

AIM element: `general_property_association.name`

Source: ISO 10303-41

Reference path: `property_definition`  
`derived_property_select = property_definition`  
`derived_property_select <-`  
`general_property_association.derived_definition`  
`general_property_association`  
`general_property_association.name`  
`{#1: (general_property_association.name = 'definitional')}`

```
#2: (general_property_association.name = 'non-definitional')
#3: (general_property_association.name = "")}
```

#### 6.1.8.11.2 description

AIM element: property\_definition.description

Source: ISO 10303-41

#### 6.1.8.11.3 property\_value\_association to property\_value\_representation (as describing\_property\_value)

#1 If the described\_element is not a material\_designation.

#2 If the described\_element is a material\_designation

AIM element: PATH

Reference path: property\_definition  
 represented\_definition = property\_definition  
 represented\_definition <-  
 #1: (property\_definition\_representation.definition  
 property\_definition\_representation  
 property\_definition\_representation.used\_representation ->)  
 #2: (material\_property\_representation.definition  
 material\_property\_representation  
 material\_property\_representation.used\_representation ->)  
 representation

#### 6.1.8.11.4 property\_value\_association to multi\_language\_string (as description)

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([property\_definition.description]  
 [PATH])  
 #2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_property\_definition

Reference path: property\_definition  
 #1: (attribute\_language\_item = property\_definition  
 attribute\_language\_item <-  
 attribute\_language\_assignment.items[i]  
 attribute\_language\_assignment  
 {attribute\_language\_assignment <=  
 attribute\_classification\_assignment  
 attribute\_classification\_assignment.attribute\_name = 'description'})  
 #2: (multi\_language\_attribute\_item = property\_definition  
 multi\_language\_attribute\_item <-  
 multi\_language\_attribute\_assignment.items[i]  
 multi\_language\_attribute\_assignment  
 {multi\_language\_attribute\_assignment <=  
 attribute\_value\_assignment  
 attribute\_value\_assignment.attribute\_name = 'description'})

**6.1.8.11.5 property\_value\_association to coupling (as described\_element)**

AIM element: property\_definition\_relationship

Source: ISO 10303-45

Reference path: property\_definition <-  
 property\_definition\_relationship.relating\_property\_definition  
 {property\_definition\_relationship  
 [property\_definition\_relationship.name = 'coupling']  
 [property\_definition\_relationship.related\_property\_definition ->  
 property\_definition  
 {property\_definition.name = 'coupling property'}}}

**6.1.8.11.6 property\_value\_association to cutting\_condition (as described\_element)**

AIM element: property\_definition\_relationship

Source: ISO 10303-45

Reference path: property\_definition <-  
 property\_definition\_relationship.relating\_property\_definition  
 {property\_definition\_relationship  
 [property\_definition\_relationship.name = 'cutting condition']  
 [property\_definition\_relationship.related\_property\_definition ->  
 property\_definition  
 {property\_definition.name = 'cutting condition'}}}

**6.1.8.11.7 property\_value\_association to cutting\_data\_association (as described\_element)**

AIM element: property\_definition\_relationship

Source: ISO 10303-45

Reference path: property\_definition <-  
 property\_definition\_relationship.relating\_property\_definition  
 {property\_definition\_relationship  
 [property\_definition\_relationship.name = 'cutting data association']  
 [property\_definition\_relationship.related\_property\_definition ->  
 property\_definition  
 {property\_definition.name = 'cutting data association'}}}

**6.1.8.11.8 property\_value\_association to document\_file (as described\_element)**

AIM element: PATH

Reference path: property\_definition  
 property\_definition.definition ->  
 characterized\_definition  
 characterized\_definition = characterized\_object  
 characterized\_object =>  
 document\_file

**6.1.8.11.9 property\_value\_association to document\_representation (as described\_element)**

AIM element: PATH

Reference path: property\_definition  
property\_definition.definition ->  
characterized\_definition  
characterized\_definition = characterized\_product\_definition  
characterized\_product\_definition  
characterized\_product\_definition = product\_definition  
product\_definition

**6.1.8.11.10 property\_value\_association to item\_definition (as described\_element)**

AIM element: PATH

Reference path: property\_definition  
property\_definition.definition ->  
characterized\_definition  
characterized\_definition = characterized\_product\_definition  
characterized\_product\_definition = product\_definition  
product\_definition  
{product\_definition.frame\_of\_reference ->  
product\_definition\_context <=  
application\_context\_element  
application\_context\_element.name = 'part definition'}

**6.1.8.11.11 property\_value\_association to item\_instance (as described\_element)**

AIM element: PATH

Reference path: property\_definition  
property\_definition.definition ->  
characterized\_definition  
characterized\_definition = characterized\_product\_definition  
characterized\_product\_definition  
characterized\_product\_definition = product\_definition\_relationship  
product\_definition\_relationship =>  
product\_definition\_usage =>  
assembly\_component\_usage

**6.1.8.11.12 property\_value\_association to item\_structure\_association (as described\_element)**

AIM element: PATH

Reference path: property\_definition  
property\_definition.definition ->  
characterized\_definition  
characterized\_definition = characterized\_product\_definition  
characterized\_product\_definition = product\_definition\_relationship  
product\_definition\_relationship

**6.1.8.11.13 property\_value\_association to grade (as described\_element)**

AIM element: PATH

Reference path: property\_definition  
property\_definition.definition ->  
characterized\_definition  
characterized\_definition = characterized\_product\_definition  
characterized\_product\_definition

characterized\_product\_definition = product\_definition  
product\_definition

#### 6.1.8.11.14 property\_value\_association to mated\_item\_relationship (as described\_element)

AIM element: PATH

Reference path: property\_definition  
property\_definition.definition ->  
characterized\_definition  
characterized\_definition = characterized\_product\_definition  
characterized\_product\_definition = product\_definition\_relationship  
product\_definition\_relationship

#### 6.1.8.11.15 property\_value\_association to material\_designation (as described\_element)

AIM element: PATH

Reference path: property\_definition <-  
material\_property\_representation.definition  
material\_property\_representation  
characterized\_material\_property = material\_property\_representation  
characterized\_material\_property <-  
material\_designation\_characterization.property  
material\_designation\_characterization  
material\_designation\_characterization.designation ->  
material\_designation

#### 6.1.8.11.16 property\_value\_association to organization (as validity\_context)

AIM element: PATH

Reference path: property\_definition  
organization\_item = property\_definition  
organization\_item <-  
applied\_organization\_assignment.items[i]  
applied\_organization\_assignment <=  
organization\_assignment  
{organization\_assignment.role ->  
organization\_role  
organization\_role.name = 'validity context'}  
organization\_assignment.assigned\_organization ->  
organization

#### 6.1.8.11.17 property\_value\_association to physical\_item\_structure\_association (as described\_element)

AIM element: PATH

Reference path: property\_definition  
property\_definition.definition ->  
characterized\_definition  
characterized\_definition = characterized\_product\_definition  
characterized\_product\_definition = product\_definition\_relationship  
product\_definition\_relationship  
{[product\_definition\_relationship.relating\_product\_definition ->  
product\_definition  
product\_definition.frame\_of\_reference ->

```

product_definition_context <=
  application_context_element
  application_context_element.name = 'physical occurrence']
[product_definition_relationship.related_product_definition ->
  product_definition
  product_definition.frame_of_reference ->
  product_definition_context <=
  application_context_element
  application_context_element.name = 'physical occurrence']]

```

**6.1.8.11.18 property\_value\_association\_to\_workpiece\_feature (as\_described\_element)**

AIM element: PATH

Reference path: property\_definition  
 property\_definition.definition ->  
 characterized\_definition  
 characterized\_definition = characterized\_object  
 characterized\_object =>  
 feature\_definition =>  
 general\_feature

**6.1.8.12 property\_value\_representation**

AIM element: representation

Source: ISO 10303-43

**6.1.8.12.1 qualifier**

AIM element: type\_qualifier.name

Source: ISO 10303-45

Rules: dependent\_instantiable\_type\_qualifier

Reference path: representation  
 representation.items[i] ->  
 representation\_item =>  
 {representation\_item.name = 'qualifier'}  
 qualified\_representation\_item  
 qualified\_representation\_item.qualifiers[i] ->  
 value\_qualifier  
 value\_qualifier = type\_qualifier  
 type\_qualifier  
 type\_qualifier.name  
 {(type\_qualifier.name = 'nominal')  
 (type\_qualifier.name = 'specified')  
 (type\_qualifier.name = 'typical')}

**6.1.8.12.2 value\_determination**

AIM element: type\_qualifier.name

Source: ISO 10303-45

Rules: dependent\_instantiable\_type\_qualifier

Reference path: representation  
 representation.items[i] ->  
 representation\_item =>  
 {representation\_item.name = 'value interpretation'}  
 qualified\_representation\_item  
 qualified\_representation\_item.qualifiers[i] ->  
 value\_qualifier  
 value\_qualifier = type\_qualifier  
 type\_qualifier  
 type\_qualifier.name  
 {(type\_qualifier.name)  
 (type\_qualifier.name = 'calculated')  
 (type\_qualifier.name = 'designed')  
 (type\_qualifier.name = 'estimated')  
 (type\_qualifier.name = 'measured')  
 (type\_qualifier.name = 'required')  
 (type\_qualifier.name = 'set point')}

#### 6.1.8.12.3 property\_value\_representation to property (as definition)

AIM element: PATH

Reference path: representation <-  
 property\_definition\_representation.used\_representation  
 property\_definition\_representation  
 property\_definition\_representation.definition ->  
 represented\_definition  
 represented\_definition = property\_definition  
 property\_definition  
 derived\_property\_select = property\_definition  
 derived\_property\_select <-  
 general\_property\_association.derived\_definition  
 general\_property\_association  
 general\_property\_association.base\_definition ->  
 general\_property

#### 6.1.8.12.4 property\_value\_representation to unit (as global\_unit)

#1: If the unit has a commonly used descriptor.

#2: If the unit descriptor is constructed of more than one unit name.

AIM element: PATH

Reference path: representation  
 representation.context\_of\_items ->  
 representation\_context =>  
 global\_unit\_assigned\_context  
 global\_unit\_assigned\_context.units[i] ->  
 unit  
 #1: (unit = named\_unit  
 named\_unit)  
 #2: (unit = derived\_unit  
 derived\_unit)

#### 6.1.8.12.5 property\_value\_representation to property\_value (as specified\_value)

#1: If the specified\_value is a numerical\_value or a value\_limit and the unit is not assigned globally.

#2: If the specified\_value is a value\_range.

#3: If the specified\_value is a string\_value.

#4: If no significant\_digits are given for specified\_value.

#5: If significant\_digits are given for specified\_value.

#6: If the specified\_value is a numerical\_value or a value\_limit and the unit is specified globally.

#7: If the specified\_value is a value\_list.

AIM element: PATH

Reference path: representation  
representation.items[i] ->  
representation\_item =>  
#3: (descriptive\_representation\_item  
{descriptive\_representation\_item <=  
representation\_item =>  
qualified\_representation\_item  
qualified\_representation\_item.qualifiers[1] ->  
value\_qualifier  
value\_qualifier = type\_qualifier  
type\_qualifier  
type\_qualifier.name  
type\_qualifier.name = 'string'})  
#1: (#4: (measure\_representation\_item)  
#5: ([measure\_representation\_item]  
[qualified\_representation\_item]))  
#2: (#4: (compound\_representation\_item =>  
value\_range)  
#5: ([compound\_representation\_item =>  
value\_range]  
[qualified\_representation\_item]))  
#6: (#4: (value\_representation\_item)  
#5: ([value\_representation\_item]  
[qualified\_representation\_item]))  
#7: (compound\_representation\_item)

#### 6.1.8.13 property\_value\_representation\_relationship

AIM element: representation\_relationship

Source: ISO 10303-43

##### 6.1.8.13.1 description

AIM element: representation\_relationship.description

Source: ISO 10303-43

##### 6.1.8.13.2 relation\_type

AIM element: representation\_relationship.name

Source: ISO 10303-43

**6.1.8.13.3 property\_value\_representation\_relationship to multi\_language\_string (as description)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([representation\_relationship.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-43

Rules: restrict\_multi\_language\_for\_representation\_relationship

Reference path: representation\_relationship  
#1: (attribute\_language\_item = representation\_relationship  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#2: (multi\_language\_attribute\_item = representation\_relationship  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

**6.1.8.13.4 property\_value\_representation\_relationship to property\_value\_representation (as related)**

AIM element: PATH

Reference path: representation\_relationship  
representation\_relationship.rep\_2 ->  
representation

**6.1.8.13.5 property\_value\_representation\_relationship to property\_value\_representation (as relating)**

AIM element: PATH

Reference path: representation\_relationship  
representation\_relationship.rep\_1 ->  
representation

**6.1.8.14 substrate**

AIM element: product\_definition

Source: ISO 10303-41

Rules: substrate\_requires\_product\_definition,  
restrict\_product\_definition\_context\_for\_product

Reference path: {product\_definition  
[product\_definition.frame\_of\_reference ->  
product\_definition\_context <=

```

application_context_element
application_context_element.name = 'material definition']
[product_definition.formation ->
product_definition_formation
product_definition_formation.of_product ->
product <-
product_related_product_category.products[i]
product_related_product_category <=
product_category
product_category.name = 'substrate']}]
    
```

#### 6.1.8.14.1 name

AIM element: PATH

Reference path: product\_definition <-  
 material\_designation.definitions[i]  
 material\_designation  
 material\_designation.name

#### 6.1.8.15 workpiece\_feature

AIM element: general\_feature

Source: ISO 10303-214

Reference path: {general\_feature <=  
 [shape\_aspect  
 shape\_aspect.of\_shape ->  
 product\_definition\_shape <=  
 property\_definition  
 property\_definition.definition ->  
 characterized\_definition  
 characterized\_definition = characterized\_product\_definition  
 characterized\_product\_definition  
 characterized\_product\_definition = product\_definition  
 product\_definition product\_definition.formation ->  
 product\_definition\_formation  
 product\_definition\_formation.of\_product ->  
 product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=  
 product\_category  
 product\_category.name = 'workpiece']  
 [feature\_definition <=  
 characterized\_object]}

#### 6.1.8.15.1 description

AIM element: characterized\_object.description

Source: ISO 10303-41

#### 6.1.8.15.2 id

AIM element: characterized\_object.name

Source: ISO 10303-41

**6.1.8.15.3 workpiece\_feature to multi\_language\_string (as description)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([characterized\_object.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_general\_feature

Reference path: general\_feature  
#1: (attribute\_language\_item = general\_feature  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#2: (multi\_language\_attribute\_item = general\_feature  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

**6.1.8.15.4 workpiece\_feature to external\_model (as representation)**

AIM element: applied\_document\_reference

Source: ISO 10303-214

Reference path: {applied\_document\_reference <=  
document\_reference  
[applied\_document\_reference.items[i] ->  
document\_reference\_item  
document\_reference\_item = general\_feature  
general\_feature]  
[document\_reference.assigned\_document ->  
document\_file <=  
characterized\_object  
characterized\_definition = characterized\_object  
characterized\_definition <-  
property\_definition.definition  
property\_definition  
{property\_definition.name = 'external definition'}  
represented\_definition = property\_definition  
represented\_definition <-  
property\_definition\_representation.definition  
property\_definition\_representation.used\_representation ->  
representation  
{representation.context\_of\_items ->  
representation\_context  
representation\_context.context\_type = 'external'}}}

### 6.1.9 product\_management\_data UoF

#### 6.1.9.1 alias\_identification

AIM element: applied\_identification\_assignment

Source: ISO 10303-214

Reference path: applied\_identification\_assignment <=  
identification\_assignment  
{identification\_assignment  
identification\_assignment.role ->  
identification\_role  
identification\_role.name = 'alias'}

##### 6.1.9.1.1 alias\_id

AIM element: identification\_assignment.assigned\_id

Source: ISO 10303-41

Reference path: applied\_identification\_assignment <=  
identification\_assignment  
identification\_assignment.assigned\_id

##### 6.1.9.1.2 alias\_version\_id

AIM element: identification\_assignment.assigned\_id

Source: ISO 10303-41

Rules: dependent\_instantiable\_identification\_role,  
restrict\_version\_assignment\_for\_applied\_identification\_assignment

Reference path: applied\_identification\_assignment  
identification\_item = applied\_identification\_assignment  
identification\_item <-  
applied\_identification\_assignment.items[i]  
applied\_identification\_assignment <=  
identification\_assignment  
{identification\_assignment.role ->  
identification\_role  
identification\_role.name = 'version'}  
identification\_assignment.assigned\_id

##### 6.1.9.1.3 description

AIM element: identification\_role.description

Source: ISO 10303-41

Rules: dependent\_instantiable\_identification\_role

Reference path: applied\_identification\_assignment <=  
identification\_assignment  
identification\_assignment.role ->  
identification\_role  
identification\_role.description

**6.1.9.1.4 alias\_identification to organization (as alias\_scope)**

AIM element: PATH

Reference path: applied\_identification\_assignment  
 organization\_item = applied\_identification\_assignment  
 organization\_item <-  
 applied\_organization\_assignment.items[i]  
 applied\_organization\_assignment <=  
 organization\_assignment  
 {organization\_assignment.role ->  
 organization\_role  
 organization\_role.name = 'alias scope'}  
 organization\_assignment.assigned\_organization ->  
 organization

**6.1.9.1.5 alias\_identification to multi\_language\_string (as description)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([identification\_role.description]  
 [PATH])  
 #2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_identification\_role

Reference path: identification\_role  
 #1: (attribute\_language\_item = identification\_role  
 attribute\_language\_item <-  
 attribute\_language\_assignment.items[i]  
 attribute\_language\_assignment  
 {attribute\_language\_assignment <=  
 attribute\_classification\_assignment  
 attribute\_classification\_assignment.attribute\_name = 'description'})  
 #2: (multi\_language\_attribute\_item = identification\_role  
 multi\_language\_attribute\_item <-  
 multi\_language\_attribute\_assignment.items[i]  
 multi\_language\_attribute\_assignment  
 {multi\_language\_attribute\_assignment <=  
 attribute\_value\_assignment  
 attribute\_value\_assignment.attribute\_name = 'description'})

**6.1.9.1.6 alias\_identification to classification\_attribute (as is\_applied\_to)**

AIM element: PATH

Reference path: applied\_identification\_assignment  
 applied\_identification\_assignment.items[i] ->  
 identification\_item  
 identification\_item = property\_definition  
 property\_definition

**6.1.9.1.7 alias\_identification to classification\_system (as is\_applied\_to)**

AIM element: PATH

Reference path: applied\_identification\_assignment  
applied\_identification\_assignment.items[i] ->  
identification\_item  
identification\_item = class\_system  
class\_system

**6.1.9.1.8 alias\_identification to document (as is\_applied\_to)**

AIM element: PATH

Reference path: applied\_identification\_assignment  
applied\_identification\_assignment.items[i] ->  
identification\_item  
identification\_item = product  
product  
{product <-  
product\_related\_product\_category.products[i]  
product\_related\_product\_category <=  
product\_category  
product\_category.name = 'document'}

**6.1.9.1.9 alias\_identification to document\_representation (as is\_applied\_to)**

AIM element: PATH

Reference path: applied\_identification\_assignment  
applied\_identification\_assignment.items[i] ->  
identification\_item  
identification\_item = product\_definition  
product\_definition

**6.1.9.1.10 alias\_identification to document\_type\_property (as is\_applied\_to)**

AIM element: PATH

Reference path: applied\_identification\_assignment  
applied\_identification\_assignment.items[i] ->  
identification\_item  
identification\_item = document\_type  
document\_type

**6.1.9.1.11 alias\_identification to document\_version (as is\_applied\_to)**

AIM element: PATH

Reference path: applied\_identification\_assignment  
applied\_identification\_assignment.items[i] ->  
identification\_item  
identification\_item = product\_definition\_formation  
product\_definition\_formation  
{product\_definition\_formation  
product\_definition\_formation.of\_product ->  
product <-  
product\_related\_product\_category.products[i]  
product\_related\_product\_category <=  
product\_category  
product\_category.name = 'document'}

**6.1.9.1.12 alias\_identification to general\_classification (as is\_applied\_to)**

AIM element: PATH

Reference path: applied\_identification\_assignment  
 applied\_identification\_assignment.items[i] ->  
 identification\_item  
 identification\_item = class  
 class

**6.1.9.1.13 alias\_identification to grade (as is\_applied\_to)**

AIM element: PATH

Reference path: applied\_identification\_assignment  
 applied\_identification\_assignment.items[i] ->  
 identification\_item  
 identification\_item = product\_definition  
 product\_definition  
 {product\_definition  
 [product\_definition.frame\_of\_reference ->  
 product\_definition\_context <=  
 application\_context\_element  
 application\_context\_element.name = 'material definition']  
 [product\_definition\_formation ->  
 product\_definition\_formation  
 product\_definition\_formation.of\_product ->  
 product <-  
 product\_related\_product\_category.products[i]  
 product\_related\_product\_category <=  
 product\_category  
 product\_category.name = 'grade']}

**6.1.9.1.14 alias\_identification to item (as is\_applied\_to)**

AIM element: PATH

Reference path: applied\_identification\_assignment  
 applied\_identification\_assignment.items[i] ->  
 identification\_item  
 identification\_item = product  
 product

**6.1.9.1.15 alias\_identification to item\_definition (as is\_applied\_to)**

AIM element: PATH

Reference path: applied\_identification\_assignment  
 applied\_identification\_assignment.items[i] ->  
 identification\_item  
 identification\_item = product\_definition  
 product\_definition

**6.1.9.1.16 alias\_identification to item\_instance (as is\_applied\_to)**

AIM element: PATH

Reference path: applied\_identification\_assignment  
applied\_identification\_assignment.items[i] ->  
identification\_item  
identification\_item = product\_definition\_relationship  
product\_definition\_relationship =>  
product\_definition\_usage =>  
assembly\_component\_usage

#### 6.1.9.1.17 alias\_identification to item\_version (as is\_applied\_to)

AIM element: PATH

Reference path: applied\_identification\_assignment  
applied\_identification\_assignment.items[i] ->  
identification\_item  
identification\_item = product\_definition\_formation  
product\_definition\_formation

#### 6.1.9.1.18 alias\_identification to organization (as is\_applied\_to)

AIM element: PATH

Reference path: applied\_identification\_assignment  
applied\_identification\_assignment.items[i] ->  
identification\_item  
identification\_item = organization  
organization

#### 6.1.9.1.19 alias\_identification to property (as is\_applied\_to)

AIM element: PATH

Reference path: applied\_identification\_assignment  
applied\_identification\_assignment.items[i] ->  
identification\_item  
identification\_item = general\_property  
general\_property

#### 6.1.9.2 application\_context

AIM element: product\_definition\_context

Source: ISO 10303-41

##### 6.1.9.2.1 application\_domain

AIM element: application\_context.application

Source: ISO 10303-41

Reference path: product\_definition\_context <=  
application\_context\_element  
application\_context\_element.frame\_of\_reference ->  
application\_context  
application\_context.application  
{(application\_context.application)  
(application\_context.application = 'machining')  
(application\_context.application = 'manufacturing simulation')}

```
(application_context.application = 'nc preparation')
(application_context.application = 'process planning')
(application_context.application = 'product planning')
(application_context.application = 'purchasing')}
```

#### 6.1.9.2.2 description

AIM element: application\_context.description

Source: ISO 10303-41

Reference path: product\_definition\_context <=  
application\_context\_element  
application\_context\_element.frame\_of\_reference ->  
application\_context  
application\_context.description

#### 6.1.9.2.3 application\_context to multi\_language\_string (as description)

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([application\_context.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_application\_context

Reference path: product\_definition\_context <=  
application\_context\_element  
application\_context\_element.frame\_of\_reference ->  
application\_context  
#1: (attribute\_language\_item = application\_context  
attribute\_language\_item <=  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#2: (multi\_language\_attribute\_item = application\_context  
multi\_language\_attribute\_item <=  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

#### 6.1.9.3 date\_time

#1: If only a certain day is known.

#2: If a certain day and the time of day is known.

AIM element: #1: (calendar\_date)  
#2: (date\_and\_time)

## ISO 13399-1:2006(E)

Source: ISO 10303-41  
ISO 10303-41

Rules: dependent\_instantiable\_date,  
dependent\_instantiable\_date\_and\_time

### 6.1.9.3.1 date

#1: If only a certain day is known.

#2: If a certain day and the time of day is known.

AIM element: #1: (IDENTICAL MAPPING)  
#2: (calendar\_date)

Reference path: #2: (date\_and\_time  
date\_and\_time.date\_component ->  
date =>  
calendar\_date)

### 6.1.9.3.2 time

AIM element: local\_time

Source: ISO 10303-41

Reference path: date\_and\_time  
date\_and\_time.time\_component ->  
local\_time

### 6.1.9.4 item

AIM element: product

Source: ISO 10303-41

Rules: product\_requires\_category,  
product\_requires\_id\_owner,  
product\_requires\_version,  
restrict\_product\_category\_for\_product

Reference path: {product <-  
product\_related\_product\_category.products[i]  
product\_related\_product\_category <=  
product\_category  
product\_category.name  
(product\_category.name = 'part')  
(product\_category.name = 'raw material')  
(product\_category.name = 'tool')}

#### 6.1.9.4.1 description

AIM element: product.description

Source: ISO 10303-41

**6.1.9.4.2 id**

AIM element: product.id

Source: ISO 10303-41

**6.1.9.4.3 name**

AIM element: product.name

Source: ISO 10303-41

**6.1.9.4.4 item to multi\_language\_string (as description)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([product.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product

Reference path: product  
#1: (attribute\_language\_item = product  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#2: (multi\_language\_attribute\_item = product  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

**6.1.9.4.5 item to multi\_language\_string (as name)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([product.name]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product

Reference path: product  
#1: (attribute\_language\_item = product

```

attribute_language_item <-
attribute_language_assignment.items[i]
attribute_language_assignment
{attribute_language_assignment <=
attribute_classification_assignment
attribute_classification_assignment.attribute_name = 'name'})
#2: (multi_language_attribute_item = product
multi_language_attribute_item <-
multi_language_attribute_assignment.items[i]
multi_language_attribute_assignment
{multi_language_attribute_assignment <=
attribute_value_assignment
attribute_value_assignment.attribute_name = 'name'})

```

#### 6.1.9.5 item\_definition

AIM element: product\_definition

Source: ISO 10303-41

Rules: restrict\_product\_definition\_context\_for\_product

Reference path: product\_definition  
 {product\_definition.frame\_of\_reference ->  
 product\_definition\_context <=  
 application\_context\_element  
 application\_context\_element.name = 'part definition'}

##### 6.1.9.5.1 id

AIM element: product\_definition.id

Source: ISO 10303-41

##### 6.1.9.5.2 name

AIM element: product\_definition.description

Source: ISO 10303-41

##### 6.1.9.5.3 item\_definition to application\_context (as additional\_context)

AIM element: PATH

Reference path: product\_definition <-  
 product\_definition\_context\_association.definition  
 product\_definition\_context\_association  
 {product\_definition\_context\_association.role  
 product\_definition\_context\_role.name = 'additional context'}  
 product\_definition\_context\_association.frame\_of\_reference ->  
 product\_definition\_context

##### 6.1.9.5.4 item\_definition to item\_version (as associated\_item\_version)

AIM element: PATH

Reference path: product\_definition  
 product\_definition.formation ->  
 product\_definition\_formation

#### 6.1.9.5.5 item\_definition to application\_context (as initial\_context)

AIM element: PATH

Reference path: product\_definition  
 product\_definition.frame\_of\_reference ->  
 product\_definition\_context

#### 6.1.9.5.6 item\_definition to multi\_language\_string (as name)

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([product\_definition.description]  
 [PATH])  
 #2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product\_definition

Reference path: product\_definition  
 #1: (attribute\_language\_item = product\_definition  
 attribute\_language\_item <-  
 attribute\_language\_assignment.items[i]  
 attribute\_language\_assignment  
 {attribute\_language\_assignment <=  
 attribute\_classification\_assignment  
 attribute\_classification\_assignment.attribute\_name = 'description'})  
 #2: (multi\_language\_attribute\_item = product\_definition  
 multi\_language\_attribute\_item <-  
 multi\_language\_attribute\_assignment.items[i]  
 multi\_language\_attribute\_assignment  
 {multi\_language\_attribute\_assignment <=  
 attribute\_value\_assignment  
 attribute\_value\_assignment.attribute\_name = 'description'})

#### 6.1.9.6 item\_version

AIM element: product\_definition\_formation

Source: ISO 10303-41

##### 6.1.9.6.1 description

AIM element: product\_definition\_formation.description

Source: ISO 10303-41

##### 6.1.9.6.2 id

AIM element: product\_definition\_formation.id

Source: ISO 10303-41

**6.1.9.6.3 item\_version to item (as associated\_item)**

AIM element: PATH

Reference path: product\_definition\_formation  
product\_definition\_formation.of\_product ->  
product

**6.1.9.6.4 item\_version to multi\_language\_string (as description)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([product\_definition\_formation.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product\_definition\_formation

Reference path: product\_definition\_formation  
#1: (attribute\_language\_item = product\_definition\_formation  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#2: (multi\_language\_attribute\_item = product\_definition\_formation  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

**6.1.9.7 item\_version relationship**

AIM element: product\_definition\_formation\_relationship

Source: ISO 10303-41

**6.1.9.7.1 description**

AIM element: product\_definition\_formation\_relationship.description

Source: ISO 10303-41

**6.1.9.7.2 relation\_type**

AIM element: product\_definition\_formation\_relationship.name

Source: ISO 10303-41

Reference path: {product\_definition\_formation\_relationship  
 (product\_definition\_formation\_relationship.name)  
 (product\_definition\_formation\_relationship.name = 'derivation')  
 (product\_definition\_formation\_relationship.name = 'hierarchy')  
 (product\_definition\_formation\_relationship.name = 'sequence')  
 (product\_definition\_formation\_relationship.name = 'supplied item')}

#### 6.1.9.7.3 item\_version\_relationship to multi\_language\_string (as description)

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string. [D:\Work\Documents\Standards\STEP\10303-214\html\\_with\\_change\\_marks\htmls\clause5\aim\\_index\pxref.htm](D:\Work\Documents\Standards\STEP\10303-214\html_with_change_marks\htmls\clause5\aim_index\pxref.htm)

AIM element: #1: ([product\_definition\_formation\_relationship.description]  
 [PATH])  
 #2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product\_definition\_formation\_relationship

Reference path: product\_definition\_formation\_relationship  
 #1: (attribute\_language\_item = product\_definition\_formation\_relationship  
 attribute\_language\_item <-  
 attribute\_language\_assignment.items[i]  
 attribute\_language\_assignment  
 {attribute\_language\_assignment <=  
 attribute\_classification\_assignment  
 attribute\_classification\_assignment.attribute\_name = 'description'})  
 #2: (multi\_language\_attribute\_item = product\_definition\_formation\_relationship  
 multi\_language\_attribute\_item <-  
 multi\_language\_attribute\_assignment.items[i]  
 multi\_language\_attribute\_assignment  
 {multi\_language\_attribute\_assignment <=  
 attribute\_value\_assignment  
 attribute\_value\_assignment.attribute\_name = 'description'})

#### 6.1.9.7.4 item\_version\_relationship to item\_version (as related)

AIM element: PATH

Reference path: product\_definition\_formation\_relationship  
 product\_definition\_formation\_relationship.related\_product\_definition\_formation ->  
 product\_definition\_formation

#### 6.1.9.7.5 item\_version\_relationship to item\_version (as relating)

AIM element: PATH

Reference path: product\_definition\_formation\_relationship  
 product\_definition\_formation\_relationship.relatng\_product\_definition\_formation ->  
 product\_definition\_formation

#### 6.1.9.8 language

AIM element: language

## ISO 13399-1:2006(E)

Source: ISO 10303-214

Reference path: language <=  
group

### 6.1.9.8.1 country\_code

AIM element: group.description

Source: ISO 10303-41

Reference path: language <=  
group  
group.description

### 6.1.9.8.2 language\_code

AIM element: group.name

Source: ISO 10303-41

Reference path: language <=  
group  
group.name

### 6.1.9.9 multi\_language\_string

#1: If additional\_language\_dependent\_string is not populated.

#2: If additional\_language\_dependent\_string is populated.

AIM element: #1: (attribute\_language\_assignment)  
#2: ([attribute\_language\_assignment]  
[multi\_language\_attribute\_assignment])

Source: ISO 10303-214  
ISO 10303-214  
ISO 10303-214

Rules: dependent\_instantiable\_attribute\_value\_role

Reference path: #1: (attribute\_language\_assignment <=  
attribute\_classification\_assignment  
{attribute\_classification\_assignment.role ->  
classification\_role  
classification\_role.name = 'primary'})  
#2: ([attribute\_language\_assignment <=  
attribute\_classification\_assignment  
{attribute\_classification\_assignment.role ->  
classification\_role  
classification\_role.name = 'primary'})  
[multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
{attribute\_value\_assignment.role ->  
attribute\_value\_role  
attribute\_value\_role.name = 'alternate language'}  
attribute\_language\_item = attribute\_value\_assignment

```

attribute_language_item <-
attribute_language_assignment.items[i]
attribute_language_assignment <=
attribute_classification_assignment
{[attribute_classification_assignment.role ->
classification_role
classification_role.name = 'translated']
[attribute_classification_assignment.attribute_name = 'attribute_value']}]

```

#### 6.1.9.9.1 multi\_language\_string to string\_with\_language (as additional\_language\_dependent\_string)

AIM element: IDENTICAL MAPPING

#### 6.1.9.9.2 multi\_language\_string to string\_with\_language (as primary\_language\_dependent\_string)

AIM element: IDENTICAL MAPPING

#### 6.1.9.10 numerical\_value

#1: If significant digits are given for the value\_with\_unit.

#2: If no significant digits are given for the value\_with\_unit.

#3: If the unit is not assigned globally.

#4: If the unit is assigned globally.

#5: For the quantity of a quantified\_instance.

AIM element: #3: (#1: ([measure\_representation\_item]  
[qualified\_representation\_item])  
#2: (measure\_representation\_item))  
#4: (#1: ([value\_representation\_item]  
[qualified\_representation\_item])  
#2: (value\_representation\_item))  
#5: ([measure\_representation\_item])

Source: ISO 10303-45  
ISO 10303-45  
ISO 10303-45  
ISO 10303-43  
ISO 10303-45  
ISO 10303-43  
ISO 10303-45

Rules: dependent\_instantiable\_measure\_with\_unit

#### 6.1.9.10.1 value\_component

#1: If the unit is not assigned globally.

#2: If the unit is assigned globally

AIM element: #1: (measure\_with\_unit.value\_component)  
#2: (value\_representation\_item.value\_component)

Source: ISO 10303-41  
ISO 10303-43

Reference path: #1: ({measure\_representation\_item <=  
measure\_with\_unit  
{(|measure\_with\_unit|)  
(measure\_with\_unit =>  
(amount\_of\_substance\_measure\_with\_unit)  
(area\_measure\_with\_unit)  
(celsius\_temperature\_measure\_with\_unit)  
(electric\_current\_measure\_with\_unit)  
(length\_measure\_with\_unit)  
(luminous\_intensity\_measure\_with\_unit)  
(mass\_measure\_with\_unit)  
(plane\_angle\_measure\_with\_unit)  
(ratio\_measure\_with\_unit)  
(solid\_angle\_measure\_with\_unit)  
(thermodynamic\_temperature\_measure\_with\_unit)  
(time\_measure\_with\_unit)  
(volume\_measure\_with\_unit))}  
measure\_with\_unit.value\_component  
{measure\_with\_unit.value\_component ->  
measure\_value  
(measure\_value = area\_measure  
area\_measure)  
(measure\_value = amount\_of\_substance\_measure  
amount\_of\_substance\_measure)  
(measure\_value = celsius\_temperature\_measure  
celsius\_temperature\_measure)  
(measure\_value = length\_measure  
length\_measure)  
(measure\_value = electric\_current\_measure  
electric\_current\_measure)  
(measure\_value = plane\_angle\_measure  
plane\_angle\_measure)  
(measure\_value = ratio\_measure  
ratio\_measure)  
(measure\_value = parameter\_value  
parameter\_value)  
(measure\_value = descriptive\_measure  
descriptive\_measure)  
(measure\_value = positive\_plane\_angle\_measure  
positive\_plane\_angle\_measure)  
(measure\_value = count\_measure  
count\_measure)  
(measure\_value = mass\_measure  
mass\_measure)  
(measure\_value = time\_measure  
time\_measure)  
(measure\_value = thermodynamic\_temperature\_measure  
thermodynamic\_temperature\_measure)  
(measure\_value = luminous\_intensity\_measure  
luminous\_intensity\_measure)  
(measure\_value = solid\_angle\_measure  
solid\_angle\_measure)  
(measure\_value = volume\_measure  
volume\_measure)  
(measure\_value = numeric\_measure  
numeric\_measure)  
(measure\_value = context\_dependent\_measure

```

context_dependent_measure)
(measure_value = positive_length_measure
positive_length_measure)
(measure_value = positive_ratio_measure
positive_ratio_measure)}})
#2: (value_representation_item
value_representation_item.value_component)

```

#### 6.1.9.11 organization

AIM element: organization

Source: ISO 10303-41

##### 6.1.9.11.1 delivery\_address

AIM element: organizational\_address

Source: ISO 10303-41

Reference path: organization <-  
organizational\_address.organizations[i]  
organizational\_address  
{organizational\_address.description = 'delivery address'}

##### 6.1.9.11.2 id

AIM element: organization.id

Source: ISO 10303-41

##### 6.1.9.11.3 organization\_name

AIM element: organization.name

Source: ISO 10303-41

##### 6.1.9.11.4 organization\_type

AIM element: organization.description

Source: ISO 10303-41

##### 6.1.9.11.5 postal\_address

AIM element: organizational\_address

Source: ISO 10303-41

Reference path: organization <-  
organizational\_address.organizations[i]  
organizational\_address  
{organizational\_address.description = 'postal address'}

##### 6.1.9.11.6 visitor\_address

AIM element: organizational\_address

## ISO 13399-1:2006(E)

Source: ISO 10303-41

Reference path: organization <-  
organizational\_address.organizations[i]  
organizational\_address  
{organizational\_address.description = 'visitor address'}

### 6.1.9.12 person

AIM element: person

Source: ISO 10303-41

Rules: person\_requires\_person\_and\_organization

Reference path: {person <-  
person\_and\_organization.the\_person  
person\_and\_organization}

#### 6.1.9.12.1 person\_name

AIM element: <person.last\_name>  
<person.first\_name>  
<person.middle\_names>

Source: ISO 10303-41  
ISO 10303-41  
ISO 10303-41

#### 6.1.9.12.2 preferred\_business\_address

AIM element: personal\_address

Source: ISO 10303-41

Reference path: person <-  
personal\_address.people[i]  
personal\_address

#### 6.1.9.13 person\_in\_organization

AIM element: person\_and\_organization

Source: ISO 10303-41

##### 6.1.9.13.1 id

AIM element: identification\_assignment.assigned\_id

Source: ISO 10303-41

Reference path: person\_and\_organization  
identification\_item = person\_and\_organization  
identification\_item <-  
applied\_identification\_assignment.items[i]  
applied\_identification\_assignment <=  
identification\_assignment  
{identification\_assignment.role ->

```

identification_role
identification_role.name = 'id'}
identification_assignment.assigned_id

```

#### 6.1.9.13.2 location

AIM element: person\_and\_organization\_address

Source: ISO 10303-214

Reference path: person\_and\_organization  
 [person\_and\_organization.the\_person ->  
 person <-  
 personal\_address.people[1]  
 personal\_address =>]  
 [person\_and\_organization.the\_organization ->  
 organization <-  
 organizational\_address.organizations[1]  
 organizational\_address =>]  
 person\_and\_organization\_address  
 {person\_and\_organization\_address <=  
 [organizational\_address]  
 [personal\_address]}

#### 6.1.9.13.3 role

AIM element: person\_and\_organization.name

Source: ISO 10303-41

#### 6.1.9.13.4 person\_in\_organization to organization (as associated\_organization)

AIM element: PATH

Reference path: person\_and\_organization  
 person\_and\_organization.the\_organization ->  
 organization

#### 6.1.9.13.5 person\_in\_organization to person (as associated\_person)

AIM element: PATH

Reference path: person\_and\_organization  
 person\_and\_organization.the\_person ->  
 person

#### 6.1.9.14 person\_organization\_assignment

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: #1: (applied\_person\_and\_organization\_assignment)  
 #2: (applied\_organization\_assignment)

Source: ISO 10303-214  
 ISO 10303-214

Reference path: #1: (applied\_person\_and\_organization\_assignment <=  
person\_and\_organization\_assignment)  
#2: (applied\_organization\_assignment <=  
organization\_assignment)

#### 6.1.9.14.1 description

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: #1: (person\_and\_organization\_role.description)  
#2: (organization\_role.description)

Source: ISO 10303-41  
ISO 10303-41

Rules: dependent\_instantiable\_organization\_role,  
dependent\_instantiable\_person\_and\_organization\_role

Reference path: #1: (applied\_person\_and\_organization\_assignment <=  
person\_and\_organization\_assignment  
person\_and\_organization\_assignment.role ->  
person\_and\_organization\_role  
person\_and\_organization\_role.description)  
#2: (applied\_organization\_assignment <=  
organization\_assignment  
organization\_assignment.role ->  
organization\_role  
organization\_role.description)

#### 6.1.9.14.2 role

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: #1: (person\_and\_organization\_role.description)  
#2: (organization\_role.description)

Source: ISO 10303-41  
ISO 10303-41

Rules: dependent\_instantiable\_organization\_role,  
dependent\_instantiable\_person\_and\_organization\_role

Reference path: #2: (applied\_person\_and\_organization\_assignment <=  
person\_and\_organization\_assignment  
person\_and\_organization\_assignment.role ->  
person\_and\_organization\_role  
person\_and\_organization\_role.name  
{(person\_and\_organization\_role.name)  
(person\_and\_organization\_role.name = 'author')  
(person\_and\_organization\_role.name = 'classification officer')  
(person\_and\_organization\_role.name = 'creator')  
(person\_and\_organization\_role.name = 'custodian')  
(person\_and\_organization\_role.name = 'customer')  
(person\_and\_organization\_role.name = 'design supplier')  
(person\_and\_organization\_role.name = 'editor')}

```

(person_and_organization_role.name = 'editor rights')
(person_and_organization_role.name = 'id owner')
(person_and_organization_role.name = 'location')
(person_and_organization_role.name = 'owner')
(person_and_organization_role.name = 'supplier'))
#3: (applied_organization_assignment <=
organization_assignment
organization_assignment.role ->
organization_role
organization_role.name
{(organization_role.name)
(organization_role.name = 'author')
(organization_role.name = 'classification officer')
(organization_role.name = 'creator')
(organization_role.name = 'custodian')
(organization_role.name = 'customer')
(organization_role.name = 'design supplier')
(organization_role.name = 'editor')
(organization_role.name = 'id owner')
(organization_role.name = 'location')
(organization_role.name = 'owner')
(organization_role.name = 'supplier'))}

```

#### 6.1.9.14.3 person\_organization\_assignment to organization (as assigned\_person\_organization)

AIM element: PATH

Reference path: applied\_organization\_assignment <=  
organization\_assignment  
organization\_assignment.assigned\_organization ->  
organization

#### 6.1.9.14.4 person\_organization\_assignment to person\_in\_organization (as assigned\_person\_organization)

AIM element: PATH

Reference path: applied\_person\_and\_organization\_assignment <=  
person\_and\_organization\_assignment  
person\_and\_organization\_assignment.assigned\_person\_and\_organization ->  
person\_and\_organization

#### 6.1.9.14.5 person\_organization\_assignment to multi\_language\_string (as description)

#1: If used for a person as member of an organization.

#2: If used for an organization.

#3: For the primary\_language\_dependent\_string.

#4: For any additional\_language\_dependent\_string.

