ISO TC 184/SC4/WG 3 N 988

Date: 2000-12-01

Supersedes ISO TC 184/SC4/WG 3 N 854

ISO/IS 10303-224.2
Product data representation and exchange: Application protocol: Mechanical product definition for process planning using machining features

COPYRIGHT NOTICE:

This document is an International Standard and is copyright-protected by ISO. Except as permitted under the applicable laws of the user's country, neither this ISO draft nor any extract from it may be reproduced, stored in a retrieval system or transmitted in any form or by any means, electronic, photocopying, recording, or otherwise, without prior written permission being secured.

Requests for permission to reproduce should be addressed to ISO at the address below or ISO's member body in the country of the requester:

Copyright Manager
ISO Central Secretariat
1 rue de Varembe
1211 Geneva 20 Switzerland
telephone: +41 22 749 0111
telefacsimile: +41 22 734 0179
Internet: central@isocs.iso.ch
X.400: c=ch; a=400net; p=iso; o=isocs; s=central

Reproduction for sales purposes for any of the above-mentioned documents may be subject to royalty payments or a licensing agreement. Violators may be prosecuted.

ABSTRACT:

This part of the ISO 10303 documents the application protocol for the representation and exchange of information for manufacturing single piece mechanical parts. This part includes explicit and implicit shape representation of machining features. It addresses manufacturing part properties, process control documentation, manufacturing specifications, administration data, and requisitions.

KEYWORDS:

Application Protocol, machining features, process planning

COMMENTS TO READER:

This document has been reviewed using the internal review checklist (see wg3m984), the project leader checklist (see wg3n985) and the convener checklist (see wg3n986), and has been determined to be ready for publication.

This is the second edition of ISO 10303-224. The content of this document incorporates the resolutions approved by WG12 to SEDS issues 417, 418, 419, 420 raised against the 1999 edition and new requirements contained in the NWI proposal document (SC4 N840). These changes do not always ensure strict "upward compatibility" with the previous edition.

Details of the differences between this document and the 1999 edition are given in the technical discussion annex of this document.

Project Leader:	Len Slovensky Team SCRA	**Project Editor:**	Len Slovensky Team SCRA
Address:	1398 Gumbert Dr. Amelia, Ohio 45102 USA	**Address:**	1398 Gumbert Dr. Amelia, Ohio 45102 USA
Telephone:	+01 (513) - 753 - 7832	**Telephone:**	+01 (513) - 753 - 7832
Telefacsimile:	(513) - 753 - 7832	**Telefacsimile:**	(513) - 753 - 7832
Electronic mail:	slovensky@scra.org	**Electronic mail:**	slovensky@scra.org

© ISO 2000
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 from the publisher:

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

 ©ISO 2000 - All rights reserved

Contents

Page

1 Scope 1

2 Normative references 2

3 Terms, definitions, and abbreviations 4
3.1 Terms defined in ISO 1101 4
3.2 Terms defined in ISO 5459 4
3.3 Terms defined in ISO 10303-1 4
3.4 Terms defined in ISO 10303-31 5
3.5 Terms defined in ISO 10303-42 5
3.6 Terms defined in ISO 10303-213 5
3.7 Other definitions 5
3.8 Abbreviations 6

4 Information requirements 6
4.1 Units of functionality 7
4.1.1 design_exception 8
4.1.2 feature_definition_item 8
4.1.3 feature_profile 10
4.1.4 manufacturing_feature 10
4.1.5 manufacturing_part_properties 14
4.1.6 manufacturing_process_control_documentation 14
4.1.7 manufacturing_process_requirement_documents 15
4.1.8 measurement_limitations 15
4.1.9 part_administration_data 17
4.1.10 part_model 17
4.1.11 requisitions 18
4.1.12 shape_representation_for_machining 18
4.2 Application objects 19
4.3 Application assertions 186

5 Application interpreted model 224
5.1 Mapping table 224
5.2 AIM EXPRESS short listing 535

6 Conformance requirements 729

Annex A (normative) AIM EXPRESS expanded listing 730

Annex B (normative) AIM short names 885

Annex C (normative) Implementation method specific requirements 897

Annex D (normative) Protocol Implementation Conformance Statement (proforma) 898

Annex E (normative) Information object registration 899

©ISO 2000 - All rights reserved

Annex F (informative) Application activity model 900
F.1 Application activity model definitions 901
F.2 Application activity model diagrams 911