AIM element: #1: (#3: ([person\_and\_organization\_role.description]  
[PATH])  
#4: (PATH))  
#2: (#3: ([organization\_role.description]  
[PATH])  
#4: (PATH))

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_organization\_role,  
restrict\_multi\_language\_for\_person\_and\_organization\_role

Reference path: #1: (applied\_person\_and\_organization\_assignment <=  
person\_and\_organization\_assignment  
person\_and\_organization\_assignment.role ->  
person\_and\_organization\_role  
#3: (attribute\_language\_item = person\_and\_organization\_role  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#4: (multi\_language\_attribute\_item = person\_and\_organization\_role  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'}))  
#2: (applied\_organization\_assignment <=  
organization\_assignment  
organization\_assignment.role ->  
organization\_role  
#3: (attribute\_language\_item = organization\_role  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#4: (multi\_language\_attribute\_item = organization\_role  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'}))

**6.1.9.14.6 person\_organization\_assignment to classification\_association (as is\_applied\_to)**

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

Reference path: #1: (applied\_person\_and\_organization\_assignment  
applied\_person\_and\_organization\_assignment.items[i] ->  
person\_and\_organization\_item  
person\_and\_organization\_item = applied\_classification\_assignment)  
#2: (applied\_organization\_assignment  
applied\_organization\_assignment.items[i] ->  
organization\_item  
organization\_item = applied\_classification\_assignment)

applied\_classification\_assignment <=  
classification\_assignment

#### 6.1.9.14.7 person\_organization\_assignment to classification\_system (as is\_applied\_to)

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

Reference path: #1: (applied\_person\_and\_organization\_assignment  
applied\_person\_and\_organization\_assignment.items[i] ->  
person\_and\_organization\_item  
person\_and\_organization\_item = class\_system)  
#2: (applied\_organization\_assignment  
applied\_organization\_assignment.items[i] ->  
organization\_item  
organization\_item = class\_system)  
class\_system

#### 6.1.9.14.8 person\_organization\_assignment to document (as is\_applied\_to)

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

Reference path: #1: (applied\_person\_and\_organization\_assignment  
applied\_person\_and\_organization\_assignment.items[i] ->  
person\_and\_organization\_item  
person\_and\_organization\_item = product)  
#2: (applied\_organization\_assignment  
applied\_organization\_assignment.items[i] ->  
organization\_item  
organization\_item = product)  
product  
{product <-  
product\_related\_product\_category.products[i]  
product\_related\_product\_category <=  
product\_category  
product\_category.name = 'document'}

#### 6.1.9.14.9 person\_organization\_assignment to document\_file (as is\_applied\_to)

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

Reference path: #1: (applied\_person\_and\_organization\_assignment  
applied\_person\_and\_organization\_assignment.items[i] ->  
person\_and\_organization\_item  
person\_and\_organization\_item = document\_file)  
#2: (applied\_organization\_assignment  
applied\_organization\_assignment.items[i] ->  
organization\_item

organization\_item = document\_file)  
document\_file

**6.1.9.14.10 person\_organization\_assignment to document\_representation (as is\_applied\_to)**

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

Reference path: #1: (applied\_person\_and\_organization\_assignment  
applied\_person\_and\_organization\_assignment.items[i] ->  
person\_and\_organization\_item  
person\_and\_organization\_item = product\_definition  
product\_definition)  
#2: (applied\_organization\_assignment  
applied\_organization\_assignment.items[i] ->  
organization\_item  
organization\_item = product\_definition  
product\_definition)

**6.1.9.14.11 person\_organization\_assignment to document\_version (as is\_applied\_to)**

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

Reference path: #1: (applied\_person\_and\_organization\_assignment  
applied\_person\_and\_organization\_assignment.items[i] ->  
person\_and\_organization\_item  
person\_and\_organization\_item = product\_definition\_formation)  
#2: (applied\_organization\_assignment  
applied\_organization\_assignment.items[i] ->  
organization\_item  
organization\_item = product\_definition\_formation)  
product\_definition\_formation  
{product\_definition\_formation.of\_product ->  
product <-  
product\_related\_product\_category.products[i]  
product\_related\_product\_category <=  
product\_category  
product\_category.name = 'document'}

**6.1.9.14.12 person\_organization\_assignment to general\_classification (as is\_applied\_to)**

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

Reference path: #1: (applied\_person\_and\_organization\_assignment  
applied\_person\_and\_organization\_assignment.items[i] ->  
person\_and\_organization\_item  
person\_and\_organization\_item = class)  
#2: (applied\_organization\_assignment

```

applied_organization_assignment.items[i] ->
organization_item
organization_item = class)
class

```

#### 6.1.9.14.13 person\_organization\_assignment to item (as is\_applied\_to)

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

```

Reference path: #1: (applied_person_and_organization_assignment
applied_person_and_organization_assignment.items[i] ->
person_and_organization_item
person_and_organization_item = product)
#2: (applied_organization_assignment
applied_organization_assignment.items[i] ->
organization_item
organization_item = product)
product

```

#### 6.1.9.14.14 person\_organization\_assignment to item\_definition (as is\_applied\_to)

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

```

Reference path: #1: (applied_person_and_organization_assignment
applied_person_and_organization_assignment.items[i] ->
person_and_organization_item
person_and_organization_item = product_definition)
#2: (applied_organization_assignment
applied_organization_assignment.items[i] ->
organization_item
organization_item = product_definition)
product_definition
{product_definition.frame_of_reference ->
product_definition_context <=
application_context_element
application_context_element.name = 'part definition'}

```

#### 6.1.9.14.15 person\_organization\_assignment to item\_structure\_association (as is\_applied\_to)

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

```

Reference path: #1: (applied_person_and_organization_assignment
applied_person_and_organization_assignment.items[i] ->
person_and_organization_item
person_and_organization_item = product_definition_relationship)
#2: (applied_organization_assignment
applied_organization_assignment.items[i] ->

```

organization\_item  
organization\_item = product\_definition\_relationship)  
product\_definition\_relationship

**6.1.9.14.16 person\_organization\_assignment to item\_instance (as is\_applied\_to)**

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

Reference path: #1: (applied\_person\_and\_organization\_assignment  
applied\_person\_and\_organization\_assignment.items[i] ->  
person\_and\_organization\_item  
person\_and\_organization\_item = product\_definition\_relationship)  
#2: (applied\_organization\_assignment  
applied\_organization\_assignment.items[i] ->  
organization\_item  
organization\_item = product\_definition\_relationship)  
product\_definition\_relationship =>  
product\_definition\_usage =>  
assembly\_component\_usage

**6.1.9.14.17 person\_organization\_assignment to item\_version (as is\_applied\_to)**

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

Reference path: #1: (applied\_person\_and\_organization\_assignment  
applied\_person\_and\_organization\_assignment.items[i] ->  
person\_and\_organization\_item  
person\_and\_organization\_item = product\_definition\_formation)  
#2: (applied\_organization\_assignment  
applied\_organization\_assignment.items[i] ->  
organization\_item  
organization\_item = product\_definition\_formation)  
product\_definition\_formation

**6.1.9.14.18 person\_organization\_assignment to item\_version\_relationship (as is\_applied\_to)**

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

Reference path: #1: (applied\_person\_and\_organization\_assignment  
applied\_person\_and\_organization\_assignment.items[i] ->  
person\_and\_organization\_item  
person\_and\_organization\_item = product\_definition\_formation\_relationship)  
#2: (applied\_organization\_assignment  
applied\_organization\_assignment.items[i] ->  
organization\_item  
organization\_item = product\_definition\_formation\_relationship)  
product\_definition\_formation\_relationship

**6.1.9.14.19 person\_organization\_assignment to material\_designation (as is\_applied\_to)**

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

Reference path: #1: (applied\_person\_and\_organization\_assignment  
applied\_person\_and\_organization\_assignment.items[i] ->  
person\_and\_organization\_item  
person\_and\_organization\_item = material\_designation)  
#2: (applied\_organization\_assignment  
applied\_organization\_assignment.items[i] ->  
organization\_item  
organization\_item = material\_designation)  
material\_designation

**6.1.9.14.20 person\_organization\_assignment to person\_in\_organization (as is\_applied\_to)**

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

Reference path: #1: (applied\_person\_and\_organization\_assignment  
applied\_person\_and\_organization\_assignment.items[i] ->  
person\_and\_organization\_item  
person\_and\_organization\_item = person\_and\_organization)  
#2: (applied\_organization\_assignment  
applied\_organization\_assignment.items[i] ->  
organization\_item  
organization\_item = person\_and\_organization)  
person\_and\_organization

**6.1.9.14.21 person\_organization\_assignment to physical\_item\_structure\_association (as is\_applied\_to)**

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

Reference path: #1: (applied\_person\_and\_organization\_assignment  
applied\_person\_and\_organization\_assignment.items[i] ->  
person\_and\_organization\_item  
person\_and\_organization\_item = product\_definition\_relationship)  
#2: (applied\_organization\_assignment  
applied\_organization\_assignment.items[i] ->  
organization\_item  
organization\_item = product\_definition\_relationship)  
product\_definition\_relationship

**6.1.9.14.22 person\_organization\_assignment to property (as is\_applied\_to)**

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

Reference path: #1: (applied\_person\_and\_organization\_assignment  
applied\_person\_and\_organization\_assignment.items[i] ->  
person\_and\_organization\_item  
person\_and\_organization\_item = general\_property)  
#2: (applied\_organization\_assignment  
applied\_organization\_assignment.items[i] ->  
organization\_item  
organization\_item = general\_property)  
general\_property

#### 6.1.9.14.23 person\_organization\_assignment to property\_value\_association (as is\_applied\_to)

#1: If used for a person as member of an organization.

#2: If used for an organization.

AIM element: PATH

Reference path: #1: (applied\_person\_and\_organization\_assignment  
applied\_person\_and\_organization\_assignment.items[i] ->  
person\_and\_organization\_item  
person\_and\_organization\_item = property\_definition  
property\_definition)  
#2: (applied\_organization\_assignment  
applied\_organization\_assignment.items[i] ->  
organization\_item  
organization\_item = property\_definition  
property\_definition)

#### 6.1.9.15 property\_value

#1: If property\_value is a value\_list.

#2: If property\_value is a string\_value.

#3: If property\_value is a numerical\_value or a value\_limit and the unit is not assigned globally.

#4: If property\_value is a value\_range.

#5: If no significant\_digits are given for the value\_with\_unit.

#6: If significant\_digits are given for the value\_with\_unit.

#7: If the property\_value is a numerical\_value or a value\_limit and the unit is specified globally.

AIM element: #1: (compound\_representation\_item)  
#2: (descriptive\_representation\_item)  
#3: (#5: (measure\_representation\_item)  
#6: ([measure\_representation\_item]  
[qualified\_representation\_item]))  
#4: (#5: (value\_range)  
#6: ([value\_range]  
[qualified\_representation\_item]))  
#7: (#5: (value\_representation\_item)

#6: ([value\_representation\_item]  
[qualified\_representation\_item]))

Source: ISO 10303-43  
ISO 10303-45  
ISO 10303-45  
ISO 10303-45  
ISO 10303-45  
ISO 10303-214  
ISO 10303-214  
ISO 10303-45  
ISO 10303-43  
ISO 10303-43  
ISO 10303-45

Reference path: #2: (descriptive\_representation\_item  
{descriptive\_representation\_item <=  
representation\_item =>  
qualified\_representation\_item  
qualified\_representation\_item.qualifiers[1] ->  
value\_qualifier  
value\_qualifier = type\_qualifier  
type\_qualifier  
type\_qualifier.name  
type\_qualifier.name = 'string'})  
#4: (#5: (value\_range <=  
compound\_representation\_item)  
#6: ([value\_range <=  
compound\_representation\_item]  
[qualified\_representation\_item]))

#### 6.1.9.15.1 value\_name

#1: If property\_value is a value\_list.

#2: If property\_value is a string\_value.

#3: If property\_value is a numerical\_value or a value\_limit and the unit is not assigned globally.

#4: If property\_value is a value\_range.

#5: If the property\_value is a numerical\_value or a value\_limit and the unit is specified globally.

AIM element: representation\_item.name

Source: ISO 10303-43

Reference path: #1: (compound\_representation\_item <=)  
#2: (descriptive\_representation\_item <=)  
#3: (measure\_representation\_item <=)  
#4: (value\_range <=  
compound\_representation\_item <=)  
#5: (value\_representation\_item <=)  
representation\_item  
representation\_item.name

#### 6.1.9.16 specific\_item\_classification

AIM element: product\_related\_product\_category

Source: ISO 10303-41

**6.1.9.16.1 classification\_name**

AIM element: product\_category.name

Source: ISO 10303-41

Reference path: product\_related\_product\_category <=  
product\_category  
product\_category.name  
{(product\_category.name)  
(product\_category.name = 'accessory item')  
(product\_category.name = 'adaptive item')  
(product\_category.name = 'assembly')  
(product\_category.name = 'assembly item')  
(product\_category.name = 'cutting item')  
(product\_category.name = 'cutting tool')  
(product\_category.name = 'detail')  
(product\_category.name = 'part')  
(product\_category.name = 'tool item')}

**6.1.9.16.2 description**

AIM element: product\_category.description

Source: ISO 10303-41

Reference path: product\_related\_product\_category <=  
product\_category  
product\_category.description

**6.1.9.16.3 specific\_item\_classification to item (as associated\_item)**

AIM element: PATH

Reference path: product\_related\_product\_category  
product\_related\_product\_category.products[i] ->  
product

**6.1.9.16.4 specific\_item\_classification to multi\_language\_string (as description)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([product\_category.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-41

Rules: restrict\_multi\_language\_for\_product\_related\_product\_category

Reference path: product\_related\_product\_category  
#1: (attribute\_language\_item = product\_related\_product\_category  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i])

```

attribute_language_assignment
{attribute_language_assignment <=
attribute_classification_assignment
attribute_classification_assignment.attribute_name = 'description'})
#2: (multi_language_attribute_item = product_related_product_category
multi_language_attribute_item <-
multi_language_attribute_assignment.items[i]
multi_language_attribute_assignment
{multi_language_attribute_assignment <=
attribute_value_assignment
attribute_value_assignment.attribute_name = 'description'})

```

#### 6.1.9.17 specific\_item\_classification\_hierarchy

AIM element: product\_category\_relationship

Source: ISO 10303-41

Reference path: {product\_category\_relationship  
product\_category\_relationship.name = 'hierarchy'}

##### 6.1.9.17.1 specific\_item\_classification\_hierarchy to specific\_item\_classification (as sub\_classification)

AIM element: PATH

Reference path: product\_category\_relationship  
product\_category\_relationship.sub\_category ->  
product\_category =>  
product\_related\_product\_category

##### 6.1.9.17.2 specific\_item\_classification\_hierarchy to specific\_item\_classification (as super\_classification)

AIM element: PATH

Reference path: product\_category\_relationship  
product\_category\_relationship.category ->  
product\_category =>  
product\_related\_product\_category

#### 6.1.9.18 string\_value

AIM element: descriptive\_representation\_item

Source: ISO 10303-45

Reference path: {descriptive\_representation\_item <=  
representation\_item =>  
qualified\_representation\_item  
qualified\_representation\_item.qualifiers[1] ->  
value\_qualifier  
value\_qualifier = type\_qualifier  
type\_qualifier  
type\_qualifier.name  
type\_qualifier.name = 'string'}

**6.1.9.18.1 value\_specification**

AIM element: descriptive\_representation\_item.description

Source: ISO 10303-45

**6.1.9.18.2 string\_value to multi\_language\_string (as value\_specification)**

#1: For the primary\_language\_dependent\_string.

#2: For any additional\_language\_dependent\_string.

AIM element: #1: ([descriptive\_representation\_item.description]  
[PATH])  
#2: (PATH)

Source: ISO 10303-45

Reference path: descriptive\_representation\_item  
#1: (attribute\_language\_item = descriptive\_representation\_item  
attribute\_language\_item <-  
attribute\_language\_assignment.items[i]  
attribute\_language\_assignment  
{attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.attribute\_name = 'description'})  
#2: (multi\_language\_attribute\_item = descriptive\_representation\_item  
multi\_language\_attribute\_item <-  
multi\_language\_attribute\_assignment.items[i]  
multi\_language\_attribute\_assignment  
{multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_name = 'description'})

**6.1.9.19 string\_with\_language**

#1: If the string\_with\_language does not play the role of a additional\_language\_dependent\_string.

#2: if the string\_with\_language play the role of an additional\_language\_dependent\_string.

AIM element: #1: (attribute\_language\_assignment)  
#2: ([attribute\_language\_assignment]  
[multi\_language\_attribute\_assignment])

Source: ISO 10303-214  
ISO 10303-214  
ISO 10303-214

Reference path: #1: (attribute\_language\_assignment <=  
attribute\_classification\_assignment  
{attribute\_classification\_assignment.role ->  
classification\_role  
classification\_role.name = 'primary'})  
#2: ([attribute\_language\_assignment <=  
attribute\_classification\_assignment  
{attribute\_classification\_assignment.role ->  
classification\_role  
classification\_role.name = 'primary'})  
[multi\_language\_attribute\_assignment <=

```

attribute_value_assignment
{attribute_value_assignment.role ->
attribute_value_role
attribute_value_role.name = 'alternate language'}
attribute_language_item = attribute_value_assignment
attribute_language_item <-
attribute_language_assignment.items[]
attribute_language_assignment <=
attribute_classification_assignment
{[attribute_classification_assignment.role ->
classification_role
classification_role.name = 'translated']
[attribute_classification_assignment.attribute_name = 'attribute_value']}]

```

#### 6.1.9.19.1 value

AIM element: #1: (IDENTICAL MAPPING)  
#2: (attribute\_value\_assignment.attribute\_value)

Reference path: #2: (multi\_language\_attribute\_assignment <=  
attribute\_value\_assignment  
attribute\_value\_assignment.attribute\_value)

#### 6.1.9.19.2 string\_with\_language to language (as language\_specification)

AIM element: PATH

Reference path: attribute\_language\_assignment <=  
attribute\_classification\_assignment  
attribute\_classification\_assignment.assigned\_class ->  
group =>  
language

#### 6.1.9.20 unit

- #1: If the unit has a commonly used descriptor.
- #2: If the unit descriptor is constructed of more than one unit name.
- #3: If the unit is defined with respect to the system of units defined in ISO 10403-41.
- #4: If the unit is defined with respect to another named\_unit by means of a conversion factor.
- #5: If the unit is dependent on the context of its usage.
- #6: In order to express a distance.
- #7: In order to express an amount of matter.
- #8: In order to express the duration of periods.
- #9: In order to express the movement of electrically charged particles.
- #10: In order to express the degree of heat of a body.
- #11: In order to express a quantity of substance in numbers of atoms.
- #12: In order to express the brightness of a body.

#13: In order to express angles in planes.

#14: In order to express solid angles.

#15: In order to express the extent of a surface.

#16: In order to express the solid content of a body.

#17: In order to express the ratio between two quantities of the same kind.

#18: If the named unit is represented by an explicit expression.

#19: If the context dependent unit is represented by an explicit expression.

#20: If the derived unit is represented by an explicit expression.

AIM element: #1: (named\_unit)  
#2: (derived\_unit)

Source: ISO 10303-41  
ISO 10303-41

Rules: dependent\_instantiable\_derived\_unit,  
dependent\_instantiable\_named\_unit

Reference path: {#1: (named\_unit)  
#18: (named\_unit =>  
named\_unit\_variable <=  
variable\_semantics)  
#19: (named\_unit =>  
context\_dependent\_unit =>  
expression\_conversion\_based\_unit <=  
variable\_semantics)  
(named\_unit =>  
<#3: (si\_unit)  
#4: (conversion\_based\_unit)  
#5: (context\_dependent\_unit)>  
<#6: (length\_unit)  
#7: (mass\_unit)  
#8: (time\_unit)  
#9: (electric\_current\_unit)  
#10: (thermodynamic\_temperature\_unit)  
#11: (amount\_of\_substance\_unit)  
#12: (luminous\_intensity\_unit)  
#13: (plane\_angle\_unit)  
#14: (solid\_angle\_unit)  
#15: (area\_unit)  
#16: (volume\_unit)  
#17: (ratio\_unit)>  
#2: (derived\_unit)  
#20: (derived\_unit =>  
derived\_unit\_variable <=  
variable\_semantics)}

#### 6.1.9.20.1 unit\_name

AIM element: #1: (#3: (si\_unit.name)  
#4: (conversion\_based\_unit.name)

#5: (context\_dependent\_unit.name))  
 #2: (derived\_unit.name)

Source: ISO 10303-41  
 ISO 10303-41  
 ISO 10303-41  
 ISO 10303-41

Reference path: #1: (named\_unit =>  
 #3: (si\_unit  
 si\_unit.name)  
 #4: (conversion\_based\_unit  
 conversion\_based\_unit.name)  
 #6: (context\_dependent\_unit  
 context\_dependent\_unit.name))  
 #2: (derived\_unit  
 derived\_unit.name)

### 6.1.9.21 value\_limit

#1: If the unit is assigned globally.

#2: If the unit is not assigned globally.

AIM element: #2: ([qualified\_representation\_item]  
 [measure\_representation\_item])  
 #1: ([value\_representation\_item]  
 [qualified\_representation\_item])

Source: ISO 10303-45  
 ISO 10303-45  
 ISO 10303-43  
 ISO 10303-45

Rules: dependent\_instantiable\_measure\_with\_unit

#### 6.1.9.21.1 limit

#1: If the unit is assigned globally.

#2: If the unit is not assigned globally.

AIM element: #1: (value\_representation\_item.value\_component)  
 #2: (measure\_with\_unit.value\_component)

Source: ISO 10303-43  
 ISO 10303-45

Reference path: #1: (value\_representation\_item  
 value\_representation\_item.value\_component)  
 #2: ({measure\_representation\_item <=  
 measure\_with\_unit  
 {(measure\_with\_unit)  
 (measure\_with\_unit =>  
 (amount\_of\_substance\_measure\_with\_unit)  
 (area\_measure\_with\_unit)  
 (electric\_current\_measure\_with\_unit)  
 (length\_measure\_with\_unit)  
 (luminous\_intensity\_measure\_with\_unit)

```

(mass_measure_with_unit)
(plane_angle_measure_with_unit)
(ratio_measure_with_unit)
(solid_angle_measure_with_unit)
(thermodynamic_temperature_measure_with_unit)
(time_measure_with_unit)
(volume_measure_with_unit))}
measure_with_unit.value_component
{measure_with_unit.value_component ->
measure_value
(measure_value = area_measure
area_measure)
(measure_value = amount_of_substance_measure
amount_of_substance_measure)
(measure_value = length_measure
length_measure)
(measure_value = electric_current_measure
electric_current_measure)
(measure_value = plane_angle_measure
plane_angle_measure)
(measure_value = ratio_measure
ratio_measure)
(measure_value = parameter_value
parameter_value)
(measure_value = descriptive_measure
descriptive_measure)
(measure_value = positive_plane_angle_measure
positive_plane_angle_measure)
(measure_value = count_measure
count_measure)
(measure_value = mass_measure
mass_measure)
(measure_value = time_measure
time_measure)
(measure_value = thermodynamic_temperature_measure
thermodynamic_temperature_measure)
(measure_value = luminous_intensity_measure
luminous_intensity_measure)
(measure_value = solid_angle_measure
solid_angle_measure)
(measure_value = volume_measure
volume_measure)
(measure_value = numeric_measure
numeric_measure)
(measure_value = context_dependent_measure
context_dependent_measure)
(measure_value = positive_length_measure
positive_length_measure)
(measure_value = positive_ratio_measure
positive_ratio_measure)}})

```

#### 6.1.9.21.2 limit\_qualifier

#1: If the limit is an upper bound.

#2: If the limit is a lower bound.

AIM element: type\_qualifier.name

Source: ISO 10303-45

Rules: dependent\_instantiable\_type\_qualifier

Reference path: qualified\_representation\_item  
 qualified\_representation\_item.qualifiers[i] ->  
 value\_qualifier = type\_qualifier  
 type\_qualifier  
 {#1: (type\_qualifier.name = 'maximum')  
 #2: (type\_qualifier.name = 'minimum')}

### 6.1.9.22 value\_list

AIM element: compound\_representation\_item

Source: ISO 10303-43

#### 6.1.9.22.1 value\_list to property\_value (as values)

#1: If values[i] is a value\_list.

#2: If values[i] is a string\_value.

#3: If values[i] is a numerical\_value or a value\_limit and the unit is not assigned globally.

#4: If values[i] is a value\_range.

#5: If no significant\_digits are given for values[i].

#6: If significant\_digits are given for values[i].

#7: If values[i] is a numerical\_value or a value\_limit and the unit is specified globally.

AIM element: PATH

Reference path: compound\_representation\_item  
 compound\_representation\_item.item\_element ->  
 compound\_item\_definition  
 compound\_item\_definition = list\_representation\_item  
 list\_representation\_item[i] ->  
 representation\_item =>  
 #1: (compound\_representation\_item)  
 #2: (descriptive\_representation\_item  
 {descriptive\_representation\_item <=  
 representation\_item =>  
 qualified\_representation\_item  
 qualified\_representation\_item.qualifiers[1] ->  
 value\_qualifier  
 value\_qualifier = type\_qualifier  
 type\_qualifier  
 type\_qualifier.name  
 type\_qualifier.name = 'string'})  
 #3: (#5: (measure\_representation\_item)  
 #6: ([measure\_representation\_item]  
 [qualified\_representation\_item]))  
 #4: (#5: (compound\_representation\_item =>  
 value\_range)  
 #6: ([compound\_representation\_item =>  
 value\_range])

[qualified\_representation\_item]))  
 #7: (#5: (value\_representation\_item)  
 #6: ([value\_representation\_item]  
 [qualified\_representation\_item]))

**6.1.9.23 value\_range**

#1: If no significant digits are given for the value\_with\_unit.

#2: If significant digits are given for the value\_with\_unit.

AIM element: #1: (value\_range)  
 #2: ([value\_range]  
 [qualified\_representation\_item])

Source: ISO 10303-214  
 ISO 10303-214  
 ISO 10303-45

Reference path: {value\_range <=  
 compound\_representation\_item}

**6.1.9.23.1 lower\_limit**

#1: If the unit is not assigned globally.

#2: If the unit is assigned globally.

AIM element: #1: (measure\_with\_unit.value\_component)  
 #2: (value\_representation\_item.value\_component)

Source: ISO 10303-41  
 ISO 10303-41

Rules: dependent\_instantiable\_measure\_with\_unit

Reference path: value\_range <=  
 compound\_representation\_item  
 compound\_representation\_item.item\_element ->  
 set\_representation\_item  
 #1: (set\_representation\_item[i] ->  
 representation\_item =>  
 {representation\_item.name = 'lower limit'}  
 measure\_representation\_item  
 measure\_representation\_item <=  
 measure\_with\_unit  
 {(measure\_with\_unit)  
 (measure\_with\_unit =>  
 (amount\_of\_substance\_measure\_with\_unit)  
 (area\_measure\_with\_unit)  
 (electric\_current\_measure\_with\_unit)  
 (length\_measure\_with\_unit)  
 (luminous\_intensity\_measure\_with\_unit)  
 (mass\_measure\_with\_unit)  
 (plane\_angle\_measure\_with\_unit)  
 (ratio\_measure\_with\_unit)  
 (solid\_angle\_measure\_with\_unit)  
 (thermodynamic\_temperature\_measure\_with\_unit)  
 (time\_measure\_with\_unit)

```

(volume_measure_with_unit)}}
measure_with_unit.value_component
{measure_with_unit.value_component ->
measure_value
(measure_value = area_measure
area_measure)
(measure_value = amount_of_substance_measure
amount_of_substance_measure)
(measure_value = length_measure
length_measure)
(measure_value = electric_current_measure
electric_current_measure)
(measure_value = plane_angle_measure
plane_angle_measure)
(measure_value = ratio_measure
ratio_measure)
(measure_value = parameter_value
parameter_value)
(measure_value = descriptive_measure
descriptive_measure)
(measure_value = positive_plane_angle_measure
positive_plane_angle_measure)
(measure_value = count_measure
count_measure)
(measure_value = mass_measure
mass_measure)
(measure_value = time_measure
time_measure)
(measure_value = thermodynamic_temperature_measure
thermodynamic_temperature_measure)
(measure_value = luminous_intensity_measure
luminous_intensity_measure)
(measure_value = solid_angle_measure
solid_angle_measure)
(measure_value = volume_measure
volume_measure)
(measure_value = numeric_measure
numeric_measure)
(measure_value = context_dependent_measure
context_dependent_measure)
(measure_value = positive_length_measure
positive_length_measure)
(measure_value = positive_ratio_measure
positive_ratio_measure)}}
#2: (set_representation_item[i] ->
representation_item =>
{representation_item.name = 'lower limit'}
value_representation_item
value_representation_item.value_component)

```

#### 6.1.9.23.2 upper\_limit

#1: If the unit is not assigned globally.

#2: If the unit is assigned globally.

AIM element: #1: (measure\_with\_unit.value\_component)  
#2: (value\_representation\_item.value\_component)

Source: ISO 10303-41  
ISO 10303-41

Rules: dependent\_instantiable\_measure\_with\_unit

Reference path: value\_range <=  
compound\_representation\_item  
compound\_representation\_item.item\_element ->  
set\_representation\_item  
#1: (set\_representation\_item[i] ->  
representation\_item =>  
{representation\_item.name = 'upper limit'}  
measure\_representation\_item  
measure\_representation\_item <=  
measure\_with\_unit  
{(measure\_with\_unit)}  
(measure\_with\_unit =>  
(amount\_of\_substance\_measure\_with\_unit)  
(area\_measure\_with\_unit)  
(electric\_current\_measure\_with\_unit)  
(length\_measure\_with\_unit)  
(luminous\_intensity\_measure\_with\_unit)  
(mass\_measure\_with\_unit)  
(plane\_angle\_measure\_with\_unit)  
(ratio\_measure\_with\_unit)  
(solid\_angle\_measure\_with\_unit)  
(thermodynamic\_temperature\_measure\_with\_unit)  
(time\_measure\_with\_unit)  
(volume\_measure\_with\_unit))}  
measure\_with\_unit.value\_component  
{measure\_with\_unit.value\_component ->  
measure\_value  
(measure\_value = area\_measure  
area\_measure)  
(measure\_value = amount\_of\_substance\_measure  
amount\_of\_substance\_measure)  
(measure\_value = length\_measure  
length\_measure)  
(measure\_value = electric\_current\_measure  
electric\_current\_measure)  
(measure\_value = plane\_angle\_measure  
plane\_angle\_measure)  
(measure\_value = ratio\_measure  
ratio\_measure)  
(measure\_value = parameter\_value  
parameter\_value)  
(measure\_value = descriptive\_measure  
descriptive\_measure)  
(measure\_value = positive\_plane\_angle\_measure  
positive\_plane\_angle\_measure)  
(measure\_value = count\_measure  
count\_measure)  
(measure\_value = mass\_measure  
mass\_measure)  
(measure\_value = time\_measure  
time\_measure)  
(measure\_value = thermodynamic\_temperature\_measure  
thermodynamic\_temperature\_measure)  
(measure\_value = luminous\_intensity\_measure  
luminous\_intensity\_measure)

```

(measure_value = solid_angle_measure
solid_angle_measure)
(measure_value = volume_measure
volume_measure)
(measure_value = numeric_measure
numeric_measure)
(measure_value = context_dependent_measure
context_dependent_measure)
(measure_value = positive_length_measure
positive_length_measure)
(measure_value = positive_ratio_measure
positive_ratio_measure))
#2: (set_representation_item[i] ->
representation_item =>
{representation_item.name = 'upper limit'}
value_representation_item
value_representation_item.value_component)

```

#### 6.1.9.24 value\_with\_unit

#1: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is not assigned globally.

#2: If the value\_with\_unit is a value\_range.

#3: If no significant digits are given for the value\_with\_unit.

#4: If significant digits are given for the value\_with\_unit.

#5: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is specified globally.

AIM element: #1: (#3: (measure\_representation\_item)  
#4: ([measure\_representation\_item]  
[qualified\_representation\_item]))  
#2: (#3: (value\_range)  
#4: ([value\_range]  
[qualified\_representation\_item]))  
#5: (#3: (value\_representation\_item)  
#4: ([value\_representation\_item]  
[qualified\_representation\_item]))

Source: ISO 10303-45  
ISO 10303-45

Reference path: #2: ({value\_range <=  
compound\_representation\_item})

##### 6.1.9.24.1 significant\_digits

AIM element: precision\_qualifier.precision\_value

Source: ISO 10303-45

Rules: dependent\_instantiable\_precision\_qualifier

Reference path: qualified\_representation\_item  
 qualified\_representation\_item.qualifiers[i] ->  
 value\_qualifier  
 value\_qualifier = precision\_qualifier  
 precision\_qualifier  
 precision\_qualifier.precision\_value

**6.1.9.24.2 value\_with\_unit to unit (as unit\_component)**

#1: If the value\_with\_unit is a numerical\_value or a value\_limit and the unit is not assigned globally.

#2: If the value\_with\_unit is a value\_range.

#3: If the unit has a commonly used descriptor.

#4: If the unit descriptor is constructed of more than one unit name.

AIM element: PATH

Reference path: #1: (measure\_representation\_item <=)  
 #2: (compound\_representation\_item  
 compound\_representation\_item.item\_element ->  
 compound\_item\_definition  
 compound\_item\_definition = set\_representation\_item  
 set\_representation\_item  
 set\_representation\_item[i] ->  
 representation\_item =>  
 measure\_representation\_item <=)  
 measure\_with\_unit  
 measure\_with\_unit.unit\_component ->  
 unit  
 #3: (unit = named\_unit  
 named\_unit)  
 #4: (unit = derived\_unit  
 derived\_unit)

**6.2 AIM EXPRESS short listing**

**6.2.1 cutting\_tool\_schema types**

**6.2.1.1 applied\_state\_type\_of\_item**

The applied\_state\_type\_of\_item type is an extension of the state\_type\_of\_item type. It adds the data type product\_definition to the list of alternate data types.

EXPRESS specification:

```
TYPE applied_state_type_of_item = SELECT BASED_ON state_type_of_item WITH;
    (product_definition);
END_TYPE;
```

**6.2.1.2 attribute\_language\_item**

An attribute\_language\_item type specifies those objects to which a language may be assigned through an attribute\_language\_assignment.

EXPRESS specification:

```

TYPE attribute_language_item = SELECT
  (property_definition,
   group,
   external_source,
   product_related_product_category,
   effectivity,
   effectivity_relationship,
   product,
   product_definition,
   product_definition_formation,
   product_definition_formation_relationship,
   representation,
   product_definition_relationship,
   general_property,
   general_property_relationship,
   representation_relationship,
   general_feature,
   identification_role,
   application_context,
   attribute_value_assignment,
   person_and_organization_role,
   organization_role,
   descriptive_representation_item);
END_TYPE;

```

**6.2.1.3 classification\_item**

A `classification_item` type specifies those objects to which a class or a class\_system may be assigned through an `applied_classification_assignment`.

EXPRESS specification:

```

TYPE classification_item = SELECT
  (product,
   document_file,
   product_definition,
   product_definition_formation,
   product_definition_relationship,
   material_designation,
   general_property,
   property_definition,
   class,
   plus_minus_tolerance,
   document_type,
   planar_extent);
END_TYPE;

```

**6.2.1.4 document\_reference\_item**

A `document_reference_item` type specifies those objects to which a `document_reference` may be assigned through an `applied_document_reference`.

EXPRESS specification:

```

TYPE document_reference_item = SELECT
  (property_definition,

```

```

class_system,
class,
product,
product_definition,
product_definition_relationship,
product_definition_formation,
material_designation,
organization,
person,
general_property,
product_related_product_category,
general_feature);
END_TYPE;

```

#### 6.2.1.5 effectivity\_item

An effectivity\_item type specifies those objects to which an effectivity may be assigned through an applied\_effectivity\_assignment.

##### EXPRESS specification:

```

TYPE effectivity_item = SELECT
(class_system,
product,
product_definition,
document_file,
product_definition_formation,
product_definition_relationship,
material_designation,
applied_location_assignment,
general_property,
property_definition,
applied_state_type_assignment);
END_TYPE;

```

#### 6.2.1.6 external\_identification\_item

An external\_identification\_item type specifies those objects to which an external identifier and an external source may be assigned through an applied\_external\_identification\_assignment.

##### EXPRESS specification:

```

TYPE external_identification_item = SELECT
(document_file,
product_definition,
externally_defined_class,
externally_defined_general_property);
END_TYPE;

```

#### 6.2.1.7 identification\_item

An identification\_item type specifies those objects to which an identifier and its usage may be assigned through an applied\_identification\_assignment.

##### EXPRESS specification:

```

TYPE identification_item = SELECT

```

```

(document_file,
product,
general_property,
applied_identification_assignment,
property_definition,
class_system,
product_definition,
document_type,
class,
product_definition_formation,
product_definition_relationship,
organization,
person_and_organization,
effectivity);
END_TYPE;

```

#### 6.2.1.8 language\_item

A `language_item` type specifies those objects to which a classification of a language may be assigned through a `language_assignment`.

##### EXPRESS specification:

```

TYPE language_item = SELECT
    (representation);
END_TYPE;

```

#### 6.2.1.9 location\_assignment\_item

A `location_assignment_item` type specifies those objects to which a location may be assigned through an `applied_location_assignment`.

##### EXPRESS specification:

```

TYPE location_assignment_item = SELECT
    (product_definition);
END_TYPE;

```

#### 6.2.1.10 multi\_language\_attribute\_item

A `multi_language_attribute_item` type specifies those objects to which a label in an alternate language for an existing attribute may be assigned through a `multi_language_attribute_assignment`.

##### EXPRESS specification:

```

TYPE multi_language_attribute_item = SELECT
    (property_definition,
    group,
    external_source,
    product_related_product_category,
    effectivity,
    effectivity_relationship,
    product_definition,
    product,
    product_definition_formation,
    product_definition_formation_relationship,
    representation,

```

```

product_definition_relationship,
general_property,
general_property_relationship,
representation_relationship,
general_feature,
identification_role,
application_context,
person_and_organization_role,
organization_role,
descriptive_representation_item);
END_TYPE;

```

#### 6.2.1.11 organization\_item

An organization\_item type specifies those objects to which an organization may be assigned through an applied\_organization\_assignment.

##### EXPRESS specification:

```

TYPE organization_item = SELECT
(class,
property_definition,
applied_identification_assignment,
applied_classification_assignment,
class_system,
product,
document_file,
product_definition,
product_definition_formation,
product_definition_relationship,
product_definition_formation_relationship,
material_designation,
person_and_organization,
general_property);
END_TYPE;

```

#### 6.2.1.12 person\_and\_organization\_item

A person\_and\_organization\_item type specifies those objects to which a person\_and\_organization may be assigned through an applied\_person\_and\_organization\_assignment.

##### EXPRESS specification:

```

TYPE person_and_organization_item = SELECT
(applied_classification_assignment,
class_system,
product,
document_file,
product_definition,
product_definition_formation,
class,
product_definition_relationship,
product_definition_formation_relationship,
material_designation,
person_and_organization,
general_property,
property_definition);
END_TYPE;

```

## 6.2.2 cutting\_tool\_schema entities

### 6.2.2.1 applied\_location\_assignment

The `applied_location_assignment` entity allows the specification of a location for a product\_definition of a physical product.

#### EXPRESS specification:

```
ENTITY applied_location_assignment
  SUBTYPE OF(location_assignment);
  items : SET [1:?] OF location_assignment_item;
END_ENTITY;
```

#### 6.2.2.1.1 items

Specifies the product\_definition of a physical product that is located.

#### 6.2.2.2 general\_feature

A `general_feature` is a type of feature\_definition that provides a mechanism to categorize an area of interest in a shape and to associate additional information with it.

#### EXPRESS specification:

```
ENTITY general_feature
  SUBTYPE OF(feature_definition, shape_aspect);
  WHERE
    WR1 :
      (SIZEOF(get_property_definition_representations(SEL)) <= 1)
  AND (SIZEOF(QUERY (pdr <* get_property_definition_representations(SEL) |
    ('CUTTING_TOOL_SCHEMA.' + 'SHAPE_REPRESENTATION_WITH_PARAMETERS' IN
    TYPEOF(pdr.used_representation)))) <= 1);

    WR2 :
      SIZEOF(QUERY (sa <* get_shape_aspects(SEL) | (sa.description =
      'course of travel occurrence')) = SIZEOF(QUERY (sa <*
      get_shape_aspects(SEL) | (sa.description = 'course of travel
      occurrence') AND (SIZEOF(QUERY (sar <* USEDIN(sa, 'CUTTING_TOOL_SCHEMA.'
      + 'SHAPE_ASPECT_RELATIONSHIP.' + 'RELATED_SHAPE_ASPECT') |
      ('CUTTING_TOOL_SCHEMA.' + 'SHAPE_DEFINING_RELATIONSHIP' IN TYPEOF(sar))
      AND (sar.name = 'course of travel')) = 1)))));

    WR3 :
      SIZEOF(QUERY (sa <* get_shape_aspects(SEL) | (sa.description =
      'course of travel occurrence')) = SIZEOF(QUERY (sa <*
      get_shape_aspects(SEL) | (sa.description = 'course of travel
      occurrence') AND (SIZEOF(QUERY (sar <* USEDIN(sa, 'CUTTING_TOOL_SCHEMA.'
      + 'SHAPE_ASPECT_RELATIONSHIP.' + 'RELATED_SHAPE_ASPECT') |
      ('CUTTING_TOOL_SCHEMA.' + 'SHAPE_DEFINING_RELATIONSHIP' IN TYPEOF(sar))
      AND ('CUTTING_TOOL_SCHEMA.' + 'PATH_FEATURE_COMPONENT' IN
      TYPEOF(sar.relating_shape_aspect)))) = 1)))));

    WR4 :
      SIZEOF(QUERY (sa <* get_shape_aspects(SEL) | (sa.description =
      'boundary occurrence')) = SIZEOF(QUERY (sa <* get_shape_aspects(SEL) |
      (sa.description = 'boundary occurrence') AND (SIZEOF(QUERY (sar <*
      USEDIN(sa, 'CUTTING_TOOL_SCHEMA.' + 'SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') | ('CUTTING_TOOL_SCHEMA.' +
```

```
'SHAPE_DEFINING_RELATIONSHIP' IN TYPEOF(sar)) AND (sar.description =
'profile usage')) = 1)))));
    WR5 :
        (SIZEOF(QUERY (sa <* get_shape_aspects(SELF) |
('CUTTING_TOOL_SCHEMA.' + 'COMPOSITE_SHAPE_ASPECT' IN TYPEOF(sa))) <=
1) AND (SIZEOF(QUERY (sa <* get_shape_aspects(SELF) |
(('CUTTING_TOOL_SCHEMA.' + 'COMPOSITE_SHAPE_ASPECT' IN TYPEOF(sa)) AND
(sa.name = 'general compound feature')) AND (SIZEOF(QUERY (sar <*
USEDIN(sa, 'CUTTING_TOOL_SCHEMA.' + 'SHAPE_ASPECT_RELATIONSHIP.' +
'RELATING_SHAPE_ASPECT') | ('CUTTING_TOOL_SCHEMA.' +
'FEATURE_COMPONENT_RELATIONSHIP' IN TYPEOF(sar)) AND
('CUTTING_TOOL_SCHEMA.' + 'INSTANCED_FEATURE' IN
TYPEOF(sar.related_shape_aspect)))) >= 2))) <= 1);
END_ENTITY;
```

### 6.2.3 cutting\_tool\_schema rules

#### 6.2.3.1 coating\_requires\_product\_definition

The coating\_requires\_product\_definition rule ensures that each instance of product\_definition\_formation which has an associated of\_product that is referenced by a product\_related\_product\_category with a name of 'coating', is referenced as formation by exactly one instance of product\_definition that references as its frame\_of\_reference a product\_definition\_context with a name of 'material definition'.

#### EXPRESS specification:

```
RULE coating_requires_product_definition FOR
    (product_definition,
    product_definition_formation,
    product_related_product_category);
WHERE
    WR1: SIZEOF ( QUERY ( pdf <* product_definition_formation | (
        SIZEOF ( QUERY ( prpc <* USEDIN ( pdf. of_product ,
        'CUTTING_TOOL_SCHEMA.'+'PRODUCT_RELATED_PRODUCT_CATEGORY.'+
        'PRODUCTS' ) | prpc.name = 'coating' )
        ) >0 ) AND ( SIZEOF ( QUERY ( pd <* USEDIN ( pdf ,
        'ISO13399_AIM_SHORT_FORM.'+'PRODUCT_DEFINITION.'+ 'FORMATION' ) |
        pd.frame_of_reference.name = 'material definition' ) ) <>1
        ) ) ) =0;
END_RULE;
```

#### Argument definitions:

**product\_definition:** the set of all instances of product\_definition.

**product\_definition\_formation:** the set of all instances of product\_definition\_formation.

**product\_related\_product\_category:** the set of all instances of product\_related\_product\_category.

**product\_definition\_formation:** the set of all instances of product\_definition\_formation.

**product\_related\_product\_category:** the set of all instances of product\_related\_product\_category.

#### Formal propositions:

WR1: Each instance of product\_definition\_formation which has an associated of\_product that is referenced by a product\_related\_product\_category with a name of 'coating', is referenced as formation by exactly one

instance of product\_definition that references as its frame\_of\_reference a product\_definition\_context with a name of 'material definition'.

### 6.2.3.2 dependent\_instantiable\_attribute\_value\_role

The dependent\_instantiable\_attribute\_value\_role rule specifies that each instance of attribute\_value\_role is dependent on the usage to define another entity.

#### EXPRESS specification:

```
RULE dependent_instantiable_attribute_value_role FOR
  (attribute_value_role);
WHERE
  WR1: SIZEOF ( QUERY ( a <* attribute_value_role | NOT ( SIZEOF
    ( USEDIN ( a , ' ' ) ) >0 ) ) ) =0;
END_RULE;
```

#### Argument definitions:

**attribute\_value\_role:** the set of all instances of attribute\_value\_role.

#### Formal propositions:

**WR1:** For each instance of attribute\_value\_role, there shall be a reference to the attribute\_value\_role instance from an attribute of another entity.

### 6.2.3.3 dependent\_instantiable\_classification\_role

The dependent\_instantiable\_classification\_role rule specifies that each instance of classification\_role is dependent on the usage to define another entity.

#### EXPRESS specification:

```
RULE dependent_instantiable_classification_role FOR
  (classification_role);
WHERE
  WR1: SIZEOF ( QUERY ( c <* classification_role | NOT ( SIZEOF (
    USEDIN ( c , ' ' ) ) >0 ) ) ) =0;
END_RULE;
```

#### Argument definitions:

**classification\_role:** the set of all instances of classification\_role.

#### Formal propositions:

**WR1:** For each instance of classification\_role, there shall be a reference to the classification\_role instance from an attribute of another entity.

### 6.2.3.4 dependent\_instantiable\_date

The dependent\_instantiable\_date rule specifies that each instance of date is dependent on the usage to define another entity.

#### EXPRESS specification:

```
RULE dependent_instantiable_date FOR
  (date);
```

```
WHERE
  WR1: SIZEOF ( QUERY ( d <* date | NOT ( SIZEOF ( USEDIN ( d ,
    ' ' ) ) >0 ) ) ) =0;
END_RULE;
```

**Argument definitions:**

**date:** the set of all instances of date.

**Formal propositions:**

**WR1:** For each instance of date, there shall be a reference to the date instance from an attribute of another entity.

**6.2.3.5 dependent\_instantiable\_date\_and\_time**

The dependent\_instantiable\_date\_and\_time rule specifies that each instance of date\_and\_time is dependent on the usage to define another entity.

**EXPRESS specification:**

```
RULE dependent_instantiable_date_and_time FOR
  (date_and_time);
WHERE
  WR1: SIZEOF ( QUERY ( d <* date_and_time | NOT ( SIZEOF (
    USEDIN ( d , ' ' ) ) >0 ) ) ) =0;
END_RULE;
```

**Argument definitions:**

**date\_and\_time:** the set of all instances of date\_and\_time.

**Formal propositions:**

**WR1:** For each instance of date\_and\_time, there shall be a reference to the date\_and\_time instance from an attribute of another entity.

**6.2.3.6 dependent\_instantiable\_derived\_unit**

The dependent\_instantiable\_derived\_unit rule specifies that each instance of derived\_unit is dependent on the usage to define another entity.

**EXPRESS specification:**

```
RULE dependent_instantiable_derived_unit FOR
  (derived_unit);
WHERE
  WR1: SIZEOF ( QUERY ( d <* derived_unit | NOT ( SIZEOF ( USEDIN
    ( d , ' ' ) ) >0 ) ) ) =0;
END_RULE;
```

**Argument definitions:**

**derived\_unit:** the set of all instances of derived\_unit.

**Formal propositions:**

**WR1:** For each instance of `derived_unit`, there shall be a reference to the `derived_unit` instance from an attribute of another entity.

#### 6.2.3.7 `dependent_instantiable_identification_role`

The `dependent_instantiable_identification_role` rule specifies that each instance of `identification_role` is dependent on the usage to define another entity.

##### **EXPRESS specification:**

```
RULE dependent_instantiable_identification_role FOR
  (identification_role);
WHERE
  WR1: SIZEOF ( QUERY ( i <* identification_role | NOT ( SIZEOF (
    USEDIN ( i , ' ' ) ) >0 ) ) ) =0;
END_RULE;
```

##### **Argument definitions:**

**identification\_role:** the set of all instances of `identification_role`.

##### **Formal propositions:**

**WR1:** For each instance of `identification_role`, there shall be a reference to the `identification_role` instance from an attribute of another entity.

#### 6.2.3.8 `dependent_instantiable_measure_with_unit`

The `dependent_instantiable_measure_with_unit` rule specifies that each instance of `measure_with_unit` is dependent on the usage to define another entity.

##### **EXPRESS specification:**

```
RULE dependent_instantiable_measure_with_unit FOR
  (measure_with_unit);
WHERE
  WR1: SIZEOF ( QUERY ( m <* measure_with_unit | NOT ( SIZEOF (
    USEDIN ( m , ' ' ) ) >0 ) ) ) =0;
END_RULE;
```

##### **Argument definitions:**

**measure\_with\_unit:** the set of all instances of `measure_with_unit`.

##### **Formal propositions:**

**WR1:** For each instance of `measure_with_unit`, there shall be a reference to the `measure_with_unit` instance from an attribute of another entity.

#### 6.2.3.9 `dependent_instantiable_named_unit`

The `dependent_instantiable_named_unit` rule specifies that each instance of `named_unit` is dependent on the usage to define another entity.

##### **EXPRESS specification:**

```
RULE dependent_instantiable_named_unit FOR
  (named_unit);
```

```
WHERE
  WR1: SIZEOF ( QUERY ( n <* named_unit | NOT ( SIZEOF ( USEDIN (
    n , ' ' ) ) >0 ) ) ) =0;
END_RULE;
```

**Argument definitions:**

**named\_unit:** the set of all instances of named\_unit.

**Formal propositions:**

**WR1:** For each instance of named\_unit, there shall be a reference to the named\_unit instance from an attribute of another entity.

**6.2.3.10 dependent\_instantiable\_object\_role**

The dependent\_instantiable\_object\_role rule specifies that each instance of object\_role is dependent on the usage to define another entity.

**EXPRESS specification:**

```
RULE dependent_instantiable_object_role FOR
  (object_role);
WHERE
  WR1: SIZEOF ( QUERY ( o <* object_role | NOT ( SIZEOF ( USEDIN
    ( o , ' ' ) ) >0 ) ) ) =0;
END_RULE;
```

**Argument definitions:**

**object\_role:** the set of all instances of object\_role.

**Formal propositions:**

**WR1:** For each instance of object\_role, there shall be a reference to the object\_role instance from an attribute of another entity.

**6.2.3.11 dependent\_instantiable\_organization\_role**

The dependent\_instantiable\_organization\_role rule specifies that each instance of organization\_role is dependent on the usage to define another entity.

**EXPRESS specification:**

```
RULE dependent_instantiable_organization_role FOR
  (organization_role);
WHERE
  WR1: SIZEOF ( QUERY ( o <* organization_role | NOT ( SIZEOF (
    USEDIN ( o , ' ' ) ) >0 ) ) ) =0;
END_RULE;
```

**Argument definitions:**

**organization\_role:** the set of all instances of organization\_role.

**Formal propositions:**

**WR1:** For each instance of organization\_role, there shall be a reference to the organization\_role instance from an attribute of another entity.

#### 6.2.3.12 dependent\_instantiable\_person\_and\_organization\_role

The dependent\_instantiable\_person\_and\_organization\_role rule specifies that each instance of person\_and\_organization\_role is dependent on the usage to define another entity.

##### EXPRESS specification:

```
RULE dependent_instantiable_person_and_organization_role FOR
  (person_and_organization_role);
WHERE
  WR1: SIZEOF ( QUERY ( p <* person_and_organization_role | NOT (
    SIZEOF ( USEDIN ( p , ' ' ) ) >0 ) ) ) =0;
END_RULE;
```

##### Argument definitions:

**person\_and\_organization\_role:** the set of all instances of person\_and\_organization\_role.

##### Formal propositions:

**WR1:** For each instance of person\_and\_organization\_role, there shall be a reference to the person\_and\_organization\_role instance from an attribute of another entity.

#### 6.2.3.13 dependent\_instantiable\_precision\_qualifier

The dependent\_instantiable\_precision\_qualifier rule specifies that each instance of precision\_qualifier is dependent on the usage to define another entity.

##### EXPRESS specification:

```
RULE dependent_instantiable_precision_qualifier FOR
  (precision_qualifier);
WHERE
  WR1: SIZEOF ( QUERY ( p <* precision_qualifier | NOT ( SIZEOF (
    USEDIN ( p , ' ' ) ) >0 ) ) ) =0;
END_RULE;
```

##### Argument definitions:

**precision\_qualifier:** the set of all instances of precision\_qualifier.

##### Formal propositions:

**WR1:** For each instance of precision\_qualifier, there shall be a reference to the precision\_qualifier instance from an attribute of another entity.

#### 6.2.3.14 dependent\_instantiable\_product\_definition\_context\_role

The dependent\_instantiable\_product\_definition\_context\_role rule specifies that each instance of product\_definition\_context\_role is dependent on the usage to define another entity.

##### EXPRESS specification:

```
RULE dependent_instantiable_product_definition_context_role FOR
  (product_definition_context_role);
```

```
WHERE
  WR1: SIZEOF ( QUERY ( p <* product_definition_context_role |
    NOT ( SIZEOF ( USEDIN ( p , ' ' ) ) >0 ) ) ) =0;
END_RULE;
```

**Argument definitions:**

**product\_definition\_context\_role:** the set of all instances of product\_definition\_context\_role.

**Formal propositions:**

**WR1:** For each instance of product\_definition\_context\_role, there shall be a reference to the product\_definition\_context\_role instance from an attribute of another entity.

**6.2.3.15 dependent\_instantiable\_tolerance\_value**

The dependent\_instantiable\_tolerance\_value rule specifies that each instance of tolerance\_value is dependent on the usage to define another entity.

**EXPRESS specification:**

```
RULE dependent_instantiable_tolerance_value FOR
  (tolerance_value);
WHERE
  WR1: SIZEOF ( QUERY ( t <* tolerance_value | NOT ( SIZEOF (
    USEDIN ( t , ' ' ) ) >0 ) ) ) =0;
END_RULE;
```

**Argument definitions:**

**tolerance\_value:** the set of all instances of tolerance\_value.

**Formal propositions:**

**WR1:** For each instance of tolerance\_value, there shall be a reference to the tolerance\_value instance from an attribute of another entity.

**6.2.3.16 dependent\_instantiable\_type\_qualifier**

The dependent\_instantiable\_type\_qualifier rule specifies that each instance of type\_qualifier is dependent on the usage to define another entity.

**EXPRESS specification:**

```
RULE dependent_instantiable_type_qualifier FOR
  (type_qualifier);
WHERE
  WR1: SIZEOF ( QUERY ( t <* type_qualifier | NOT ( SIZEOF (
    USEDIN ( t , ' ' ) ) >0 ) ) ) =0;
END_RULE;
```

**Argument definitions:**

**type\_qualifier:** the set of all instances of type\_qualifier.

**Formal propositions:**

**WR1:** For each instance of `type_qualifier`, there shall be a reference to the `type_qualifier` instance from an attribute of another entity.

### 6.2.3.17 externally\_defined\_class\_with\_known\_source\_requirement

The `externally_defined_class_with_known_source_requirement` rule specifies that each instance of `externally_defined_class` which references as the source a `known_source` is contained in the set of items of exactly one `applied_organization_assignment` which references as its role an `organization_role` with a name of 'class supplier'. This rule enforces the requirement for every `plib_class_reference` to have a `supplier_bsu`.

#### EXPRESS specification:

```
RULE externally_defined_class_with_known_source_requirement FOR
  (externally_defined_class);
WHERE
  WR1: SIZEOF ( QUERY ( edc <* externally_defined_class | (
    'CUTTING_TOOL_SCHEMA.'+ 'KNOWN_SOURCE' IN TYPEOF ( edc.
    source ) ) AND ( SIZEOF ( QUERY ( aoa <* USEDIN ( edc ,
    'CUTTING_TOOL_SCHEMA.APPLIED_ORGANIZATION_ASSIGNMENT.ITEMS' )
    | aoa. role.name = 'class supplier' ) ) =0 ) ) ) =0;
END_RULE;
```

#### Argument definitions:

**externally\_defined\_class:** the set of all instances of `externally_defined_class`.

#### Formal propositions:

**WR1:** Each instance of `externally_defined_class` which references as the source a `known_source` shall be member in the set of items of an `applied_organization_assignment`. This `organization_assignment` shall reference as its role an `organization_role` with a name of 'class supplier'.

### 6.2.3.18 grade\_requires\_product\_definition

The `grade_requires_product_definition` rule ensures that each instance of `product_definition_formation` which has an associated `of_product` that is referenced by a `product_related_product_category` with a name of 'grade', is referenced as formation by exactly one instance of `product_definition` that references as its `frame_of_reference` a `product_definition_context` with a name of 'material definition'.

#### EXPRESS specification:

```
RULE grade_requires_product_definition FOR
  (product_definition,
  product_definition_formation,
  product_related_product_category);
WHERE
  WR1: SIZEOF ( QUERY ( pdf <* product_definition_formation | (
    SIZEOF ( QUERY ( prpc <* USEDIN ( pdf. of_product ,
    'CUTTING_TOOL_SCHEMA.'+'PRODUCT_RELATED_PRODUCT_CATEGORY.'+
    'PRODUCTS' ) | prpc. name = 'grade' )
    ) >0 ) AND ( SIZEOF ( QUERY ( pd <* USEDIN ( pdf ,
    'CUTTING_TOOL_SCHEMA.'+'PRODUCT_DEFINITION.'+ 'FORMATION' ) |
    pd. frame_of_reference.name = 'material definition' ) ) <>1
    ) ) ) =0;
END_RULE;
```

#### Argument definitions:

**product\_definition:** the set of all instances of product\_definition.

**product\_definition\_formation:** the set of all instances of product\_definition\_formation.

**product\_related\_product\_category:** the set of all instances of product\_related\_product\_category.

**Formal propositions:**

WR1: Each instance of product\_definition\_formation which has an associated of\_product that is referenced by a product\_related\_product\_category with a name of 'grade', is referenced as formation by exactly one instance of product\_definition that references as its frame\_of\_reference a product\_definition\_context with a name of 'material definition'.

**6.2.3.19 person\_requires\_person\_and\_organization**

The person\_requires\_person\_and\_organization rule ensures that each instance of person shall be referenced by at least one instance of person\_and\_organization as the\_person. This rule enforces the requirement for each person to have at least one organization specified the person belongs to.

**EXPRESS specification:**

```
RULE person_requires_person_and_organization FOR
    (person);
WHERE
    WR1: SIZEOF ( QUERY ( p <* person |      SIZEOF ( USEDIN ( p ,
        'CUTTING_TOOL_SCHEMA.PERSON_AND_ORGANIZATION.THE_PERSON' )
        ) =0 ) ) =0;
END_RULE;
```

**Argument definitions:**

**person:** the set of all instances of person.

**Formal propositions:**

WR1: Each instance of person shall be referenced by at least one instance of person\_and\_organization as the\_person.

**6.2.3.20 plib\_class\_reference\_requires\_version**

The plib\_class\_reference\_requires\_version rule specifies that each instance of externally\_defined\_class that has as its source a known\_source is contained in the set of items of exactly one applied\_external\_identification\_assignment which references as its role an identification\_role with a name of 'version'. This rule enforces the requirement for every Plib\_class\_reference to have exactly one version.

**EXPRESS specification:**

```
RULE plib_class_reference_requires_version FOR
    (externally_defined_class);
WHERE
    WR1: SIZEOF ( QUERY ( edc <* externally_defined_class | (
        'CUTTING_TOOL_SCHEMA.'+ 'KNOWN_SOURCE' IN TYPEOF ( edc.
        source ) ) AND ( SIZEOF ( QUERY ( aei <* USEDIN ( edc ,
        'CUTTING_TOOL_SCHEMA.APPLIED_EXTERNAL_IDENTIFICATION_ASSIGNMENT.ITEMS'
        ) | aei.role.name = 'version' ) ) <>1 ) ) ) =0;
    WR2: SIZEOF ( QUERY ( edc <* externally_defined_class | (
        'CUTTING_TOOL_SCHEMA.'+ 'KNOWN_SOURCE' IN TYPEOF ( edc.
        source ) ) AND ( SIZEOF ( QUERY ( aei <* USEDIN ( edc ,
```

```
'CUTTING_TOOL_SCHEMA.APPLIED_IDENTIFICATION_ASSIGNMENT.ITEMS'
) | aei.role.name = 'version' ) ) >0 ) ) ) =0;
END_RULE;
```

#### **Argument definitions:**

**externally\_defined\_class:** the set of all instances of externally\_defined\_class.

#### **Formal propositions:**

**WR1:** Each instance of externally\_defined\_class that has as its source a known\_source is contained in the set of items of exactly one applied\_external\_identification\_assignment which references as its role an identification\_role with a name of 'version'.

**WR2:** An instance of externally\_defined\_class that has as its source a known\_source may not be contained in the set of items of an applied\_identification\_assignment which references as its role an identification\_role with a name of 'version'.

#### **6.2.3.21 plib\_property\_reference\_requires\_name\_scope**

The plib\_property\_reference\_requires\_name\_scope rule specifies that each instance of externally\_defined\_general\_property that has as its source a known\_source is referenced as the relating\_item by exactly one instance of externally\_defined\_item\_relationship which has a name of 'name scope' and which references as the related\_item an externally\_defined\_class that has as its source a known\_source. This rule enforces the requirement for every plib\_property\_reference to have a name\_scope.

#### **EXPRESS specification:**

```
RULE plib_property_reference_requires_name_scope FOR
  (externally_defined_general_property);
WHERE
  WR1: SIZEOF ( QUERY ( edgp <*
    externally_defined_general_property | (
      'CUTTING_TOOL_SCHEMA.' + 'KNOWN_SOURCE' IN TYPEOF ( edgp.
      source ) ) AND ( SIZEOF ( QUERY ( edir <* USEDIN ( edgp ,
      'CUTTING_TOOL_SCHEMA.' + 'EXTERNALLY_DEFINED_ITEM_RELATIONSHIP.' +
      'RELATING_ITEM' ) | ( edir.name = 'name scope' ) AND (
      'CUTTING_TOOL_SCHEMA.' + 'EXTERNALLY_DEFINED_CLASS' IN TYPEOF
      ( edir.related_item ) ) AND ( 'CUTTING_TOOL_SCHEMA.' +
      'KNOWN_SOURCE' IN TYPEOF ( edir.related_item.source ) ) )
      ) <>1 ) ) ) =0;
END_RULE;
```

#### **Argument definitions:**

**externally\_defined\_general\_property:** the set of all instances of externally\_defined\_general\_property.

#### **Formal propositions:**

**WR1:** Each instance of externally\_defined\_general\_property that has as its source a known\_source is referenced as the relating\_item by exactly one instance of externally\_defined\_item\_relationship which has a name of 'name scope' and which references as the related\_item an externally\_defined\_class that has as its source a known\_source.

#### **6.2.3.22 plib\_property\_reference\_requires\_version**

The plib\_property\_reference\_requires\_version rule specifies that each instance of externally\_defined\_general\_property that has as its source a known\_source is contained in the set of items of

exactly one applied\_external\_identification\_assignment which references as its role an identification\_role with a name of 'version'. This rule enforces the requirement for every plib\_property\_reference to have a version.

**EXPRESS specification:**

```

RULE plib_property_reference_requires_version FOR
  (externally_defined_general_property);
WHERE
  WR1: SIZEOF ( QUERY ( edgp <*
    externally_defined_general_property | (
      'CUTTING_TOOL_SCHEMA.'+ 'KNOWN_SOURCE' IN TYPEOF ( edgp.
      source ) ) AND ( SIZEOF ( QUERY ( edir <* USEDIN ( edgp ,
      'CUTTING_TOOL_SCHEMA.'+'APPLIED_EXTERNAL_IDENTIFICATION_ASSIGNMENT.'+
      'ITEMS' ) | ( edir. role.name = 'version' ) ) ) <>1 ) ) )
    =0;
END_RULE;

```

**Argument definitions:**

**externally\_defined\_general\_property:** the set of all instances of externally\_defined\_general\_property.

**Formal propositions:**

**WR1:** Each instance of externally\_defined\_general\_property that has as its source a known\_source is contained in the set of items of exactly one applied\_external\_identification\_assignment which references as its role an identification\_role with a name of 'version'.

**6.2.3.23 product\_requires\_category**

The product\_requires\_category rule specifies that each instance of product shall be referenced by at least one product\_related\_product\_category.

**EXPRESS specification:**

```

RULE product_requires_category FOR
  (product,
  product_related_product_category);
WHERE
  WR1: SIZEOF ( QUERY ( p <* product | SIZEOF ( USEDIN ( p ,
    'CUTTING_TOOL_SCHEMA.PRODUCT_RELATED_PRODUCT_CATEGORY.PRODUCTS'
    ) ) =0 ) ) =0;
END_RULE;

```

**Argument definitions:**

**product:** the set of all instances of product.

**product\_related\_product\_category:** the set of all instances of product\_related\_product\_category.

**product\_related\_product\_category:** the set of all instances of product\_related\_product\_category.

**Formal propositions:**

**WR1:** Each instance of product shall be referenced by at least one product\_related\_product\_category.

### 6.2.3.24 product\_requires\_id\_owner

The `product_requires_id_owner` rule specifies that each instance of `product` that is referenced by an `product_related_product_category` with a name of 'part', 'tool', or 'raw material' shall be either referenced by an instance of `applied_person_and_organization_assignment` or by an instance of `applied_organization_assignment`. This rule enforces the requirement that for every item an owner shall be specified for its id.

#### **EXPRESS specification:**

```

RULE product_requires_id_owner FOR
  (product);
WHERE
  WR1: SIZEOF ( QUERY ( prod <* product | ( SIZEOF ( QUERY ( prpc
    <* USEDIN ( prod ,
      'CUTTING_TOOL_SCHEMA.PRODUCT_RELATED_PRODUCT_CATEGORY.PRODUCTS'
    ) | prpc.name IN ['part' , 'tool' , 'raw material' ] ) )
    >0 ) AND ( SIZEOF ( QUERY ( apoa <* USEDIN ( prod ,
      'CUTTING_TOOL_SCHEMA.APPLIED_PERSON_AND_ORGANIZATION_ASSIGNMENT.ITEMS'
    ) | apoa\person_and_organization_assignment.role.name<>
      'id owner' ) ) =1 ) ) ) =0; +
    (SIZEOF( QUERY (oa <* USEDIN ( prod,
      'CUTTING_TOOL_SCHEMA.APPLIED_ORGANIZATION_ASSIGNMENT.ITEMS' ) |
      oa\organization_assignment.role.name = 'id owner' ) ) <> 1)
  )
END_RULE;

```

#### **Argument definitions:**

**product:** the set of all instances of `product`.

#### **Formal propositions:**

**WR1:** Each instance of `product` that is referenced by a `product_related_product_category` with a name of 'part', 'tool', or 'raw material' is contained in the set of items of either exactly one `applied_person_and_organization_assignment` which references as its role an `person_and_organization_role` with a name of 'id owner' or exactly one `applied_organization_assignment` which references as its role an `organization_role` with a name of 'id owner'.

### 6.2.3.25 product\_requires\_version

The `product_requires_version` rule ensures that each instance of `product` is referenced by at least one instance of `product_definition_formation`. This rule enforces the requirement for every item to have at least one item version associated to it.

#### **EXPRESS specification:**

```

RULE product_requires_version FOR
  (product);
WHERE
  WR1: SIZEOF ( QUERY ( prod <* product | ( SIZEOF ( USEDIN (
    prod ,
    'CUTTING_TOOL_SCHEMA.'+'PRODUCT_DEFINITION_FORMATION.'+
    'OF_PRODUCT' ) ) =0 ) ) ) =0;
  )
END_RULE;

```

#### **Argument definitions:**

**product:** the set of all instances of `product`.

**Formal propositions:**

**WR1:** For each instance of product there shall be one or more instances of product\_definition\_formation that refers as of\_product attribute to this instance of product.

**6.2.3.26 restrict\_applied\_classification\_assignment\_role**

The restrict\_applied\_classification\_assignment\_role rule ensures the correct correlation between the assigned\_class of the applied\_classification\_assignment and the name of the classification\_role referenced as the role.

**EXPRESS specification:**

```

RULE restrict_applied_classification_assignment_role FOR
  (applied_classification_assignment);
WHERE
  WR1: SIZEOF ( QUERY ( aca <* applied_classification_assignment
    | ( 'CUTTING_TOOL_SCHEMA.'+ 'CLASS' IN TYPEOF ( aca.
      assigned_class ) ) AND NOT ( aca. role.name IN
      ['definitional' , 'non-definitional' , '' ] ) ) ) ) =0;
  WR2: SIZEOF ( QUERY ( aca <* applied_classification_assignment
    | ( aca. role.name IN ['definitional' , 'non-definitional'
      , '' ] ) AND NOT ( 'CUTTING_TOOL_SCHEMA.'+ 'CLASS' IN TYPEOF (
      aca. assigned_class ) ) ) ) ) =0;
  WR3: SIZEOF ( QUERY ( aca <* applied_classification_assignment
    | ( 'CUTTING_TOOL_SCHEMA.'+ 'CLASS_SYSTEM' IN TYPEOF ( aca.
      assigned_class ) ) AND ( aca. role.name<>
      'class system membership' ) ) ) =0;
  WR4: SIZEOF ( QUERY ( aca <* applied_classification_assignment
    | ( aca. role.name = 'class system membership' ) AND NOT (
      'CUTTING_TOOL_SCHEMA.'+ 'CLASS_SYSTEM' IN TYPEOF ( aca.
      assigned_class ) ) ) ) ) =0;
END_RULE;

```

**Argument definitions:**

**applied\_classification\_assignment:** the set of all instances of applied\_classification\_assignment.

**Formal propositions:**

**WR1:** If the assigned\_class of an applied\_classification\_assignment is of type class, the classification\_role referenced by the role attribute has a value of either 'definitional', 'non-definitional', or ''.

**WR2:** If an applied\_classification\_assignment references as its role a classification\_role with a name of either 'definitional', 'non-definitional', or '', the assigned\_class shall be of type class.

**WR3:** If the assigned\_class of an applied\_classification\_assignment is of type class\_system, the classification\_role referenced by the role attribute has a value of 'class system membership'.

**WR4:** If an applied\_classification\_assignment references as its role a classification\_role with a name of 'class system membership', the assigned\_class shall be of type class\_system.

**6.2.3.27 restrict\_class\_system\_assignment\_for\_class**

The restrict\_class\_system\_assignment\_for\_class rule specifies that each instance of class shall be referenced by at most one instance of applied\_classification\_assignment that specifies the classification system for the class.

**EXPRESS specification:**

```

RULE restrict_class_system_assignment_for_class FOR
  (applied_classification_assignment,
   class);
WHERE
  WR1: SIZEOF ( QUERY ( c <* class | SIZEOF ( QUERY ( aca <*
    USEDIN ( c ,
      'CUTTING_TOOL_SCHEMA.'+'CLASSIFICATION_ASSIGNMENT.'+
      'ASSIGNED_CLASS' ) | ( ( aca. role.name =
        'class system membership' ) AND ( 'CUTTING_TOOL_SCHEMA.'+
        'APPLIED_CLASSIFICATION_ASSIGNMENT' IN TYPEOF ( aca ) ) ) )
    ) >1 ) ) =0;
END_RULE;

```

**Argument definitions:**

**applied\_classification\_assignment:** the set of all instances of applied\_classification\_assignment.

**class:** the set of all instances of class.

**Formal propositions:**

**WR1:** For each instance of class there shall be only at most one instance of applied\_classification\_assignment with a role name of 'class system membership' that references that instance of class.

**6.2.3.28 restrict\_class\_system\_assignment\_for\_planar\_extent**

The restrict\_class\_system\_assignment\_for\_planar\_extent rule specifies that each instance of planar\_extent with a name of 'size format' shall be referenced by at most one instance of applied\_classification\_assignment that specifies the referenced standard for the named\_size.

**EXPRESS specification:**

```

RULE restrict_class_system_assignment_for_planar_extent FOR
  (applied_classification_assignment,
   planar_extent);
WHERE
  WR1: SIZEOF ( QUERY ( pe <* planar_extent | ( pe\
    representation_item.name = 'size format' ) AND ( SIZEOF (
    QUERY ( aca <* USEDIN ( pe ,
      'CUTTING_TOOL_SCHEMA.APPLIED_CLASSIFICATION_ASSIGNMENT.ITEMS'
      | aca\ classification_assignment.role.name =
        'class system membership' ) ) >1 ) ) ) =0;
END_RULE;

```

**Argument definitions:**

**applied\_classification\_assignment:** the set of all instances of applied\_classification\_assignment.

**planar\_extent:** the set of all instances of planar\_extent.

**Formal propositions:**

**WR1:** For each instance of planar\_extent with a name of 'size format' there shall be at most one instance of applied\_classification\_assignment with a role name of 'class system membership'.

### 6.2.3.29 restrict\_externally\_defined\_item\_relationship

The restrict\_externally\_defined\_item\_relationship rule ensures the correct correlation between the name of an externally\_defined\_item\_relationship and the related externally\_defined\_items.

#### EXPRESS specification:

```
RULE restrict_externally_defined_item_relationship FOR
  (externally_defined_item_relationship);
WHERE
  WR1: SIZEOF ( QUERY ( edir <*
    externally_defined_item_relationship | ( edir. name =
    'name scope' ) AND ( NOT ( 'CUTTING_TOOL_SCHEMA.'+
    'EXTERNALLY_DEFINED_GENERAL_PROPERTY' IN TYPEOF ( edir.
    relating_item ) ) OR NOT ( 'CUTTING_TOOL_SCHEMA.'+
    'KNOWN_SOURCE' IN TYPEOF ( edir. relating_item.source ) )
    OR NOT ( 'CUTTING_TOOL_SCHEMA.'+ 'EXTERNALLY_DEFINED_CLASS'
    IN TYPEOF ( edir. related_item ) ) OR NOT (
    'CUTTING_TOOL_SCHEMA.'+ 'KNOWN_SOURCE' IN TYPEOF ( edir.
    related_item.source ) ) ) ) ) =0;
END_RULE;
```

#### Argument definitions:

**externally\_defined\_item\_relationship:** the set of all instances of externally\_defined\_item\_relationship.

#### Formal propositions:

**WR1:** If an externally\_defined\_item\_relationship has a name of 'name scope', the relating\_item shall be of type externally\_defined\_general\_property and shall reference as the source a known\_source and the related\_item shall be of type externally\_defined\_class and shall reference as its source a known\_source.

### 6.2.3.30 restrict\_group\_relationship\_for\_general\_classification\_hierarchy

The restrict\_group\_relationship\_for\_general\_classification\_hierarchy rule ensures the correct correlation between the name of a group\_relationship and the related groups of type class.

#### EXPRESS specification:

```
RULE restrict_group_relationship_for_general_classification_hierarchy FOR
  (class,
  group_relationship);
WHERE
  WR1: SIZEOF ( QUERY ( gr <* group_relationship | ( gr. name =
    'class hierarchy' ) AND ( NOT ( 'CUTTING_TOOL_SCHEMA.'+
    'CLASS' IN TYPEOF ( gr. related_group ) ) OR NOT (
    'CUTTING_TOOL_SCHEMA.'+ 'CLASS' IN TYPEOF ( gr.
    relating_group ) ) ) ) ) =0;
END_RULE;
```

#### Argument definitions:

**class:** the set of all instances of class.

**group\_relationship:** the set of all instances of group\_relationship.

#### Formal propositions:

**WR1:** If the group\_relationship has a name of 'class hierarchy', the related\_group and the relating\_group shall be of type class.

### 6.2.3.31 restrict\_multi\_language\_for\_application\_context

The restrict\_multi\_language\_for\_application\_context rule specifies that each instance of application\_context shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

#### **EXPRESS specification:**

```

RULE restrict_multi_language_for_application_context FOR
  (application_context);
WHERE
  WR1: SIZEOF ( QUERY ( ent <* application_context | SIZEOF (
    QUERY ( mlaa <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | ( mlaa. attribute_name<>'description' ) ) ) ) >0
    ) ) =0;
  WR2: SIZEOF ( QUERY ( ent <* application_context | SIZEOF (
    QUERY ( ala <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | ( ala. attribute_name<>'description' ) ) ) ) >0 )
    ) =0;
  WR3: SIZEOF ( QUERY ( ent <* application_context | ( SIZEOF (
    QUERY ( mlaa1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | ( mlaa1. attribute_name =mlaa2. attribute_name
    ) AND ( mlaa1. language =mlaa2. language ) ) ) >1 ) ) +
    SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | ( ala1. attribute_name =ala2. attribute_name )
    ) AND ( ala1. language =ala2. language ) ) ) >1 ) ) ) >0 ) )
    ) =0;
END_RULE;

```

#### **Argument definitions:**

**application\_context:** the set of all instances of application\_context.

#### **Formal propositions:**

**WR1:** Each instance of application\_context shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of 'description'.

**WR2:** Each instance of application\_context shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**WR3:** For each instance of application\_context, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

### 6.2.3.32 restrict\_multi\_language\_for\_descriptive\_representation\_item

The restrict\_multi\_language\_for\_descriptive\_representation\_item rule specifies that each instance of descriptive\_representation\_item shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

#### **EXPRESS specification:**

```

RULE restrict_multi_language_for_descriptive_representation_item FOR
  (descriptive_representation_item);
WHERE
  WR1: SIZEOF ( QUERY ( ent <* descriptive_representation_item |
    SIZEOF ( QUERY ( mlaa <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+'
      'ITEMS' ) | ( mlaa.attribute_name<>'description' ) ) ) ) >0
    ) ) =0;
  WR2: SIZEOF ( QUERY ( ent <* descriptive_representation_item |
    SIZEOF ( QUERY ( ala <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+'
      'ITEMS' ) | ( ala.attribute_name<>'description' ) ) ) ) >0 )
    ) =0;
  WR3: SIZEOF ( QUERY ( ent <* descriptive_representation_item |
    ( SIZEOF ( QUERY ( mlaa1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+'
      'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+'
      'ITEMS' ) | ( mlaa1.attribute_name =mlaa2.attribute_name
      ) AND ( mlaa1.language =mlaa2.language ) ) ) >1 ) ) +
    SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+'
      'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+'
      'ITEMS' ) | ( ala1.attribute_name =ala2.attribute_name )
      AND ( ala1.language =ala2.language ) ) ) >1 ) ) ) >0 ) )
    ) =0;
END_RULE;

```

#### **Argument definitions:**

**descriptive\_representation\_item:** the set of all instances of descriptive\_representation\_item.

#### **Formal propositions:**

**WR1:** Each instance of descriptive\_representation\_item shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of 'description'.

**WR2:** Each instance of descriptive\_representation\_item shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**WR3:** For each instance of descriptive\_representation\_item, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

### 6.2.3.33 restrict\_multi\_language\_for\_effectivity

The `restrict_multi_language_for_effectivity` rule specifies that each instance of `effectivity` shall only be referenced by instances of `multi_language_attribute_assignment` or `attribute_language_assignment` that have an attribute `attribute_name` of 'description'.

#### **EXPRESS specification:**

```

RULE restrict_multi_language_for_effectivity FOR
  (effectivity);
WHERE
  WR1: SIZEOF ( QUERY ( ent <* effectivity | SIZEOF ( QUERY (
    mlaa <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | ( mlaa.attribute_name<>'description' ) ) ) ) >0
    ) ) =0;
  WR2: SIZEOF ( QUERY ( ent <* effectivity | SIZEOF ( QUERY ( ala
    <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | ( ala.attribute_name<>'description' ) ) ) ) >0 )
    ) =0;
  WR3: SIZEOF ( QUERY ( ent <* effectivity | ( SIZEOF ( QUERY (
    mlaa1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | ( mlaa1.attribute_name =mlaa2.attribute_name
      ) AND ( mlaa1.language =mlaa2.language ) ) ) >1 ) ) +
    SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | ( ala1.attribute_name =ala2.attribute_name )
      AND ( ala1.language =ala2.language ) ) ) >1 ) ) ) ) >0 ) )
    ) =0;
END_RULE;

```

#### **Argument definitions:**

**effectivity:** the set of all instances of `effectivity`.

#### **Formal propositions:**

**WR1:** Each instance of `effectivity` shall be only in the set of items of those instances of `multi_language_attribute_assignment` that have an attribute `attribute_name` of 'description'.

**WR2:** Each instance of `effectivity` shall be only in the set of items of those instances of `attribute_language_assignment` that have an attribute `attribute_name` of 'description'.

**WR3:** For each instance of `effectivity`, there shall not be more than one `multi_language_attribute_assignment` or `attribute_language_assignment` with the same `language` and the same `attribute_name`.

### 6.2.3.34 restrict\_multi\_language\_for\_effectivity\_relationship

The `restrict_multi_language_for_effectivity_relationship` rule specifies that each instance of `effectivity_relationship` shall only be referenced by instances of `multi_language_attribute_assignment` or `attribute_language_assignment` that have an attribute `attribute_name` of 'description'.

**EXPRESS specification:**

```

RULE restrict_multi_language_for_effectivity_relationship FOR
  (effectivity_relationship);
WHERE
  WR1: SIZEOF ( QUERY ( ent <* effectivity_relationship | SIZEOF
    ( QUERY ( mlaa <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | ( mlaa. attribute_name<>'description' ) ) ) ) >0
    ) ) =0;
  WR2: SIZEOF ( QUERY ( ent <* effectivity_relationship | SIZEOF
    ( QUERY ( ala <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | ( ala. attribute_name<>'description' ) ) ) ) >0 )
    ) =0;
  WR3: SIZEOF ( QUERY ( ent <* effectivity_relationship | (
    SIZEOF ( QUERY ( mlaa1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | ( mlaa1. attribute_name =mlaa2. attribute_name
      ) AND ( mlaa1. language =mlaa2. language ) ) ) >1 ) ) +
    SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | ( ala1. attribute_name =ala2. attribute_name )
      AND ( ala1. language =ala2. language ) ) ) >1 ) ) ) >0 ) )
    ) =0;
END_RULE;

```

**Argument definitions:**

**effectivity\_relationship:** the set of all instances of effectivity\_relationship.

**Formal propositions:**

**WR1:** Each instance of effectivity\_relationship shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of 'description'.

**WR2:** Each instance of effectivity\_relationship shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**WR3:** For each instance of effectivity\_relationship, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

**6.2.3.35 restrict\_multi\_language\_for\_external\_source**

The restrict\_multi\_language\_for\_external\_source rule specifies that each instance of external\_source shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**EXPRESS specification:**

```

RULE restrict_multi_language_for_external_source FOR
  (external_source);
WHERE

```

```

WR1: SIZEOF ( QUERY ( ent <* external_source | SIZEOF ( QUERY (
  mlaa <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
    'ITEMS' ) | ( mlaa. attribute_name<>'description' ) ) ) >0
  ) ) =0;
WR2: SIZEOF ( QUERY ( ent <* external_source | SIZEOF ( QUERY (
  ala <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
    'ITEMS' ) | ( ala. attribute_name<>'description' ) ) ) >0 )
  ) =0;
WR3: SIZEOF ( QUERY ( ent <* external_source | ( SIZEOF ( QUERY
  ( mlaa1 <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
    'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
    'ITEMS' ) | ( mlaa1. attribute_name =mlaa2. attribute_name
    ) AND ( mlaa1. language =mlaa2. language ) ) ) >1 ) +
  SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
    'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
    'ITEMS' ) | ( ala1. attribute_name =ala2. attribute_name )
    AND ( ala1. language =ala2. language ) ) ) >1 ) ) ) >0 ) )
  ) =0;
END_RULE;

```

**Argument definitions:**

**external\_source:** the set of all instances of external\_source.

**Formal propositions:**

**WR1:** Each instance of external\_source shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of 'description'.

**WR2:** Each instance of external\_source shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**WR3:** For each instance of external\_source, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

**6.2.3.36 restrict\_multi\_language\_for\_general\_feature**

The restrict\_multi\_language\_for\_general\_feature rule specifies that each instance of general\_feature shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**EXPRESS specification:**

```

RULE restrict_multi_language_for_general_feature FOR
  (general_feature);
WHERE
  WR1: SIZEOF ( QUERY ( ent <* general_feature | SIZEOF ( QUERY (
    mlaa <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | ( mlaa. attribute_name<>'description' ) ) ) >0
    ) ) =0;

```

```

WR2: SIZEOF ( QUERY ( ent <* general_feature | SIZEOF ( QUERY (
    ala <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
    'ITEMS' ) | ( ala.attribute_name<>'description' ) ) ) >0 )
) =0;
WR3: SIZEOF ( QUERY ( ent <* general_feature | ( SIZEOF ( QUERY
( mlaa1 <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
    'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
    'ITEMS' ) | ( mlaa1.attribute_name =mlaa2.attribute_name
) AND ( mlaa1.language =mlaa2.language ) ) ) >1 ) ) +
SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
    'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
    'ITEMS' ) | ( ala1.attribute_name =ala2.attribute_name )
AND ( ala1.language =ala2.language ) ) ) >1 ) ) ) >0 ) ) )
=0;
END_RULE;

```

**Argument definitions:**

**general\_feature:** the set of all instances of general\_feature.

**Formal propositions:**

**WR1:** Each instance of general\_feature shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of 'description'.

**WR2:** Each instance of general\_feature shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**WR3:** For each instance of general\_feature, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

**6.2.3.37 restrict\_multi\_language\_for\_general\_property**

The restrict\_multi\_language\_for\_general\_property rule specifies that each instance of general\_property shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**EXPRESS specification:**

```

RULE restrict_multi_language_for_general_property FOR
    (general_property);
WHERE
WR1: SIZEOF ( QUERY ( ent <* general_property | SIZEOF ( QUERY
    ( mlaa <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
    'ITEMS' ) | ( mlaa.attribute_name<>'description' ) ) ) >0
) ) =0;
WR2: SIZEOF ( QUERY ( ent <* general_property | SIZEOF ( QUERY
    ( ala <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
    'ITEMS' ) | ( ala.attribute_name<>'description' ) ) ) >0 )
) =0;

```

```

WR3: SIZEOF ( QUERY ( ent <* general_property | ( SIZEOF (
  QUERY ( mlaa1 <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
    'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
    'ITEMS' ) | ( mlaa1. attribute_name =mlaa2. attribute_name
    ) AND ( mlaa1. language =mlaa2. language ) ) ) >1 ) ) +
  SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
    'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
    'ITEMS' ) | ( ala1. attribute_name =ala2. attribute_name )
    AND ( ala1. language =ala2. language ) ) ) >1 ) ) ) >0 ) )
=0;
END_RULE;

```

**Argument definitions:**

**general\_property:** the set of all instances of general\_property.

**Formal propositions:**

**WR1:** Each instance of general\_property shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of 'description'.

**WR2:** Each instance of general\_property shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**WR3:** For each instance of general\_property, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

**6.2.3.38 restrict\_multi\_language\_for\_general\_property\_relationship**

The restrict\_multi\_language\_for\_general\_property\_relationship rule specifies that each instance of general\_property\_relationship shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**EXPRESS specification:**

```

RULE restrict_multi_language_for_general_property_relationship FOR
  (general_property_relationship);
WHERE
  WR1: SIZEOF ( QUERY ( ent <* general_property_relationship |
    SIZEOF ( QUERY ( mlaa <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | ( mlaa. attribute_name<>'description' ) ) ) >0
    ) ) =0;
  WR2: SIZEOF ( QUERY ( ent <* general_property_relationship |
    SIZEOF ( QUERY ( ala <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | ( ala. attribute_name<>'description' ) ) ) >0 )
    ) =0;
  WR3: SIZEOF ( QUERY ( ent <* general_property_relationship | (
    SIZEOF ( QUERY ( mlaa1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+

```

```

'ITEMS' ) | ( mlaa1. attribute_name =mlaa2. attribute_name
) AND ( mlaa1. language =mlaa2. language ) ) ) >1 ) ) +
SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
'ITEMS' ) | ( ala1. attribute_name =ala2. attribute_name )
AND ( ala1. language =ala2. language ) ) ) >1 ) ) ) >0 ) )
=0;
END_RULE;

```

**Argument definitions:**

**general\_property\_relationship:** the set of all instances of general\_property\_relationship.

**Formal propositions:**

**WR1:** Each instance of general\_property\_relationship shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of 'description'.

**WR2:** Each instance of general\_property\_relationship shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**WR3:** For each instance of general\_property\_relationship, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

**6.2.3.39 restrict\_multi\_language\_for\_group**

The restrict\_multi\_language\_for\_group rule specifies that each instance of group shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of either 'description' or 'name'.