Annex G (informative) Application reference model 926

Annex H (informative) AIM EXPRESS-G 955

Annex J (informative) Computer interpretable listings 993

Annex K (informative) Application protocol usage guide 994

Annex L (informative) Technical discussion 995

Bibliography 1013

Index 1014

Figures

Figure 1 - Data planning model xvi
Figure 2 - Angle_taper 20
Figure 3 - Angular_dimension_tolerance attribute usage 21
Figure 4 - Angular_dimension_tolerance 22
Figure 5 - Angular_size_dimension_tolerance 23
Figure 6 - Angularity_tolerance for a plane surface 24
Figure 7 - Angularity_tolerance for an axis 25
Figure 8 - Block_base_shape 27
Figure 9 - Catalogue_knurl 30
Figure 10 - Catalogue_marking 31
Figure 11 - Catalogue_thread 31
Figure 12 - Chamfer 32
Figure 13 - Circular_boss 34
Figure 14 - Circular_boss with taper 34
Figure 15 - Circular_closed_profile 35
Figure 16 - Circular_offset_pattern 36
Figure 17 - Circular_omit_pattern 37
Figure 18 - Circular_pattern with rotation of base feature 38
Figure 19 - Circular_pattern without rotation of base feature 39
Figure 20 - Circular_runout_tolerance 41
Figure 21 - Circularity_tolerance 42
Figure 22 - Complete_circular_path 42
Figure 23 - Compound_feature 43
Figure 24 - Concentricity_tolerance 45
Figure 25 - Conical_hole_bottom 46
Figure 26 - Constant_radius_edge_round 47
Figure 27 - Constant_radius_fillet 48
Figure 28 - Counterbore_hole 50
Figure 29 - Countersunk_hole 50

 ©ISO 2000 - All rights reserved

Figure 30 - Curved_dimension_tolerance 51
Figure 31 - Cutout 53
Figure 32 - Cylindrical_base_shape 54
Figure 33 - Cylindricity_tolerance 55
Figure 34 - Defined_marking 57
Figure 35 - Thread and Defined_thread attributes 59
Figure 36 - Diagonal_knurl 61
Figure 37 - Diameter_dimension_tolerance 62
Figure 38 - Diameter_taper 63
Figure 39 - Diamond_knurl 64
Figure 40 - Distance_along_curve_tolerance 66
Figure 41 - Edge_round 67
Figure 42 - Fillet 70
Figure 43 - Flat hole bottom types 72
Figure 44 - Flat_slot_end_type 72
Figure 45 - Flatness_tolerance 74
Figure 46 - General_boss 75
Figure 47 - General_closed_profile 76
Figure 48 - General_open_profile 77
Figure 49 - General_outside_profile 78
Figure 50 - General_path 78
Figure 51 - General_pattern 79
Figure 52 - General_pocket 80
Figure 53 - General_removal_volume 82
Figure 54 - General_revolution for outer shape 83
Figure 55 - General_revolution for volume removal 85
Figure 56 - Groove 87
Figure 57 - Implicit_base_shape_representation 89
Figure 58 - Knurl 90
Figure 59 - Linear_path 92
Figure 60 - Linear_profile 93
Figure 61 - Linear_profile_tolerance 94
Figure 62 - Location_dimension_tolerance 95
Figure 63 - Manufactured_assembly and Mating_defintion 97
Figure 64 - Ngon_base_shape 103
Figure 65 - Ngon_profile 105
Figure 66 - Open_slot_end_type 107
Figure 67 - Outer_diameter 109
Figure 68 - Outer_diameter_to_shoulder 111
Figure 69 - Parallelism_tolerance for a plane 112
Figure 70 - Parallelism_tolerance for an axis 113
Figure 71 - Partial_circular_path 118
Figure 72 - Partial_circular_profile 119
Figure 73 - Perpendicularity_tolerance 121
Figure 74 - Planar_face 125
Figure 75 - Pocket_bottom_condition with start_or_end attribute 129
Figure 76 - Position_tolerance 130
Figure 77 - Protrusion 136
Figure 78 - Radial_dimension_tolerance 137

©ISO 2000 - All rights reserved

Figure 79 - Radiused_slot_end_type 137
Figure 80 - Recess 138
Figure 81 - Rectangular_boss 139
Figure 82 - Rectangular_closed_pocket 140
Figure 83 - Rectangular_closed_profile 141
Figure 84 - Rectangular_offset_pattern 142
Figure 85 - Rectangular_omit_pattern 143
Figure 86 - Rectangular_open_pocket 144
Figure 87 - Rectangular_pattern 145
Figure 88 - Revolved_flat 150
Figure 89 - Revolved_round 151
Figure 90 - Rib_top 152
Figure 91 - Round_hole 153
Figure 92 - Rounded_end 154
Figure 93 - Rounded_U_profile 156
Figure 94 - Shape_profile 158
Figure 95 - Slot 161
Figure 96 - Spherical_cap 164
Figure 97 - Spherical_hole_bottom 164
Figure 98 - Square_U_profile 165
Figure 99 - Step 167
Figure 100 - Straight_knurl 168
Figure 101 - Straightness_tolerance 168
Figure 102 - Surface_profile_tolerance 169
Figure 103 - Symmetry_tolerance 170
Figure 104 - Tee_profile 173
Figure 105 - Thread 175
Figure 106 - Through_bottom_condition 177
Figure 107 - Total_runout_tolerance 181
Figure 108 - Turned_knurl 182
Figure 109 - Vee_profile 184
Figure 110 - Slot with Woodruff_slot_end_type at each end 185
Figure 111 - Pattern offset required instances 628
Figure 112 - Pattern omit required instances 632
Figure F.1 - IDEF0 Basic notation 900
Figure F.2 - Mechanical products definition for process planning using machining features 912
Figure F.3 - A0 manufacture mechanical parts 913
Figure F.4 - A1 manage manufacturing process 914
Figure F.5 - A13 manage equipment and materials 915
Figure F.6 - A131 manage inventory 916
Figure F.7 - A2 capture digital product definition 917
Figure F.8 - A21 capture part data definition 918
Figure F.9 - A214 create part model 919
Figure F.10 - A3 generate manufacturing data 920
Figure F.11 - A31 generate process plan 921
Figure F.12 - A311 define resources 922
Figure F.13 - A312 define operator data 923
Figure F.14 - A315 define machine instructions 924
Figure F.15 - A4 operate shop floor 925