**EXPRESS specification:**

```

RULE restrict_multi_language_for_group FOR
(group);
WHERE
WR1: SIZEOF ( QUERY ( ent <* group | SIZEOF ( QUERY ( mlaa <*
USEDIN ( ent
'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
'ITEMS' ) | ( mlaa. attribute_name<>'description' ) AND (
mlaa. attribute_name<>'name' ) ) ) >0 ) ) =0;
WR2: SIZEOF ( QUERY ( ent <* group | SIZEOF ( QUERY ( ala <*
USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
'ITEMS' ) | ( ala. attribute_name<>'description' ) AND (
ala. attribute_name<>'name' ) ) ) >0 ) ) =0;
WR3: SIZEOF ( QUERY ( ent <* group | ( SIZEOF ( QUERY ( mlaa1
<* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
'ITEMS' ) | ( mlaa1. attribute_name =mlaa2. attribute_name
) AND ( mlaa1. language =mlaa2. language ) ) ) >1 ) ) +
SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,

```

```

'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
'ITEMS' ) | ( ala1. attribute_name =ala2. attribute_name )
AND ( ala1. language =ala2. language ) ) ) >1 ) ) ) >0 ) )
=0;
END_RULE;

```

**Argument definitions:**

**group:** the set of all instances of group.

**Formal propositions:**

**WR1:** Each instance of group shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of either 'description' or 'name'.

**WR2:** Each instance of group shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of either 'description' or 'name'.

**WR3:** For each instance of group, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

**6.2.3.40 restrict\_multi\_language\_for\_identification\_role**

The restrict\_multi\_language\_for\_identification\_role rule specifies that each instance of identification\_role shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**EXPRESS specification:**

```

RULE restrict_multi_language_for_identification_role FOR
  (identification_role);
WHERE
  WR1: SIZEOF ( QUERY ( ent <* identification_role | SIZEOF (
    QUERY ( mlaa <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | ( mlaa. attribute_name<>'description' ) ) ) ) >0
    ) ) =0;
  WR2: SIZEOF ( QUERY ( ent <* identification_role | SIZEOF (
    QUERY ( ala <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | ( ala. attribute_name<>'description' ) ) ) ) >0 )
    ) =0;
  WR3: SIZEOF ( QUERY ( ent <* identification_role | ( SIZEOF (
    QUERY ( mlaa1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
        'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
        'ITEMS' ) | ( mlaa1. attribute_name =mlaa2. attribute_name
        ) AND ( mlaa1. language =mlaa2. language ) ) ) >1 ) ) +
    SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
        'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
        'ITEMS' ) | ( ala1. attribute_name =ala2. attribute_name
        ) AND ( ala1. language =ala2. language ) ) ) >1 ) ) ) >0 ) )
    ) =0;
END_RULE;

```

**Argument definitions:**

**identification\_role:** the set of all instances of identification\_role.

**Formal propositions:**

**WR1:** Each instance of identification\_role shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of 'description'.

**WR2:** Each instance of identification\_role shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**WR3:** For each instance of identification\_role, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

**6.2.3.41 restrict\_multi\_language\_for\_organization\_role**

The restrict\_multi\_language\_for\_organization\_role rule specifies that each instance of organization\_role shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**EXPRESS specification:**

```

RULE restrict_multi_language_for_organization_role FOR
  (organization_role);
WHERE
  WR1: SIZEOF ( QUERY ( ent <* organization_role | SIZEOF ( QUERY
    ( mlaa <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | ( mlaa. attribute_name<>'description' ) ) ) ) >0
    ) ) =0;
  WR2: SIZEOF ( QUERY ( ent <* organization_role | SIZEOF ( QUERY
    ( ala <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | ( ala. attribute_name<>'description' ) ) ) ) >0 )
    ) =0;
  WR3: SIZEOF ( QUERY (ent <* organization_role | ( SIZEOF (
    QUERY ( mlaa1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | ( mlaa1. attribute_name =mlaa2. attribute_name
    ) AND ( mlaa1. language =mlaa2. language ) ) ) >1 ) ) +
    SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | ( ala1. attribute_name =ala2. attribute_name )
    ) AND ( ala1. language =ala2. language ) ) ) >1 ) ) ) >0 ) )
    ) =0;
END_RULE;

```

**Argument definitions:**

**organization\_role:** the set of all instances of organization\_role.

**Formal propositions:**

**WR1:** Each instance of organization\_role shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of 'description'.

**WR2:** Each instance of organization\_role shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**WR3:** For each instance of organization\_role, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

#### 6.2.3.42 restrict\_multi\_language\_for\_person\_and\_organization\_role

The restrict\_multi\_language\_for\_person\_and\_organization\_role rule specifies that each instance of person\_and\_organization\_role shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

##### **EXPRESS specification:**

```

RULE restrict_multi_language_for_person_and_organization_role FOR
  (person_and_organization_role);
WHERE
  WR1: SIZEOF ( QUERY ( ent <* person_and_organization_role |
    SIZEOF ( QUERY ( mlaa <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | ( mlaa.attribute_name<>'description' ) ) ) >0
    ) ) =0;
  WR2: SIZEOF ( QUERY ( ent <* person_and_organization_role |
    SIZEOF ( QUERY ( ala <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | ( ala.attribute_name<>'description' ) ) ) >0 )
    ) =0;
  WR3: SIZEOF ( QUERY ( ent <* person_and_organization_role | (
    SIZEOF ( QUERY ( mlaa1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | ( mlaa1.attribute_name =mlaa2.attribute_name
      ) AND ( mlaa1.language =mlaa2.language ) ) ) >1 ) ) +
    SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | ( ala1.attribute_name =ala2.attribute_name )
      AND ( ala1.language =ala2.language ) ) ) >1 ) ) ) >0 ) )
    ) =0;
  END_RULE;

```

##### **Argument definitions:**

**person\_and\_organization\_role:** the set of all instances of person\_and\_organization\_role.

##### **Formal propositions:**

**WR1:** Each instance of person\_and\_organization\_role shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of 'description'.

**WR2:** Each instance of person\_and\_organization\_role shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**WR3:** For each instance of `person_and_organization_role`, there shall not be more than one `multi_language_attribute_assignment` or `attribute_language_assignment` with the same language and the same `attribute_name`.

### 6.2.3.43 restrict\_multi\_language\_for\_product

The `restrict_multi_language_for_product` rule specifies that each instance of `product` shall only be referenced by instances of `multi_language_attribute_assignment` or `attribute_language_assignment` that have an `attribute_name` of either 'description' or 'name'.

#### **EXPRESS specification:**

```

RULE restrict_multi_language_for_product FOR
    (product);
WHERE
    WR1: SIZEOF ( QUERY ( ent <* product | SIZEOF ( QUERY ( mlaa <*
        USEDIN ( ent ,
        'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
        'ITEMS' ) | ( mlaa. attribute_name<>'description' ) AND (
        mlaa. attribute_name<>'name' ) ) ) >0 ) ) =0;
    WR2: SIZEOF ( QUERY ( ent <* product | SIZEOF ( QUERY ( ala <*
        USEDIN ( ent ,
        'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
        'ITEMS' ) | ( ala. attribute_name<>'description' ) AND (
        ala. attribute_name<>'name' ) ) ) >0 ) ) =0;
    WR3: SIZEOF ( QUERY ( ent <* product | ( SIZEOF ( QUERY ( mlaa1
        <* USEDIN ( ent ,
        'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
        'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
        'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
        'ITEMS' ) | ( mlaa1. attribute_name =mlaa2. attribute_name
        ) AND ( mlaa1. language =mlaa2. language ) ) ) >1 ) ) +
        SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
        'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
        'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
        'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
        'ITEMS' ) | ( ala1. attribute_name =ala2. attribute_name )
        AND ( ala1. language =ala2. language ) ) ) >1 ) ) ) >0 ) )
        =0;
END_RULE;

```

#### **Argument definitions:**

**product:** the set of all instances of `product`.

#### **Formal propositions:**

**WR1:** Each instance of `product` shall be only in the set of items of those instances of `multi_language_attribute_assignment` that have an `attribute_name` of either 'description' or 'name'.

**WR2:** Each instance of `product` shall be only in the set of items of those instances of `attribute_language_assignment` that have an `attribute_name` of either 'description' or 'name'.

**WR3:** For each instance of `product`, there shall not be more than one `multi_language_attribute_assignment` or `attribute_language_assignment` with the same language and the same `attribute_name`.

#### 6.2.3.44 restrict\_multi\_language\_for\_product\_definition

The restrict\_multi\_language\_for\_product\_definition rule specifies that each instance of product\_definition shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of either 'description' or 'name'.

##### **EXPRESS specification:**

```

RULE restrict_multi_language_for_product_definition FOR
  (product_definition);
WHERE
  WR1: SIZEOF ( QUERY ( ent <* product_definition | SIZEOF (
    QUERY ( mlaa <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | ( mlaa.attribute_name<>'description' ) AND (
      mlaa.attribute_name<>'name' ) ) ) >0 ) ) =0;
  WR2: SIZEOF ( QUERY ( ent <* product_definition | SIZEOF (
    QUERY ( ala <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | ( ala.attribute_name<>'description' ) AND (
      ala.attribute_name<>'name' ) ) ) >0 ) ) =0;
  WR3: SIZEOF ( QUERY ( ent <* product_definition | ( SIZEOF (
    QUERY ( mlaa1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | ( mlaa1.attribute_name =mlaa2.attribute_name
      ) AND ( mlaa1.language =mlaa2.language ) ) ) >1 ) ) +
    SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | ( ala1.attribute_name =ala2.attribute_name
      ) AND ( ala1.language =ala2.language ) ) ) >1 ) ) ) >0 ) )
    =0;
  END_RULE;

```

##### **Argument definitions:**

**product\_definition:** the set of all instances of product\_definition.

##### **Formal propositions:**

**WR1:** Each instance of product\_definition shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of either 'description' or 'name'.

**WR2:** Each instance of product\_definition shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of either 'description' or 'name'.

**WR3:** For each instance of product\_definition, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

#### 6.2.3.45 restrict\_multi\_language\_for\_product\_definition\_formation

The restrict\_multi\_language\_for\_product\_definition\_formation rule specifies that each instance of product\_definition\_formation shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**EXPRESS specification:**

```

RULE restrict_multi_language_for_product_definition_formation FOR
  (product_definition_formation);
WHERE
  WR1: SIZEOF ( QUERY ( ent <* product_definition_formation |
    SIZEOF ( QUERY ( mlaa <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | ( mlaa. attribute_name<>'description' ) ) ) ) >0
    ) ) =0;
  WR2: SIZEOF ( QUERY ( ent <* product_definition_formation |
    SIZEOF ( QUERY ( ala <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | ( ala. attribute_name<>'description' ) ) ) ) >0 )
    ) =0;
  WR3: SIZEOF ( QUERY ( ent <* product_definition_formation | (
    SIZEOF ( QUERY ( mlaa1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
      'ITEMS' ) | ( mlaa1. attribute_name =mlaa2. attribute_name
      ) AND ( mlaa1. language =mlaa2. language ) ) ) >1 ) ) +
    SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
      'ITEMS' ) | ( ala1. attribute_name =ala2. attribute_name )
      AND ( ala1. language =ala2. language ) ) ) >1 ) ) ) >0 ) )
    ) =0;
END_RULE;

```

**Argument definitions:**

**product\_definition\_formation:** the set of all instances of product\_definition\_formation.

**Formal propositions:**

**WR1:** Each instance of product\_definition\_formation shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of 'description'.

**WR2:** Each instance of product\_definition\_formation shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**WR3:** For each instance of product\_definition\_formation, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

**6.2.3.46 restrict\_multi\_language\_for\_product\_definition\_formation\_relationship**

The restrict\_multi\_language\_for\_product\_definition\_formation\_relationship rule specifies that each instance of product\_definition\_formation\_relationship shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**EXPRESS specification:**

```

RULE restrict_multi_language_for_product_definition_formation_relationship
  FOR

```

```

(product_definition_formation_relationship);
WHERE
WR1: SIZEOF ( QUERY ( ent <*
product_definition_formation_relationship | SIZEOF ( QUERY
( mlaa <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
'ITEMS' ) | ( mlaa.attribute_name<>'description' ) ) ) >0
) ) =0;
WR2: SIZEOF ( QUERY ( ent <*
product_definition_formation_relationship | SIZEOF ( QUERY
( ala <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
'ITEMS' ) | ( ala.attribute_name<>'description' ) ) ) >0
) ) =0;
WR3: SIZEOF ( QUERY ( ent <*
product_definition_formation_relationship | ( SIZEOF (
QUERY ( mlaa1 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
'ITEMS' ) | ( mlaa1.attribute_name =mlaa2.attribute_name
) AND ( mlaa1.language =mlaa2.language ) ) ) >1 ) ) +
SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
'ITEMS' ) | ( ala1.attribute_name =ala2.attribute_name
) AND ( ala1.language =ala2.language ) ) ) >1 ) ) ) >0 ) )
=0;
END_RULE;

```

### **Argument definitions:**

**product\_definition\_formation\_relationship:** the set of all instances of product\_definition\_formation\_relationship.

### **Formal propositions:**

**WR1:** Each instance of product\_definition\_formation\_relationship shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of 'description'.

**WR2:** Each instance of product\_definition\_formation\_relationship shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**WR3:** For each instance of product\_definition\_formation\_relationship, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

#### **6.2.3.47 restrict\_multi\_language\_for\_product\_definition\_relationship**

The restrict\_multi\_language\_for\_product\_definition\_relationship rule specifies that each instance of product\_definition\_relationship shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

### **EXPRESS specification:**

```

RULE restrict_multi_language_for_product_definition_relationship FOR
(product_definition_relationship);

```

WHERE

```

WR1: SIZEOF ( QUERY ( ent <* product_definition_relationship |
  SIZEOF ( QUERY ( mlaa <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
    'ITEMS' ) | ( mlaa. attribute_name<>'description' ) ) ) >0
  ) ) =0;
WR2: SIZEOF ( QUERY ( ent <* product_definition_relationship |
  SIZEOF ( QUERY ( ala <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
    'ITEMS' ) | ( ala. attribute_name<>'description' ) ) ) >0 )
  ) ) =0;
WR3: SIZEOF ( QUERY ( ent <* product_definition_relationship |
  ( SIZEOF ( QUERY ( mlaa1 <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
    'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
    'ITEMS' ) | ( mlaa1. attribute_name =mlaa2. attribute_name
    ) AND ( mlaa1. language =mlaa2. language ) ) ) >1 ) ) +
  SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
    'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
    'ITEMS' ) | ( ala1. attribute_name =ala2. attribute_name )
    AND ( ala1. language =ala2. language ) ) ) >1 ) ) ) >0 ) )
  ) ) =0;
END_RULE;

```

#### **Argument definitions:**

**product\_definition\_relationship:** the set of all instances of product\_definition\_relationship.

#### **Formal propositions:**

**WR1:** Each instance of product\_definition\_relationship shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of 'description'.

**WR2:** Each instance of product\_definition\_relationship shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**WR3:** For each instance of product\_definition\_relationship, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

#### **6.2.3.48 restrict\_multi\_language\_for\_product\_related\_product\_category**

The restrict\_multi\_language\_for\_product\_related\_product\_category rule specifies that each instance of product\_related\_product\_category shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

#### **EXPRESS specification:**

```

RULE restrict_multi_language_for_product_related_product_category FOR
  (product_related_product_category);
WHERE
  WR1: SIZEOF ( QUERY ( ent <* product_related_product_category |
    SIZEOF ( QUERY ( mlaa <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+

```

```

        'ITEMS' ) | ( mlaa. attribute_name<>'description' ) ) ) >0
    ) ) =0;
WR2: SIZEOF ( QUERY ( ent <* product_related_product_category |
    SIZEOF ( QUERY ( ala <* USEDIN ( ent ,
        'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
        'ITEMS' ) | ( ala. attribute_name<>'description' ) ) ) >0 )
    ) =0;
WR3: SIZEOF ( QUERY ( ent <* product_related_product_category |
    ( SIZEOF ( QUERY ( mlaa1 <* USEDIN ( ent ,
        'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
        'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
        'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
        'ITEMS' ) | ( mlaa1. attribute_name =mlaa2. attribute_name
        ) AND ( mlaa1. language =mlaa2. language ) ) ) >1 ) ) +
    SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
        'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
        'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
        'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
        'ITEMS' ) | ( ala1. attribute_name =ala2. attribute_name )
        AND ( ala1. language =ala2. language ) ) ) >1 ) ) ) >0 ) )
    =0;
END_RULE;

```

### **Argument definitions:**

**product\_related\_product\_category:** the set of all instances of product\_related\_product\_category.

### **Formal propositions:**

**WR1:** Each instance of product\_related\_product\_category shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of 'description'.

**WR2:** Each instance of product\_related\_product\_category shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**WR3:** For each instance of product\_related\_product\_category, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

### **6.2.3.49 restrict\_multi\_language\_for\_property\_definition**

The restrict\_multi\_language\_for\_property\_definition rule specifies that each instance of property\_definition shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of either 'description' or 'name'.

### **EXPRESS specification:**

```

RULE restrict_multi_language_for_property_definition FOR
    (property_definition);
WHERE
    WR1: SIZEOF ( QUERY ( ent <* property_definition | SIZEOF (
        QUERY ( mlaa <* USEDIN ( ent ,
            'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+
            'ITEMS' ) | ( mlaa. attribute_name<>'description' ) AND (
            mlaa. attribute_name<>'name' ) ) ) >0 ) ) =0;
    WR2: SIZEOF ( QUERY ( ent <* property_definition | SIZEOF (
        QUERY ( ala <* USEDIN ( ent ,
            'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+

```

```

'ITEMS' ) | ( ala. attribute_name<>'description' ) AND (
ala. attribute_name<>'name' ) ) ) >0 ) ) =0;
WR3: SIZEOF ( QUERY ( ent <* property_definition | ( SIZEOF (
QUERY ( mlaa1 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+'
'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+'
'ITEMS' ) | ( mlaa1. attribute_name =mlaa2. attribute_name
) AND ( mlaa1. language =mlaa2. language ) ) ) >1 ) ) +
SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+'
'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+'
'ITEMS' ) | ( ala1. attribute_name =ala2. attribute_name )
AND ( ala1. language =ala2. language ) ) ) >1 ) ) ) >0 ) )
=0;
END_RULE;

```

**Argument definitions:**

**property\_definition:** the set of all instances of property\_definition.

**Formal propositions:**

**WR1:** Each instance of property\_definition shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of either 'description' or 'name'.

**WR2:** Each instance of property\_definition shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of either 'description' or 'name'.

**WR3:** For each instance of property\_definition, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

**6.2.3.50 restrict\_multi\_language\_for\_representation**

The restrict\_multi\_language\_for\_representation rule specifies that each instance of representation shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of either 'description' or 'name'.

**EXPRESS specification:**

```

RULE restrict_multi_language_for_representation FOR
(representation);
WHERE
WR1: SIZEOF ( QUERY ( ent <* representation | SIZEOF ( QUERY (
mlaa <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+'
'ITEMS' ) | ( mlaa. attribute_name<>'description' ) AND (
mlaa. attribute_name<>'name' ) ) ) >0 ) ) =0;
WR2: SIZEOF ( QUERY ( ent <* representation | SIZEOF ( QUERY (
ala <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+'
'ITEMS' ) | ( ala. attribute_name<>'description' ) AND (
ala. attribute_name<>'name' ) ) ) >0 ) ) =0;
WR3: SIZEOF ( QUERY ( ent <* representation | ( SIZEOF ( QUERY
( mlaa1 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+'

```

```

'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+'
'ITEMS' ) | ( mlaa1. attribute_name =mlaa2. attribute_name
) AND ( mlaa1. language =mlaa2. language ) ) ) >1 ) ) +
SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+'
'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+'
'ITEMS' ) | ( ala1. attribute_name =ala2. attribute_name )
AND ( ala1. language =ala2. language ) ) ) >1 ) ) ) >0 ) )
=0;
END_RULE;

```

**Argument definitions:**

**representation:** the set of all instances of representation.

**Formal propositions:**

**WR1:** Each instance of representation shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of either 'description' or 'name'.

**WR2:** Each instance of representation shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of either 'description' or 'name'.

**WR3:** For each instance of representation, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

**6.2.3.51 restrict\_multi\_language\_for\_representation\_relationship**

The restrict\_multi\_language\_for\_representation\_relationship rule specifies that each instance of representation\_relationship shall only be referenced by instances of multi\_language\_attribute\_assignment or attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**EXPRESS specification:**

```

RULE restrict_multi_language_for_representation_relationship FOR
(representation_relationship);
WHERE
WR1: SIZEOF ( QUERY ( ent <* representation_relationship |
SIZEOF ( QUERY ( mlaa <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+'
'ITEMS' ) | ( mlaa. attribute_name<>'description' ) ) ) >0
) ) =0;
WR2: SIZEOF ( QUERY ( ent <* representation_relationship |
SIZEOF ( QUERY ( ala <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+'
'ITEMS' ) | ( ala. attribute_name<>'description' ) ) ) >0 )
) =0;
WR3: SIZEOF ( QUERY ( ent <* representation_relationship | (
SIZEOF ( QUERY ( mlaa1 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+'
'ITEMS' ) | SIZEOF ( QUERY ( mlaa2 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.'+'
'ITEMS' ) | ( mlaa1. attribute_name =mlaa2. attribute_name
) AND ( mlaa1. language =mlaa2. language ) ) ) >1 ) ) +
SIZEOF ( QUERY ( ala1 <* USEDIN ( ent ,

```

```
'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
'ITEMS' ) | SIZEOF ( QUERY ( ala2 <* USEDIN ( ent ,
'CUTTING_TOOL_SCHEMA.'+'ATTRIBUTE_LANGUAGE_ASSIGNMENT.'+
'ITEMS' ) | ( ala1. attribute_name =ala2. attribute_name )
AND ( ala1. language =ala2. language ) ) ) >1 ) ) >0 ) )
=0;
END_RULE;
```

**Argument definitions:**

**representation\_relationship:** the set of all instances of representation\_relationship.

**Formal propositions:**

**WR1:** Each instance of representation\_relationship shall be only in the set of items of those instances of multi\_language\_attribute\_assignment that have an attribute attribute\_name of 'description'.

**WR2:** Each instance of representation\_relationship shall be only in the set of items of those instances of attribute\_language\_assignment that have an attribute attribute\_name of 'description'.

**WR3:** For each instance of representation\_relationship, there shall not be more than one multi\_language\_attribute\_assignment or attribute\_language\_assignment with the same language and the same attribute\_name.

**6.2.3.52 restrict\_name\_for\_known\_source**

The restrict\_name\_for\_known\_source rule ensures that each instance of known\_source has a name of 'ISO 13584 library'.

**EXPRESS specification:**

```
RULE restrict_name_for_known_source FOR
(known_source);
WHERE
WR1: SIZEOF ( QUERY ( ks <* known_source | ( ks. name<>
'ISO 13584 library' ) ) ) =0;
END_RULE;
```

**Argument definitions:**

**known\_source:** the set of all instances of known\_source.

**Formal propositions:**

**WR1:** Each instance of known\_source shall have a name of 'ISO 13584 library'.

**6.2.3.53 restrict\_part\_occurrence**

The restrict\_part\_occurrence rule specifies the restrictions that apply to product\_definitions which reference as frame\_of\_reference a product\_definition\_context with a name of 'part occurrence'.

**EXPRESS specification:**

```
RULE restrict_part_occurrence FOR
(product_definition,
product_definition_occurrence_relationship);
WHERE
WR1: SIZEOF ( QUERY ( pd <* product_definition | ( pd.
```

```

frame_of_reference.name = 'part occurrence' ) AND ( NOT (
pd.name IN [ 'single instance' ,
'quantified instance' ] ) ) ) ) =0;
WR2: SIZEOF ( QUERY ( pd <* product_definition | ( pd.
frame_of_reference.name = 'part occurrence' ) AND ( SIZEOF
( QUERY ( cd <* USEDIN ( pd ,
'CUTTING_TOOL_SCHEMA.'+'PRODUCT_DEFINITION_RELATIONSHIP.'+
'RELATED_PRODUCT_DEFINITION' ) | ( 'CUTTING_TOOL_SCHEMA.'+
'PRODUCT_DEFINITION_USAGE' IN TYPEOF ( cd ) ) ) ) ) =0 ) AND
( SIZEOF ( USEDIN ( pd ,
'CUTTING_TOOL_SCHEMA.'+'PRODUCT_DEFINITION_OCCURRENCE_RELATIONSHIP.'+
'OCCURRENCE' ) ) ) =0 ) ) ) =0;
WR3: SIZEOF ( QUERY ( pd <* product_definition | ( pd.
frame_of_reference.name = 'part occurrence' ) AND ( pd.
name = 'quantified instance' ) AND ( SIZEOF ( QUERY ( ppd
<* USEDIN ( pd ,
'CUTTING_TOOL_SCHEMA.'+'PROPERTY_DEFINITION.'+ 'DEFINITION' )
| ( ppd.name = 'occurrence quantity' ) AND ( SIZEOF (
QUERY ( pdr <* USEDIN ( ppd ,
'CUTTING_TOOL_SCHEMA.'+'PROPERTY_DEFINITION_REPRESENTATION.'+
'DEFINITION' ) | ( pdr.used_representation.name
='quantity' ) AND ( SIZEOF ( pdr.used_representation.items
) =1 ) AND ( SIZEOF ( QUERY ( i <* pdr.
used_representation.items | ( 'CUTTING_TOOL_SCHEMA.'+
'MEASURE_REPRESENTATION_ITEM' IN TYPEOF ( i ) ) AND (
i.name = 'quantity measure' ) ) ) =1 ) ) ) =1 ) ) ) =0 ) )
) =0;
END_RULE;

```

### **Argument definitions:**

**product\_definition:** the set of all instances of product\_definition.

**product\_definition\_occurrence\_relationship:** the set of all instances of product\_definition\_occurrence\_relationship.

### **Formal propositions:**

**WR1:** Each instance of product\_definition which references as its frame\_of\_reference a product\_definition\_context with a name of 'part occurrence', shall either have a name of 'single instance' or 'quantified instance'.

**WR2:** Each instance of product\_definition which references as its frame\_of\_reference a product\_definition\_context with a name of 'part occurrence' shall (at least) participate in a product\_definition\_usage relationship as the related\_product\_definition or in a product\_definition\_occurrence\_relationship as occurrence.

**WR3:** Each instance of product\_definition which references as its frame\_of\_reference a product\_definition\_context with a name of 'part occurrence' and which has a name of 'quantified instance', shall be referenced by a property\_definition which has a name of 'occurrence quantity', and which is referenced as definition by a property\_definition\_representation which has as used\_representation a representation which has a name of 'quantity' and which contains exactly one representation\_item in its set of items. This representation\_item is of type measure\_representation\_item and has a name of 'quantity measure'.

### 6.2.3.54 restrict\_product\_category\_for\_product

The restrict\_product\_category\_for\_product rule enforces the existence of a product\_related\_product\_category with a name of either 'cutting item', 'tool item', 'adaptive item', 'accessory item', 'assembly item', 'document', 'grade', 'coating', 'physically realized product', or 'substrate' for each product.

#### EXPRESS specification:

```

RULE restrict_product_category_for_product FOR
  (product);
WHERE
  WR1: SIZEOF ( QUERY ( p <* product | SIZEOF ( QUERY ( prpc <*
    USEDIN ( p ,
      'CUTTING_TOOL_SCHEMA.'+'PRODUCT_RELATED_PRODUCT_CATEGORY.'+
      'PRODUCTS' ) | prpc.name IN ['cutting item' , 'tool item' ,
      'adaptive item' , 'accessory item' , 'assembly item' ,
      'document' , 'grade' , 'coating' , 'physically realized product' ,
      'substrate' ] ) ) ) =0 ) )
    =0;
  WR2: SIZEOF ( QUERY ( p <* product | SIZEOF ( QUERY ( prpc <*
    USEDIN ( p ,
      'CUTTING_TOOL_SCHEMA.'+'PRODUCT_RELATED_PRODUCT_CATEGORY.'+
      'PRODUCTS' ) | prpc.name IN ['cutting item' , 'tool item' ,
      'adaptive item' , 'accessory item' , 'assembly item' ,
      'document' , 'grade' , 'coating' , 'substrate' ]
    ) ) >1 ) ) =0;
  WR3: SIZEOF ( QUERY ( p <* product | SIZEOF ( QUERY ( prpc <*
    USEDIN ( p ,
      'CUTTING_TOOL_SCHEMA.'+'PRODUCT_RELATED_PRODUCT_CATEGORY.'+
      'PRODUCTS' ) | prpc.name IN [ 'physically realized product' ]
    ) ) >1 ) ) =0;
END_RULE;

```

#### Argument definitions:

**product:** the set of all instances of product.

#### Formal propositions:

**WR1:** Each instance of product shall be referenced by a product\_related\_product\_category with a name of either 'cutting item', 'tool item', 'adaptive item', 'accessory item', 'assembly item', 'document', 'grade', 'coating', 'physically realized product', or 'substrate'.

**WR2:** Each instance of product shall be referenced by at most one product\_related\_product\_category with a name of either 'cutting item', 'tool item', 'adaptive item', 'accessory item', 'assembly item', 'document', 'grade', 'coating', or 'substrate'.

**WR3:** Each instance of product shall be referenced by at most one product\_related\_product\_category with a name of 'physically realized product'.

### 6.2.3.55 restrict\_product\_definition\_context\_for\_applied\_location\_assignment

The restrict\_product\_definition\_context\_for\_applied\_location\_assignment rule constrains that the name of the product\_definition\_context which is referenced as the frame\_of\_reference by a product\_definition that is referenced as the located\_product by an applied\_location\_assignment.

**EXPRESS specification:**

```

RULE restrict_product_definition_context_for_applied_location_assignment
  FOR
    (applied_location_assignment);
WHERE
  WR1: SIZEOF ( QUERY ( ala <* applied_location_assignment |
    ( ala.located_product.frame_of_reference.name<>
      'physical occurrence' ) ) ) =0;
END_RULE;

```

**Argument definitions:**

**applied\_location\_assignment:** the set of all instances of product\_definition.

**Formal propositions:**

**WR1:** Each instance of applied\_location\_assignment shall reference as located\_product a product\_definition that references a product\_definition\_context with a name of 'physical occurrence'.

**6.2.3.56 restrict\_product\_definition\_context\_for\_applied\_state\_type\_assignment**

The restrict\_product\_definition\_context\_for\_applied\_location\_assignment rule constrains that the name of the product\_definition\_context which is referenced as the frame\_of\_reference by a product\_definition that is referenced as the located\_product by an applied\_location\_assignment.

**EXPRESS specification:**

```

RULE restrict_product_definition_context_for_applied_state_type_assignment
  FOR
    (applied_state_type_assignment);
WHERE
  WR1: SIZEOF ( QUERY ( pd <* product_definition | ( pd.
    frame_of_reference.name <> 'physical occurrence' ) AND USEDIN ( pd ,
    'CUTTING_TOOL_SCHEMA.'+'APPLIED_STATE_TYPE_ASSIGNMENT.'+'ITEM_SET' )
    ) ) = 0;
END_RULE;

```

**Argument definitions:**

**product\_definition:** the set of all instances of product\_definition.

**Formal propositions:**

**WR1:** Each instance of applied\_state\_type\_assignment shall reference as its item\_set only product\_definitions that reference a product\_definition\_context with a name of 'physical occurrence'.

**6.2.3.57 restrict\_product\_definition\_context\_for\_product**

The restrict\_product\_definition\_context\_for\_product rule ensures the correct correlation between the name of the product\_definition\_context which is referenced as the frame\_of\_reference by a product\_definition and the product\_related\_product\_category of the associated product.

**EXPRESS specification:**

```

RULE restrict_product_definition_context_for_product FOR
  (product_definition,
  product_related_product_category);
WHERE
  WR1: SIZEOF ( QUERY ( pd <* product_definition | ( pd.
    frame_of_reference.name IN [ 'part definition' ,

```

```

'part occurrence' ] ) AND ( SIZEOF ( QUERY ( prpc <* USEDIN
( pd. formation.of_product ,
'CUTTING_TOOL_SCHEMA.PRODUCT_RELATED_PRODUCT_CATEGORY.PRODUCTS'
) | prpc. name IN ['cutting item' , 'tool item' ,
'adaptive item' , 'accessory item' , 'assembly item' ] ) ) =0
) ) ) =0;
WR2: SIZEOF ( QUERY ( pd <* product_definition | ( pd.
frame_of_reference.name = 'physical occurrence' ) AND (
SIZEOF ( QUERY ( prpc <* USEDIN ( pd. formation.of_product
, 'CUTTING_TOOL_SCHEMA.'+'PRODUCT_RELATED_PRODUCT_CATEGORY.'+
'PRODUCTS' ) | prpc. name = 'physically realized product' )
) =0 ) ) ) =0;
WR3: SIZEOF ( QUERY ( pd <* product_definition | ( pd.
frame_of_reference.name IN[ 'physical document definition' ,
'digital document definition' ] )
AND ( SIZEOF ( QUERY ( prpc <* USEDIN ( pd.
formation.of_product ,
'CUTTING_TOOL_SCHEMA.'+'PRODUCT_RELATED_PRODUCT_CATEGORY.'+
'PRODUCTS' ) | prpc. name = 'document' ) ) =0 ) ) ) =0;
WR4: SIZEOF ( QUERY ( pd <* product_definition | ( pd.
frame_of_reference.name = 'material definition' ) AND (
SIZEOF ( QUERY ( prpc <* USEDIN ( pd. formation.of_product
, 'CUTTING_TOOL_SCHEMA.'+'PRODUCT_RELATED_PRODUCT_CATEGORY.'+
'PRODUCTS' ) | prpc. name IN [ 'grade' , 'substrate' , 'coating' ] )
) =0 ) ) ) =0;
END_RULE;

```

**Argument definitions:**

**product\_definition:** the set of all instances of product\_definition.

**product\_related\_product\_category:** the set of all instances of product\_related\_product\_category.

**Formal propositions:**

**WR1:** If the product\_definition references as its frame\_of\_reference a product\_definition\_context with a name of 'part definition' or 'part occurrence', this product\_definition shall reference as its formation a product\_definition\_formation which references as of\_product a product which is in the set of products of a product\_related\_product\_category with a name of 'cutting item', 'tool item', 'adaptive item'.

**WR2:** If the product\_definition references as its frame\_of\_reference a product\_definition\_context with a name of 'physical occurrence', this product\_definition shall reference as its formation a product\_definition\_formation which references as of\_product a product which is in the set of products of a product\_related\_product\_category with a name of 'physically realized product'.

**WR3:** If the product\_definition references as its frame\_of\_reference a product\_definition\_context with a name of 'physical document definition' or 'digital document definition', this product\_definition shall reference as its formation a product\_definition\_formation which references as of\_product a product which is in the set of products of a product\_related\_product\_category with a name of 'document'.

**WR4:** If the product\_definition references as its frame\_of\_reference a product\_definition\_context with a name of 'material definition', this product\_definition shall reference as its formation a product\_definition\_formation which references as of\_product a product which is in the set of products of a product\_related\_product\_category with a name of 'grade', 'substrate', or 'coating'.

### 6.2.3.58 restrict\_product\_definitions\_for\_product\_definition\_relationship

The restrict\_product\_definitions\_for\_product\_definition\_relationship rule ensures the correct correlation between the name or type of the product\_definition\_relationship and the product\_definition\_context of the related product\_definitions.

#### EXPRESS specification:

```

RULE restrict_product_definitions_for_product_definition_relationship FOR
  (assembly_component_usage,
   product_definition_relationship);
WHERE
  WR1: SIZEOF ( QUERY ( pdr <* product_definition_relationship |
    ( 'CUTTING_TOOL_SCHEMA.'+ 'ASSEMBLY_COMPONENT_USAGE' IN
    TYPEOF ( pdr ) ) AND ( ( pdr.
    relating_product_definition.frame_of_reference.name<>
    'part definition' ) OR ( pdr.
    related_product_definition.frame_of_reference.name<>
    'part definition' ) ) AND ( ( pdr.
    relating_product_definition.frame_of_reference.name<>
    'physical occurrence' ) OR ( pdr.
    related_product_definition.frame_of_reference.name<>
    'physical occurrence' ) ) ) ) =0;
  WR2: SIZEOF ( QUERY ( pdr <* product_definition_relationship |
    ( pdr.name = 'physical realization' ) AND ( ( pdr.
    relating_product_definition.frame_of_reference.name<>
    'part definition' ) OR ( pdr.
    related_product_definition.frame_of_reference.name<>
    'physical occurrence' ) ) ) ) =0;
  WR3: SIZEOF ( QUERY ( pdr <* product_definition_relationship |
    ( pdr.name = 'physical occurrence usage' ) AND ( NOT (
    'CUTTING_TOOL_SCHEMA.'+ 'ASSEMBLY_COMPONENT_USAGE' IN TYPEOF
    ( pdr ) ) OR ( pdr.
    relating_product_definition.frame_of_reference.name<>
    'physical occurrence' ) OR ( pdr.
    related_product_definition.frame_of_reference.name<>
    'physical occurrence' ) ) ) ) =0;
  WR4: SIZEOF ( QUERY ( pdr <* product_definition_relationship |
    ( pdr.name = 'substrate' ) AND ( ( pdr.
    relating_product_definition.frame_of_reference.name<>
    'material definition' ) OR ( pdr.
    related_product_definition.frame_of_reference.name<>
    'material definition' ) ) ) ) =0;
  WR5: SIZEOF ( QUERY ( pdr <* product_definition_relationship |
    ( pdr.name = 'coating' ) AND ( ( pdr.
    relating_product_definition.frame_of_reference.name<>
    'material definition' ) OR ( pdr.
    related_product_definition.frame_of_reference.name<>
    'material definition' ) ) ) ) =0;
END_RULE;

```

#### Argument definitions:

**assembly\_component\_usage:** the set of all instances of assembly\_component\_usage.

**product\_definition\_relationship:** the set of all instances of product\_definition\_relationship.

**Formal propositions:**

**WR1:** If the `product_definition_relationship` is of type `assembly_component_usage`, the `relating_product_definition` and the `related_product_definition` shall either both reference as their `frame_of_reference` a `product_definition_context` with a name of 'part definition', or shall both reference as their `frame_of_reference` a `product_definition_context` with a name of 'physical occurrence'.

**WR2:** If the `product_definition_relationship` has a name of 'physical realization', the `relating_product_definition` shall reference as its `frame_of_reference` a `product_definition_context` with a name of 'part definition', and the `related_product_definition` shall reference as its `frame_of_reference` a `product_definition_context` with a name of 'physical occurrence'.

**WR3:** If the `product_definition_relationship` has a name of 'physical occurrence usage', it shall be of type `assembly_component_usage` and both the `relating_product_definition` and the `related_product_definition` shall reference as `frame_of_reference` a `product_definition_context` with a name of 'physical occurrence'.

**WR4:** If the `product_definition_relationship` has a name of 'substrate', the `relating_product_definition` shall reference as its `frame_of_reference` a `product_definition_context` with a name of 'material definition', and the `related_product_definition` shall reference as its `frame_of_reference` a `product_definition_context` with a name of 'material definition'.

**WR5:** If the `product_definition_relationship` has a name of 'coating', the `relating_product_definition` shall reference as its `frame_of_reference` a `product_definition_context` with a name of 'material definition', and the `related_product_definition` shall reference as its `frame_of_reference` a `product_definition_context` with a name of 'material definition'.

**6.2.3.59 restrict\_properties\_of\_document\_file**

The `restrict_properties_of_document_file` rule specifies the restrictions that apply for properties of `digital_files`.

**EXPRESS specification:**

```

RULE restrict_properties_of_document_file FOR
  (document_file);
WHERE
  WR1: SIZEOF ( QUERY ( df <*> document_file | SIZEOF ( QUERY ( pd
    <*> USEDIN ( df ,
      'CUTTING_TOOL_SCHEMA.'+'PROPERTY_DEFINITION.'+ 'DEFINITION' )
    | pd. name = 'document property' ) ) >1 ) ) =0;
  WR2: SIZEOF ( QUERY ( df <*> document_file | SIZEOF ( QUERY ( rt
    <*> df. representation_types | ( 'CUTTING_TOOL_SCHEMA.'+
      'DOCUMENT REPRESENTATION_TYPE' IN TYPEOF ( rt ) ) AND ( rt.
      name IN [ 'digital' , 'physical' ] ) ) ) =0 ) ) =0;
END_RULE;

```

**Argument definitions:**

**document\_file:** the set of all instances of `document_file`.

**Formal propositions:**

**WR1:** Each instance of `document_file` shall be referenced as definition by at most one `property_definition` with a name of 'document property'.

**WR2:** Each instance of `document_file` shall reference as `representation_types` an instance of `document_representation_type` with a name of either 'digital', or 'physical'.

### 6.2.3.60 restrict\_properties\_of\_document\_representation

The restrict\_properties\_of\_document\_representation rule specifies the restrictions that apply for properties of document\_representations.

#### EXPRESS specification:

```
RULE restrict_properties_of_document_representation FOR
  (product_definition);
WHERE
  WR1: SIZEOF ( QUERY ( pd <* product_definition | ( pd.
    frame_of_reference.name IN [ 'physical document definition'
    , 'digital document definition' ] ) AND (
    SIZEOF ( QUERY ( pd1 <* USEDIN ( pd ,
    'CUTTING_TOOL_SCHEMA.'+'PROPERTY_DEFINITION.'+ 'DEFINITION' )
    | pd1. name = 'document property' ) ) >1 ) ) ) =0;
END_RULE;
```

#### Argument definitions:

**product\_definition:** the set of all instances of product\_definition.

#### Formal propositions:

**WR1:** Each instance of product\_definition that references as frame\_of\_reference a application\_context\_element with a name of either 'physical document definition' or 'digital document definition', shall be referenced as definition by at most one property\_definition with a name of 'document property'.

### 6.2.3.61 restrict\_representation\_for\_coupling

The restrict\_representation\_for\_document\_content\_property rule specifies the restrictions that apply for representations with a name of 'coupling'.

#### EXPRESS specification:

```
RULE restrict_representation_for_coupling FOR
  (representation,
  representation_item);
WHERE
  WR1: SIZEOF ( QUERY ( r <* representation | ( r.name =
    'coupling' ) AND ( ( SIZEOF ( r.items ) <4 ) OR (
    SIZEOF ( r.items ) >5 ) ) ) ) =0;
  WR2: SIZEOF ( QUERY ( ri <* representation_item | ( SIZEOF (
    QUERY ( r <* USEDIN ( ri ,
    'CUTTING_TOOL_SCHEMA.REPRESENTATION.ITEMS' ) | r.name =
    'coupling' ) ) >0 ) AND NOT ( ri. name IN [
    'coupling type' , 'pieces' , 'side' , 'size' , 'style' ] ) )
    ) =0;
  WR3: SIZEOF ( QUERY ( r <* representation | ( r.name =
    'coupling' ) AND ( SIZEOF ( QUERY ( i <* r.items |
    ( i.name = 'coupling type' ) AND ( 'CUTTING_TOOL_SCHEMA.'+
    'DESCRIPTIVE_REPRESENTATION_ITEM' IN TYPEOF ( i ) ) ) ) >1
    ) ) ) =0;
  WR4: SIZEOF ( QUERY ( r <* representation | ( r.name =
    'coupling' ) AND ( SIZEOF ( QUERY ( i <* r.items |
    ( i.name = 'pieces' ) AND ( SIZEOF (
    ['CUTTING_TOOL_SCHEMA.'+ 'MEASURE_REPRESENTATION_ITEM' ,
```

```

        'CUTTING_TOOL_SCHEMA.'+ 'VALUE_REPRESENTATION_ITEM']* TYPEOF
        ( i ) ) =1 ) ) ) >1 ) ) ) =0;
WR5: SIZEOF ( QUERY ( r <* representation | ( r.name =
        'coupling' ) AND ( SIZEOF ( QUERY ( i <* r.items |
        ( i.name = 'side' ) AND ( 'CUTTING_TOOL_SCHEMA.'+
        'DESCRIPTIVE_REPRESENTATION_ITEM' IN TYPEOF ( i ) ) ) ) >1
        ) ) ) =0;
WR6: SIZEOF ( QUERY ( r <* representation | ( r.name =
        'coupling' ) AND ( SIZEOF ( QUERY ( i <* r.items |
        ( i.name = 'size' ) AND ( SIZEOF (
        ['CUTTING_TOOL_SCHEMA.'+ 'MEASURE_REPRESENTATION_ITEM' ,
        'CUTTING_TOOL_SCHEMA.'+ 'VALUE_REPRESENTATION_ITEM']* TYPEOF
        ( i ) ) =1 ) ) ) >1 ) ) ) =0;
WR7: SIZEOF ( QUERY ( r <* representation | ( r.name =
        'coupling' ) AND ( SIZEOF ( QUERY ( i <* r.items |
        ( i.name = 'style' ) AND ( 'CUTTING_TOOL_SCHEMA.'+
        'DESCRIPTIVE_REPRESENTATION_ITEM' IN TYPEOF ( i ) ) ) ) >1
        ) ) ) =0;
END_RULE;

```

**Argument definitions:**

**representation:** the set of all instances of representation.

**representation\_item:** the set of all instances of representation\_item.

**Formal propositions:**

**WR1:** For each instance of representation which has a name of 'coupling' there shall be at least four and at most five instances of representation\_items in the set of items.

**WR2:** Each instance of representation\_item that is referenced as items by an instance of representation with a name of 'coupling' shall have a name of either 'coupling type', 'pieces', 'side', 'size', or 'style'.

**WR3:** There shall be at most one instance of representation\_item that have a name of 'coupling type' and which is of type descriptive\_representation\_item that is referenced as items by an instance of representation with a name of 'coupling'.

**WR4:** There shall be at most one instance of representation\_item that have a name of 'pieces' and which is either of type measure\_representation\_item, or value\_representation\_item that is referenced as items by an instance of representation with a name of 'coupling'.

**WR5:** There shall be at most one instance of representation\_item that have a name of 'side' and which is of type descriptive\_representation\_item that is referenced as items by an instance of representation with a name of 'coupling'.

**WR6:** There shall be at most one instance of representation\_item that have a name of 'size' and which is either of type measure\_representation\_item, or value\_representation\_item that is referenced as items by an instance of representation with a name of 'coupling'.

**WR7:** There shall be at most one instance of representation\_item that have a name of 'style' and which is of type descriptive\_representation\_item that is referenced as items by an instance of representation with a name of 'coupling'.

**6.2.3.62 restrict\_representation\_for\_cutting\_condition**

The restrict\_representation\_for\_document\_content\_property rule specifies the restrictions that apply for representations with a name of 'cutting condition'.

**EXPRESS specification:**

```

RULE restrict_representation_for_cutting_condition FOR
  (representation,
   representation_item);
WHERE
  WR1: SIZEOF ( QUERY ( r <* representation | ( r.name =
    'cutting condition' ) AND ( SIZEOF ( r.items ) <1 ) ) ) =0;
  WR2: SIZEOF ( QUERY ( ri <* representation_item | ( SIZEOF (
    QUERY ( r <* USEDIN ( ri ,
    'CUTTING_TOOL_SCHEMA.REPRESENTATION.ITEMS' ) | r.name =
    'cutting condition' ) ) >0 ) AND NOT ( ri. name IN [
    'condition name' ] ) )
    ) =0;
  WR3: SIZEOF ( QUERY ( r <* representation | ( r.name =
    'cutting condition' ) AND ( SIZEOF ( QUERY ( i <* r.items |
    ( i.name = 'condition name' ) AND ( 'CUTTING_TOOL_SCHEMA.'+
    'DESCRIPTIVE_REPRESENTATION_ITEM' IN TYPEOF ( i ) ) ) ) >1
    ) ) ) =0;
END_RULE;

```

**Argument definitions:**

**representation:** the set of all instances of representation.

**representation\_item:** the set of all instances of representation\_item.

**Formal propositions:**

**WR1:** For each instance of representation which has a name of 'cutting condition' there shall be at least one instance of representation\_items in the set of items.

**WR2:** Each instance of representation\_item that is referenced as items by an instance of representation with a name of 'cutting condition' shall have a name of 'condition name'.

**WR3:** There shall be at most one instance of representation\_item that have a name of 'condition name' and which is of type descriptive\_representation\_item that is referenced as items by an instance of representation with a name of 'cutting condition'.

**6.2.3.63 restrict\_representation\_for\_cutting\_data\_association**

The restrict\_representation\_for\_document\_content\_property rule specifies the restrictions that apply for representations with a name of 'cutting data association'.

**EXPRESS specification:**

```

RULE restrict_representation_for_cutting_data_association FOR
  (representation,
   representation_item);
WHERE
  WR1: SIZEOF ( QUERY ( r <* representation | ( r.name =
    'cutting data association' ) AND ( SIZEOF ( r.items ) <3 ) ) ) =0;
  WR2: SIZEOF ( QUERY ( ri <* representation_item | ( SIZEOF (
    QUERY ( r <* USEDIN ( ri ,
    'CUTTING_TOOL_SCHEMA.REPRESENTATION.ITEMS' ) | r.name =
    'cutting data association' ) ) >0 ) AND NOT ( ri. name IN [
    'depth of cut' , 'feed' , 'speed' ] ) )
    ) =0;
  WR3: SIZEOF ( QUERY ( r <* representation | ( r.name =

```

```

'cutting data association' ) AND ( SIZEOF ( QUERY ( i <* r.items |
( i.name = 'depth of cut' ) AND ( SIZEOF (
['CUTTING_TOOL_SCHEMA.'+ 'MEASURE_REPRESENTATION_ITEM' ,
'CUTTING_TOOL_SCHEMA.'+ 'VALUE_RANGE' ,
'CUTTING_TOOL_SCHEMA.'+ 'VALUE_REPRESENTATION_ITEM']* TYPEOF
( i ) ) =1 ) ) ) >1 ) ) ) =0;
WR4: SIZEOF ( QUERY ( r <* representation | ( r.name =
'cutting data association' ) AND ( SIZEOF ( QUERY ( i <* r.items |
( i.name = 'feed' ) AND ( SIZEOF (
['CUTTING_TOOL_SCHEMA.'+ 'MEASURE_REPRESENTATION_ITEM' ,
'CUTTING_TOOL_SCHEMA.'+ 'VALUE_RANGE' ,
'CUTTING_TOOL_SCHEMA.'+ 'VALUE_REPRESENTATION_ITEM']* TYPEOF
( i ) ) =1 ) ) ) >1 ) ) ) =0;
WR5: SIZEOF ( QUERY ( r <* representation | ( r.name =
'cutting data association' ) AND ( SIZEOF ( QUERY ( i <* r.items |
( i.name = 'speed' ) AND ( SIZEOF (
['CUTTING_TOOL_SCHEMA.'+ 'MEASURE_REPRESENTATION_ITEM' ,
'CUTTING_TOOL_SCHEMA.'+ 'VALUE_RANGE' ,
'CUTTING_TOOL_SCHEMA.'+ 'VALUE_REPRESENTATION_ITEM']* TYPEOF
( i ) ) =1 ) ) ) >1 ) ) ) =0;
END_RULE;

```

**Argument definitions:**

**representation:** the set of all instances of representation.

**representation\_item:** the set of all instances of representation\_item.

**Formal propositions:**

**WR1:** For each instance of representation which has a name of 'cutting data association' there shall be at least three instances of representation\_items in the set of items.

**WR2:** Each instance of representation\_item that is referenced as items by an instance of representation with a name of 'cutting data association' shall have a name of either 'depth of cut', 'feed', or 'speed'.

**WR4:** There shall be at most one instance of representation\_item that have a name of 'depth of cut' and which is either of type measure\_representation\_item, value\_range, or value\_representation\_item that is referenced as items by an instance of representation with a name of 'cutting data association'.

**WR5:** There shall be at most one instance of representation\_item that have a name of 'feed' and which is either of type measure\_representation\_item, value\_range, or value\_representation\_item that is referenced as items by an instance of representation with a name of 'cutting data association'.

**WR6:** There shall be at most one instance of representation\_item that have a name of 'speed' and which is either of type measure\_representation\_item, value\_range, or value\_representation\_item that is referenced as items by an instance of representation with a name of 'cutting data association'.

**6.2.3.64 restrict\_representation\_for\_document\_content\_property**

The restrict\_representation\_for\_document\_content\_property rule specifies the restrictions that apply for representations with a name of 'document content'.

**EXPRESS specification:**

```

RULE restrict_representation_for_document_content_property FOR
(representation,
representation_item);

```

```

WHERE
WR1: SIZEOF ( QUERY ( r <* representation | ( r.name =
  'document content' ) AND ( ( SIZEOF ( r.items ) <1 ) OR (
  SIZEOF ( r.items ) >3 ) ) ) ) =0;
WR2: SIZEOF ( QUERY ( ri <* representation_item | ( SIZEOF (
  QUERY ( r <* USEDIN ( ri ,
  'CUTTING_TOOL_SCHEMA.REPRESENTATION.ITEMS' ) | r.name =
  'document content' ) ) >0 ) AND NOT ( ri. name IN [
  'detail level' , 'geometry type' , 'real world scale' ] ) )
  ) =0;
WR3: SIZEOF ( QUERY ( r <* representation | ( r.name =
  'document content' ) AND ( SIZEOF ( QUERY ( i <* r.items |
  ( i.name = 'detail level' ) AND ( 'CUTTING_TOOL_SCHEMA.'+
  'DESCRIPTIVE_REPRESENTATION_ITEM' IN TYPEOF ( i ) ) ) ) >1
  ) ) ) =0;
WR4: SIZEOF ( QUERY ( r <* representation | ( r.name =
  'document content' ) AND ( SIZEOF ( QUERY ( i <* r.items |
  ( i.name = 'geometry type' ) AND ( 'CUTTING_TOOL_SCHEMA.'+
  'DESCRIPTIVE_REPRESENTATION_ITEM' IN TYPEOF ( i ) ) ) ) >1
  ) ) ) =0;
WR5: SIZEOF ( QUERY ( r <* representation | ( r.name =
  'document content' ) AND ( SIZEOF ( QUERY ( i <* r.items |
  ( i.name = 'real world scale' ) AND ( SIZEOF (
  ['CUTTING_TOOL_SCHEMA.'+ 'MEASURE_REPRESENTATION_ITEM' ,
  'CUTTING_TOOL_SCHEMA.'+ 'VALUE_REPRESENTATION_ITEM']* TYPEOF
  ( i ) ) =1 ) ) ) >1 ) ) ) =0;
END_RULE;

```

### **Argument definitions:**

**representation:** the set of all instances of representation.

**representation\_item:** the set of all instances of representation\_item.

### **Formal propositions:**

**WR1:** For each instance of representation which has a name of 'document content' there shall be at least one and at most three instances of representation\_items in the set of items.

**WR2:** Each instance of representation\_item that is referenced as items by an instance of representation with a name of 'document content' shall have a name of either 'detail level', 'geometry type', or 'real world scale'.

**WR3:** There shall be at most one instance of representation\_item that have a name of 'detail level' and which is of type descriptive\_representation\_item that is referenced as items by an instance of representation with a name of 'document content'.

**WR4:** There shall be at most one instance of representation\_item that have a name of 'geometry type' and which is of type descriptive\_representation\_item that is referenced as items by an instance of representation with a name of 'document content'.

**WR5:** There shall be at most one instance of representation\_item that have a name of 'real world scale' and which is either of type measure\_representation\_item, or value\_representation\_item that is referenced as items by an instance of representation with a name of 'document content'.

### **6.2.3.65 restrict\_representation\_for\_document\_creation\_property**

The restrict\_representation\_for\_document\_creation\_property rule specifies the restrictions that apply for representations with a name of 'document creation'.

**EXPRESS specification:**

```

RULE restrict_representation_for_document_creation_property FOR
    (representation,
     representation_item);
WHERE
    WR1: SIZEOF ( QUERY ( r <* representation | ( r.name =
        'document creation' ) AND ( ( SIZEOF ( r.items ) <2 ) OR (
        SIZEOF ( r.items ) >3 ) ) ) ) =0;
    WR2: SIZEOF ( QUERY ( ri <* representation_item | ( SIZEOF (
        QUERY ( r <* USEDIN ( ri ,
        'CUTTING_TOOL_SCHEMA.REPRESENTATION.ITEMS' ) | r.name =
        'document creation' ) ) >0 ) AND NOT ( ri. name IN [
        'creating interface' , 'creating system' ,
        'operating system' ] ) ) ) =0;
    WR3: SIZEOF ( QUERY ( r <* representation | ( r.name =
        'document creation' ) AND ( SIZEOF ( QUERY ( i <* r.items |
        ( i.name = 'creating interface' ) AND (
        'CUTTING_TOOL_SCHEMA.'+ 'DESCRIPTIVE_REPRESENTATION_ITEM' IN
        TYPEOF ( i ) ) ) ) >1 ) ) ) =0;
    WR4: SIZEOF ( QUERY ( r <* representation | ( r.name =
        'document creation' ) AND ( SIZEOF ( QUERY ( i <* r.items |
        ( i.name = 'creating system' ) AND ( 'CUTTING_TOOL_SCHEMA.'+
        'DESCRIPTIVE_REPRESENTATION_ITEM' IN TYPEOF ( i ) ) ) ) <>1
        ) ) ) =0;
    WR5: SIZEOF ( QUERY ( r <* representation | ( r.name =
        'document creation' ) AND ( SIZEOF ( QUERY ( i <* r.items |
        ( i.name = 'operating system' ) AND ( 'CUTTING_TOOL_SCHEMA.'+
        'DESCRIPTIVE_REPRESENTATION_ITEM' IN TYPEOF ( i ) ) ) ) >1
        ) ) ) =0;
END_RULE;

```

**Argument definitions:**

**representation:** the set of all instances of representation.

**representation\_item:** the set of all instances of representation\_item.

**Formal propositions:**

**WR1:** For each instance of representation which has a name of 'document creation' there shall be at least two one and at most three instances of representation\_items in the set of items.

**WR2:** Each instance of representation\_item that is referenced as items by an instance of representation with a name of 'document creation' shall have a name of either 'creating interface', 'creating system', or 'operating system'.

**WR3:** There shall be at most one instance of representation\_item that have a name of 'creating interface' and which is of type descriptive\_representation\_item that is referenced as items by an instance of representation with a name of 'document creation'.

**WR4:** There shall be exactly one instance of representation\_item that have a name of 'creating system' and which is of type descriptive\_representation\_item that is referenced as items by an instance of representation with a name of 'document creation'.

**WR5:** There shall be at most one instance of representation\_item that have a name of 'operating system' and which is of type descriptive\_representation\_item that is referenced as items by an instance of representation with a name of 'document creation'.

### 6.2.3.66 restrict\_representation\_for\_document\_format\_property

The restrict\_representation\_for\_document\_format\_property rule specifies the restrictions that apply for representations with a name of 'document format'.

#### EXPRESS specification:

```

RULE restrict_representation_for_document_format_property FOR
  (representation,
   representation_item);
WHERE
WR1: SIZEOF ( QUERY ( r <* representation | ( r.name =
  'document format' ) AND ( ( SIZEOF ( r.items ) <1 ) OR (
  SIZEOF ( r.items ) >3 ) ) ) ) =0;
WR2: SIZEOF ( QUERY ( ri <* representation_item | ( SIZEOF (
  QUERY ( r <* USEDIN ( ri ,
  'CUTTING_TOOL_SCHEMA.REPRESENTATION.ITEMS' ) | r.name =
  'document format' ) ) >0 ) AND NOT ( ri.name IN [
  'character code' , 'data format' ] ) ) ) ) =0;
WR3: SIZEOF ( QUERY ( r <* representation | ( r.name =
  'document format' ) AND ( SIZEOF ( QUERY ( i <* r.items | (
  i.name = 'character code' ) AND ( 'CUTTING_TOOL_SCHEMA.'+
  'DESCRIPTIVE_REPRESENTATION_ITEM' IN TYPEOF ( i ) ) ) ) ) >1
  ) ) ) =0;
WR4: SIZEOF ( QUERY ( r <* representation | ( r.name =
  'document format' ) AND ( SIZEOF ( QUERY ( i <* r.items | (
  i.name = 'data format' ) AND ( 'CUTTING_TOOL_SCHEMA.'+
  'DESCRIPTIVE_REPRESENTATION_ITEM' IN TYPEOF ( i ) ) ) ) ) >1
  ) ) ) =0;
WR5: SIZEOF( QUERY( r <* representation |
  (r.name = 'document format') AND
  (SIZEOF( QUERY( i <* r.items |
  (i.name = 'size format') AND
  ('CUTTING_TOOL_SCHEMA.PLANAR_EXTENT' IN
  TYPEOF(i))
  )) > 1)
  )) = 0;
END_RULE;

```

#### Argument definitions:

**representation:** the set of all instances of representation.

**representation\_item:** the set of all instances of representation\_item.

#### Formal propositions:

**WR1:** For each instance of representation which has a name of 'document format' there shall be at least one and at most three instances of representation\_items in the set of items.

**WR2:** Each instance of representation\_item that is referenced as items by an instance of representation with a name of 'document format' shall have a name of either 'character code', or 'data format'.

**WR3:** There shall be at most one instance of representation\_item that have a name of 'character code' and which is of type descriptive\_representation\_item that is referenced as items by an instance of representation with a name of 'document format'.

**WR4:** There shall be at most one instance of `representation_item` that have a name of 'data format' and which is of type `descriptive_representation_item` that is referenced as items by an instance of `representation` with a name of 'document format'.

**WR5:** There shall be at most one instance of `representation_item` that have a name of 'size format' and which is of type `planar_extent` that is referenced as items by an instance of `representation` with a name of 'document format'.

### 6.2.3.67 restrict\_representation\_for\_document\_properties

The `restrict_representation_for_document_properties` rule specifies that each instance of `property_definition` with a name of 'document property' shall be represented by representations with a name of either 'document content', 'document creation', 'document format', or 'document size'. At the same time, this rule enforces that each instance of `representation` with a name of either 'document content', 'document creation', 'document format', or 'document size', shall be referenced by one instance of `property_definition` with a name of 'document property'.

#### EXPRESS specification:

```

RULE restrict_representation_for_document_properties FOR
  (property_definition,
   representation,
   representation_context);
WHERE
  WR1: SIZEOF ( QUERY ( pd <* property_definition | ( pd.name =
    'document property' ) AND ( SIZEOF ( QUERY ( pdr <* USEDIN
      ( pd ,
        'CUTTING_TOOL_SCHEMA.'+'PROPERTY_DEFINITION_REPRESENTATION.'+'DEFINITION
        ) | NOT ( pdr.used_representation.name IN [
          'document content' , 'document creation' ,
          'document format' , 'document size' ] ) ) ) >0 ) ) ) =0;
  WR2: SIZEOF ( QUERY ( r <* representation | ( r.name IN [
    'document content' , 'document creation' ,
    'document format' , 'document size' ] ) AND ( SIZEOF (
      QUERY ( pdr <* USEDIN ( r ,
        'CUTTING_TOOL_SCHEMA.'+'PROPERTY_DEFINITION_REPRESENTATION.'
        +'USED_REPRESENTATION'
      ) | ( 'CUTTING_TOOL_SCHEMA.'+'PROPERTY_DEFINITION' IN TYPEOF
        ( pdr.definition ) ) AND ( pdr.definition.name =
        'document property' ) ) ) =0 ) ) ) =0;
  WR3: SIZEOF ( QUERY ( r <* representation | ( r.name IN [
    'document content' , 'document creation' ,
    'document format' , 'document size' ] ) AND (
      r.context_of_items.context_type <> 'document parameters' )
    ) ) =0;
  WR4: SIZEOF ( QUERY ( rc <* representation_context | ( rc.
    context_type = 'document parameters' ) AND ( SIZEOF ( QUERY
      ( r <* USEDIN ( rc ,
        'CUTTING_TOOL_SCHEMA.REPRESENTATION.CONTEXT_OF_ITEMS' ) | NOT
      ( r.name IN [ 'document content' , 'document creation' ,
        'document format' , 'document size' ] ) ) ) >0 ) ) ) =0;
END_RULE;

```

#### Argument definitions:

**property\_definition:** the set of all instances of `property_definition`.

**representation:** the set of all instances of representation.

**representation\_context:** the set of all instances of representation\_context.

**Formal propositions:**

**WR1:** Each instance of property\_definition with a name of 'document property' shall be referenced as definition by a property\_definition\_representation, which has as used\_representation a representation with a name of either 'document content', 'document creation', 'document format', or 'document size'.

**WR2:** Each instance of representation with a name of either 'document content', 'document creation', 'document format', or 'document size' shall be referenced as used\_representation by a property\_definition\_representation, which has as definition a property\_definition with a name of 'document property'.

**WR3:** Each instance of representation with a name of either 'document content', 'document creation', 'document format', or 'document size' shall reference as context\_of\_items a representation\_context, which a context\_type of 'document parameters'.

**WR4:** Each instance of representation\_context with a context\_type of 'document parameters' shall be referenced as context\_of\_items by a representation with a name of either 'document content', 'document creation', 'document format', or 'document size'.

**6.2.3.68 restrict\_representation\_for\_document\_size\_property**

The restrict\_representation\_for\_document\_size\_property rule specifies the restrictions that apply for representations with a name of 'document size'.

**EXPRESS specification:**

```

RULE restrict_representation_for_document_size_property FOR
  (representation,
   representation_item);
WHERE
  WR1: SIZEOF ( QUERY ( r <* representation | ( r.name =
    'document size' ) AND ( ( SIZEOF ( r.items ) <1 ) OR (
    SIZEOF ( r.items ) >2 ) ) ) ) =0;
  WR2: SIZEOF ( QUERY ( ri <* representation_item | ( SIZEOF (
    QUERY ( r <* USEDIN ( ri ,
    'CUTTING_TOOL_SCHEMA.REPRESENTATION.ITEMS' ) | r.name =
    'document size' ) ) >0 ) AND NOT ( ri.name IN [
    'file size' , 'page count' ] ) ) ) =0;
  WR3: SIZEOF ( QUERY ( r <* representation | ( r.name =
    'document size' ) AND ( SIZEOF ( QUERY ( i <* r.items | (
    i.name = 'file size' ) AND ( SIZEOF (
    ['CUTTING_TOOL_SCHEMA.'+ 'MEASURE REPRESENTATION_ITEM' ,
    'CUTTING_TOOL_SCHEMA.'+ 'VALUE_RANGE' , 'CUTTING_TOOL_SCHEMA.'+
    'VALUE REPRESENTATION_ITEM']* TYPEOF ( i ) ) =1 ) ) ) >1 )
    ) ) =0;
  WR4: SIZEOF ( QUERY ( r <* representation | ( r.name =
    'document size' ) AND ( SIZEOF ( QUERY ( i <* r.items | (
    i.name = 'page count' ) AND ( SIZEOF (
    ['CUTTING_TOOL_SCHEMA.'+ 'MEASURE REPRESENTATION_ITEM' ,
    'CUTTING_TOOL_SCHEMA.'+ 'VALUE_RANGE' , 'CUTTING_TOOL_SCHEMA.'+
    'VALUE REPRESENTATION_ITEM']* TYPEOF ( i ) ) =1 ) ) ) >1 )
    ) ) =0;
END_RULE;

```

**Argument definitions:**

**representation:** the set of all instances of representation.

**representation\_item:** the set of all instances of representation\_item.

**Formal propositions:**

**WR1:** For each instance of representation which has a name of 'document size' there shall be at least one and at most two instances of representation\_items in the set of items.

**WR2:** Each instance of representation\_item that is referenced as items by an instance of representation with a name of 'document size' shall have a name of either 'file size', or 'page count'.

**WR3:** There shall be at most one instance of representation\_item that have a name of 'file size' and which is either of type measure\_representation\_item, value\_range, or value\_representation\_item that is referenced as items by an instance of representation with a name of 'document size'.

**WR4:** There shall be at most one instance of representation\_item that have a name of 'page count' and which is either of type measure\_representation\_item, value\_range, or value\_representation\_item that is referenced as items by an instance of representation with a name of 'document size'.

**6.2.3.69 restrict\_version\_assignment\_for\_applied\_identification\_assignment**

The restrict\_version\_assignment\_for\_applied\_identification\_assignment rule specifies that each instance of applied\_identification\_assignment shall be referenced by at most one instance of applied\_identification\_assignment that specifies the version of the applied\_identification\_assignment.

**EXPRESS specification:**

```

RULE restrict_version_assignment_for_applied_identification_assignment FOR
    (applied_identification_assignment);
WHERE
    WR1: SIZEOF ( QUERY ( ent <* applied_identification_assignment
        | SIZEOF ( QUERY ( ia <* USEDIN ( ent ,
            'CUTTING_TOOL_SCHEMA.'+'APPLIED_IDENTIFICATION_ASSIGNMENT.'+
            'ITEMS' ) | ia.role.name ='version' ) ) >1 ) ) =0;
END_RULE;

```

**Argument definitions:**

**applied\_identification\_assignment:** the set of all instances of applied\_identification\_assignment.

**Formal propositions:**

**WR1:** For each instance of applied\_identification\_assignment there shall be only at most one instance of applied\_identification\_assignment with a role name of 'version'.

**6.2.3.70 restrict\_version\_assignment\_for\_class**

The restrict\_version\_assignment\_for\_class rule specifies that each instance of class shall be referenced by at most one instance of applied\_identification\_assignment that specifies the version of the class.

**EXPRESS specification:**

```

RULE restrict_version_assignment_for_class FOR
    (class);
WHERE

```

```

WR1: SIZEOF ( QUERY ( ent <* class | SIZEOF ( QUERY ( ia <*
  USEDIN ( ent ,
    'CUTTING_TOOL_SCHEMA.'+'APPLIED_IDENTIFICATION_ASSIGNMENT.'+
    'ITEMS' ) | ia. role.name ='version' ) ) >1 ) ) =0;
END_RULE;

```

**Argument definitions:**

**class:** the set of all instances of class.

**Formal propositions:**

**WR1:** For each instance of class there shall be only at most one instance of applied\_identification\_assignment with a role name of 'version'.

**6.2.3.71 restrict\_version\_assignment\_for\_document\_file**

The restrict\_version\_assignment\_for\_document\_file rule specifies that each instance of document\_file shall be referenced by at most one instance of applied\_identification\_assignment that specifies the version of the document\_file.

**EXPRESS specification:**

```

RULE restrict_version_assignment_for_document_file FOR
  (document_file);
WHERE
  WR1: SIZEOF ( QUERY ( ent <* document_file | SIZEOF ( QUERY (
    ia <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'APPLIED_IDENTIFICATION_ASSIGNMENT.'+
      'ITEMS' ) | ia. role.name ='version' ) ) >1 ) ) =0;
END_RULE;

```

**Argument definitions:**

**document\_file:** the set of all instances of document\_file.

**Formal propositions:**

**WR1:** For each instance of document\_file there shall be at most one instance of applied\_identification\_assignment with a role name of 'version'.

**6.2.3.72 restrict\_version\_assignment\_for\_effectivity**

The restrict\_version\_assignment\_for\_effectivity rule specifies that each instance of effectivity shall be referenced by at most one instance of applied\_identification\_assignment that specifies the version of the effectivity.

**EXPRESS specification:**

```

RULE restrict_version_assignment_for_effectivity FOR
  (effectivity);
WHERE
  WR1: SIZEOF ( QUERY ( ent <* effectivity | SIZEOF ( QUERY ( ia
    <* USEDIN ( ent ,
      'CUTTING_TOOL_SCHEMA.'+'APPLIED_IDENTIFICATION_ASSIGNMENT.'+
      'ITEMS' ) | ia. role.name ='version' ) ) >1 ) ) =0;
END_RULE;

```

**Argument definitions:**

**effectivity:** the set of all instances of effectivity.

**Formal propositions:**

**WR1:** For each instance of effectivity there shall be only at most one instance of applied\_identification\_assignment with a role name of 'version'.

**6.2.3.73 substrate\_requires\_product\_definition**

The substrate\_requires\_product\_definition rule ensures that each instance of product\_definition\_formation which has an associated of\_product that is referenced by a product\_related\_product\_category with a name of 'substrate', is referenced as formation by exactly one instance of product\_definition that references as its frame\_of\_reference a product\_definition\_context with a name of 'material definition'.

**EXPRESS specification:**

```

RULE grade_requires_product_definition FOR
  (product_definition,
   product_definition_formation,
   product_related_product_category);
WHERE
  WR1: SIZEOF ( QUERY ( pdf <* product_definition_formation | (
    SIZEOF ( QUERY ( prpc <* USEDIN ( pdf. of product ,
      'CUTTING_TOOL_SCHEMA.'+'PRODUCT_RELATED_PRODUCT_CATEGORY.'+
      'PRODUCTS' ) | prpc. name = 'substrate')
    ) >0 ) AND ( SIZEOF ( QUERY ( pd <* USEDIN ( pdf ,
      'CUTTING_TOOL_SCHEMA.'+'PRODUCT_DEFINITION.'+ 'FORMATION' ) |
      pd. frame_of_reference.name = 'material definition' ) ) <>1
    ) ) ) =0;
END_RULE;

```

**Argument definitions:**

**product\_definition:** the set of all instances of product\_definition.

**product\_definition\_formation:** the set of all instances of product\_definition\_formation.

**product\_related\_product\_category:** the set of all instances of product\_related\_product\_category.

**Formal propositions:**

**WR1:** Each instance of product\_definition\_formation which has an associated of\_product that is referenced by a product\_related\_product\_category with a name of 'substrate', is referenced as formation by exactly one instance of product\_definition that references as its frame\_of\_reference a product\_definition\_context with a name of 'material definition'.

## Annex A (normative)

### AIM longform

```

SCHEMA CUTTING_TOOL_SCHEMA;

CONSTANT
  DUMMY_GRI : GEOMETRIC_REPRESENTATION_ITEM := representation_item('') ||
  geometric_representation_item();
END_CONSTANT;

TYPE action_item = SELECT
  (applied_classification_assignment,
   applied_person_and_organization_assignment,
   class,
   class_system,
   document_file);
END_TYPE;

TYPE language_item = SELECT
  (representation);
END_TYPE;

TYPE label = STRING;
END_TYPE;

TYPE length_measure = REAL;
END_TYPE;

TYPE text = STRING;
END_TYPE;

TYPE dimension_count = INTEGER;
  WHERE
    WR1 :
      SELF > 0;
END_TYPE;

TYPE compound_item_definition = SELECT
  (list_representation_item,
   set_representation_item);
END_TYPE;

TYPE list_representation_item = LIST [1:?] OF representation_item;
END_TYPE;

TYPE set_representation_item = SET [1:?] OF representation_item;
END_TYPE;

TYPE measure_value = SELECT
  (length_measure,
   mass_measure,
   time_measure,
   electric_current_measure,

```

```

thermodynamic_temperature_measure,
celsius_temperature_measure,
amount_of_substance_measure,
luminous_intensity_measure,
plane_angle_measure,
solid_angle_measure,
area_measure,
volume_measure,
ratio_measure,
parameter_value,
numeric_measure,
context_dependent_measure,
descriptive_measure,
positive_length_measure,
positive_plane_angle_measure,
positive_ratio_measure,
count_measure);
END_TYPE;

TYPE mass_measure = REAL;
END_TYPE;

TYPE time_measure = REAL;
END_TYPE;

TYPE electric_current_measure = REAL;
END_TYPE;

TYPE thermodynamic_temperature_measure = REAL;
END_TYPE;

TYPE celsius_temperature_measure = REAL;
END_TYPE;

TYPE amount_of_substance_measure = REAL;
END_TYPE;

TYPE luminous_intensity_measure = REAL;
END_TYPE;

TYPE plane_angle_measure = REAL;
END_TYPE;

TYPE solid_angle_measure = REAL;
END_TYPE;

TYPE area_measure = REAL;
END_TYPE;

TYPE volume_measure = REAL;
END_TYPE;

TYPE ratio_measure = REAL;
END_TYPE;

TYPE parameter_value = REAL;
END_TYPE;

```

```
TYPE numeric_measure = NUMBER;
END_TYPE;

TYPE context_dependent_measure = REAL;
END_TYPE;

TYPE descriptive_measure = STRING;
END_TYPE;

TYPE positive_length_measure = length_measure;
WHERE
  WR1 :
    SELF > 0.00000;
END_TYPE;

TYPE positive_plane_angle_measure = plane_angle_measure;
WHERE
  WR1 :
    SELF > 0.00000;
END_TYPE;

TYPE positive_ratio_measure = ratio_measure;
WHERE
  WR1 :
    SELF > 0.00000;
END_TYPE;

TYPE count_measure = NUMBER;
END_TYPE;

TYPE unit = SELECT
  (named_unit,
   derived_unit);
END_TYPE;

TYPE si_prefix = ENUMERATION OF
  (EXA,
   PETA,
   TERA,
   GIGA,
   MEGA,
   KILO,
   HECTO,
   DECA,
   DECI,
   CENTI,
   MILLI,
   MICRO,
   NANO,
   PICO,
   FEMTO,
   ATTO);
END_TYPE;

TYPE si_unit_name = ENUMERATION OF
  (METRE,
   GRAM,
   SECOND,
```

```

    AMPERE,
    KELVIN,
    MOLE,
    CANDELA,
    RADIAN,
    STERADIAN,
    HERTZ,
    NEWTON,
    PASCAL,
    JOULE,
    WATT,
    COULOMB,
    VOLT,
    FARAD,
    OHM,
    SIEMENS,
    WEBER,
    TESLA,
    HENRY,
    DEGREE_CELSIUS,
    LUMEN,
    LUX,
    BECQUEREL,
    GRAY,
    SIEVERT);
END_TYPE;

TYPE value_qualifier = SELECT
    (precision_qualifier,
     type_qualifier);
END_TYPE;

TYPE identifier = STRING;
END_TYPE;

TYPE source_item = SELECT
    (identifier);
END_TYPE;

TYPE classification_item = SELECT
    (product,
     document_file,
     product_definition,
     product_definition_formation,
     product_definition_relationship,
     material_designation,
     general_property,
     property_definition,
     class,
     plus_minus_tolerance,
     document_type,
     planar_extent);
END_TYPE;

TYPE characterized_definition = SELECT
    (characterized_object,
     characterized_product_definition,
     shape_definition);

```

[www.iso.org/iso/13399-1:2006](http://www.iso.org/iso/13399-1:2006) : Click to view the full PDF of ISO 13399-1:2006

```

END_TYPE;

TYPE characterized_product_definition = SELECT
  (product_definition,
   product_definition_relationship);
END_TYPE;

TYPE shape_definition = SELECT
  (product_definition_shape,
   shape_aspect);
END_TYPE;

TYPE tolerance_method_definition = SELECT
  (tolerance_value,
   limits_and_fits);
END_TYPE;

TYPE dimensional_characteristic = SELECT
  (dimensional_size);
END_TYPE;

TYPE person_and_organization_item = SELECT
  (applied_classification_assignment,
   class_system,
   product,
   document_file,
   product_definition,
   product_definition_formation,
   class,
   product_definition_relationship,
   product_definition_formation_relationship,
   material_designation,
   person_and_organization,
   general_property,
   property_definition);
END_TYPE;

TYPE action_request_item = SELECT
  (document_file);
END_TYPE;

TYPE ahead_or_behind = ENUMERATION OF
  (AHEAD,
   EXACT,
   BEHIND);
END_TYPE;

TYPE applied_state_type_of_item = SELECT
  (product_definition);
END_TYPE;

TYPE approval_item = SELECT
  (applied_classification_assignment,
   class,
   class_system,
   document_file);
END_TYPE;

```

```

TYPE attribute_language_item = SELECT
  (property_definition,
   group,
   external_source,
   product_related_product_category,
   effectivity,
   effectivity_relationship,
   product,
   product_definition,
   product_definition_formation,
   product_definition_formation_relationship,
   representation,
   product_definition_relationship,
   general_property,
   general_property_relationship,
   representation_relationship,
   general_feature,
   identification_role,
   application_context,
   attribute_value_assignment,
   person_and_organization_role,
   organization_role,
   descriptive_representation_item);
END_TYPE;

```

```

TYPE date_time_or_event_occurrence = SELECT
  (date_time_select);
END_TYPE;

```

```

TYPE date_time_select = SELECT
  (date,
   local_time,
   date_and_time);
END_TYPE;

```

```

TYPE day_in_month_number = INTEGER;
WHERE
  WR1 :
    (1 <= SELF) AND (SELF <= 31);
END_TYPE;

```

```

TYPE month_in_year_number = INTEGER;
WHERE
  WR1 :
    (1 <= SELF) AND (SELF <= 12);
END_TYPE;

```

```

TYPE year_number = INTEGER;
END_TYPE;

```

```

TYPE hour_in_day = INTEGER;
WHERE
  WR1 :
    (0 <= SELF) AND (SELF < 24);
END_TYPE;

```

```

TYPE minute_in_hour = INTEGER;
WHERE

```

```

        WR1 :
            (0 <= SELF) AND (SELF <= 59);
END_TYPE;

TYPE second_in_minute = REAL;
    WHERE
        WR1 :
            (0 <= SELF) AND (SELF <= 60.0000);
END_TYPE;

TYPE transformation = SELECT
    (item_defined_transformation,
     functionally_defined_transformation);
END_TYPE;

TYPE multi_language_attribute_item = SELECT
    (property_definition,
     group,
     external_source,
     product_related_product_category,
     effectivity,
     effectivity_relationship,
     product_definition,
     product,
     product_definition_formation,
     product_definition_formation_relationship,
     representation,
     product_definition_relationship,
     general_property,
     general_property_relationship,
     representation_relationship,
     general_feature,
     identification_role,
     application_context,
     person_and_organization_role,
     organization_role,
     descriptive_representation_item);
END_TYPE;

TYPE attribute_type = SELECT
    (label,
     text);
END_TYPE;

TYPE axis2_placement = SELECT
    (axis2_placement_2d,
     axis2_placement_3d);
END_TYPE;

TYPE character_spacing_select = SELECT
    (length_measure,
     ratio_measure,
     measure_with_unit,
     descriptive_measure);
END_TYPE;

TYPE configuration_design_item = SELECT
    (product_definition,

```

```

    product_definition_formation);
END_TYPE;

TYPE date_and_time_item = SELECT
    (applied_classification_assignment,
    applied_organization_assignment,
    applied_person_and_organization_assignment,
    class,
    class_system,
    document_file);
END_TYPE;

TYPE organization_item = SELECT
    (class,
    property_definition,
    applied_identification_assignment,
    applied_classification_assignment,
    class_system,
    product,
    document_file,
    product_definition,
    product_definition_formation,
    product_definition_relationship,
    product_definition_formation_relationship,
    material_designation,
    person_and_organization,
    general_property);
END_TYPE;

TYPE external_identification_item = SELECT
    (document_file,
    product_definition,
    externally_defined_class,
    externally_defined_general_property);
END_TYPE;

TYPE identification_item = SELECT
    (document_file,
    product,
    general_property,
    applied_identification_assignment,
    property_definition,
    class_system,
    product_definition,
    document_type,
    class,
    product_definition_formation,
    product_definition_relationship,
    organization,
    person_and_organization,
    effectivity);
END_TYPE;

TYPE date_item = SELECT
    (applied_classification_assignment,
    applied_organization_assignment,
    applied_person_and_organization_assignment,
    class,

```

Click to view the full PDF of ISO 13399-1:2006

```

        class_system,
        document_file);
END_TYPE;

TYPE derived_property_select = SELECT
    (property_definition);
END_TYPE;

TYPE description_attribute_select = SELECT
    (application_context,
    context_dependent_shape_representation,
    effectivity,
    external_source,
    organization_role,
    person_and_organization_role,
    person_and_organization,
    property_definition_representation,
    representation);
END_TYPE;

TYPE represented_definition = SELECT
    (general_property,
    property_definition,
    shape_aspect);
END_TYPE;

TYPE document_reference_item = SELECT
    (property_definition,
    class_system,
    class,
    product,
    product_definition,
    product_definition_relationship,
    product_definition_formation,
    material_designation,
    organization,
    person,
    general_property,
    product_related_product_category,
    general_feature);
END_TYPE;

TYPE effectivity_item = SELECT
    (class_system,
    product,
    product_definition,
    document_file,
    product_definition_formation,
    product_definition_relationship,
    material_designation,
    applied_location_assignment,
    general_property,
    property_definition,
    applied_state_type_assignment);
END_TYPE;

TYPE location_assignment_item = SELECT
    (product_definition);

```

```

END_TYPE;

TYPE event_occurrence_item = SELECT
    (applied_classification_assignment,
     applied_person_and_organization_assignment,
     class,
     class_system,
     document_file);
END_TYPE;

TYPE founded_item_select = SELECT
    (representation_item);
END_TYPE;

TYPE geometric_set_select = SELECT
    (point);
END_TYPE;

TYPE id_attribute_select = SELECT
    (address,
     product_category,
     property_definition,
     shape_aspect,
     application_context,
     group,
     representation);
END_TYPE;

TYPE invisible_item = SELECT
    (representation);
END_TYPE;

TYPE layered_item = SELECT
    (representation_item);
END_TYPE;

TYPE name_attribute_select = SELECT
    (address,
     context_dependent_shape_representation,
     derived_unit,
     effectivity,
     person_and_organization,
     product_definition,
     property_definition_representation);
END_TYPE;

TYPE person_organization_select = SELECT
    (person,
     organization,
     person_and_organization);
END_TYPE;

TYPE product_or_formation_or_definition = SELECT
    (product,
     product_definition_formation,
     product_definition);
END_TYPE;

```

STANDARDS1.COM : Click to view the full PDF of ISO 13399-1:2006

```

TYPE property_or_shape_select = SELECT
    (property_definition,
     shape_definition);
END_TYPE;

TYPE role_select = SELECT
    (effectivity_assignment);
END_TYPE;

TYPE security_classification_item = SELECT
    (class_system,
     document_file);
END_TYPE;

TYPE shape_tolerance_select = SELECT
    (plus_minus_tolerance);
END_TYPE;

TYPE size_select = SELECT
    (positive_length_measure,
     measure_with_unit,
     descriptive_measure);
END_TYPE;

TYPE style_context_select = SELECT
    (group,
     representation,
     representation_item);
END_TYPE;

TYPE time_interval_item = SELECT
    (applied_classification_assignment,
     applied_person_and_organization_assignment,
     class,
     class_system,
     document_file);
END_TYPE;

TYPE trimming_select = SELECT
    (cartesian_point,
     parameter_value);
END_TYPE;

TYPE vector_or_direction = SELECT
    (vector,
     direction);
END_TYPE;

ENTITY applied_classification_assignment
    SUBTYPE OF(classification_assignment);
    items : SET [1:?] OF classification_item;
    WHERE
        WR1 :
            NOT ('CUTTING_TOOL_SCHEMA.' + 'CLASS' IN
TYPEOF(SELF.assigned_class)) OR (SIZEOF(QUERY (i <* SELF.items|
(SIZEOF([ ('CUTTING_TOOL_SCHEMA.' + 'ACTION'), ('CUTTING_TOOL_SCHEMA.' +
'ACTION_DIRECTIVE'), ('CUTTING_TOOL_SCHEMA.' + 'ACTION_METHOD'),
('CUTTING_TOOL_SCHEMA.' + 'ACTION_PROPERTY'), ('CUTTING_TOOL_SCHEMA.' +

```

```

'APPLICATION_CONTEXT'), ('CUTTING_TOOL_SCHEMA.' + 'APPROVAL_STATUS'),
('CUTTING_TOOL_SCHEMA.' + 'ASSEMBLY_COMPONENT_USAGE'),
('CUTTING_TOOL_SCHEMA.' + 'CONFIGURATION_ITEM'), ('CUTTING_TOOL_SCHEMA.'
+ 'CONTRACT'), ('CUTTING_TOOL_SCHEMA.' + 'DOCUMENT_TYPE'),
('CUTTING_TOOL_SCHEMA.' + 'FEATURE_DEFINITION'), ('CUTTING_TOOL_SCHEMA.'
+ 'GENERAL_PROPERTY'), ('CUTTING_TOOL_SCHEMA.' +
'MATERIAL_DESIGNATION'), ('CUTTING_TOOL_SCHEMA.' +
'ORGANIZATIONAL_PROJECT'), ('CUTTING_TOOL_SCHEMA.' + 'PRODUCT'),
('CUTTING_TOOL_SCHEMA.' + 'PRODUCT_CONCEPT'), ('CUTTING_TOOL_SCHEMA.' +
'PRODUCT_CONCEPT_FEATURE_CATEGORY'), ('CUTTING_TOOL_SCHEMA.' +
'PRODUCT_DEFINITION'), ('CUTTING_TOOL_SCHEMA.' +
'PRODUCT_DEFINITION_FORMATION'), ('CUTTING_TOOL_SCHEMA.' +
'PROPERTY_DEFINITION'), ('CUTTING_TOOL_SCHEMA.' + 'RESOURCE_PROPERTY'),
('CUTTING_TOOL_SCHEMA.' + 'SECURITY_CLASSIFICATION_LEVEL'),
('CUTTING_TOOL_SCHEMA.' + 'SHAPE_ASPECT'), ('CUTTING_TOOL_SCHEMA.' +
'VERSIONED_ACTION_REQUEST'] * TYPEOF(i)) (<> 1))) = 0);
    WR2 :
        NOT (('CUTTING_TOOL_SCHEMA.' + 'GROUP' IN
TYPEOF(SELF.assigned_class)) AND (SELF.assigned_class\group.name = 'E'))
AND (SELF.assigned_class\group.description = 'dimensioning principle')
OR (SIZEOF(QUERY (i <* SELF.items| NOT ('CUTTING_TOOL_SCHEMA.' +
'PLUS_MINUS_TOLERANCE' IN TYPEOF(i)))) = 0);
    WR3 :
        NOT (('CUTTING_TOOL_SCHEMA.' + 'CLASS_SYSTEM' IN
TYPEOF(SELF.assigned_class)) AND (SELF.role.name = 'class system
membership')) OR (SIZEOF(QUERY (i <* SELF.items| (SIZEOF([
('CUTTING_TOOL_SCHEMA.' + 'APPROVAL_STATUS'), ('CUTTING_TOOL_SCHEMA.' +
'CLASS'), ('CUTTING_TOOL_SCHEMA.' + 'DESCRIPTIVE_REPRESENTATION_ITEM'),
('CUTTING_TOOL_SCHEMA.' + 'DOCUMENT_TYPE'), ('CUTTING_TOOL_SCHEMA.' +
'SECURITY_CLASSIFICATION_LEVEL'), ('CUTTING_TOOL_SCHEMA.' +
'APPROVAL_STATUS'] * TYPEOF(i)) (<> 1)))) = 0);
END_ENTITY;

ENTITY classification_assignment
    ABSTRACT SUPERTYPE;
    assigned_class : group;
    role           : classification_role;
END_ENTITY;

ENTITY language_assignment
    SUBTYPE OF(classification_assignment);
    items : SET [1:?] OF language_item;
    WHERE
        WR1 : 'CUTTING_TOOL_SCHEMA.' + 'LANGUAGE' IN
TYPEOF(SELF.assigned_class);
        WR2 :
            SELF.role.name = 'language';
        WR3 :
            SIZEOF(SELF.items) = SIZEOF(QUERY (i <* SELF.items|
('CUTTING_TOOL_SCHEMA.' + 'REPRESENTATION' IN TYPEOF(i)) AND
(i\representation.name = 'document content')));
END_ENTITY;

ENTITY representation;
    name           : label;
    items          : SET [1:?] OF representation_item;
    context_of_items : representation_context;