 ©ISO 2000 - All rights reserved

Figure G.1 - ARM EXPRESS-G diagram 1 of 28 927
Figure G.2 - ARM EXPRESS-G diagram 2 of 28 928
Figure G.3 - ARM EXPRESS-G diagram 3 of 28 929
Figure G.4 - ARM EXPRESS-G diagram 4 of 28 930
Figure G.5 - ARM EXPRESS-G diagram 5 of 28 931
Figure G.6 - ARM EXPRESS-G diagram 6 of 28 932
Figure G.7 - ARM EXPRESS-G diagram 7 of 28 933
Figure G.8 - ARM EXPRESS-G diagram 8 of 28 934
Figure G.9 - ARM EXPRESS-G diagram 9 of 28 935
Figure G.10 - ARM EXPRESS-G diagram 10 of 28 936
Figure G.11 - ARM EXPRESS-G diagram 11 of 28 937
Figure G.12 - ARM EXPRESS-G diagram 12 of 28 938
Figure G.13 - ARM EXPRESS-G diagram 13 of 28 939
Figure G.14 - ARM EXPRESS-G diagram 14 of 28 940
Figure G.15 - ARM EXPRESS-G diagram 15 of 28 941
Figure G.16 - ARM EXPRESS-G diagram 16 of 28 942
Figure G.17 - ARM EXPRESS-G diagram 17 of 28 943
Figure G.18 - ARM EXPRESS-G diagram 18 of 28 944
Figure G.19 - ARM EXPRESS-G diagram 19 of 28 945
Figure G.20 - ARM EXPRESS-G diagram 20 of 28 946
Figure G.21 - ARM EXPRESS-G diagram 21 of 28 947
Figure G.22 - ARM EXPRESS-G diagram 22 of 28 948
Figure G.23 - ARM EXPRESS-G diagram 23 of 28 949
Figure G.24 - ARM EXPRESS-G diagram 24 of 28 950
Figure G.25 - ARM EXPRESS-G diagram 25 of 28 951
Figure G.26 - ARM EXPRESS-G diagram 26 of 28 952
Figure G.27 - ARM EXPRESS-G diagram 27 of 28 953
Figure G.28 - ARM EXPRESS-G diagram 28 of 28 954
Figure H.1 - application context - AIM EXPRESS-G diagram 1 of 37 956
Figure H.2 - product definition - AIM EXPRESS-G diagram 2 of 37 957
Figure H.3 - property definition - AIM EXPRESS-G diagram 3 of 37 958
Figure H.4 - shape representation - AIM EXPRESS-G diagram 4 of 37 959
Figure H.5 - representation - AIM EXPRESS-G diagram 5 of 37 960
Figure H.6 - shape aspect - AIM EXPRESS-G diagram 6 of 37 961
Figure H.7 - instanced feature - AIM EXPRESS-G diagram 7 of 37 962
Figure H.8 - replicate feature and group - AIM EXPRESS-G diagram 8 of 37 963
Figure H.9 - transition feature - AIM EXPRESS-G diagram 9 of 37 964
Figure H.10 - geometry and topology - AIM EXPRESS-G diagram 10 of 37 965
Figure H.11 - point - AIM EXPRESS-G diagram 11 of 37 966
Figure H.12 - geometric orientation - AIM EXPRESS-G diagram 12 of 37 967
Figure H.13 - curve - AIM EXPRESS-G diagram 13 of 37 968
Figure H.14 - bounded curve - AIM EXPRESS-G diagram 14 of 37 969
Figure H.15 - surface curve - AIM EXPRESS-G diagram 15 of 37 970
Figure H.16 - surface - AIM EXPRESS-G diagram 16 of 37 971
Figure H.17 - elementary surface - AIM EXPRESS-G diagram 17 of 37 972
Figure H.18 - bounded surface - AIM EXPRESS-G diagram 18 of 37 973
Figure H.19 - topology - AIM EXPRESS-G diagram 19 of 37 974
Figure H.20 - shell - AIM EXPRESS-G diagram 20 of 37 975
Figure H.21 - document - AIM EXPRESS-G diagram 21 of 37 976