```

```

DERIVE
  id          : identifier := get_id_value(SELF);
  description : text := get_description_value(SELF);
WHERE
  WR1 :
    SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
'ID_ATTRIBUTE.IDENTIFIED_ITEM')) <= 1;
  WR2 :
    SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1;
END_ENTITY;

ENTITY shape_representation
  SUBTYPE OF(representation);
END_ENTITY;

ENTITY shape_dimension_representation
  SUBTYPE OF(shape_representation);
WHERE
  WR1 :
    SIZEOF(QUERY (temp <* SELF\representation.items| NOT
('CUTTING_TOOL_SCHEMA.MEASURE_REPRESENTATION_ITEM' IN TYPEOF(temp)))) =
0;
  WR2 :
    SIZEOF(SELF\representation.items) <= 3;
  WR3 :
    SIZEOF(QUERY (pos_mri <* QUERY (real_mri <*
SELF\representation.items| ('REAL' IN
TYPEOF(real_mri\measure_with_unit.value_component)))| NOT
(pos_mri\measure_with_unit.value_component > 0.00000))) = 0;
END_ENTITY;

ENTITY representation_item;
  name : label;
WHERE
  WR1 :
    SIZEOF(using_representations(SELF)) > 0;
END_ENTITY;

ENTITY geometric_representation_item
  SUPERTYPE OF (ONEOF(point, direction, vector, placement,
cartesian_transformation_operator) ANDOR planar_extent)
  SUBTYPE OF(representation_item);
DERIVE
  dim : dimension_count := dimension_of(SELF);
WHERE
  WR1 :
    SIZEOF(QUERY (using_rep <* using_representations(SELF)| NOT
('CUTTING_TOOL_SCHEMA.GEOMETRIC_REPRESENTATION_CONTEXT' IN
TYPEOF(using_rep.context_of_items)))) = 0;
END_ENTITY;

ENTITY point
  SUBTYPE OF(geometric_representation_item);
END_ENTITY;

ENTITY cartesian_point
  SUBTYPE OF(point);

```

```

    coordinates : LIST [1:3] OF length_measure;
END_ENTITY;

ENTITY direction
  SUBTYPE OF(geometric_representation_item);
  direction_ratios : LIST [2:3] OF REAL;
  WHERE
    WR1 :
      SIZEOF(QUERY (tmp <* direction_ratios| (tmp <> 0.00000))) > 0;
END_ENTITY;

ENTITY vector
  SUBTYPE OF(geometric_representation_item);
  orientation : direction;
  magnitude : length_measure;
  WHERE
    WR1 :
      magnitude >= 0.00000;
END_ENTITY;

ENTITY placement
  SUPERTYPE OF (ONEOF(axis2_placement_2d, axis2_placement_3d))
  SUBTYPE OF(geometric_representation_item);
  location : cartesian_point;
END_ENTITY;

ENTITY axis2_placement_2d
  SUBTYPE OF(placement);
  ref_direction : OPTIONAL direction;
  DERIVE
    p : LIST [2:2] OF direction :=
  build_2axes(ref_direction);
  WHERE
    WR1 :
      SELF\geometric_representation_item.dim = 2;
END_ENTITY;

ENTITY axis2_placement_3d
  SUBTYPE OF(placement);
  axis : OPTIONAL direction;
  ref_direction : OPTIONAL direction;
  DERIVE
    p : LIST [3:3] OF direction := build_axes(axis,
  ref_direction);
  WHERE
    WR1 :
      SELF\placement.location.dim = 3;
    WR2 :
      NOT EXISTS(axis) OR (axis.dim = 3);
    WR3 :
      NOT EXISTS(ref_direction) OR (ref_direction.dim = 3);
    WR4 :
      (NOT EXISTS(axis) OR NOT EXISTS(ref_direction)) OR
      (cross_product(axis, ref_direction).magnitude > 0.00000);
END_ENTITY;

ENTITY cartesian_transformation_operator

```

```

    SUPERTYPE OF (ONEOF(cartesian_transformation_operator_2d,
cartesian_transformation_operator_3d))
    SUBTYPE OF(geometric_representation_item,
functionally_defined_transformation);
    axis1      : OPTIONAL direction;
    axis2      : OPTIONAL direction;
    local_origin : cartesian_point;
    scale      : OPTIONAL REAL;
    DERIVE
    scl        : REAL := NVL(scale, 1.00000);
    WHERE
    WR1 :
        scl > 0.00000;
END_ENTITY;

ENTITY functionally_defined_transformation;
    name      : label;
    description : OPTIONAL text;
END_ENTITY;

ENTITY cartesian_transformation_operator_2d
    SUBTYPE OF(cartesian_transformation_operator);
    DERIVE
    u : LIST [2:2] OF direction := base_axis(2,
SELF\cartesian_transformation_operator.axis1,
SELF\cartesian_transformation_operator.axis2, ?);
    WHERE
    WR1 :
        SELF\geometric_representation_item.dim = 2;
END_ENTITY;

ENTITY cartesian_transformation_operator_3d
    SUBTYPE OF(cartesian_transformation_operator);
    axis3 : OPTIONAL direction;
    DERIVE
    u : LIST [3:3] OF direction := base_axis(3,
SELF\cartesian_transformation_operator.axis1,
SELF\cartesian_transformation_operator.axis2, axis3);
    WHERE
    WR1 :
        SELF\geometric_representation_item.dim = 3;
END_ENTITY;

ENTITY planar_extent
    SUBTYPE OF(geometric_representation_item);
    size_in_x : length_measure;
    size_in_y : length_measure;
END_ENTITY;

ENTITY descriptive_representation_item
    SUBTYPE OF(representation_item);
    description : text;
END_ENTITY;

ENTITY compound_representation_item
    SUBTYPE OF(representation_item);
    item_element : compound_item_definition;
END_ENTITY;

```

```

ENTITY value_range
  SUBTYPE OF(compound_representation_item);
  WHERE
    WR1 :
      ('CUTTING_TOOL_SCHEMA.' + 'SET_REPRESENTATION_ITEM' IN
TYPEOF(item_element)) AND value_range_wr1(item_element);
    WR2 :
      value_range_wr2(item_element);
    WR3 :
      value_range_wr3(item_element);
END_ENTITY;

ENTITY mapped_item
  SUBTYPE OF(representation_item);
  mapping_source : representation_map;
  mapping_target : representation_item;
  WHERE
    WR1 :
      acyclic_mapped_representation(using_representations(SELF), [
SELF ]);
END_ENTITY;

ENTITY representation_map;
  mapping_origin      : representation_item;
  mapped_representation : representation;
  INVERSE
  map_usage          : SET [1:?] OF mapped_item FOR mapping_source;
  WHERE
    WR1 :
      item_in_context(SELF.mapping_origin,
SELF.mapped_representation.context_of_items);
END_ENTITY;

ENTITY measure_representation_item
  SUBTYPE OF(representation_item, measure_with_unit);
END_ENTITY;

ENTITY measure_with_unit
  SUPERTYPE OF (ONEOF(length_measure_with_unit, mass_measure_with_unit,
time_measure_with_unit, electric_current_measure_with_unit,
thermodynamic_temperature_measure_with_unit,
celsius_temperature_measure_with_unit,
amount_of_substance_measure_with_unit,
luminous_intensity_measure_with_unit, plane_angle_measure_with_unit,
solid_angle_measure_with_unit, area_measure_with_unit,
volume_measure_with_unit, ratio_measure_with_unit) ANDOR
measure_representation_item);
  value_component : measure_value;
  unit_component  : unit;
  WHERE
    WR1 :
      valid_units(SELF);
END_ENTITY;

ENTITY length_measure_with_unit
  SUBTYPE OF(measure_with_unit);
  WHERE

```

```

WR1 :
    'CUTTING_TOOL_SCHEMA.LENGTH_UNIT' IN
TYPEOF(SELF\measure_with_unit.unit_component);
END_ENTITY;

ENTITY mass_measure_with_unit
    SUBTYPE OF(measure_with_unit);
    WHERE
        WR1 :
            'CUTTING_TOOL_SCHEMA.MASS_UNIT' IN
TYPEOF(SELF\measure_with_unit.unit_component);
END_ENTITY;

ENTITY time_measure_with_unit
    SUBTYPE OF(measure_with_unit);
    WHERE
        WR1 :
            'CUTTING_TOOL_SCHEMA.TIME_UNIT' IN
TYPEOF(SELF\measure_with_unit.unit_component);
END_ENTITY;

ENTITY electric_current_measure_with_unit
    SUBTYPE OF(measure_with_unit);
    WHERE
        WR1 :
            'CUTTING_TOOL_SCHEMA.ELECTRIC_CURRENT_UNIT' IN
TYPEOF(SELF\measure_with_unit.unit_component);
END_ENTITY;

ENTITY thermodynamic_temperature_measure_with_unit
    SUBTYPE OF(measure_with_unit);
    WHERE
        WR1 :
            'CUTTING_TOOL_SCHEMA.THERMODYNAMIC_TEMPERATURE_UNIT' IN
TYPEOF(SELF\measure_with_unit.unit_component);
END_ENTITY;

ENTITY celsius_temperature_measure_with_unit
    SUBTYPE OF(measure_with_unit);
    WHERE
        WR1 :
            'CUTTING_TOOL_SCHEMA.THERMODYNAMIC_TEMPERATURE_UNIT' IN
TYPEOF(SELF\measure_with_unit.unit_component);
END_ENTITY;

ENTITY amount_of_substance_measure_with_unit
    SUBTYPE OF(measure_with_unit);
    WHERE
        WR1 :
            'CUTTING_TOOL_SCHEMA.AMOUNT_OF_SUBSTANCE_UNIT' IN
TYPEOF(SELF\measure_with_unit.unit_component);
END_ENTITY;

ENTITY luminous_intensity_measure_with_unit
    SUBTYPE OF(measure_with_unit);
    WHERE
        WR1 :

```

```

        'CUTTING_TOOL_SCHEMA.LUMINOUS_INTENSITY_UNIT' IN
    TYPEOF(SELF\measure_with_unit.unit_component);
END_ENTITY;

```

```

ENTITY plane_angle_measure_with_unit
    SUBTYPE OF(measure_with_unit);
    WHERE
        WR1 :
            'CUTTING_TOOL_SCHEMA.PLANE_ANGLE_UNIT' IN
    TYPEOF(SELF\measure_with_unit.unit_component);
END_ENTITY;

```

```

ENTITY solid_angle_measure_with_unit
    SUBTYPE OF(measure_with_unit);
    WHERE
        WR1 :
            'CUTTING_TOOL_SCHEMA.SOLID_ANGLE_UNIT' IN
    TYPEOF(SELF\measure_with_unit.unit_component);
END_ENTITY;

```

```

ENTITY area_measure_with_unit
    SUBTYPE OF(measure_with_unit);
    WHERE
        WR1 :
            'CUTTING_TOOL_SCHEMA.AREA_UNIT' IN
    TYPEOF(SELF\measure_with_unit.unit_component);
END_ENTITY;

```

```

ENTITY volume_measure_with_unit
    SUBTYPE OF(measure_with_unit);
    WHERE
        WR1 :
            'CUTTING_TOOL_SCHEMA.VOLUME_UNIT' IN
    TYPEOF(SELF\measure_with_unit.unit_component);
END_ENTITY;

```

```

ENTITY ratio_measure_with_unit
    SUBTYPE OF(measure_with_unit);
    WHERE
        WR1 :
            'CUTTING_TOOL_SCHEMA.RATIO_UNIT' IN
    TYPEOF(SELF\measure_with_unit.unit_component);
END_ENTITY;

```

```

ENTITY named_unit
    SUPERTYPE OF (ONEOF(si_unit, conversion_based_unit,
    context_dependent_unit) ANDOR ONEOF(length_unit, mass_unit, time_unit,
    electric_current_unit, thermodynamic_temperature_unit,
    amount_of_substance_unit, luminous_intensity_unit, plane_angle_unit,
    solid_angle_unit, area_unit, volume_unit, ratio_unit) ANDOR
    named_unit_variable);
    dimensions : dimensional_exponents;
END_ENTITY;

```

```

ENTITY si_unit
    SUBTYPE OF(named_unit);
    prefix      : OPTIONAL si_prefix;
    name       : si_unit_name;

```

```

DERIVE
  SELF\named_unit.dimensions : dimensional_exponents :=
dimensions_for_si_unit(name);
END_ENTITY;

ENTITY dimensional_exponents;
  length_exponent           : REAL;
  mass_exponent             : REAL;
  time_exponent             : REAL;
  electric_current_exponent : REAL;
  thermodynamic_temperature_exponent : REAL;
  amount_of_substance_exponent : REAL;
  luminous_intensity_exponent : REAL;
END_ENTITY;

ENTITY conversion_based_unit
  SUBTYPE OF(named_unit);
  name          : label;
  conversion_factor : measure_with_unit;
END_ENTITY;

ENTITY context_dependent_unit
  SUBTYPE OF(named_unit);
  name : label;
END_ENTITY;

ENTITY expression_conversion_based_unit
  SUBTYPE OF(context_dependent_unit, variable_semantics);
  INVERSE
    associated_variable_environment : environment FOR semantics;
END_ENTITY;

ENTITY variable_semantics
  ABSTRACT SUPERTYPE;
END_ENTITY;

ENTITY derived_unit_variable
  SUBTYPE OF(derived_unit, variable_semantics);
  INVERSE
    associated_variable_environment : environment FOR semantics;
END_ENTITY;

ENTITY derived_unit;
  elements : SET [1:?] OF derived_unit_element;
  DERIVE
    name : label := get_name_value(SELF);
  WHERE
    WR1 :
      (SIZEOF(elements) > 1) OR (SIZEOF(elements) = 1) AND
      (elements[1].exponent <> 1.00000);
    WR2 :
      SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
'NAME_ATTRIBUTE.NAMED_ITEM')) <= 1;
END_ENTITY;

ENTITY derived_unit_element;
  unit : named_unit;
  exponent : REAL;

```

```

END_ENTITY;

ENTITY environment;
    syntactic_representation : generic_variable;
    semantics                 : variable_semantics;
END_ENTITY;

ENTITY generic_variable
    ABSTRACT SUPERTYPE
    SUBTYPE OF (simple_generic_expression);
    INVERSE
        interpretation : environment FOR syntactic_representation;
END_ENTITY;

ENTITY simple_generic_expression
    ABSTRACT SUPERTYPE
    SUBTYPE OF (generic_expression);
END_ENTITY;

ENTITY generic_expression
    ABSTRACT SUPERTYPE OF (ONEOF (simple_generic_expression,
    unary_generic_expression, binary_generic_expression,
    multiple_arity_generic_expression));
    WHERE
        WR1 :
            is_acyclic (SELF);
END_ENTITY;

ENTITY unary_generic_expression
    ABSTRACT SUPERTYPE
    SUBTYPE OF (generic_expression);
    operand : generic_expression;
END_ENTITY;

ENTITY binary_generic_expression
    ABSTRACT SUPERTYPE
    SUBTYPE OF (generic_expression);
    operands : LIST [2:2] OF generic_expression;
END_ENTITY;

ENTITY multiple_arity_generic_expression
    ABSTRACT SUPERTYPE
    SUBTYPE OF (generic_expression);
    operands : LIST [2:?] OF generic_expression;
END_ENTITY;

ENTITY named_unit_variable
    SUBTYPE OF (named_unit, variable_semantics);
    INVERSE
        associated_variable_environment : environment FOR semantics;
END_ENTITY;

ENTITY length_unit
    SUBTYPE OF (named_unit);
    WHERE
        WR1 :
            (((((SELF\named_unit.dimensions.length_exponent = 1.00000) AND
            (SELF\named_unit.dimensions.mass_exponent = 0.00000)) AND

```

```

(SELF\named_unit.dimensions.time_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.electric_current_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.thermodynamic_temperature_exponent =
0.00000)) AND (SELF\named_unit.dimensions.amount_of_substance_exponent =
0.00000)) AND (SELF\named_unit.dimensions.luminous_intensity_exponent =
0.00000);
END_ENTITY;

ENTITY mass_unit
  SUBTYPE OF(named_unit);
  WHERE
    WR1 :
      ((((((SELF\named_unit.dimensions.length_exponent = 0.00000) AND
(SELF\named_unit.dimensions.mass_exponent = 1.00000)) AND
(SELF\named_unit.dimensions.time_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.electric_current_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.thermodynamic_temperature_exponent =
0.00000)) AND (SELF\named_unit.dimensions.amount_of_substance_exponent =
0.00000)) AND (SELF\named_unit.dimensions.luminous_intensity_exponent =
0.00000);
END_ENTITY;

ENTITY time_unit
  SUBTYPE OF(named_unit);
  WHERE
    WR1 :
      ((((((SELF\named_unit.dimensions.length_exponent = 0.00000) AND
(SELF\named_unit.dimensions.mass_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.time_exponent = 1.00000)) AND
(SELF\named_unit.dimensions.electric_current_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.thermodynamic_temperature_exponent =
0.00000)) AND (SELF\named_unit.dimensions.amount_of_substance_exponent =
0.00000)) AND (SELF\named_unit.dimensions.luminous_intensity_exponent =
0.00000);
END_ENTITY;

ENTITY electric_current_unit
  SUBTYPE OF(named_unit);
  WHERE
    WR1 :
      ((((((SELF\named_unit.dimensions.length_exponent = 0.00000) AND
(SELF\named_unit.dimensions.mass_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.time_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.electric_current_exponent = 1.00000)) AND
(SELF\named_unit.dimensions.thermodynamic_temperature_exponent =
0.00000)) AND (SELF\named_unit.dimensions.amount_of_substance_exponent =
0.00000)) AND (SELF\named_unit.dimensions.luminous_intensity_exponent =
0.00000);
END_ENTITY;

ENTITY thermodynamic_temperature_unit
  SUBTYPE OF(named_unit);
  WHERE
    WR1 :
      ((((((SELF\named_unit.dimensions.length_exponent = 0.00000) AND
(SELF\named_unit.dimensions.mass_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.time_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.electric_current_exponent = 0.00000)) AND

```

```

(SELF\named_unit.dimensions.thermodynamic_temperature_exponent =
1.00000) AND (SELF\named_unit.dimensions.amount_of_substance_exponent =
0.00000) AND (SELF\named_unit.dimensions.luminous_intensity_exponent =
0.00000);
END_ENTITY;

ENTITY amount_of_substance_unit
  SUBTYPE OF(named_unit);
  WHERE
    WR1 :
      (((((SELF\named_unit.dimensions.length_exponent = 0.00000) AND
(SELF\named_unit.dimensions.mass_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.time_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.electric_current_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.thermodynamic_temperature_exponent =
0.00000) AND (SELF\named_unit.dimensions.amount_of_substance_exponent =
1.00000) AND (SELF\named_unit.dimensions.luminous_intensity_exponent =
0.00000);
END_ENTITY;

ENTITY luminous_intensity_unit
  SUBTYPE OF(named_unit);
  WHERE
    WR1 :
      (((((SELF\named_unit.dimensions.length_exponent = 0.00000) AND
(SELF\named_unit.dimensions.mass_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.time_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.electric_current_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.thermodynamic_temperature_exponent =
0.00000) AND (SELF\named_unit.dimensions.amount_of_substance_exponent =
0.00000) AND (SELF\named_unit.dimensions.luminous_intensity_exponent =
1.00000);
END_ENTITY;

ENTITY plane_angle_unit
  SUBTYPE OF(named_unit);
  WHERE
    WR1 :
      (((((SELF\named_unit.dimensions.length_exponent = 0.00000) AND
(SELF\named_unit.dimensions.mass_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.time_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.electric_current_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.thermodynamic_temperature_exponent =
0.00000) AND (SELF\named_unit.dimensions.amount_of_substance_exponent =
0.00000) AND (SELF\named_unit.dimensions.luminous_intensity_exponent =
0.00000);
END_ENTITY;

ENTITY solid_angle_unit
  SUBTYPE OF(named_unit);
  WHERE
    WR1 :
      (((((SELF\named_unit.dimensions.length_exponent = 0.00000) AND
(SELF\named_unit.dimensions.mass_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.time_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.electric_current_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.thermodynamic_temperature_exponent =
0.00000) AND (SELF\named_unit.dimensions.amount_of_substance_exponent =

```

```

0.00000)) AND (SELF\named_unit.dimensions.luminous_intensity_exponent =
0.00000);
END_ENTITY;

ENTITY area_unit
  SUBTYPE OF(named_unit);
  WHERE
    WR1 :
      ((((((SELF\named_unit.dimensions.length_exponent = 2.00000) AND
(SELF\named_unit.dimensions.mass_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.time_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.electric_current_exponent = 0.00000) AND
(SELF\named_unit.dimensions.thermodynamic_temperature_exponent =
0.00000)) AND (SELF\named_unit.dimensions.amount_of_substance_exponent =
0.00000)) AND (SELF\named_unit.dimensions.luminous_intensity_exponent =
0.00000);
END_ENTITY;

ENTITY volume_unit
  SUBTYPE OF(named_unit);
  WHERE
    WR1 :
      ((((((SELF\named_unit.dimensions.length_exponent = 3.00000) AND
(SELF\named_unit.dimensions.mass_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.time_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.electric_current_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.thermodynamic_temperature_exponent =
0.00000)) AND (SELF\named_unit.dimensions.amount_of_substance_exponent =
0.00000)) AND (SELF\named_unit.dimensions.luminous_intensity_exponent =
0.00000);
END_ENTITY;

ENTITY ratio_unit
  SUBTYPE OF(named_unit);
  WHERE
    WR1 :
      ((((((SELF\named_unit.dimensions.length_exponent = 0.00000) AND
(SELF\named_unit.dimensions.mass_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.time_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.electric_current_exponent = 0.00000)) AND
(SELF\named_unit.dimensions.thermodynamic_temperature_exponent =
0.00000)) AND (SELF\named_unit.dimensions.amount_of_substance_exponent =
0.00000)) AND (SELF\named_unit.dimensions.luminous_intensity_exponent =
0.00000);
END_ENTITY;

ENTITY qualified_representation_item
  SUBTYPE OF(representation_item);
  qualifiers : SET [1:?] OF value_qualifier;
  WHERE
    WR1 :
      SIZEOF(QUERY (temp <* qualifiers|
('CUTTING_TOOL_SCHEMA.PRECISION_QUALIFIER' IN TYPEOF(temp)))) < 2;
END_ENTITY;

ENTITY precision_qualifier;
  precision_value : INTEGER;
END_ENTITY;

```

```

ENTITY type_qualifier;
    name : label;
END_ENTITY;

ENTITY value_representation_item
    SUBTYPE OF(representation_item);
    value_component : measure_value;
    WHERE
        WR1 :
            SIZEOF(QUERY (rep <* using_representations(SELF) | NOT
('CUTTING_TOOL_SCHEMA.GLOBAL_UNIT_ASSIGNED_CONTEXT' IN
TYPEOF(rep.context_of_items)))) = 0;
END_ENTITY;

ENTITY representation_context;
    context_identifier : identifier;
    context_type : text;
    INVERSE
        representations_in_context : SET [1:?] OF representation FOR
context_of_items;
END_ENTITY;

ENTITY geometric_representation_context
    SUBTYPE OF(representation_context);
    coordinate_space_dimension : dimension count;
END_ENTITY;

ENTITY global_unit_assigned_context
    SUBTYPE OF(representation_context);
    units : SET [1:?] OF unit;
END_ENTITY;

ENTITY group;
    name : label;
    description : OPTIONAL text;
    DERIVE
        id : identifier := get_id_value(SELF);
    WHERE
        WR1 :
            SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
'ID_ATTRIBUTE_IDENTIFIED_ITEM')) <= 1;
END_ENTITY;

ENTITY class
    SUBTYPE OF(group);
END_ENTITY;

ENTITY externally_defined_class
    SUBTYPE OF(class, externally_defined_item);
END_ENTITY;

ENTITY externally_defined_item;
    item_id : source_item;
    source : external_source;
END_ENTITY;

ENTITY externally_defined_general_property

```

```

    SUBTYPE OF(general_property, externally_defined_item);
END_ENTITY;

ENTITY general_property;
    id          : identifier;
    name        : label;
    description : OPTIONAL text;
END_ENTITY;

ENTITY external_source;
    source_id   : source_item;
    DERIVE
        description : text := get_description_value(SELF);
    WHERE
        WR1 :
            SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
                'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1;
END_ENTITY;

ENTITY known_source
    SUBTYPE OF(external_source, pre_defined_item);
END_ENTITY;

ENTITY pre_defined_item;
    name : label;
END_ENTITY;

ENTITY characterized_class
    SUBTYPE OF(characterized_object, class);
END_ENTITY;

ENTITY characterized_object;
    name          : label;
    description   : OPTIONAL text;
END_ENTITY;

ENTITY document_file
    SUBTYPE OF(document, characterized_object);
    WHERE
        WR1 :
            SELF\characterized_object.name = '';
        WR2 :
            NOT EXISTS(SELF\characterized_object.description);
        WR3 :
            SIZEOF(QUERY (drt <* SELF\document.representation_types |
                (drt.name IN [ 'digital', 'physical' ]))) = 1;
END_ENTITY;

ENTITY document;
    id          : identifier;
    name        : label;
    description : OPTIONAL text;
    kind        : document_type;
    INVERSE
        representation_types : SET OF document_representation_type FOR
            represented_document;
END_ENTITY;

```

```

ENTITY document_type;
    product_data_type : label;
END_ENTITY;

ENTITY document_representation_type;
    name                : label;
    represented_document : document;
END_ENTITY;

ENTITY feature_definition
    SUBTYPE OF(characterized_object);
    WHERE
        WR1 :
            SIZEOF(QUERY (pdr <*
get_property_definition_representations(SELF)| ('CUTTING_TOOL_SCHEMA.' +
'SHAPE_REPRESENTATION_WITH_PARAMETERS' IN
TYPEOF(pdr.used_representation)))) <= 1;
        WR2 :
            (SIZEOF(QUERY (pdr <*
get_property_definition_representations(SELF)| ('CUTTING_TOOL_SCHEMA.' +
'SHAPE_REPRESENTATION_WITH_PARAMETERS' IN
TYPEOF(pdr.used_representation)))) = 0) OR (SIZEOF(QUERY (pdr <*
get_property_definition_representations(SELF)| ('CUTTING_TOOL_SCHEMA.' +
'SHAPE_REPRESENTATION_WITH_PARAMETERS' IN
TYPEOF(pdr.used_representation)) AND (SIZEOF(QUERY (i <*
pdr.used_representation.items| (i.name = 'orientation') AND
('CUTTING_TOOL_SCHEMA.' + 'PLACEMENT' IN TYPEOF(i)))) <= 1))) = 1);
        WR3 :
            NOT ((SIZEOF([ ('CUTTING_TOOL_SCHEMA.' + 'BARRING_HOLE'),
('CUTTING_TOOL_SCHEMA.' + 'BEAD'), ('CUTTING_TOOL_SCHEMA.' +
'HOLE_IN_PANEL'), ('CUTTING_TOOL_SCHEMA.' + 'FEATURE_IN_PANEL'),
('CUTTING_TOOL_SCHEMA.' + 'JOGGLE'), ('CUTTING_TOOL_SCHEMA.' +
'LOCATOR') ] * TYPEOF(SELF)) = 1) OR ('CUTTING_TOOL_SCHEMA.' +
'COMPOUND_FEATURE' IN TYPEOF(SELF)) AND (SIZEOF(QUERY (sa <*
get_shape_aspects(SELF)| (sa.name = 'compound feature in panel')) = 1))
OR (SIZEOF(QUERY (pdr <*get_property_definition_representations(SELF)|
('CUTTING_TOOL_SCHEMA.' + 'DIRECTION_SHAPE_REPRESENTATION' IN
TYPEOF(pdr.used_representation)))) <= 1) AND ((SIZEOF(QUERY (pdr <*
get_property_definition_representations(SELF)| ('CUTTING_TOOL_SCHEMA.' +
'DIRECTION_SHAPE_REPRESENTATION' IN TYPEOF(pdr.used_representation)))) =
0) OR (SIZEOF(QUERY (pdr <*
get_property_definition_representations(SELF)| ('CUTTING_TOOL_SCHEMA.' +
'DIRECTION_SHAPE_REPRESENTATION' IN TYPEOF(pdr.used_representation)) AND
(pdr.used_representation.name = 'punch direction')) = 1)));
        WR4 :
            (SIZEOF(QUERY (pdr <*
get_property_definition_representations(SELF)| ('CUTTING_TOOL_SCHEMA.' +
'SHAPE_REPRESENTATION_WITH_PARAMETERS' IN
TYPEOF(pdr.used_representation)))) = 0) OR (SIZEOF(QUERY (pdr <*
get_property_definition_representations(SELF)| (('CUTTING_TOOL_SCHEMA.'
+ 'SHAPE_REPRESENTATION_WITH_PARAMETERS' IN
TYPEOF(pdr.used_representation)) AND ('CUTTING_TOOL_SCHEMA.' +
'GEOMETRIC_REPRESENTATION_CONTEXT' IN
TYPEOF(pdr.used_representation.context_of_items))) AND
(pdr.used_representation.context_of_items\geometric_representation_conte
xt.coordinate_space_dimension = 3))) = 1);
        WR5 :

```

```

NOT ((SIZEOF([ ('CUTTING_TOOL_SCHEMA.' + 'BOSS'),
('CUTTING_TOOL_SCHEMA.' + 'RIB'), ('CUTTING_TOOL_SCHEMA.' + 'SLOT'),
('CUTTING_TOOL_SCHEMA.' + 'ROUND_HOLE'), ('CUTTING_TOOL_SCHEMA.' +
'POCKET') ] * TYPEOF(SELF)) = 1) OR ('CUTTING_TOOL_SCHEMA.' +
'COMPOUND_FEATURE' IN TYPEOF(SELF)) AND (SIZEOF(QUERY (sa <*
get_shape_aspects(SELF)| ('CUTTING_TOOL_SCHEMA.' +
'COMPOSITE_SHAPE_ASPECT' IN TYPEOF(sa)) AND (sa.name = 'compound feature
in solid')) = 1)) OR (SIZEOF(QUERY (pdr <*
get_property_definition_representations(SELF)| ('CUTTING_TOOL_SCHEMA.' +
'DIRECTION_SHAPE_REPRESENTATION' IN TYPEOF(pdr.used_representation))) =
1) AND (SIZEOF(QUERY (pdr <*
get_property_definition_representations(SELF)| ('CUTTING_TOOL_SCHEMA.' +
'DIRECTION_SHAPE_REPRESENTATION' IN TYPEOF(pdr.used_representation)) AND
(SIZEOF(QUERY (i <* pdr.used_representation.items| (i.name =
'orientation') AND ('CUTTING_TOOL_SCHEMA.' + 'PLACEMENT' IN TYPEOF(i)))
= 1))) = 1));

```

WR6 :

```

SIZEOF([ ('CUTTING_TOOL_SCHEMA.' + 'BEAD'),
('CUTTING_TOOL_SCHEMA.' + 'BARRING_HOLE'), ('CUTTING_TOOL_SCHEMA.' +
'BOSS'), ('CUTTING_TOOL_SCHEMA.' + 'COMPOUND_FEATURE'),
('CUTTING_TOOL_SCHEMA.' + 'EXTERNALLY_DEFINED_FEATURE_DEFINITION'),
('CUTTING_TOOL_SCHEMA.' + 'FEATURE_IN_PANEL'), ('CUTTING_TOOL_SCHEMA.' +
'GENERAL_FEATURE'), ('CUTTING_TOOL_SCHEMA.' + 'HOLE_IN_PANEL'),
('CUTTING_TOOL_SCHEMA.' + 'JOGGLE'), ('CUTTING_TOOL_SCHEMA.' +
'LOCATOR'), ('CUTTING_TOOL_SCHEMA.' + 'ROCKET'), ('CUTTING_TOOL_SCHEMA.'
+ 'RIB'), ('CUTTING_TOOL_SCHEMA.' + 'REPLICATE_FEATURE'),
('CUTTING_TOOL_SCHEMA.' + 'ROUND_HOLE'), ('CUTTING_TOOL_SCHEMA.' +
'SLOT'), ('CUTTING_TOOL_SCHEMA.' + 'THREAD') ] * TYPEOF(SELF)) <= 1;
END_ENTITY;

```

ENTITY general\_feature

SUBTYPE OF(feature\_definition);

WHERE

WR1 :

```

(SIZEOF(get_property_definition_representations(SELF)) <= 1)
AND (SIZEOF(QUERY (pdr <* get_property_definition_representations(SELF)|
('CUTTING_TOOL_SCHEMA.' + 'SHAPE_REPRESENTATION_WITH_PARAMETERS' IN
TYPEOF(pdr.used_representation)))) <= 1);

```

WR2 :

```

SIZEOF(QUERY (sa <* get_shape_aspects(SELF)| (sa.description =
'course of travel occurrence')) = SIZEOF(QUERY (sa <*
get_shape_aspects(SELF)| (sa.description = 'course of travel
occurrence') AND (SIZEOF(QUERY (sar <* USEDIN(sa, 'CUTTING_TOOL_SCHEMA.'
+ 'SHAPE_ASPECT_RELATIONSHIP.' + 'RELATED_SHAPE_ASPECT')|
('CUTTING_TOOL_SCHEMA.' + 'SHAPE_DEFINING_RELATIONSHIP' IN TYPEOF(sar))
AND (sar.name = 'course of travel')) = 1)))));

```

WR3 :

```

SIZEOF(QUERY (sa <* get_shape_aspects(SELF)| (sa.description =
'course of travel occurrence')) = SIZEOF(QUERY (sa <*
get_shape_aspects(SELF)| (sa.description = 'course of travel
occurrence') AND (SIZEOF(QUERY (sar <* USEDIN(sa, 'CUTTING_TOOL_SCHEMA.'
+ 'SHAPE_ASPECT_RELATIONSHIP.' + 'RELATED_SHAPE_ASPECT')|
('CUTTING_TOOL_SCHEMA.' + 'SHAPE_DEFINING_RELATIONSHIP' IN TYPEOF(sar))
AND ('CUTTING_TOOL_SCHEMA.' + 'PATH_FEATURE_COMPONENT' IN
TYPEOF(sar.relatng_shape_aspect)))) = 1)))));

```

WR4 :

```

        SIZEOF(QUERY (sa <* get_shape_aspects(SELf) | (sa.description =
'boundary occurrence')) = SIZEOF(QUERY (sa <* get_shape_aspects(SELf) |
(sa.description = 'boundary occurrence') AND (SIZEOF(QUERY (sar <*
USEDIN(sa, 'CUTTING_TOOL_SCHEMA.' + 'SHAPE_ASPECT_RELATIONSHIP.' +
'RELATED_SHAPE_ASPECT') | ('CUTTING_TOOL_SCHEMA.' +
'SHAPE_DEFINING_RELATIONSHIP' IN TYPEOF(sar)) AND (sar.description =
'profile usage')) = 1)))));
    WR5 :
        (SIZEOF(QUERY (sa <* get_shape_aspects(SELf) |
('CUTTING_TOOL_SCHEMA.' + 'COMPOSITE_SHAPE_ASPECT' IN TYPEOF(sa))) <=
1) AND (SIZEOF(QUERY (sa <* get_shape_aspects(SELf) |
(('CUTTING_TOOL_SCHEMA.' + 'COMPOSITE_SHAPE_ASPECT' IN TYPEOF(sa)) AND
(sa.name = 'general compound feature')) AND (SIZEOF(QUERY (sar <*
USEDIN(sa, 'CUTTING_TOOL_SCHEMA.' + 'SHAPE_ASPECT_RELATIONSHIP.' +
'RELATING_SHAPE_ASPECT') | ('CUTTING_TOOL_SCHEMA.' +
'FEATURE_COMPONENT_RELATIONSHIP' IN TYPEOF(sar)) AND
('CUTTING_TOOL_SCHEMA.' + 'INSTANCED_FEATURE' IN
TYPEOF(sar.related_shape_aspect)))) >= 2))) <= 1);
END_ENTITY;

ENTITY class_system
    SUBTYPE OF(group);
END_ENTITY;

ENTITY language
    SUBTYPE OF(group);
    WHERE
        WR1 :
            (SIZEOF(QUERY (ca <* USEDIN(SELf, 'CUTTING_TOOL_SCHEMA.' +
'CLASSIFICATION_ASSIGNMENT.' + 'ASSIGNED_CLASS') |
('CUTTING_TOOL_SCHEMA.' + 'LANGUAGE_ASSIGNMENT' IN TYPEOF(ca)))) > 0) OR
(SIZEOF(QUERY (aca <* USEDIN(SELf, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_CLASSIFICATION_ASSIGNMENT.' + 'ASSIGNED_CLASS') |
('CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT' IN
TYPEOF(aca)))) > 0);
END_ENTITY;

ENTITY classification_role;
    name          : label;
    description   : OPTIONAL text;
END_ENTITY;

ENTITY product;
    id            : identifier;
    name         : label;
    description   : OPTIONAL text;
    frame_of_reference : SET [1:?] OF product_context;
END_ENTITY;

ENTITY product_context
    SUBTYPE OF(application_context_element);
    discipline_type : label;
END_ENTITY;

ENTITY application_context_element
    SUPERTYPE OF (ONEOF(product_context, product_definition_context));
    name          : label;
    frame_of_reference : application_context;

```

```

END_ENTITY;

ENTITY product_definition_context
  SUBTYPE OF(application_context_element);
  life_cycle_stage : label;
END_ENTITY;

ENTITY application_context;
  application      : label;
  DERIVE
    description    : text := get_description_value(SELF);
    id             : identifier := get_id_value(SELF);
  INVERSE
    context_elements : SET [1:?] OF application_context_element FOR
frame_of_reference;
  WHERE
    WR1 :
      SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1;
    WR2 :
      SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
'ID_ATTRIBUTE.IDENTIFIED_ITEM')) <= 1;
END_ENTITY;

ENTITY product_definition;
  id             : identifier;
  description    : OPTIONAL text;
  formation      : product_definition_formation;
  frame_of_reference : product_definition_context;
  DERIVE
    name         : label := get_name_value(SELF);
  WHERE
    WR1 :
      SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
'NAME_ATTRIBUTE.NAMED_ITEM')) <= 1;
END_ENTITY;

ENTITY product_definition_with_associated_documents
  SUBTYPE OF(product_definition);
  documentation_ids : SET [1:?] OF document;
END_ENTITY;

ENTITY product_definition_formation;
  id             : identifier;
  description    : OPTIONAL text;
  of_product    : product;
  UNIQUE
    UR1 : id, of_product;
END_ENTITY;

ENTITY product_definition_relationship;
  id             : identifier;
  name          : label;
  description    : OPTIONAL text;
  relating_product_definition : product_definition;
  related_product_definition  : product_definition;
END_ENTITY;

```

```

ENTITY product_definition_usage
  SUBTYPE OF (product_definition_relationship);
  UNIQUE
  UR1 : id, relating_product_definition, related_product_definition;
  WHERE
  WR1 :
    acyclic_product_definition_relationship(SELF, [
  SELF\product_definition_relationship.related_product_definition ],
  'CUTTING_TOOL_SCHEMA.PRODUCT_DEFINITION_USAGE');
END_ENTITY;

ENTITY assembly_component_usage
  SUBTYPE OF (product_definition_usage);
  reference_designator : OPTIONAL identifier;
END_ENTITY;

ENTITY quantified_assembly_component_usage
  SUBTYPE OF (assembly_component_usage);
  quantity : measure_with_unit;
  WHERE
  WR1 :
    NOT ('NUMBER' IN TYPEOF(quantity.value_component)) OR
    (quantity.value_component > 0);
END_ENTITY;

ENTITY material_designation;
  name : label;
  definitions : SET [1:?] OF characterized_definition;
END_ENTITY;

ENTITY product_definition_shape
  SUBTYPE OF (property_definition);
  UNIQUE
  UR1 : definition;
  WHERE
  WR1 :
    SIZEOF([
  'CUTTING_TOOL_SCHEMA.CHARACTERIZED_PRODUCT_DEFINITION',
  'CUTTING_TOOL_SCHEMA.CHARACTERIZED_OBJECT'] *
  TYPEOF(SELF\property_definition.definition)) > 0;
END_ENTITY;

ENTITY property_definition;
  name : label;
  description : OPTIONAL text;
  definition : characterized_definition;
  DERIVE
  id : identifier := get_id_value(SELF);
  WHERE
  WR1 :
    SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
  'ID_ATTRIBUTE.IDENTIFIED_ITEM')) <= 1;
END_ENTITY;

ENTITY shape_aspect;
  name : label;
  description : OPTIONAL text;
  of_shape : product_definition_shape;

```

```

    product_definitional : LOGICAL;
  DERIVE
    id : identifier := get_id_value(SELF);
  WHERE
    WR1 :
      SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
        'ID_ATTRIBUTE.IDENTIFIED_ITEM')) <= 1;
  END_ENTITY;

  ENTITY plus_minus_tolerance;
    range : tolerance_method_definition;
    toleranced_dimension : dimensional_characteristic;
  UNIQUE
    UR1 : toleranced_dimension;
  END_ENTITY;

  ENTITY tolerance_value;
    lower_bound : measure_with_unit;
    upper_bound : measure_with_unit;
  WHERE
    WR1 :
      upper_bound\measure_with_unit.value_component >
      lower_bound\measure_with_unit.value_component;
    WR2 :
      upper_bound\measure_with_unit.unit_component =
      lower_bound\measure_with_unit.unit_component;
  END_ENTITY;

  ENTITY limits_and_fits;
    form_variance : label;
    zone_variance : label;
    grade : label;
    source : text;
  END_ENTITY;

  ENTITY dimensional_size;
    applies_to : shape_aspect;
    name : label;
  WHERE
    WR1 :
      applies_to.product_definitional = TRUE;
  END_ENTITY;

  ENTITY applied_person_and_organization_assignment
  SUBTYPE OF(person_and_organization_assignment);
    items : SET [1:?] OF person_and_organization_item;
  WHERE
    WR1 :
      NOT (SELF.role.name = 'signing for contract') OR
      item_correlation(SELF.items, [ 'APPLIED_ORGANIZATION_ASSIGNMENT' ]);
  END_ENTITY;

  ENTITY person_and_organization_assignment
  ABSTRACT SUPERTYPE;
    assigned_person_and_organization : person_and_organization;
    role : person_and_organization_role;
  END_ENTITY;

```

```

ENTITY person_and_organization;
    the_person      : person;
    the_organization : organization;
DERIVE
    name            : label := get_name_value(SELF);
    description     : text  := get_description_value(SELF);
WHERE
    WR1 :
        SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
'NAME_ATTRIBUTE.NAMED_ITEM')) <= 1;
    WR2 :
        SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1;
END_ENTITY;

ENTITY person;
    id              : identifier;
    last_name       : OPTIONAL label;
    first_name      : OPTIONAL label;
    middle_names    : OPTIONAL LIST [1:?] OF label;
    prefix_titles   : OPTIONAL LIST [1:?] OF label;
    suffix_titles   : OPTIONAL LIST [1:?] OF label;
WHERE
    WR1 :
        EXISTS(last_name) OR EXISTS(first_name);
END_ENTITY;

ENTITY organization;
    id              : OPTIONAL identifier;
    name            : label;
    description     : OPTIONAL text;
END_ENTITY;

ENTITY person_and_organization_role;
    name           : label;
DERIVE
    description : text := get_description_value(SELF);
WHERE
    WR1 :
        SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1;
END_ENTITY;

ENTITY product_definition_formation_relationship;
    id              : identifier;
    name            : label;
    description     : OPTIONAL text;
    relating_product_definition_formation : product_definition_formation;
    related_product_definition_formation : product_definition_formation;
END_ENTITY;

ENTITY product_related_product_category
    SUBTYPE OF(product_category);
    products : SET [1:?] OF product;
END_ENTITY;

ENTITY product_category;
    name          : label;