©ISO 2000 - All rights reserved

Figure H.22 - approval - AIM EXPRESS-G diagram 22 of 37 977
Figure H.23 - person and organization - AIM EXPRESS-G diagram 23 of 37 978
Figure H.24 - person and organization assignment - AIM EXPRESS-G diagram 24 of 37 979
Figure H.25 - date - AIM EXPRESS-G diagram 25 of 37 980
Figure H.26 - action - AIM EXPRESS-G diagram 26 of 37 981
Figure H.27 - security classification - AIM EXPRESS-G diagram 27 of 37 982
Figure H.28 - units - AIM EXPRESS-G diagram 28 of 37 983
Figure H.29 - measure with units - AIM EXPRESS-G diagram 29 of 37 984
Figure H.30 - measures - AIM EXPRESS-G diagram 30 of 37 985
Figure H.31 - datums - AIM EXPRESS-G diagram 31 of 37 986
Figure H.32 - tolerances - AIM EXPRESS-G diagram 32 of 37 987
Figure H.33 - geometric tolerances - AIM EXPRESS-G diagram 33 of 37 988
Figure H.34 - tolerance zone - AIM EXPRESS-G diagram 34 of 37 989
Figure H.35 - materials - AIM EXPRESS-G diagram 35 of 37 990
Figure H.36 - material property and identification - AIM EXPRESS-G diagram 36 of 37 991
Figure H.37 - attributes - AIM EXPRESS-G diagram 37 of 37 992

Tables

Table 1 - Mapping table for design_exception UoF 226
Table 2 - Mapping table for feature_definition_item UoF 229
Table 3 - Mapping table for feature_profile UoF 272
Table 4 - Mapping table for manufacturing_part_properties UoF 305
Table 5 - Mapping table for manufacturing_part_properties UoF 463
Table 6 - Mapping table for manufacturing_process_control_documentation UoF 471
Table 7 - Mapping table for manufacturing_process_requirement_documents UoF 478
Table 8 - Mapping table for measurement_limitations UoF 479
Table 9 - Mapping table for part_administration_data UoF 509
Table 10 - Mapping table for part model UoF 511
Table 11 - Mapping table for requisitions UoF 521
Table 12 - Mapping table for shape_representation_for_machining UoF 523
Table L.1 Summary of AP224 2nd edition ARM changes 996
Table L.2 Summary of AP224 2nd edition AIM changes 1003

 ©ISO 2000 - All rights reserved

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 ISO/IEC Directives, Part 3.

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 part of ISO 10303 may be the subject of patent rights. ISO shall not be held responsible for identifying any or all such patent rights.

International Standard ISO 10303-224 was prepared by ny Technical Committee ISO TC 184/SC4, *Industrial automation systems and integration*, Subcommittee 4 *Industrial data*. This second edition of ISO 10303-224 cancels and replaces the first edition (ISO 10303-224:1999), of which it constitutes a technical revision.

This International Standard is organized as a series of parts, each published separately. The structure of this standard is described in ISO 10303-1. The numbering of the parts of this International Standard reflects its structure:

— Parts 11 to 14 specify the description methods;

— Parts 21 to 29 specify the implementation methods;

— Parts 31 to 35 specify the conformance testing methodology and framework;

— Parts 41 to 50 specify the integrated generic resources;

— Parts 101 to 107 specify the integrated application resources;

— Parts 201 to 237 specify the application protocols;

— Parts 301 to 337 specify the abstract test suites;

— Parts 501 to 520 specify the application interpreted constructs.

A complete list of parts of ISO 10303 is available from the Internet:

`<http://www.nist.gov/sc4/editing/step/titles/>`

©ISO 2000 - All rights reserved

Should further parts of ISO 10303 be published, they will follow the same numbering pattern.

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

 ©ISO 2000 - All rights reserved

Introduction

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

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

This part of ISO 10303 specifies an application protocol (AP) for the representation of information needed to produce a mechanical part definition for process planning of a single piece or an assembly of piece parts for machining operations, and specifies the integrated resources necessary to satisfy these requirements.

This application protocol defines the context, scope, and information requirements for the representation of information needed to produce a mechanical part definition. These requirements specify the part identification, tracking, shape, representation of the shape and material data necessary for the definition of a part for process planning. The process planning function in an organization can be assisted a great deal by identifying machining oriented part shape features so that the process planner can more readily identify machining tools and processes to manufacture a part.

This application protocol identifies specific characteristics of part shape used in manufacturing. These characteristics are used to define manufacturing features. These shapes may be represented either by machining features defined in this application protocol, or by a boundary representation solid model, shared by other application protocols and used as application interpreted constructs in this part. The purpose of manufacturing features is to facilitate the identification of manufacturing shapes that are human and computer interpretable. Manufacturing features allow information about the shape to be used for decisions in computerized process planning systems.