```

```

        description : OPTIONAL text;
    DERIVE
        id          : identifier := get_id_value(SELF);
    WHERE
        WR1 :
            SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
                'ID_ATTRIBUTE.IDENTIFIED_ITEM')) <= 1;
    END_ENTITY;

ENTITY effectivity
    SUPERTYPE OF (ONEOF(dated_effectivity,
        time_interval_based_effectivity));
    id          : identifier;
    DERIVE
        name      : label := get_name_value(SELF);
        description : text := get_description_value(SELF);
    WHERE
        WR1 :
            SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
                'NAME_ATTRIBUTE.NAMED_ITEM')) <= 1;
        WR2 :
            SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
                'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1;
    END_ENTITY;

ENTITY dated_effectivity
    SUBTYPE OF(effectivity);
    effectivity_end_date : OPTIONAL date_time_or_event_occurrence;
    effectivity_start_date : date_time_or_event_occurrence;
    END_ENTITY;

ENTITY date;
    year_component : year_number;
    END_ENTITY;

ENTITY calendar_date
    SUBTYPE OF(date);
    day_component   : day_in_month_number;
    month_component : month_in_year_number;
    WHERE
        WR1 :
            valid_calendar_date(SELF);
    END_ENTITY;

ENTITY local_time;
    hour_component   : hour_in_day;
    minute_component : OPTIONAL minute_in_hour;
    second_component : OPTIONAL second_in_minute;
    zone             : coordinated_universal_time_offset;
    WHERE
        WR1 :
            valid_time(SELF);
    END_ENTITY;

ENTITY coordinated_universal_time_offset;
    hour_offset      : INTEGER;
    minute_offset    : OPTIONAL INTEGER;
    sense            : ahead_or_behind;

```

```

DERIVE
  actual_minute_offset : INTEGER := NVL(minute_offset, 0);
WHERE
  WR1 :
    (0 <= hour_offset) AND (hour_offset < 24);
  WR2 :
    (0 <= actual_minute_offset) AND (actual_minute_offset <= 59);
  WR3 :
    NOT (((hour_offset <> 0) OR (actual_minute_offset <> 0)) AND
(sense = exact));
END_ENTITY;

ENTITY date_and_time;
  date_component : date;
  time_component : local_time;
END_ENTITY;

ENTITY time_interval_based_effectivity
  SUBTYPE OF(effectivity);
  effectivity_period : time_interval;
END_ENTITY;

ENTITY time_interval;
  id          : identifier;
  name       : label;
  description : OPTIONAL text;
END_ENTITY;

ENTITY time_interval_with_bounds
  SUBTYPE OF(time_interval);
  primary_bound   : OPTIONAL date_time_or_event_occurrence;
  secondary_bound : OPTIONAL date_time_or_event_occurrence;
  duration        : OPTIONAL time_measure_with_unit;
WHERE
  WR1 :
    NOT (EXISTS(secondary_bound) AND EXISTS(duration));
  WR2 :
    EXISTS(primary_bound) OR EXISTS(secondary_bound);
END_ENTITY;

ENTITY effectivity_relationship;
  name          : label;
  description   : OPTIONAL text;
  related_effectivity : effectivity;
  relating_effectivity : effectivity;
END_ENTITY;

ENTITY general_property_relationship;
  name          : label;
  description   : OPTIONAL text;
  relating_property : general_property;
  related_property  : general_property;
END_ENTITY;

ENTITY representation_relationship;
  name          : label;
  description   : OPTIONAL text;
  rep_1        : representation;

```

```

        rep_2          : representation;
END_ENTITY;

ENTITY shape_representation_relationship
  SUBTYPE OF (representation_relationship);
  WHERE
    WR1 :
      'CUTTING_TOOL_SCHEMA.SHAPE_REPRESENTATION' IN
      TYPEOF (SELF\representation_relationship.rep_1) +
      TYPEOF (SELF\representation_relationship.rep_2);
END_ENTITY;

ENTITY representation_relationship_with_transformation
  SUBTYPE OF (representation_relationship);
  transformation_operator : transformation;
  WHERE
    WR1 :
      SELF\representation_relationship.rep_1.context_of_items :<>:
      SELF\representation_relationship.rep_2.context_of_items;
END_ENTITY;

ENTITY item_defined_transformation;
  name          : label;
  description   : OPTIONAL text;
  transform_item_1 : representation_item;
  transform_item_2 : representation_item;
END_ENTITY;

ENTITY identification_role;
  name          : label;
  description   : OPTIONAL text;
END_ENTITY;

ENTITY attribute_value_assignment
  ABSTRACT SUPERTYPE;
  attribute_name : label;
  attribute_value : attribute_type;
  role          : attribute_value_role;
END_ENTITY;

ENTITY multi_language_attribute_assignment
  SUBTYPE OF (attribute_value_assignment);
  items      : SET [1:?] OF multi_language_attribute_item;
  DERIVE
    language : label := get_multi_language (SELF);
  WHERE
    WR1 :
      SELF\attribute_value_assignment.role.name = 'alternate
language';
    WR2 :
      (SIZEOF (USEDIN (SELF,
'CUTTING_TOOL_SCHEMA.ATTRIBUTE_LANGUAGE_ASSIGNMENT.ITEMS')) = 1) AND
(SIZEOF (QUERY (ala <* USEDIN (SELF, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS') | (ala.attribute_name =
'attribute_value')))) = 1);
END_ENTITY;

```

```

ENTITY organization_role;
    name          : label;
    DERIVE
        description : text := get_description_value(SELF);
    WHERE
        WR1 :
            SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1;
END_ENTITY;

ENTITY attribute_value_role;
    name          : label;
    description : OPTIONAL text;
END_ENTITY;

ENTITY applied_organization_assignment
    SUBTYPE OF(organization_assignment);
    items : SET [1:?] OF organization_item;
    WHERE
        WR1 :
            NOT (SELF.role.name = 'organization in contract') OR
item_correlation(SELF.items, [ 'CONTRACT' ]);
        WR2 :
            NOT (SELF.role.name = 'signing for contract') OR
item_correlation(SELF.items, [ 'APPLIED_ORGANIZATION_ASSIGNMENT' ]);
        WR3 :
            NOT (SELF.role.name = 'validity context') OR
item_correlation(SELF.items, [ 'ACTION_PROPERTY', 'RESOURCE_PROPERTY',
'PROPERTY_DEFINITION' ]);
        WR4 :
            NOT (SELF.role.name = 'alias scope') OR
item_correlation(SELF.items, [ 'APPLIED_IDENTIFICATION_ASSIGNMENT' ]);
END_ENTITY;

ENTITY organization_assignment
    ABSTRACT SUPERTYPE;
    assigned_organization : organization;
    role                  : organization_role;
END_ENTITY;

ENTITY applied_identification_assignment
    SUBTYPE OF(identification_assignment);
    items : SET [1:?] OF identification_item;
    WHERE
        WR1 :
            NOT (SELF.role.name = 'version') OR
item_correlation(SELF.items, [ 'ACTION', 'ACTION_DIRECTIVE',
'SHAPE_REPRESENTATION', 'EFFECTIVITY', 'PRODUCT_CONCEPT',
'CONFIGURATION_ITEM', 'PRODUCT_DEFINITION', 'PRODUCT_CONCEPT_FEATURE',
'DOCUMENT_FILE', 'CLASS', 'APPLIED_IDENTIFICATION_ASSIGNMENT',
'DRAUGHTING_MODEL',
'MECHANICAL_DESIGN_GEOMETRIC_PRESENTATION_REPRESENTATION',
'PRESENTATION_AREA' ]);
        WR2 :
            NOT (SELF.role.name = 'lot context') OR
item_correlation(SELF.items, [ 'PRODUCT_DEFINITION' ]);
        WR3 :

```

```

        NOT (SELF.role.name = 'property change id context') OR
item_correlation(SELF.items, [ 'PROPERTY_DEFINITION_RELATIONSHIP',
'SHAPE_ASPECT_RELATIONSHIP', 'ACTION_PROPERTY' ]);
    WR4 :
        NOT (SELF.role.name = 'size id') OR
item_correlation(SELF.items, [ 'DIMENSIONAL_SIZE' ]);
    WR5 :
        NOT (SELF.role.name = 'model change id') OR
item_correlation(SELF.items, [ 'PROPERTY_DEFINITION', 'ACTION_PROPERTY'
]);
    WR6 :
        NOT (SELF.role.name = 'alias') OR item_correlation(SELF.items,
[ 'APPLICATION_CONTEXT', 'APPROVAL_STATUS', 'ASSEMBLY_COMPONENT_USAGE',
'CLASS', 'CLASS_SYSTEM', 'DOCUMENT_TYPE', 'DRAUGHTING_MODEL',
'GENERAL_PROPERTY',
'MECHANICAL_DESIGN_GEOMETRIC_PRESENTATION_REPRESENTATION',
'ORGANIZATION', 'PRODUCT', 'PRODUCT_CONCEPT', 'PRODUCT_CONCEPT_FEATURE',
'PRODUCT_CONCEPT_FEATURE_CATEGORY', 'PRODUCT_DEFINITION',
'PRODUCT_DEFINITION_FORMATION', 'PROPERTY_DEFINITION',
'SECURITY_CLASSIFICATION_LEVEL', 'SHAPE_REPRESENTATION' ]);
END_ENTITY;

ENTITY identification_assignment
    ABSTRACT SUPERTYPE;
    assigned_id : identifier;
    role        : identification_role;
END_ENTITY;

ENTITY external_identification_assignment
    ABSTRACT SUPERTYPE
    SUBTYPE OF(identification_assignment);
    source : external_source;
END_ENTITY;

ENTITY applied_external_identification_assignment
    SUBTYPE OF(external_identification_assignment);
    items : SET [1:?] OF external_identification_item;
    WHERE
    WR1 :
        NOT (SELF.role.name = 'alternative document id and location')
OR item_correlation(SELF.items, [ 'DOCUMENT_FILE' ]);
    WR2 :
        NOT (SELF.role.name = 'version') OR
item_correlation(SELF.items, [ 'EXTERNALLY_DEFINED_CLASS',
'EXTERNALLY_DEFINED_GENERAL_PROPERTY' ]);
END_ENTITY;

ENTITY context_dependent_shape_representation;
    representation_relation : shape_representation_relationship;
    represented_product_relation : product_definition_shape;
    DERIVE
    description : text := get_description_value(SELF);
    name : label := get_name_value(SELF);
    WHERE
    WR1 :
        'CUTTING_TOOL_SCHEMA.PRODUCT_DEFINITION_RELATIONSHIP' IN
TYPEOF(SELF.represented_product_relation.definition);
    WR2 :

```

```

        SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1;
    WR3 :
        SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
'NAME_ATTRIBUTE.NAMED_ITEM')) <= 1;
END_ENTITY;

ENTITY property_definition_representation;
    definition      : represented_definition;
    used_representation : representation;
    DERIVE
        description      : text := get_description_value(SELF);
        name              : label := get_name_value(SELF);
    WHERE
        WR1 :
            SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1;
        WR2 :
            SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
'NAME_ATTRIBUTE.NAMED_ITEM')) <= 1;
END_ENTITY;

ENTITY shape_definition_representation
    SUBTYPE OF(property_definition_representation);
    WHERE
        WR1 :
            ('CUTTING_TOOL_SCHEMA.PRODUCT_DEFINITION_SHAPE' IN
TYPEOF(SELF.definition)) OR ('CUTTING_TOOL_SCHEMA.SHAPE_DEFINITION' IN
TYPEOF(SELF.definition.definition));
        WR2 :
            'CUTTING_TOOL_SCHEMA.SHAPE_REPRESENTATION' IN
TYPEOF(SELF.used_representation);
END_ENTITY;

ENTITY applied_location_assignment
    SUBTYPE OF(location_assignment);
    items : SET [1:?] OF location_assignment_item;
END_ENTITY;

ENTITY location_assignment
    ABSTRACT SUPERTYPE;
    id      : identifier;
    name    : label;
    description : OPTIONAL text;
    assigned_location : location;
    role    : location_role;
END_ENTITY;

ENTITY location;
    id      : identifier;
    name    : label;
    description : OPTIONAL text;
END_ENTITY;

ENTITY location_role;
    id      : identifier;
    name    : label;
    description : OPTIONAL text;

```

```

END_ENTITY;

ENTITY applied_state_type_assignment;
    assigned_state_type : state_type;
    item_set            : SET [1:?] OF applied_state_type_of_item;
    role                : state_type_role;
END_ENTITY;

ENTITY state_type;
    name                : STRING;
    description         : OPTIONAL STRING;
END_ENTITY;

ENTITY state_type_role;
    name                : STRING;
    description         : OPTIONAL STRING;
END_ENTITY;

ENTITY address;
    internal_location   : OPTIONAL label;
    street_number       : OPTIONAL label;
    street              : OPTIONAL label;
    postal_box          : OPTIONAL label;
    town                : OPTIONAL label;
    region              : OPTIONAL label;
    postal_code         : OPTIONAL label;
    country              : OPTIONAL label;
    facsimile_number    : OPTIONAL label;
    telephone_number    : OPTIONAL label;
    electronic_mail_address : OPTIONAL label;
    telex_number        : OPTIONAL label;
    DERIVE
        name            : label := get_name_value(SELF);
        url             : identifier := get_id_value(SELF);
    WHERE
        WR1 :
            ((((((EXISTS(internal_location) OR EXISTS(street_number))
OR EXISTS(street)) OR EXISTS(postal_box)) OR EXISTS(town)) OR
EXISTS(region) OR EXISTS(postal_code)) OR EXISTS(country)) OR
EXISTS(facsimile_number)) OR EXISTS(telephone_number)) OR
EXISTS(electronic_mail_address)) OR EXISTS(telex_number);
END_ENTITY;

ENTITY organizational_address
    SUBTYPE OF(address);
    organizations : SET [1:?] OF organization;
    description   : OPTIONAL text;
END_ENTITY;

ENTITY person_and_organization_address
    SUBTYPE OF(organizational_address, personal_address);
    WHERE
        WR1 :
            SIZEOF(SELF\personal_address.people) = 1;
        WR2 :
            SIZEOF(SELF\organizational_address.organizations) = 1;
        WR3 :

```

```

        SIZEOF(QUERY (pao <* USEDIN(SELF.people[1],
'CUTTING_TOOL_SCHEMA.PERSON_AND_ORGANIZATION.THE_PERSON') |
(pao.the_organization :=:
SELF\organizational_address.organizations[1]))) = 1;
END_ENTITY;

ENTITY personal_address
  SUBTYPE OF(address);
  people      : SET [1:?] OF person;
  description : OPTIONAL text;
END_ENTITY;

ENTITY effectivity_assignment
  ABSTRACT SUPERTYPE;
  assigned_effectivity : effectivity;
  DERIVE
  role                  : object_role := get_role(SELF);
  WHERE
  WR1 :
    SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
'ROLE_ASSOCIATION.ITEM_WITH_ROLE')) <= 1;
END_ENTITY;

ENTITY applied_effectivity_assignment
  SUBTYPE OF(effectivity_assignment);
  items : SET [1:?] OF effectivity_item;
  WHERE
  WR1 :
    SIZEOF([ ('CUTTING_TOOL_SCHEMA.' + 'LOT_EFFECTIVITY'),
('CUTTING_TOOL_SCHEMA.' + 'SERIAL_NUMBERED_EFFECTIVITY'),
('CUTTING_TOOL_SCHEMA.' + 'PRODUCT_DEFINITION_EFFECTIVITY') ] *
TYPEOF(SELF.assigned_effectivity)) = 0;
END_ENTITY;

ENTITY applied_ineffectivity_assignment
  SUBTYPE OF(effectivity_assignment);
  items : SET [1:?] OF effectivity_item;
  WHERE
  WR1 :
    SIZEOF([ ('CUTTING_TOOL_SCHEMA.' + 'LOT_EFFECTIVITY'),
('CUTTING_TOOL_SCHEMA.' + 'SERIAL_NUMBERED_EFFECTIVITY'),
('CUTTING_TOOL_SCHEMA.' + 'PRODUCT_DEFINITION_EFFECTIVITY') ] *
TYPEOF(SELF.assigned_effectivity)) = 0;
END_ENTITY;

ENTITY object_role;
  name      : label;
  description : OPTIONAL text;
END_ENTITY;

ENTITY applied_document_reference
  SUBTYPE OF(document_reference);
  items : SET [1:?] OF document_reference_item;
  WHERE
  WR1 :
    NOT (SELF.role.name = 'general tolerance definition') OR
item_correlation(SELF.items, [ 'REPRESENTATION' ]);
  WR2 :

```

```

        (SELF.role.name = 'general tolerance definition') OR
item_correlation(SELF.items, [ 'ACTION', 'ACTION_DIRECTIVE',
'ACTION_METHOD', 'ACTION_RELATIONSHIP', 'APPLIED_ACTION_ASSIGNMENT',
'APPROVAL', 'CERTIFICATION', 'CLASS', 'CLASS_SYSTEM',
'CONFIGURATION_DESIGN', 'CONFIGURATION_ITEM', 'CONTRACT',
'FEATURE_DEFINITION', 'GENERAL_PROPERTY', 'MATERIAL_DESIGNATION',
'ORGANIZATION', 'ORGANIZATIONAL_PROJECT', 'PERSON', 'PRODUCT_CONCEPT',
'PRODUCT_CONCEPT_FEATURE', 'PRODUCT_CONCEPT_FEATURE_CATEGORY',
'PRODUCT_DEFINITION', 'PRODUCT_DEFINITION_FORMATION',
'PRODUCT_DEFINITION_FORMATION_RELATIONSHIP',
'PRODUCT_DEFINITION_RELATIONSHIP', 'PRODUCT_DEFINITION_SUBSTITUTE',
'PRODUCT_RELATED_PRODUCT_CATEGORY', 'PROPERTY_DEFINITION',
'REPRESENTATION', 'RESOURCE_REQUIREMENT_TYPE',
'SECURITY_CLASSIFICATION', 'SHAPE_ASPECT', 'SHAPE_ASPECT_RELATIONSHIP',
'VERSIONED_ACTION_REQUEST' ]);
END_ENTITY;

ENTITY document_reference
  ABSTRACT SUPERTYPE;
  assigned_document : document;
  source            : label;
  DERIVE
  role              : object_role := get_role(SELF);
  WHERE
  WR1 :
      SIZEOF(USEDIN(SELF, 'CUTTING_TOOL_SCHEMA.' +
'ROLE_ASSOCIATION.' + 'ITEM_WITH_ROLE')) <= 1;
END_ENTITY;

ENTITY attribute_classification_assignment
  ABSTRACT SUPERTYPE;
  assigned_class : group;
  attribute_name : label;
  role          : classification_role;
END_ENTITY;

ENTITY attribute_language_assignment
  SUBTYPE OF(attribute_classification_assignment);
  items : SET [1:?] OF attribute_language_item;
  DERIVE
  language : label :=
SELF\attribute_classification_assignment.assigned_class.name;
  WHERE
  WR1 :
      SELF\attribute_classification_assignment.role.name IN [
'primary', 'translated' ];
  WR2 :
      'CUTTING_TOOL_SCHEMA.' + 'LANGUAGE' IN
TYPEOF(SELF\attribute_classification_assignment.assigned_class);
END_ENTITY;

ENTITY description_attribute;
  attribute_value : text;
  described_item  : description_attribute_select;
END_ENTITY;

ENTITY dimensional_characteristic_representation;
  dimension : dimensional_characteristic;

```

```

        representation : shape_dimension_representation;
END_ENTITY;

ENTITY document_product_association;
    name                : label;
    description          : OPTIONAL text;
    relating_document    : document;
    related_product      : product_or_formation_or_definition;
END_ENTITY;

ENTITY document_product_equivalence
    SUBTYPE OF (document_product_association);
    WHERE
        WR1 :
            SELF.name = 'equivalence';
        WR2 :
            NOT ('CUTTING_TOOL_SCHEMA.' + 'PRODUCT' IN
            TYPEOF(SELF.related_product)) OR
            (SELF.relying_document.kind.product_data_type = 'configuration
            controlled document') AND (SIZEOF(QUERY (prpc <*
            USEDIN(SELF.related_product, 'CUTTING_TOOL_SCHEMA.' +
            'PRODUCT_RELATED_PRODUCT_CATEGORY.' + 'PRODUCTS') | (prpc.name =
            'document')))) = 1);
        WR3 :
            NOT ('CUTTING_TOOL_SCHEMA.' + 'PRODUCT_DEFINITION_FORMATION' IN
            TYPEOF(SELF.related_product)) OR
            (SELF.relying_document.kind.product_data_type = 'configuration
            controlled document version') AND (SIZEOF(QUERY (prpc <*
            USEDIN(SELF.related_product\product_definition_formation.of_product,
            'CUTTING_TOOL_SCHEMA.PRODUCT_RELATED_PRODUCT_CATEGORY.PRODUCTS') |
            (prpc.name = 'document')))) = 1);
        WR4 :
            NOT ('CUTTING_TOOL_SCHEMA.' + 'PRODUCT_DEFINITION' IN
            TYPEOF(SELF.related_product)) OR
            (SELF.relying_document.kind.product_data_type = 'configuration
            controlled document definition') AND (SIZEOF(QUERY (prpc <*
            USEDIN(SELF.related_product\product_definition.formation.of_product,
            'CUTTING_TOOL_SCHEMA.PRODUCT_RELATED_PRODUCT_CATEGORY.PRODUCTS') |
            (prpc.name = 'document')))) = 1);
END_ENTITY;

ENTITY externally_defined_item_relationship;
    name                : label;
    description          : OPTIONAL text;
    relating_item        : externally_defined_item;
    related_item         : externally_defined_item;
END_ENTITY;

ENTITY general_property_association;
    name                : label;
    description          : OPTIONAL text;
    base_definition      : general_property;
    derived_definition   : derived_property_select;
    WHERE
        WR1 :
            SIZEOF(USEDIN(derived_definition, 'CUTTING_TOOL_SCHEMA.' +
            'GENERAL_PROPERTY_ASSOCIATION.' + 'DERIVED_DEFINITION')) = 1;
        WR2 :

```

```

        derived_definition.name = base_definition.name;
END_ENTITY;

ENTITY group_relationship;
    name          : label;
    description    : OPTIONAL text;
    relating_group : group;
    related_group  : group;
END_ENTITY;

ENTITY id_attribute;
    attribute_value : identifier;
    identified_item  : id_attribute_select;
END_ENTITY;

ENTITY measure_qualification;
    name          : label;
    description    : text;
    qualified_measure : measure_with_unit;
    qualifiers     : SET [1:?] OF value_qualifier;
    WHERE
        WR1 :
            SIZEOF(QUERY (temp <* qualifiers
                ('CUTTING_TOOL_SCHEMA.PRECISION_QUALIFIER' IN TYPEOF(temp)))) < 2;
END_ENTITY;

ENTITY name_attribute;
    attribute_value : label;
    named_item      : name_attribute_select;
END_ENTITY;

ENTITY product_category_relationship;
    name          : label;
    description    : OPTIONAL text;
    category      : product_category;
    sub_category   : product_category;
    WHERE
        WR1 :
            acyclic_product_category_relationship(SELF, [ SELF.sub_category
                ]);
END_ENTITY;

ENTITY product_definition_context_association;
    definition      : product_definition;
    frame_of_reference : product_definition_context;
    role            : product_definition_context_role;
END_ENTITY;

ENTITY product_definition_context_role;
    name          : label;
    description    : OPTIONAL text;
END_ENTITY;

ENTITY product_definition_occurrence_relationship;
    name          : label;
    description    : OPTIONAL text;
    occurrence     : product_definition;
    occurrence_usage : assembly_component_usage;

```

```

WHERE
  WR1 :
    occurrence_usage.relying_product_definition :<>: occurrence;
  WR2 :
    occurrence_usage.related_product_definition :<>: occurrence;
  WR3 :
    occurrence.formation :=:
occurrence_usage.related_product_definition.formation;
END_ENTITY;

ENTITY role_association;
  role          : object_role;
  item_with_role : role_select;
END_ENTITY;

RULE coating_requires_product_definition FOR
  (product_definition,
   product_definition_formation,
   product_related_product_category);
WHERE
  WR1 :
    SIZEOF(QUERY (pdf <* product_definition_formation|
(SIZEOF(QUERY (prpc <* USEDIN(pdf.of_product, 'CUTTING_TOOL_SCHEMA.' +
'PRODUCT_RELATED_PRODUCT_CATEGORY.' + 'PRODUCTS')| (prpc.name =
'coating')) > 0) AND (SIZEOF(QUERY (pd <* USEDIN(pdf,
'CUTTING_TOOL_SCHEMA.' + 'PRODUCT_DEFINITION.' + 'FORMATION')|
(pd.frame_of_reference.name = 'material_definition')) <> 1))) = 0;
END_RULE;

RULE compatible_dimension FOR
  (cartesian_point,
   direction,
   representation_context,
   geometric_representation_context);
WHERE
  WR1 :
    SIZEOF(QUERY (x <* cartesian_point| (SIZEOF(QUERY (y <*
geometric_representation_context| item_in_context(x, y) AND
(HIINDEX(x.coordinates) <> y.coordinate_space_dimension))) > 0))) = 0;
  WR2 :
    SIZEOF(QUERY (x <* direction| (SIZEOF(QUERY (y <*
geometric_representation_context| item_in_context(x, y) AND
(HIINDEX(x.direction_ratios) <> y.coordinate_space_dimension))) > 0))) =
0;
END_RULE;

RULE dependent_instantiable_attribute_value_role FOR
  (attribute_value_role);
WHERE
  WR1 :
    SIZEOF(QUERY (a <* attribute_value_role| NOT (SIZEOF(USEDIN(a,
'')) > 0))) = 0;
END_RULE;

RULE dependent_instantiable_classification_role FOR
  (classification_role);
WHERE
  WR1 :

```

```

        SIZEOF(QUERY (c <* classification_role| NOT (SIZEOF(USEDIN(c,
        '')) > 0))) = 0;
END_RULE;

RULE dependent_instantiable_date FOR
    (date);
    WHERE
        WR1 :
            SIZEOF(QUERY (d <* date| NOT (SIZEOF(USEDIN(d, '')) > 0))) = 0;
END_RULE;

RULE dependent_instantiable_date_and_time FOR
    (date_and_time);
    WHERE
        WR1 :
            SIZEOF(QUERY (d <* date_and_time| NOT (SIZEOF(USEDIN(d, '')) >
    0))) = 0;
END_RULE;

RULE dependent_instantiable_derived_unit FOR
    (derived_unit);
    WHERE
        WR1 :
            SIZEOF(QUERY (d <* derived_unit| NOT (SIZEOF(USEDIN(d, '')) >
    0))) = 0;
END_RULE;

RULE dependent_instantiable_identification_role FOR
    (identification_role);
    WHERE
        WR1 :
            SIZEOF(QUERY (i <* identification_role| NOT (SIZEOF(USEDIN(i,
    '')) > 0))) = 0;
END_RULE;

RULE dependent_instantiable_measure_with_unit FOR
    (measure_with_unit);
    WHERE
        WR1 :
            SIZEOF(QUERY (m <* measure_with_unit| NOT (SIZEOF(USEDIN(m,
    '')) > 0))) = 0;
END_RULE;

RULE dependent_instantiable_named_unit FOR
    (named_unit);
    WHERE
        WR1 :
            SIZEOF(QUERY (n <* named_unit| NOT (SIZEOF(USEDIN(n, '')) >
    0))) = 0;
END_RULE;

RULE dependent_instantiable_object_role FOR
    (object_role);
    WHERE
        WR1 :
            SIZEOF(QUERY (o <* object_role| NOT (SIZEOF(USEDIN(o, '')) >
    0))) = 0;
END_RULE;

```

```

RULE dependent_instantiable_organization_role FOR
  (organization_role);
  WHERE
    WR1 :
      SIZEOF(QUERY (o <* organization_role| NOT (SIZEOF(USEDIN(o,
  '')) > 0))) = 0;
END_RULE;

RULE dependent_instantiable_person_and_organization_role FOR
  (person_and_organization_role);
  WHERE
    WR1 :
      SIZEOF(QUERY (p <* person_and_organization_role| NOT
  (SIZEOF(USEDIN(p, '')) > 0))) = 0;
END_RULE;

RULE dependent_instantiable_precision_qualifier FOR
  (precision_qualifier);
  WHERE
    WR1 :
      SIZEOF(QUERY (p <* precision_qualifier| NOT (SIZEOF(USEDIN(p,
  '')) > 0))) = 0;
END_RULE;

RULE dependent_instantiable_product_definition_context_role FOR
  (product_definition_context_role);
  WHERE
    WR1 :
      SIZEOF(QUERY (p <* product_definition_context_role| NOT
  (SIZEOF(USEDIN(p, '')) > 0))) = 0;
END_RULE;

RULE dependent_instantiable_tolerance_value FOR
  (tolerance_value);
  WHERE
    WR1 :
      SIZEOF(QUERY (t <* tolerance_value| NOT (SIZEOF(USEDIN(t, ''))
  > 0))) = 0;
END_RULE;

RULE dependent_instantiable_type_qualifier FOR
  (type_qualifier);
  WHERE
    WR1 :
      SIZEOF(QUERY (t <* type_qualifier| NOT (SIZEOF(USEDIN(t, '')) >
  0))) = 0;
END_RULE;

RULE externally_defined_class_with_known_source_requirement FOR
  (externally_defined_class);
  WHERE
    WR1 :
      SIZEOF(QUERY (edc <* externally_defined_class|
  ('CUTTING_TOOL_SCHEMA.' + 'KNOWN_SOURCE' IN TYPEOF(edc.source)) AND
  (SIZEOF(QUERY (aoa <* USEDIN(edc,
  'CUTTING_TOOL_SCHEMA.APPLIED_ORGANIZATION_ASSIGNMENT.ITEMS')|
  (aoa.role.name = 'class supplier')) = 0))) = 0;

```

```

END_RULE;

RULE grade_requires_product_definition FOR
  (product_definition,
   product_definition_formation,
   product_related_product_category);
WHERE
  WR1 :
    SIZEOF(QUERY (pdf <* product_definition_formation|
(SIZEOF(QUERY (prpc <* USEDIN(pdf.of_product, 'CUTTING_TOOL_SCHEMA.' +
'PRODUCT_RELATED_PRODUCT_CATEGORY.' + 'PRODUCTS')| (prpc.name =
'grade')) > 0) AND (SIZEOF(QUERY (pd <* USEDIN(pdf,
'CUTTING_TOOL_SCHEMA.' + 'PRODUCT_DEFINITION.' + 'FORMATION')
(pd.frame_of_reference.name = 'material definition')) <> 1))) = 0;
END_RULE;

RULE person_requires_person_and_organization FOR
  (person);
WHERE
  WR1 :
    SIZEOF(QUERY (p <* person| (SIZEOF(USEDIN(p,
'CUTTING_TOOL_SCHEMA.PERSON_AND_ORGANIZATION.THE_PERSON')) = 0))) = 0;
END_RULE;

RULE plib_class_reference_requires_version FOR
  (externally_defined_class);
WHERE
  WR1 :
    SIZEOF(QUERY (edc <* externally_defined_class|
('CUTTING_TOOL_SCHEMA.' + 'KNOWN_SOURCE' IN TYPEOF(edc.source)) AND
(SIZEOF(QUERY (aei <* USEDIN(edc,
'CUTTING_TOOL_SCHEMA.APPLIED_EXTERNAL_IDENTIFICATION_ASSIGNMENT.ITEMS')|
(aei.role.name = 'version')) <> 1))) = 0;
  WR2 :
    SIZEOF(QUERY (edc <* externally_defined_class|
('CUTTING_TOOL_SCHEMA.' + 'KNOWN_SOURCE' IN TYPEOF(edc.source)) AND
(SIZEOF(QUERY (aei <* USEDIN(edc,
'CUTTING_TOOL_SCHEMA.APPLIED_IDENTIFICATION_ASSIGNMENT.ITEMS')|
(aei.role.name = 'version')) > 0))) = 0;
END_RULE;

RULE plib_property_reference_requires_name_scope FOR
  (externally_defined_general_property);
WHERE
  WR1 :
    SIZEOF(QUERY (edgp <* externally_defined_general_property|
('CUTTING_TOOL_SCHEMA.' + 'KNOWN_SOURCE' IN TYPEOF(edgp.source)) AND
(SIZEOF(QUERY (edir <* USEDIN(edgp, 'CUTTING_TOOL_SCHEMA.' +
'EXTERNALLY_DEFINED_ITEM_RELATIONSHIP.' + 'RELATING_ITEM')| ((edir.name
= 'name scope') AND ('CUTTING_TOOL_SCHEMA.' + 'EXTERNALLY_DEFINED_CLASS'
IN TYPEOF(edir.related_item))) AND ('CUTTING_TOOL_SCHEMA.' +
'KNOWN_SOURCE' IN TYPEOF(edir.related_item.source)))) <> 1))) = 0;
END_RULE;

RULE plib_property_reference_requires_version FOR
  (externally_defined_general_property);
WHERE
  WR1 :

```

```

        SIZEOF(QUERY (edgp <* externally_defined_general_property|
('CUTTING_TOOL_SCHEMA.' + 'KNOWN_SOURCE' IN TYPEOF(edgp.source)) AND
(SIZEOF(QUERY (edir <* USEDIN(edgp, 'CUTTING_TOOL_SCHEMA.' +
'APPLIED_EXTERNAL_IDENTIFICATION_ASSIGNMENT.' + 'ITEMS')|
(edir.role.name = 'version')) <> 1))) = 0;
END_RULE;

RULE product_requires_category FOR
    (product,
    product_related_product_category);
WHERE
    WR1 :
        SIZEOF(QUERY (p <* product| (SIZEOF(USEDIN(p,
'CUTTING_TOOL_SCHEMA.PRODUCT_RELATED_PRODUCT_CATEGORY.PRODUCTS')) = 0)))
= 0;
END_RULE;

RULE product_requires_id_owner FOR
    (product);
WHERE
    WR1 :
        SIZEOF(QUERY (prod <* product| (SIZEOF(QUERY (prpc <*
USEDIN(prod,
'CUTTING_TOOL_SCHEMA.PRODUCT_RELATED_PRODUCT_CATEGORY.PRODUCTS')|
(prpc.name IN [ 'part', 'tool', 'raw material' ]))) > 0) AND
(SIZEOF(QUERY (apoa <* USEDIN(prod,
'CUTTING_TOOL_SCHEMA.APPLIED_PERSON_AND_ORGANIZATION_ASSIGNMENT.ITEMS')|
(apoa\person_and_organization_assignment.role.name <> 'id owner')) +
(SIZEOF(QUERY (oa <* USEDIN(prod,
'CUTTING_TOOL_SCHEMA.APPLIED_ORGANIZATION_ASSIGNMENT.ITEMS')|
(oa\organization_assignment.role.name = 'id owner')) <> 1)))) = 0;
END_RULE;

RULE product_requires_version FOR
    (product);
WHERE
    WR1 :
        SIZEOF(QUERY (prod <* product| (SIZEOF(USEDIN(prod,
'CUTTING_TOOL_SCHEMA.' + 'PRODUCT_DEFINITION_FORMATION.' +
'OF_PRODUCT')) = 0))) = 0;
END_RULE;

RULE restrict_applied_classification_assignment_role FOR
    (applied_classification_assignment);
WHERE
    WR1 :
        SIZEOF(QUERY (aca <* applied_classification_assignment|
('CUTTING_TOOL_SCHEMA.' + 'CLASS' IN TYPEOF(aca.assigned_class)) AND NOT
(aca.role.name IN [ 'definitional', 'non-definitional', '' ]))) = 0;
    WR2 :
        SIZEOF(QUERY (aca <* applied_classification_assignment|
(aca.role.name IN [ 'definitional', 'non-definitional', '' ]) AND NOT
('CUTTING_TOOL_SCHEMA.' + 'CLASS' IN TYPEOF(aca.assigned_class))) = 0;
    WR3 :
        SIZEOF(QUERY (aca <* applied_classification_assignment|
('CUTTING_TOOL_SCHEMA.' + 'CLASS_SYSTEM' IN TYPEOF(aca.assigned_class))
AND (aca.role.name <> 'class system membership')) = 0;
    WR4 :

```

```

        SIZEOF(QUERY (aca <* applied_classification_assignment|
(aca.role.name = 'class system membership') AND NOT
('CUTTING_TOOL_SCHEMA.' + 'CLASS_SYSTEM' IN
TYPEOF(aca.assigned_class)))) = 0;
END_RULE;

RULE restrict_class_system_assignment_for_class FOR
  (applied_classification_assignment,
   class);
WHERE
  WR1 :
        SIZEOF(QUERY (c <* class| (SIZEOF(QUERY (aca <* USEDIN(c,
'CUTTING_TOOL_SCHEMA.' + 'CLASSIFICATION_ASSIGNMENT.' +
'ASSIGNED_CLASS')| (aca.role.name = 'class system membership') AND
('CUTTING_TOOL_SCHEMA.' + 'APPLIED_CLASSIFICATION_ASSIGNMENT' IN
TYPEOF(aca)))) > 1))) = 0;
END_RULE;

RULE restrict_class_system_assignment_for_planar_extent FOR
  (applied_classification_assignment,
   planar_extent);
WHERE
  WR1 :
        SIZEOF(QUERY (pe <* planar_extent| (pe\representation_item.name
= 'size format') AND (SIZEOF(QUERY (aca <* USEDIN(pe,
'CUTTING_TOOL_SCHEMA.APPLIED_CLASSIFICATION_ASSIGNMENT.ITEMS')|
(aca\classification_assignment.role.name = 'class system membership'))
> 1))) = 0;
END_RULE;

RULE restrict_externally_defined_item_relationship FOR
  (externally_defined_item_relationship);
WHERE
  WR1 :
        SIZEOF(QUERY (edir <* externally_defined_item_relationship|
(edir.name = 'name scope') AND ((NOT ('CUTTING_TOOL_SCHEMA.' +
'EXTERNALLY_DEFINED_GENERAL_PROPERTY' IN TYPEOF(edir.relating_item)) OR
NOT ('CUTTING_TOOL_SCHEMA.' + 'KNOWN_SOURCE' IN
TYPEOF(edir.relating_item.source))) OR NOT ('CUTTING_TOOL_SCHEMA.' +
'EXTERNALLY_DEFINED_CLASS' IN TYPEOF(edir.related_item))) OR NOT
('CUTTING_TOOL_SCHEMA.' + 'KNOWN_SOURCE' IN
TYPEOF(edir.related_item.source)))) = 0;
END_RULE;

RULE restrict_group_relationship_for_general_classification_hierarchy FOR
  (group_relationship,
   class);
WHERE
  WR1 :
        SIZEOF(QUERY (gr <* group_relationship| (gr.name = 'class
hierarchy') AND (NOT ('CUTTING_TOOL_SCHEMA.' + 'CLASS' IN
TYPEOF(gr.related_group)) OR NOT ('CUTTING_TOOL_SCHEMA.' + 'CLASS' IN
TYPEOF(gr.relating_group)))) = 0;
END_RULE;

RULE restrict_multi_language_for_application_context FOR
  (application_context);
WHERE

```

```

WR1 :
    SIZEOF(QUERY (ent <* application_context| (SIZEOF(QUERY (mlaa
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description')))) > 0))) = 0;
WR2 :
    SIZEOF(QUERY (ent <* application_context| (SIZEOF(QUERY (ala <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala.attribute_name <> 'description')))) > 0))) = 0;
WR3 :
    SIZEOF(QUERY (ent <* application_context| (SIZEOF(QUERY (mlaa1
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language)) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language)) > 1))) > 0))) = 0;
END_RULE;

```

```

RULE restrict_multi_language_for_descriptive_representation_item FOR
(descriptive_representation_item);

```

```

WHERE
WR1 :
    SIZEOF(QUERY (ent <* descriptive_representation_item|
(SIZEOF(QUERY (mlaa <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description')))) > 0))) = 0;
WR2 :
    SIZEOF(QUERY (ent <* descriptive_representation_item|
(SIZEOF(QUERY (ala <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (ala.attribute_name <>
'description')))) > 0))) = 0;
WR3 :
    SIZEOF(QUERY (ent <* descriptive_representation_item|
(SIZEOF(QUERY (mlaa1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language)) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language)) > 1))) > 0))) = 0;
END_RULE;

```

```

RULE restrict_multi_language_for_effectivity FOR
(effectivity);

```

```

WHERE
WR1 :
    SIZEOF(QUERY (ent <* effectivity| (SIZEOF(QUERY (mlaa <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description')))) > 0))) = 0;

```

```

WR2 :
    SIZEOF(QUERY (ent <* effectivity| (SIZEOF(QUERY (ala <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala.attribute_name <> 'description')) > 0))) = 0;
WR3 :
    SIZEOF(QUERY (ent <* effectivity| (SIZEOF(QUERY (mlaa1 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language))) > 1))) > 0))) = 0;
END_RULE;

```

```

RULE restrict_multi_language_for_effectivity_relationship FOR
(effectivity_relationship);

```

```

WHERE

```

```

WR1 :
    SIZEOF(QUERY (ent <* effectivity_relationship| (SIZEOF(QUERY
(mlaa <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description')) > 0))) = 0;
WR2 :
    SIZEOF(QUERY (ent <* effectivity_relationship| (SIZEOF(QUERY
(ala <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (ala.attribute_name <>
'description')) > 0))) = 0;
WR3 :
    SIZEOF(QUERY (ent <* effectivity_relationship| (SIZEOF(QUERY
(mlaa1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language))) > 1))) > 0))) = 0;
END_RULE;

```

```

RULE restrict_multi_language_for_external_source FOR
(external_source);

```

```

WHERE

```

```

WR1 :
    SIZEOF(QUERY (ent <* external_source| (SIZEOF(QUERY (mlaa <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description')) > 0))) = 0;
WR2 :
    SIZEOF(QUERY (ent <* external_source| (SIZEOF(QUERY (ala <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala.attribute_name <> 'description')) > 0))) = 0;
WR3 :

```

```

        SIZEOF(QUERY (ent <* external_source| (SIZEOF(QUERY (mlaa1 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language))) > 1))) > 0))) = 0;
END_RULE;

```

```

RULE restrict_multi_language_for_general_feature FOR
  (general_feature);

```

```

  WHERE

```

```

    WR1 :

```

```

        SIZEOF(QUERY (ent <* general_feature| (SIZEOF(QUERY (mlaa <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description')) > 0))) = 0;

```

```

    WR2 :

```

```

        SIZEOF(QUERY (ent <* general_feature| (SIZEOF(QUERY (ala <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala.attribute_name <> 'description')) > 0))) = 0;

```

```

    WR3 :

```

```

        SIZEOF(QUERY (ent <* general_feature| (SIZEOF(QUERY (mlaa1 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language))) > 1))) > 0))) = 0;
END_RULE;

```

```

RULE restrict_multi_language_for_general_property FOR
  (general_property);

```

```

  WHERE

```

```

    WR1 :

```

```

        SIZEOF(QUERY (ent <* general_property| (SIZEOF(QUERY (mlaa <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description')) > 0))) = 0;

```

```

    WR2 :

```

```

        SIZEOF(QUERY (ent <* general_property| (SIZEOF(QUERY (ala <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala.attribute_name <> 'description')) > 0))) = 0;

```

```

    WR3 :

```

```

        SIZEOF(QUERY (ent <* general_property| (SIZEOF(QUERY (mlaa1 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +

```

```

SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language))) > 1))) > 0))) = 0;
END_RULE;

```

```

RULE restrict_multi_language_for_general_property_relationship FOR
  (general_property_relationship);

```

```

  WHERE

```

```

    WR1 :

```

```

      SIZEOF(QUERY (ent <* general_property_relationship|
(SIZEOF(QUERY (mlaa <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description')) > 0))) = 0;

```

```

    WR2 :

```

```

      SIZEOF(QUERY (ent <* general_property_relationship|
(SIZEOF(QUERY (ala <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (ala.attribute_name <>
'description')) > 0))) = 0;

```

```

    WR3 :

```

```

      SIZEOF(QUERY (ent <* general_property_relationship|
(SIZEOF(QUERY (mlaa1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language))) > 1))) > 0))) = 0;
END_RULE;

```

```

RULE restrict_multi_language_for_group FOR

```

```

  (group);

```

```

  WHERE

```

```

    WR1 :