Information about the part material is supplied so the process planning activity can determine equipment and material requirements. Also supplied is administrative information necessary for tracking customer information, supplier information about the part, and internal control information for the manufacturing operation to support process planning. Tracking of certain administrative information is a component of the iterative process for creating a process plan.

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

Clause 1 defines the scope of the application protocol and summarizes the functionality and data covered by the AP. Clause 3 lists the words defined in this part of ISO 10303 and gives pointers to words defined elsewhere. An application activity model that is the basis for the definition of the scope is provided in annex F. The information requirements of the application are specified in clause 4 using terminology appropriate to the application. A graphical representation of the information requirements, referred to as the application reference model, is given in annex G.

©ISO 2000 - All rights reserved

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

Figure 1 contains the data planning model that provides a high level description of the requirements for this application protocol. This planning model was created from the in-scope data from the activities of the application activity model (AAM) and grouped into logical units of functionality. This planning model is used as a guide in developing the application reference model (ARM).

This edition of this part of ISO 10303 incorporates modifications that are upwardly compatible with the previous edition, and modifications that are not upwardly compatible with the previous edition. Modifications to EXPRESS specifications are upwardly compatible if:

— the modifications do not result in changes to instances that are encoded according to ISO 10303-21; such instances conform to both the unmodified and modified EXPRESS specifications;

— the modifications do not result in changes to software that conforms to ISO 10303-22 with respect to access to the data content of data structures;

— the modifications do not invalidate mappings to the previous edition of this part of ISO 10303 that are specified in the mapping table of an ISO 10303 application protocol.

Technical modifications that are upwardly compatible or are not upwardly compatible with ISO 10303-224:1999 are summarized in annex L. A brief summary of the modifications are the following:

— New Manufacturing_features which include Cutout, Recess, Rib_top, and Shape_profile;

— Enhancements to existing features and feature_definition_items which include Planar_face, Ngon_profile, Ngon_base_shape, and new sub-type for Boss which is a Rectangular_boss;

— Increased scope from the manufacture of a single piece part to the manufacturing of an assembly of piece parts;

— Modified Manufacturing_part_properties entities found in error in the first edition (ISO 10303-224:1999);

— Modified several information requirements and corresponding application interpreted model entities to support harmonization with other application protocol;

— Normatively reference all of the current second edition and technical corrigendum parts of ISO 10303.

 ©ISO 2000 - All rights reserved

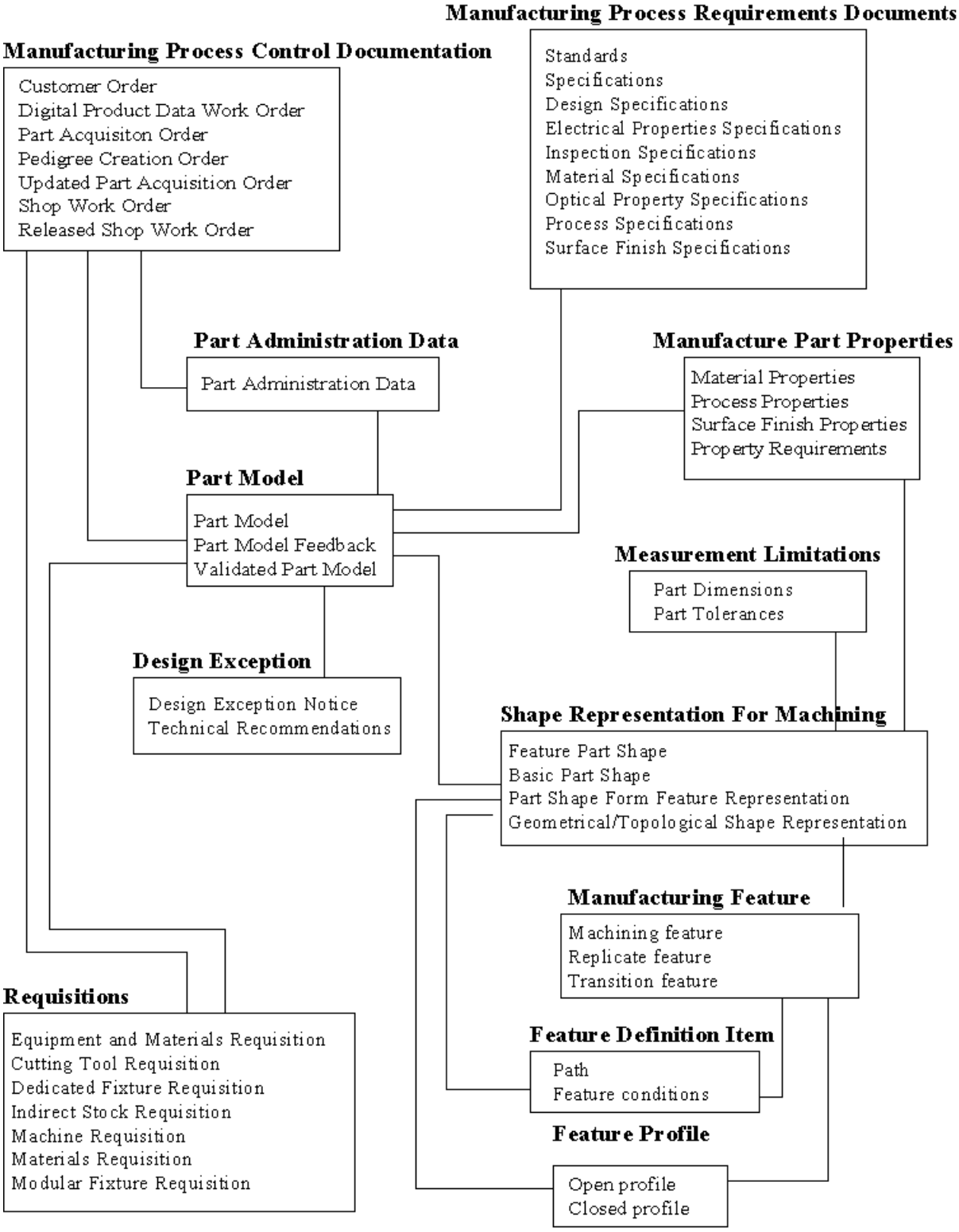


Figure 1 - Data planning model

©ISO 2000 - All rights reserved

 ©ISO 2000 - All rights reserved

Industrial automation systems and integration — Product data representation and exchange — Part 224: Application protocol: Mechanical product definition for process planning using machining features

1 Scope

This part of ISO 10303 specifies the use of the integrated resources necessary for the scope and information requirements for the representation and exchange of information needed to define product data necessary for manufacturing a single piece or assembly of mechanical parts. The product data is based on existing part designs that have their shapes represented by machining features. This part supports digital representation for computer integrated manufacturing.

NOTE 1 - The application activity model in annex F provides a graphical representation of the process and information flows which are the basis for the definition of the scope of this Part of ISO 10303.

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

— the manufacture of a single piece mechanical part, and assemblies of single piece parts for manufacturing purpose;

— parts that are to be manufactured by either milling or turning processes;

— machining features for defining shapes necessary for manufacturing;

NOTE 2 - The machining feature set is defined in this part of ISO 10303.

— Explicit representation of the 3D shape of machining features through bounded geometry representations;

— implicit representation of machining features through selection of standard parameters;

— machining feature definition elements necessary for creating machining form features;

— customer order administrative data to track receipt of an order for a part to the shop floor, but not including tracking of the order on the shop floor;

— approval data to authorize the manufacture of a part;

— requisition administrative data to identify requirements and track the status of materials and equipment needed to manufacture a part;

— work order data to track and identify the status of a part;

©ISO 2000 - All rights reserved

— tracking the state of raw stock for documenting the manufacturing history of a part;

— tracking a design exception notice of a part.

NOTE 3 - The design exception notice relates to discrepancies in the machining features used to describe a part's shape.

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

— results from process planning functions;

— exchange of data within process planning systems;

— feature order or sequence;

— representation of assemblies for design or bill of materials;

— representation of composite material parts;

— representation of sheet metal parts;

— representation of part pedigree;

— design features of a part;

— schedule for completing a work order through the manufacturing process;

— configuration control for a part.

2 Normative references

The following normative documents contain provisions which, through reference in this text, constitute provisions of this International Standard. For dated references, subsequent amendments to, or revisions of, any of these publications do not apply. However, parties to agreements based on this International Standard are encouraged to investigate the possibility of applying the most recent editions of the normative documents indicated below. For undated references, the latest edition of the normative document referred to applies. Members of ISO and IEC maintain registers of currently valid International Standards.

ISO 286–1:1988, *ISO system of limits and fits – Part 1: Bases of tolerances, deviations, and fits.*

ISO 286–2:1988, *ISO system of limits and fits – Part 2: Tables of standard tolerance grades and limit deviations for holes and shafts.*

ISO 1101:1983, *Technical drawings - Geometric tolerancing - Tolerance of form, orientation, location and run-out - Generalities definitions, symbols, indications on drawings.*

 ©ISO 2000 - All rights reserved

ISO 5459:1981, *Technical drawings - Geometric tolerancing - Datums and datum-systems for geometric tolerances.*

ISO/IEC 8824-1:1994, *Information Technology - Open Systems Interconnection - Abstract Syntax Notation One (ASN.1) - Part 1: Specification of Basic Notation.*

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

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

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

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

ISO/DIS 10303-41:1999, *Industrial automation systems and integration - Product data representation and exchange - Part 41: Integrated generic resources: Fundamentals of product description and support.*

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

ISO 10303-43:2000, *Industrial automation systems and integration - Product data representation and exchange - Part 43: Integrated generic resources: Representation structures.*