```

```

      SIZEOF(QUERY (ent <* group| (SIZEOF(QUERY (mlaa <* USEDIN(ent,
'CUTTING_TOOL_SCHEMA.' + 'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' +
'ITEMS')| (mlaa.attribute_name <> 'description') AND
(mlaa.attribute_name <> 'name')) > 0))) = 0;

```

```

    WR2 :

```

```

      SIZEOF(QUERY (ent <* group| (SIZEOF(QUERY (ala <* USEDIN(ent,
'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')|
(ala.attribute_name <> 'description') AND (ala.attribute_name <>
'name')) > 0))) = 0;

```

```

    WR3 :

```

```

      SIZEOF(QUERY (ent <* group| (SIZEOF(QUERY (mlaa1 <* USEDIN(ent,
'CUTTING_TOOL_SCHEMA.' + 'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' +
'ITEMS')| (SIZEOF(QUERY (mlaa2 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language))) > 1))) > 0))) = 0;

```

END\_RULE;

```

RULE restrict_multi_language_for_identification_role FOR
  (identification_role);
  WHERE
    WR1 :
      SIZEOF(QUERY (ent <* identification_role| (SIZEOF(QUERY (mlaa <*
        USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
        'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
        <> 'description')))) > 0))) = 0;
    WR2 :
      SIZEOF(QUERY (ent <* identification_role| (SIZEOF(QUERY (ala <*
        USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
        'ITEMS')| (ala.attribute_name <> 'description')))) > 0))) = 0;
    WR3 :
      SIZEOF(QUERY (ent <* identification_role| (SIZEOF(QUERY (mlaa1 <*
        USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
        'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
        <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
        'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
        = mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
        SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
        'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
        USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
        'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
        = ala2.language))) > 1))) > 0))) = 0;
  END_RULE;
```

```

RULE restrict_multi_language_for_organization_role FOR
  (organization_role);
  WHERE
    WR1 :
      SIZEOF(QUERY (ent <* organization_role| (SIZEOF(QUERY (mlaa <*
        USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
        'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
        <> 'description')))) > 0))) = 0;
    WR2 :
      SIZEOF(QUERY (ent <* organization_role| (SIZEOF(QUERY (ala <*
        USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
        'ITEMS')| (ala.attribute_name <> 'description')))) > 0))) = 0;
    WR3 :
      SIZEOF(QUERY (ent <* organization_role| (SIZEOF(QUERY (mlaa1 <*
        USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
        'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
        <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
        'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
        = mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
        SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
        'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
        USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
        'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
        = ala2.language))) > 1))) > 0))) = 0;
  END_RULE;
```

```

RULE restrict_multi_language_for_person_and_organization_role FOR
  (person_and_organization_role);
  WHERE
    WR1 :
```

```

        SIZEOF(QUERY (ent <* person_and_organization_role|
(SIZEOF(QUERY (mlaa <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description')) > 0))) = 0;
    WR2 :
        SIZEOF(QUERY (ent <* person_and_organization_role|
(SIZEOF(QUERY (ala <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (ala.attribute_name <>
'description')) > 0))) = 0;
    WR3 :
        SIZEOF(QUERY (ent <* person_and_organization_role|
(SIZEOF(QUERY (mlaa1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language))) > 1))) > 0))) = 0;
END_RULE;

RULE restrict_multi_language_for_product FOR
    (product);
    WHERE
        WR1 :
            SIZEOF(QUERY (ent <* product| (SIZEOF(QUERY (mlaa <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description') AND (mlaa.attribute_name <> 'name')) > 0))) = 0;
        WR2 :
            SIZEOF(QUERY (ent <* product| (SIZEOF(QUERY (ala <* USEDIN(ent,
'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')|
(ala.attribute_name <> 'description') AND (ala.attribute_name <>
'name')) > 0))) = 0;
        WR3 :
            SIZEOF(QUERY (ent <* product| (SIZEOF(QUERY (mlaa1 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language))) > 1))) > 0))) = 0;
END_RULE;

RULE restrict_multi_language_for_product_definition FOR
    (product_definition);
    WHERE
        WR1 :
            SIZEOF(QUERY (ent <* product_definition| (SIZEOF(QUERY (mlaa <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description') AND (mlaa.attribute_name <> 'name')) > 0))) = 0;

```

```

WR2 :
    SIZEOF(QUERY (ent <* product_definition| (SIZEOF(QUERY (ala <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala.attribute_name <> 'description') AND (ala.attribute_name
<> 'name')) > 0))) = 0;
WR3 :
    SIZEOF(QUERY (ent <* product_definition| (SIZEOF(QUERY (mlaa1
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language))) > 1))) > 0))) = 0;
END_RULE;

```

```

RULE restrict_multi_language_for_product_definition_formation FOR
    (product_definition_formation);

```

```

WHERE
    WR1 :
        SIZEOF(QUERY (ent <* product_definition_formation|
(SIZEOF(QUERY (mlaa <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description')) > 0))) = 0;
    WR2 :
        SIZEOF(QUERY (ent <* product_definition_formation|
(SIZEOF(QUERY (ala <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (ala.attribute_name <>
'description')) > 0))) = 0;
    WR3 :
        SIZEOF(QUERY (ent <* product_definition_formation|
(SIZEOF(QUERY (mlaa1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language))) > 1))) > 0))) = 0;
END_RULE;

```

```

RULE
restrict_multi_language_for_product_definition_formation_relationship
FOR
    (product_definition_formation_relationship);

```

```

WHERE
    WR1 :
        SIZEOF(QUERY (ent <* product_definition_formation_relationship|
(SIZEOF(QUERY (mlaa <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description')) > 0))) = 0;
    WR2 :

```

```

        SIZEOF(QUERY (ent <* product_definition_formation_relationship|
(SIZEOF(QUERY (ala <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (ala.attribute_name <>
'description')))) > 0))) = 0;
        WR3 :
        SIZEOF(QUERY (ent <* product_definition_formation_relationship|
(SIZEOF(QUERY (mlaa1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language))) > 1))) > 0))) = 0;
END_RULE;

```

```

RULE restrict_multi_language_for_product_definition_relationship FOR
(product_definition_relationship);

```

```

WHERE

```

```

    WR1 :

```

```

        SIZEOF(QUERY (ent <* product_definition_relationship|
(SIZEOF(QUERY (mlaa <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description')))) > 0))) = 0;

```

```

    WR2 :

```

```

        SIZEOF(QUERY (ent <* product_definition_relationship|
(SIZEOF(QUERY (ala <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (ala.attribute_name <>
'description')))) > 0))) = 0;

```

```

    WR3 :

```

```

        SIZEOF(QUERY (ent <* product_definition_relationship|
(SIZEOF(QUERY (mlaa1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language))) > 1))) > 0))) = 0;

```

```

END_RULE;

```

```

RULE restrict_multi_language_for_product_related_product_category FOR
(product_related_product_category);

```

```

WHERE

```

```

    WR1 :

```

```

        SIZEOF(QUERY (ent <* product_related_product_category|
(SIZEOF(QUERY (mlaa <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description')))) > 0))) = 0;

```

```

    WR2 :

```

```

        SIZEOF(QUERY (ent <* product_related_product_category|
(SIZEOF(QUERY (ala <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (ala.attribute_name <>
'description')))) > 0))) = 0;

```

```

WR3 :
    SIZEOF(QUERY (ent <* product_related_product_category|
(SIZEOF(QUERY (mlaa1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language))) > 1))) > 0))) = 0;
END_RULE;

```

```

RULE restrict_multi_language_for_property_definition FOR
(property_definition);

```

```

WHERE
WR1 :
    SIZEOF(QUERY (ent <* property_definition| (SIZEOF(QUERY (mlaa
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description') AND (mlaa.attribute_name <> 'name')))) > 0))) = 0;
WR2 :
    SIZEOF(QUERY (ent <* property_definition| (SIZEOF(QUERY (ala <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala.attribute_name <> 'description') AND (ala.attribute_name
<> 'name')))) > 0))) = 0;
WR3 :
    SIZEOF(QUERY (ent <* property_definition| (SIZEOF(QUERY (mlaa1
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language))) > 1))) > 0))) = 0;
END_RULE;

```

```

RULE restrict_multi_language_for_representation FOR
(representation);

```

```

WHERE
WR1 :
    SIZEOF(QUERY (ent <* representation| (SIZEOF(QUERY (mlaa <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description') AND (mlaa.attribute_name <> 'name')))) > 0))) = 0;
WR2 :
    SIZEOF(QUERY (ent <* representation| (SIZEOF(QUERY (ala <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala.attribute_name <> 'description') AND (ala.attribute_name
<> 'name')))) > 0))) = 0;
WR3 :
    SIZEOF(QUERY (ent <* representation| (SIZEOF(QUERY (mlaa1 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2

```

```

<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language))) > 1))) > 0))) = 0;
END_RULE;

```

```

RULE restrict_multi_language_for_representation_relationship FOR
  (representation_relationship);
  WHERE
    WR1 :
      SIZEOF(QUERY (ent <* representation_relationship| (SIZEOF(QUERY
(mlaa <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa.attribute_name
<> 'description')))) > 0))) = 0;
    WR2 :
      SIZEOF(QUERY (ent <* representation_relationship| (SIZEOF(QUERY
(ala <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (ala.attribute_name <>
'description')))) > 0))) = 0;
    WR3 :
      SIZEOF(QUERY (ent <* representation_relationship| (SIZEOF(QUERY
(mlaa1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (mlaa2
<* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'MULTI_LANGUAGE_ATTRIBUTE_ASSIGNMENT.' + 'ITEMS')| (mlaa1.attribute_name
= mlaa2.attribute_name) AND (mlaa1.language = mlaa2.language))) > 1))) +
SIZEOF(QUERY (ala1 <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' + 'ITEMS')| (SIZEOF(QUERY (ala2 <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' + 'ATTRIBUTE_LANGUAGE_ASSIGNMENT.' +
'ITEMS')| (ala1.attribute_name = ala2.attribute_name) AND (ala1.language
= ala2.language))) > 1))) > 0))) = 0;
END_RULE;

```

```

RULE restrict_name_for_known_source FOR
  (known_source);
  WHERE
    WR1 :
      SIZEOF(QUERY (ks <* known_source| (ks.name <> 'ISO 13584
library')))) = 0;
END_RULE;

```

```

RULE restrict_part_occurrence FOR
  (product_definition,
  product_definition_occurrence_relationship);
  WHERE
    WR1 :
      SIZEOF(QUERY (pd <* product_definition|
(pd.frame_of_reference.name = 'part occurrence') AND NOT (pd.name IN [
'single instance', 'quantified instance' ]))) = 0;
    WR2 :
      SIZEOF(QUERY (pd <* product_definition|
((pd.frame_of_reference.name = 'part occurrence') AND (SIZEOF(QUERY (cd
<* USEDIN(pd, 'CUTTING_TOOL_SCHEMA.' +
'PRODUCT_DEFINITION_RELATIONSHIP.' + 'RELATED_PRODUCT_DEFINITION')|

```

```
( 'CUTTING_TOOL_SCHEMA.' + 'PRODUCT_DEFINITION_USAGE' IN TYPEOF(cd))) =
0)) AND (SIZEOF(USEDIN(pd, 'CUTTING_TOOL_SCHEMA.' +
'PRODUCT_DEFINITION_OCCURRENCE_RELATIONSHIP.' + 'OCCURRENCE')) = 0)) =
0;
```

```
WR3 :
    SIZEOF(QUERY (pd <* product_definition|
((pd.frame_of_reference.name = 'part occurrence') AND (pd.name =
'quantified instance')) AND (SIZEOF(QUERY (ppd <* USEDIN(pd,
'CUTTING_TOOL_SCHEMA.' + 'PROPERTY_DEFINITION.' + 'DEFINITION')|
(ppd.name = 'occurrence quantity') AND (SIZEOF(QUERY (pdr <* USEDIN(ppd,
'CUTTING_TOOL_SCHEMA.' + 'PROPERTY_DEFINITION_REPRESENTATION.' +
'DEFINITION')| ((pdr.used_representation.name = 'quantity') AND
(SIZEOF(pdr.used_representation.items) = 1)) AND (SIZEOF(QUERY (i <*
pdr.used_representation.items| ('CUTTING_TOOL_SCHEMA.' +
'MEASURE_REPRESENTATION_ITEM' IN TYPEOF(i)) AND (i.name = 'quantity
measure')))) = 1))) = 1))) = 0))) = 0;
END_RULE;
```

RULE restrict\_product\_category\_for\_product FOR

```
(product);
WHERE
WR1 :
    SIZEOF(QUERY (p <* product| (SIZEOF(QUERY (prpc <* USEDIN(p,
'CUTTING_TOOL_SCHEMA.' + 'PRODUCT_RELATED_PRODUCT_CATEGORY.' +
'PRODUCTS')| (prpc.name IN [ 'cutting item', 'tool item', 'adaptive
item', 'accessory item', 'assembly item', 'document', 'grade', 'coating',
'physically realized product', 'substrate ])))) = 0))) = 0;
WR2 :
    SIZEOF(QUERY (p <* product| (SIZEOF(QUERY (prpc <* USEDIN(p,
'CUTTING_TOOL_SCHEMA.' + 'PRODUCT_RELATED_PRODUCT_CATEGORY.' +
'PRODUCTS')| (prpc.name IN [ 'cutting item', 'tool item', 'adaptive
item', 'accessory item', 'assembly item', 'document', 'grade', 'coating',
'substrate ])))) > 1))) = 0;
WR3 :
    SIZEOF(QUERY (p <* product| (SIZEOF(QUERY (prpc <* USEDIN(p,
'CUTTING_TOOL_SCHEMA.' + 'PRODUCT_RELATED_PRODUCT_CATEGORY.' +
'PRODUCTS')| (prpc.name IN [ 'physically realized product' ])))) > 1))) =
0;
END_RULE;
```

RULE restrict\_product\_definition\_context\_for\_applied\_location\_assignment FOR

```
(applied_location_assignment);
WHERE
WR1 :
    SIZEOF(QUERY (ala <* applied_location_assignment|
(ala.items.frame_of_reference.name <> 'physical occurrence')) = 0;
END_RULE;
```

RULE restrict\_product\_definition\_context\_for\_applied\_state\_type\_assignment FOR

```
(applied_state_type_assignment);
WHERE
WR1 :
    SIZEOF(QUERY (pd <* product_definition|
(pd.frame_of_reference.name <> 'physical occurrence') AND USEDIN(pd,
```

```

('CUTTING_TOOL_SCHEMA.' + 'APPLIED_STATE_TYPE_ASSIGNMENT.' +
'ITEM_SET')))) = 0;
END_RULE;

RULE restrict_product_definition_context_for_product FOR
  (product_definition,
   product_related_product_category);
WHERE
  WR1 :
    SIZEOF(QUERY (pd <* product_definition|
(pd.frame_of_reference.name IN [ 'part definition', 'part occurrence' ])
AND (SIZEOF(QUERY (prpc <* USEDIN(pd.formation.of_product,
'CUTTING_TOOL_SCHEMA.PRODUCT_RELATED_PRODUCT_CATEGORY.PRODUCTS')|
(prpc.name IN [ 'part', 'raw material', 'tool', 'cutting item', 'tool
item', 'adaptive item', 'accessory item', 'assembly item' ]))) = 0))) =
0;
  WR2 :
    SIZEOF(QUERY (pd <* product_definition|
(pd.frame_of_reference.name = 'physical occurrence') AND (SIZEOF(QUERY
(prpc <* USEDIN(pd.formation.of_product, 'CUTTING_TOOL_SCHEMA.' +
'PRODUCT_RELATED_PRODUCT_CATEGORY.' + 'PRODUCTS')| (prpc.name =
'physically realized product')))) = 0))) = 0;
  WR3 :
    SIZEOF(QUERY (pd <* product_definition|
(pd.frame_of_reference.name IN [ 'physical document definition',
'digital document definition' ]) AND (SIZEOF(QUERY (prpc <*
USEDIN(pd.formation.of_product, 'CUTTING_TOOL_SCHEMA.' +
'PRODUCT_RELATED_PRODUCT_CATEGORY.' + 'PRODUCTS')| (prpc.name =
'document')))) = 0))) = 0;
  WR4 :
    SIZEOF(QUERY (pd <* product_definition|
(pd.frame_of_reference.name = 'material definition') AND (SIZEOF(QUERY
(prpc <* USEDIN(pd.formation.of_product, 'CUTTING_TOOL_SCHEMA.' +
'PRODUCT_RELATED_PRODUCT_CATEGORY.' + 'PRODUCTS')| (prpc.name IN [
'grade', 'substrate', 'coating' ]))) = 0))) = 0;
END_RULE;

RULE restrict_product_definitions_for_product_definition_relationship FOR
  (assembly_component_usage,
   product_definition_relationship);
WHERE
  WR1 :
    SIZEOF(QUERY (pdr <* product_definition_relationship|
(('CUTTING_TOOL_SCHEMA.' + 'ASSEMBLY_COMPONENT_USAGE' IN TYPEOF(pdr))
AND ((pdr.relatng_product_definition.frame_of_reference.name <> 'part
definition') OR (pdr.related_product_definition.frame_of_reference.name
<> 'part definition')) AND
((pdr.relatng_product_definition.frame_of_reference.name <> 'physical
occurrence') OR (pdr.related_product_definition.frame_of_reference.name
<> 'physical occurrence')))) = 0;
  WR2 :
    SIZEOF(QUERY (pdr <* product_definition_relationship| (pdr.name
= 'physical realization') AND
((pdr.relatng_product_definition.frame_of_reference.name <> 'part
definition') OR (pdr.related_product_definition.frame_of_reference.name
<> 'physical occurrence')))) = 0;
  WR3 :

```

```

        SIZEOF(QUERY (pdr <* product_definition_relationship| (pdr.name
= 'physical occurrence usage') AND ((NOT ('CUTTING_TOOL_SCHEMA.' +
'ASSEMBLY_COMPONENT_USAGE' IN TYPEOF(pdr)) OR
(pdr.relating_product_definition.frame_of_reference.name <> 'physical
occurrence')) OR (pdr.related_product_definition.frame_of_reference.name
<> 'physical occurrence')))) = 0;

```

```

    WR4 :

```

```

        SIZEOF(QUERY (pdr <* product_definition_relationship| (pdr.name
= 'substrate') AND
((pdr.relating_product_definition.frame_of_reference.name <> 'material
definition') OR (pdr.related_product_definition.frame_of_reference.name
<> 'material definition')))) = 0;

```

```

    WR5 :

```

```

        SIZEOF(QUERY (pdr <* product_definition_relationship| (pdr.name
= 'coating') AND
((pdr.relating_product_definition.frame_of_reference.name <> 'material
definition') OR (pdr.related_product_definition.frame_of_reference.name
<> 'material definition')))) = 0;

```

```

END_RULE;

```

```

RULE restrict_properties_of_document_file FOR

```

```

    (document_file);

```

```

    WHERE

```

```

        WR1 :

```

```

            SIZEOF(QUERY (df <* document_file| (SIZEOF(QUERY (pd <*
USEDIN(df, 'CUTTING_TOOL_SCHEMA.' + 'PROPERTY_DEFINITION.' +
'DEFINITION')| (pd.name = 'document property')) > 1))) > 1))) = 0;

```

```

        WR2 :

```

```

            SIZEOF(QUERY (df <* document_file| (SIZEOF(QUERY (rt <*
df.representation_types| ('CUTTING_TOOL_SCHEMA.' +
'DOCUMENT_REPRESENTATION_TYPE' IN TYPEOF(rt)) AND (rt.name IN [
'digital', 'physical' ]))) = 0))) = 0;

```

```

END_RULE;

```

```

RULE restrict_properties_of_document_representation FOR

```

```

    (product_definition);

```

```

    WHERE

```

```

        WR1 :

```

```

            SIZEOF(QUERY (pd <* product_definition|
(pd.frame_of_reference.name IN [ 'physical document definition',
'digital document definition', 'physical model occurrence' ]) AND
(SIZEOF(QUERY (pd1 <* USEDIN(pd, 'CUTTING_TOOL_SCHEMA.' +
'PROPERTY_DEFINITION.' + 'DEFINITION')| (pd1.name = 'document
property')) > 1))) = 0;

```

```

END_RULE;

```

```

RULE restrict_representation_for_coupling FOR

```

```

    (representation,
    representation_item);

```

```

    WHERE

```

```

        WR1 :

```

```

            SIZEOF(QUERY (r <* representation| (r.name = 'coupling') AND
((SIZEOF(r.items) < 4) OR (SIZEOF(r.items) > 5))) = 0;

```

```

        WR2 :

```

```

            SIZEOF(QUERY (ri <* representation_item| (SIZEOF(QUERY (r <*
USEDIN(ri, 'CUTTING_TOOL_SCHEMA.REPRESENTATION.ITEMS')| (r.name =
'coupling')) > 0) AND NOT (ri.name IN [ 'coupling type', 'pieces',
'side', 'size', 'style' ]))) = 0;

```

```

WR3 :
    SIZEOF(QUERY (r <* representation| (r.name = 'coupling') AND
(SIZEOF(QUERY (i <* r.items| (i.name = 'coupling type') AND
('CUTTING_TOOL_SCHEMA.' + 'DESCRIPTIVE_REPRESENTATION_ITEM' IN
TYPEOF(i)))) > 1))) = 0;
WR4 :
    SIZEOF(QUERY (r <* representation| (r.name = 'coupling') AND
(SIZEOF(QUERY (i <* r.items| (i.name = 'pieces') AND (SIZEOF([
('CUTTING_TOOL_SCHEMA.' + 'MEASURE_REPRESENTATION_ITEM'),
('CUTTING_TOOL_SCHEMA.' + 'VALUE_REPRESENTATION_ITEM') ] * TYPEOF(i)) =
1))) > 1))) = 0;
WR5 :
    SIZEOF(QUERY (r <* representation| (r.name = 'coupling') AND
(SIZEOF(QUERY (i <* r.items| (i.name = 'side') AND
('CUTTING_TOOL_SCHEMA.' + 'DESCRIPTIVE_REPRESENTATION_ITEM' IN
TYPEOF(i)))) > 1))) = 0;
WR6 :
    SIZEOF(QUERY (r <* representation| (r.name = 'coupling') AND
(SIZEOF(QUERY (i <* r.items| (i.name = 'size') AND (SIZEOF([
('CUTTING_TOOL_SCHEMA.' + 'MEASURE_REPRESENTATION_ITEM'),
('CUTTING_TOOL_SCHEMA.' + 'VALUE_REPRESENTATION_ITEM') ] * TYPEOF(i)) =
1))) > 1))) = 0;
WR7 :
    SIZEOF(QUERY (r <* representation| (r.name = 'coupling') AND
(SIZEOF(QUERY (i <* r.items| (i.name = 'style') AND
('CUTTING_TOOL_SCHEMA.' + 'DESCRIPTIVE_REPRESENTATION_ITEM' IN
TYPEOF(i)))) > 1))) = 0;
END_RULE;

```

```

RULE restrict_representation_for_cutting_condition FOR
    (representation,
    representation_item);
WHERE
WR1 :
    SIZEOF(QUERY (r <* representation| (r.name = 'cutting
condition') AND (SIZEOF(r.items) = 1))) = 0;
WR2 :
    SIZEOF(QUERY (ri <* representation_item| (SIZEOF(QUERY (r <*
USEDIN(ri, 'CUTTING_TOOL_SCHEMA.REPRESENTATION.ITEMS')| (r.name =
'cutting condition')) > 0) AND NOT (ri.name IN [ 'condition name' ])))
= 0;
WR3 :
    SIZEOF(QUERY (r <* representation| (r.name = 'cutting
condition') AND (SIZEOF(QUERY (i <* r.items| (i.name = 'condition name')
AND ('CUTTING_TOOL_SCHEMA.' + 'DESCRIPTIVE_REPRESENTATION_ITEM' IN
TYPEOF(i)))) > 1))) = 0;
END_RULE;

```

```

RULE restrict_representation_for_cutting_data_association FOR
    (representation,
    representation_item);
WHERE
WR1 :
    SIZEOF(QUERY (r <* representation| (r.name = 'cutting data
association') AND (SIZEOF(r.items) = 3))) = 0;
WR2 :
    SIZEOF(QUERY (ri <* representation_item| (SIZEOF(QUERY (r <*
USEDIN(ri, 'CUTTING_TOOL_SCHEMA.REPRESENTATION.ITEMS')| (r.name =

```

```
'cutting data association')) > 0) AND NOT (ri.name IN [ 'depth of cut',
'feed', 'speed' ])) = 0;
```

```
WR3 :
```

```
SIZEOF(QUERY (r <* representation| (r.name = 'cutting data
association') AND (SIZEOF(QUERY (i <* r.items| (i.name = 'depth of cut')
AND (SIZEOF([ ('CUTTING_TOOL_SCHEMA.' + 'MEASURE_REPRESENTATION_ITEM'),
('CUTTING_TOOL_SCHEMA.' + 'VALUE_RANGE'), ('CUTTING_TOOL_SCHEMA.' +
'VALUE_REPRESENTATION_ITEM') ] * TYPEOF(i)) = 1))) > 1))) = 0;
```

```
WR4 :
```

```
SIZEOF(QUERY (r <* representation| (r.name = 'cutting data
association') AND (SIZEOF(QUERY (i <* r.items| (i.name = 'feed') AND
(SIZEOF([ ('CUTTING_TOOL_SCHEMA.' + 'MEASURE_REPRESENTATION_ITEM'),
('CUTTING_TOOL_SCHEMA.' + 'VALUE_RANGE'), ('CUTTING_TOOL_SCHEMA.' +
'VALUE_REPRESENTATION_ITEM') ] * TYPEOF(i)) = 1))) > 1))) = 0;
```

```
WR5 :
```

```
SIZEOF(QUERY (r <* representation| (r.name = 'cutting data
association') AND (SIZEOF(QUERY (i <* r.items| (i.name = 'speed') AND
(SIZEOF([ ('CUTTING_TOOL_SCHEMA.' + 'MEASURE_REPRESENTATION_ITEM'),
('CUTTING_TOOL_SCHEMA.' + 'VALUE_RANGE'), ('CUTTING_TOOL_SCHEMA.' +
'VALUE_REPRESENTATION_ITEM') ] * TYPEOF(i)) = 1))) > 1))) = 0;
```

```
END_RULE;
```

```
RULE restrict_representation_for_document_content property FOR
```

```
(representation,
representation_item);
```

```
WHERE
```

```
WR1 :
```

```
SIZEOF(QUERY (r <* representation| (r.name = 'document
content') AND ((SIZEOF(r.items) < 1) OR (SIZEOF(r.items) > 3)))) = 0;
```

```
WR2 :
```

```
SIZEOF(QUERY (ri <* representation_item| (SIZEOF(QUERY (r <*
USEDIN(ri, 'CUTTING_TOOL_SCHEMA.REPRESENTATION.ITEMS')| (r.name =
'document content')) > 0) AND NOT (ri.name IN [ 'detail level',
'geometry type', 'real world scale' ])))) = 0;
```

```
WR3 :
```

```
SIZEOF(QUERY (r <* representation| (r.name = 'document
content') AND (SIZEOF(QUERY (i <* r.items| (i.name = 'detail level') AND
('CUTTING_TOOL_SCHEMA.' + 'DESCRIPTIVE_REPRESENTATION_ITEM' IN
TYPEOF(i)))) > 1))) = 0;
```

```
WR4 :
```

```
SIZEOF(QUERY (r <* representation| (r.name = 'document
content') AND (SIZEOF(QUERY (i <* r.items| (i.name = 'geometry type')
AND ('CUTTING_TOOL_SCHEMA.' + 'DESCRIPTIVE_REPRESENTATION_ITEM' IN
TYPEOF(i)))) > 1))) = 0;
```

```
WR5 :
```

```
SIZEOF(QUERY (r <* representation| (r.name = 'document
content') AND (SIZEOF(QUERY (i <* r.items| (i.name = 'real world scale')
AND (SIZEOF([ ('CUTTING_TOOL_SCHEMA.' + 'MEASURE_REPRESENTATION_ITEM'),
('CUTTING_TOOL_SCHEMA.' + 'VALUE_REPRESENTATION_ITEM') ] * TYPEOF(i)) =
1))) > 1))) = 0;
```

```
END_RULE;
```

```
RULE restrict_representation_for_document_creation_property FOR
```

```
(representation,
representation_item);
```

```
WHERE
```

```
WR1 :
```

```

        SIZEOF(QUERY (r <* representation| (r.name = 'document
creation') AND ((SIZEOF(r.items) < 2) OR (SIZEOF(r.items) > 3)))) = 0;
    WR2 :
        SIZEOF(QUERY (ri <* representation_item| (SIZEOF(QUERY (r <*
USEDIN(ri, 'CUTTING_TOOL_SCHEMA.REPRESENTATION.ITEMS')| (r.name =
'document creation')) > 0) AND NOT (ri.name IN [ 'creating interface',
'creating system', 'operating system' ]))) = 0;
    WR3 :
        SIZEOF(QUERY (r <* representation| (r.name = 'document
creation') AND (SIZEOF(QUERY (i <* r.items| (i.name = 'creating
interface') AND ('CUTTING_TOOL_SCHEMA.' +
'DESRIPTIVE_REPRESENTATION_ITEM' IN TYPEOF(i)))) > 1))) = 0;
    WR4 :
        SIZEOF(QUERY (r <* representation| (r.name = 'document
creation') AND (SIZEOF(QUERY (i <* r.items| (i.name = 'creating system')
AND ('CUTTING_TOOL_SCHEMA.' + 'DESCRIPTIVE_REPRESENTATION_ITEM' IN
TYPEOF(i)))) <> 1))) = 0;
    WR5 :
        SIZEOF(QUERY (r <* representation| (r.name = 'document
creation') AND (SIZEOF(QUERY (i <* r.items| (i.name = 'operating
system') AND ('CUTTING_TOOL_SCHEMA.' + 'DESCRIPTIVE_REPRESENTATION_ITEM'
IN TYPEOF(i)))) > 1))) = 0;
END_RULE;

RULE restrict_representation_for_document_format_property FOR
(representation,
representation_item);
WHERE
    WR1 :
        SIZEOF(QUERY (r <* representation| (r.name = 'document format')
AND ((SIZEOF(r.items) < 1) OR (SIZEOF(r.items) > 3)))) = 0;
    WR2 :
        SIZEOF(QUERY (ri <* representation_item| (SIZEOF(QUERY (r <*
USEDIN(ri, 'CUTTING_TOOL_SCHEMA.REPRESENTATION.ITEMS')| (r.name =
'document format')) > 0) AND NOT (ri.name IN [ 'character code', 'data
format' ]))) = 0;
    WR3 :
        SIZEOF(QUERY (r <* representation| (r.name = 'document format')
AND (SIZEOF(QUERY (i <* r.items| (i.name = 'character code') AND
('CUTTING_TOOL_SCHEMA.' + 'DESCRIPTIVE_REPRESENTATION_ITEM' IN
TYPEOF(i)))) > 1))) = 0;
    WR4 :
        SIZEOF(QUERY (r <* representation| (r.name = 'document format')
AND (SIZEOF(QUERY (i <* r.items| (i.name = 'data format') AND
('CUTTING_TOOL_SCHEMA.' + 'DESCRIPTIVE_REPRESENTATION_ITEM' IN
TYPEOF(i)))) > 1))) = 0;
    WR5 :
        SIZEOF(QUERY (r <* representation| (r.name = 'document format')
AND (SIZEOF(QUERY (i <* r.items| (i.name = 'size format') AND
('CUTTING_TOOL_SCHEMA.PLANAR_EXTENT' IN TYPEOF(i)))) > 1))) = 0;
END_RULE;

RULE restrict_representation_for_document_properties FOR
(representation,
property_definition,
representation_context);
WHERE
    WR1 :

```

```

        SIZEOF(QUERY (pd <* property_definition| (pd.name = 'document
property') AND (SIZEOF(QUERY (pdr <* USEDIN(pd, 'CUTTING_TOOL_SCHEMA.' +
'PROPERTY_DEFINITION_REPRESENTATION.' + 'DEFINITION')| NOT
(pdr.used_representation.name IN [ 'document content', 'document
creation', 'document format', 'document size' ]))) > 0))) = 0;
    WR2 :
        SIZEOF(QUERY (r <* representation| (r.name IN [ 'document
content', 'document creation', 'document format', 'document size' ]) AND
(SIZEOF(QUERY (pdr <* USEDIN(r, 'CUTTING_TOOL_SCHEMA.' +
'PROPERTY_DEFINITION_REPRESENTATION.' + 'USED_REPRESENTATION')|
('CUTTING_TOOL_SCHEMA.' + 'PROPERTY_DEFINITION' IN
TYPEOF(pdr.definition)) AND (pdr.definition.name = 'document
property')))) = 0))) = 0;
    WR3 :
        SIZEOF(QUERY (r <* representation| (r.name IN [ 'document
content', 'document creation', 'document format', 'document size' ]) AND
(r.context_of_items.context_type <> 'document parameters')))) = 0;
    WR4 :
        SIZEOF(QUERY (rc <* representation_context| (rc.context_type =
'document parameters') AND (SIZEOF(QUERY (r <* USEDIN(rc,
'CUTTING_TOOL_SCHEMA.REPRESENTATION.CONTEXT_OF_ITEMS')| NOT (r.name IN [
'document content', 'document creation', 'document format', 'document
size' ]))) > 0))) = 0;
END_RULE;

```

```

RULE restrict_representation_for_document_size_property FOR
    (representation,
    representation_item);
WHERE
    WR1 :
        SIZEOF(QUERY (r <* representation| (r.name = 'document size')
AND ((SIZEOF(r.items) < 1) OR (SIZEOF(r.items) > 2)))) = 0;
    WR2 :
        SIZEOF(QUERY (ri <* representation_item| (SIZEOF(QUERY (r <*
USEDIN(ri, 'CUTTING_TOOL_SCHEMA.REPRESENTATION.ITEMS')| (r.name =
'document size')) > 0) AND NOT (ri.name IN [ 'file size', 'page count'
]))) = 0;
    WR3 :
        SIZEOF(QUERY (r <* representation| (r.name = 'document size')
AND (SIZEOF(QUERY (i <* r.items| (i.name = 'file size') AND (SIZEOF([
('CUTTING_TOOL_SCHEMA.' + 'MEASURE_REPRESENTATION_ITEM'),
('CUTTING_TOOL_SCHEMA.' + 'VALUE_RANGE'), ('CUTTING_TOOL_SCHEMA.' +
'VALUE_REPRESENTATION_ITEM') ] * TYPEOF(i)) = 1))) > 1))) = 0;
    WR4 :
        SIZEOF(QUERY (r <* representation| (r.name = 'document size')
AND (SIZEOF(QUERY (i <* r.items| (i.name = 'page count') AND (SIZEOF([
('CUTTING_TOOL_SCHEMA.' + 'MEASURE_REPRESENTATION_ITEM'),
('CUTTING_TOOL_SCHEMA.' + 'VALUE_RANGE'), ('CUTTING_TOOL_SCHEMA.' +
'VALUE_REPRESENTATION_ITEM') ] * TYPEOF(i)) = 1))) > 1))) = 0;
END_RULE;

```

```

RULE restrict_version_assignment_for_applied_identification_assignment
FOR
    (applied_identification_assignment);
WHERE
    WR1 :
        SIZEOF(QUERY (ent <* applied_identification_assignment|
(SIZEOF(QUERY (ia <* USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +

```

```

'APPLIED_IDENTIFICATION_ASSIGNMENT.' + 'ITEMS')| (ia.role.name =
'version')))) > 1))) = 0;
END_RULE;

RULE restrict_version_assignment_for_class FOR
(class);
WHERE
WR1 :
      SIZEOF(QUERY (ent <* class| (SIZEOF(QUERY (ia <* USEDIN(ent,
'CUTTING_TOOL_SCHEMA.' + 'APPLIED_IDENTIFICATION_ASSIGNMENT.' +
'ITEMS')| (ia.role.name = 'version')))) > 1))) = 0;
END_RULE;

RULE restrict_version_assignment_for_document_file FOR
(document_file);
WHERE
WR1 :
      SIZEOF(QUERY (ent <* document_file| (SIZEOF(QUERY (ia <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'APPLIED_IDENTIFICATION_ASSIGNMENT.' + 'ITEMS')| (ia.role.name =
'version')))) > 1))) = 0;
END_RULE;

RULE restrict_version_assignment_for_effectivity FOR
(effectivity);
WHERE
WR1 :
      SIZEOF(QUERY (ent <* effectivity| (SIZEOF(QUERY (ia <*
USEDIN(ent, 'CUTTING_TOOL_SCHEMA.' +
'APPLIED_IDENTIFICATION_ASSIGNMENT.' + 'ITEMS')| (ia.role.name =
'version')))) > 1))) = 0;
END_RULE;

RULE substrate_requires_product_definition FOR
(product_definition,
product_definition_formation,
product_related_product_category);
WHERE
WR1 :
      SIZEOF(QUERY (pdf <* product_definition_formation|
(SIZEOF(QUERY (prpc <* USEDIN(pdf.of_product, 'CUTTING_TOOL_SCHEMA.' +
'PRODUCT_RELATED_PRODUCT_CATEGORY.' + 'PRODUCTS')| (prpc.name =
'substrate')) > 0) AND (SIZEOF(QUERY (pd <* USEDIN(pdf,
'CUTTING_TOOL_SCHEMA.' + 'PRODUCT_DEFINITION.' + 'FORMATION')|
(pd.frame_of_reference.name = 'material_definition')) <> 1))) <> 1))) = 0;
END_RULE;

RULE subtype_exclusiveness_characterized_object FOR
(characterized_object);
WHERE
WR1 :
      SIZEOF(QUERY (co <* characterized_object| NOT
type_check_function(co, [ ('CUTTING_TOOL_SCHEMA.' +
'CHARACTERIZED_CLASS'), ('CUTTING_TOOL_SCHEMA.' + 'DOCUMENT_FILE'),
('CUTTING_TOOL_SCHEMA.' + 'FEATURE_COMPONENT_DEFINITION'),
('CUTTING_TOOL_SCHEMA.' + 'FEATURE_DEFINITION'), ('CUTTING_TOOL_SCHEMA.'
+ 'PRODUCT_CLASS'), ('CUTTING_TOOL_SCHEMA.' + 'PRODUCT_IDENTIFICATION')
], 3))) = 0;

```

```
END_RULE;
```

```
RULE subtype_exclusiveness_document_reference FOR
  (document_reference);
  WHERE
    WR1 :
      SIZEOF(QUERY (dr <* document_reference| NOT
type_check_function(dr, [ ('CUTTING_TOOL_SCHEMA.' +
'APPLIED_DOCUMENT_REFERENCE'), ('CUTTING_TOOL_SCHEMA.' +
'DRAUGHTING_SPECIFICATION_REFERENCE') ], 3))) = 0;
END_RULE;
```

```
RULE subtype_exclusiveness_feature_definition FOR
  (feature_definition);
  WHERE
    WR1 :
      SIZEOF(QUERY (fd <* feature_definition| NOT
type_check_function(fd, [ ('CUTTING_TOOL_SCHEMA.' + 'BARRING_HOLE'),
('CUTTING_TOOL_SCHEMA.' + 'BEAD'), ('CUTTING_TOOL_SCHEMA.' + 'BOSS'),
('CUTTING_TOOL_SCHEMA.' + 'COMPOUND_FEATURE'), ('CUTTING_TOOL_SCHEMA.' +
'EXTERNALLY_DEFINED_FEATURE_DEFINITION'), ('CUTTING_TOOL_SCHEMA.' +
'FEATURE_IN_PANEL'), ('CUTTING_TOOL_SCHEMA.' + 'GENERAL_FEATURE'),
('CUTTING_TOOL_SCHEMA.' + 'HOLE_IN_PANEL'), ('CUTTING_TOOL_SCHEMA.' +
'JOGGLE'), ('CUTTING_TOOL_SCHEMA.' + 'LOCATOR'), ('CUTTING_TOOL_SCHEMA.' +
'POCKET'), ('CUTTING_TOOL_SCHEMA.' + 'RIB'), ('CUTTING_TOOL_SCHEMA.' +
'ROUND_HOLE'), ('CUTTING_TOOL_SCHEMA.' + 'SLOT'),
('CUTTING_TOOL_SCHEMA.' + 'THREAD') ], 3))) = 0;
END_RULE;
```

```
RULE subtype_exclusiveness_pre_defined_item FOR
  (pre_defined_item);
  WHERE
    WR1 :
      SIZEOF(QUERY (pdi <* pre_defined_item| NOT
type_check_function(pdi, [ ('CUTTING_TOOL_SCHEMA.' + 'KNOWN_SOURCE'),
('CUTTING_TOOL_SCHEMA.' + 'PRE_DEFINED_COLOUR'), ('CUTTING_TOOL_SCHEMA.' +
'PRE_DEFINED_CURVE_FONT'), ('CUTTING_TOOL_SCHEMA.' +
'PRE_DEFINED_MARKER'), ('CUTTING_TOOL_SCHEMA.' +
'PRE_DEFINED_PRESENTATION_STYLE'), ('CUTTING_TOOL_SCHEMA.' +
'PRE_DEFINED_SYMBOL'), ('CUTTING_TOOL_SCHEMA.' +
'PRE_DEFINED_TEXT_FONT') ], 3))) = 0;
END_RULE;
```

```
RULE subtype_mandatory_pre_defined_item FOR
  (pre_defined_item);
  WHERE
    WR1 :
      SIZEOF(QUERY (pdi <* pre_defined_item| NOT
type_check_function(pdi, [ ('CUTTING_TOOL_SCHEMA.' + 'KNOWN_SOURCE'),
('CUTTING_TOOL_SCHEMA.' + 'PRE_DEFINED_COLOUR'), ('CUTTING_TOOL_SCHEMA.' +
'PRE_DEFINED_CURVE_FONT'), ('CUTTING_TOOL_SCHEMA.' +
'PRE_DEFINED_MARKER'), ('CUTTING_TOOL_SCHEMA.' +
'PRE_DEFINED_PRESENTATION_STYLE'), ('CUTTING_TOOL_SCHEMA.' +
'PRE_DEFINED_SYMBOL'), ('CUTTING_TOOL_SCHEMA.' +
'PRE_DEFINED_TEXT_FONT') ], 0))) = 0;
END_RULE;
```

```
FUNCTION acyclic
```

```

(* *****
Functions in the schema CUTTING_TOOL_SCHEMA
***** *) (arg1 : generic_expression;
          arg2 : SET OF generic_expression ) : BOOLEAN;
  LOCAL
    result : BOOLEAN;
  END_LOCAL;
  IF 'CUTTING_TOOL_SCHEMA.SIMPLE_GENERIC_EXPRESSION' IN TYPEOF(arg1)
THEN
  RETURN (TRUE);
END_IF;
IF arg1 IN arg2 THEN
  RETURN (FALSE);
END_IF;
IF 'CUTTING_TOOL_SCHEMA.UNARY_GENERIC_EXPRESSION' IN TYPEOF(arg1)
THEN
  RETURN (acyclic(arg1\unary_generic_expression.operand, arg2 + [
arg1 ]));
END_IF;
IF 'CUTTING_TOOL_SCHEMA.BINARY_GENERIC_EXPRESSION' IN TYPEOF(arg1)
THEN
  RETURN (acyclic(arg1\binary_generic_expression.operands[1],
(arg2 + [ arg1 ])) AND
acyclic(arg1\binary_generic_expression.operands[2], (arg2 + [ arg1 ])));
END_IF;
IF 'CUTTING_TOOL_SCHEMA.MULTIPLE_ARITY_GENERIC_EXPRESSION' IN
TYPEOF(arg1) THEN
  result := TRUE;
  REPEAT i := 1 TO
SIZEOF(arg1\multiple_arity_generic_expression.operands);
    result := result AND
acyclic(arg1\multiple_arity_generic_expression.operands[i], (arg2 + [
arg1 ]));
  END_REPEAT;
  RETURN (result);
END_IF;
END_FUNCTION;

FUNCTION acyclic_mapped_representation
(parent_set : SET OF representation;
 children_set : SET OF representation_item ) : BOOLEAN;
  LOCAL
    x : SET OF representation_item;
    y : SET OF representation_item;
  END_LOCAL;
  x := QUERY (z <* children_set | 'CUTTING_TOOL_SCHEMA.MAPPED_ITEM' IN
TYPEOF(z));
  IF SIZEOF(x) > 0 THEN
    REPEAT i := 1 TO HIINDEX(x);
      IF x[i]\mapped_item.mapping_source.mapped_representation IN
parent_set THEN
        RETURN (FALSE);
      END_IF;
      IF NOT acyclic_mapped_representation((parent_set +
x[i]\mapped_item.mapping_source.mapped_representation),
x[i]\mapped_item.mapping_source.mapped_representation.items) THEN
        RETURN (FALSE);
      END_IF;
    END_REPEAT;
  END_IF;
END_FUNCTION;

```