ISO 10303-44:2000, *Industrial automation systems and integration - Product data representation and exchange - Part 44: Integrated generic resources: Product structure configuration.*

ISO 10303-45:1998, *Industrial automation systems and integration - Product data representation and exchange - Part 45: Integrated generic resources: Materials.*

ISO 10303-47:1997, *Industrial automation systems and integration - Product data representation and exchange - Part 47: Integrated generic resources: Shape variation tolerances.*

ISO 10303-511:1999, *Industrial automation systems and integration - Product data representation and exchange - Part 511: Application interpreted construct: Topologically bounded surface.*

ISO 10303-514:1999, *Industrial automation systems and integration - Product data representation and exchange - Part 514: Application interpreted construct: Advanced boundary representation.*

ISO 10303-519:1999, *Industrial automation systems and integration - Product data representation and exchange - Part 519: Application interpreted construct: Geometric tolerances.*

©ISO 2000 - All rights reserved

3 Terms, definitions, and abbreviations

3.1 Terms defined in ISO 1101

For the purpose of this part of ISO 10303, the following terms defined in ISO 1101 apply.

— dimension;

— tolerance.

3.2 Terms defined in ISO 5459

For the purpose of this part of ISO 10303, the following terms defined in ISO 5459 apply.

— datum.

3.3 Terms defined in ISO 10303-1

For the purpose of this part of ISO 10303, the following terms defined in ISO 10303-1 apply.

— application;

— application activity model (AAM);

— application interpreted model (AIM);

— application object;

— application protocol (AP);

— application reference model (ARM);

— implementation method;

— integrated resource;

— model;

— product;

— product data;

— protocol implementation conformance statement (PICS);

— unit of functionality (UoF).

 ©ISO 2000 - All rights reserved

3.4 Terms defined in ISO 10303-31

For the purpose of this part of ISO 10303, the following terms defined in ISO 10303-31 apply.

— conformance class;

— conformance testing.

3.5 Terms defined in ISO 10303-42

For the purpose of this part of ISO 10303, the following terms defined in ISO 10303-42 apply.

— boundary representation (B-rep) solid model;

— manifold solid boundary representation.

3.6 Terms defined in ISO 10303-213

For the purpose of this part of ISO 10303, the following terms defined in ISO 10303-213 apply.

— process plan;

— workstation.

3.7 Other definitions

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

3.7.1
basic angle
is implied where center lines of features in a pattern or surfaces shown on a drawing are located or defined by the use of basic dimensions and no angle is specified.

3.7.2
cutting tool
an instrument used to remove material from a part.

3.7.3
dedicated fixture
a fixture designed for a particular part.

3.7.4
fixture
a device to hold and locate a part during manufacturing and inspection.

©ISO 2000 - All rights reserved

3.7.5
indirect material
consumable material used during the manufacture of a part.

EXAMPLE - Oil, cutting fluid, pallets, brooms, shop towels, etc.

3.7.6
machine
a structure consisting of a framework with various moving parts, used for material removal operations to a part.

3.7.7
modular fixture
fixture component that is interchangeable and can be assembled into different configurations as needed.

3.7.8 raw stock
any material that can have a manufacturing process applied, including castings and forgings.

3.8 Abbreviations

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

AAM	application activity model
AIC	application interpreted construct
AIM	application interpreted model
AP	application protocol
ARM	application reference model
B-rep	boundary representation
ICOM	input, control, output, mechanism
Ngon	N number of sides polygon
PICS	protocol implementation conformance statement
UoF	unit of functionality

4 Information requirements

This clause specifies the information required for the definition of product data for mechanical product definition for process planning using machining features.

 ©ISO 2000 - All rights reserved

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

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

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

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

4.1 Units of functionality

This subclause specifies the units of functionality for the mechanical product definition for process planning using machining features application protocol. This part of ISO 10303 specifies the following units of functionality:

— design_exception;

— feature_definition_item;

— feature_profile;

— manufacturing_feature;

— manufacturing_part_properties;

— manufacturing_process_control_documentation;

— manufacturing_process_requirement_documents;

— measurement_limitations;

— part_administration_data;

— part_model;

— requisitions;

— shape_representation_for_machining.

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

©ISO 2000 - All rights reserved

4.1.1 design_exception

The design_exception UoF contains the application objects used for documentation required for issuing an error report for a problem that was discovered in the creation of a process plan, and a solution to the problem as it pertains to the regeneration of input data for further process planning.

The following application objects are used by the design_exception UoF:

— Design_exception_notice;

— Engineering_change_order;

— Engineering_change_proposal.

4.1.2 feature_definition_item

The feature_definition_item UoF contains the information necessary to create a machining feature. Additionally this UoF identifies the relationship between machining features and aspects of shape.

The following application objects are used by the feature_definition_item UoF:

— Angle_taper;

— Blind_bottom_condition;

— Boss_top_condition;

— Chamfer_angle;

— Circular_path;

— Complete_circular_path;

— Conical_hole_bottom;

— Diameter_taper;

— Directed_taper;

— First_offset;

— Flat_hole_bottom;

— Flat_slot_end_type;

— Flat_with_radius_hole_bottom;

— Flat_with_taper_hole_bottom;

 ©ISO 2000 - All rights reserved

— General_path;

— General_pocket_bottom_condition;

— General_profile_floor;

— General_rib_top_floor;

— General_top_condition;

— Linear_path;

— Open_slot_end_type;

— Partial_area_definition;

— Partial_circular_path;

— Path;

— Planar_pocket_bottom_condition;

— Planar_profile_floor;

— Planar_rib_top_floor;

— Planar_top_condition;

— Pocket_bottom_condition;

— Profile_floor;

— Radiused_slot_end_type;

— Rib_top_floor;

— Second_chamfer_offset;

— Second_offset;

— Slot_end_type;

— Spherical_hole_bottom;

— Through_bottom_condition;

©ISO 2000 - All rights reserved

— Through_pocket_bottom_condition;

— Through_profile_floor;

— Woodruff_slot_end_type.

4.1.3 feature_profile

The feature_profile UoF contains the information necessary to identify 2D shapes. By sweeping a feature_profile along a path, 3D features are created.

The following application objects are used by the feature_profile UoF:

— Circular_closed_profile;

— Closed_profile;

— General_closed_profile;

— General_open_profile;

— Linear_profile;

— Ngon_profile;

— Open_profile;

— Partial_circular_profile;

— Profile;

— Rectangular_closed_profile;

— Rounded_U_profile;

— Square_U_profile;

— Tee_profile;

— Vee_profile.

4.1.4 manufacturing_feature

The manufacturing_feature UoF contains the information necessary to identify shapes which represent volumes of material that shall be removed from a part by machining or shall result from machining.

 ©ISO 2000 - All rights reserved

The following application objects are used by the manufacturing_feature UoF:

— Boss;

— Catalogue_knurl;

— Catalogue_marking;

— Catalogue_thread;

— Chamfer;

— Circular_boss;

— Circular_closed_shape_profile;

— Circular_cutout;

— Circular_offset_pattern;

— Circular_omit_pattern;

— Circular_pattern;

— Compound_feature;

— Compound_feature_element;

— Compound_feature_relationship;

— Constant_radius_edge_round;

— Constant_radius_fillet;

— Counterbore_hole;

— Countersunk_hole;

— Cutout;

— Defined_marking;

— Defined_thread;

— Diagonal_knurl;

— Diamond_knurl;

— Edge_round;

©ISO 2000 - All rights reserved

— Fillet;

— General_boss;

— General_cutout;

— General_outside_profile;

— General_pattern;

— General_pocket;

— General_removal_volume;

— General_revolution;

— General_shape_profile;

— Groove;

— Hole;

— Knurl;

— Machining_feature;

— Manufacturing_feature;

— Manufacturing_feature_group;

— Marking;

— Multi_axis_feature;

— Outer_diameter;

— Outer_diameter_to_shoulder;

— Outer_round;

— Partial_circular_shape_profile;

— Planar_face;

— Pocket;

— Profile_feature;

— Protrusion;

 ©ISO 2000 - All rights reserved

— Recess;

— Rectangular_boss;

— Rectangular_closed_pocket;

— Rectangular_closed_shape_profile;

— Rectangular_offset_pattern;

— Rectangular_omit_pattern;

— Rectangular_open_pocket;

— Rectangular_open_shape_profile;

— Rectangular_pattern;

— Replicate_base;

— Replicate_feature;

— Revolved_feature;

— Revolved_flat;

— Revolved_round;

— Rib_top;

— Round_hole;

— Rounded_end;

— Shape_profile;

— Slot;

— Spherical_cap;

— Step;

— Straight_knurl;

— Thread;

— Transition_feature;

— Turned_knurl.

©ISO 2000 - All rights reserved

4.1.5 manufacturing_part_properties

The manufacturing_part_properties UoF contains the description of characteristics of the part that is being defined. These characteristics specify requirements for manufacturing that apply to either the state of the part at a particular time prior to or after the manufacture of the part, or a process that is required to be executed during the manufacture of the part.

The following application objects are used by the manufacturing_part_properties UoF:

— Alternate_material;

— Descriptive_parameter;

— Hardness;

— Material;

— Material_property;

— Numeric_parameter;

— Numeric_parameter_with_tolerance;

— Part_property;

— Process_property;

— Property;

— Property_parameter;

— Surface_property.

4.1.6 manufacturing_process_control_documentation

The manufacturing_process_control_documentation UoF contains information that identifies product data order information. These documents include work order information from an internal or external customer and internal manufacturing operations documentation that allows for the tracking of the work on a part's manufacture.

The following application objects are used by the manufacturing_process_control_documentation UoF:

— Customer_order;

— Digital_technical_data_package_work_order;

— Ordered_part;

— Pedigree_creation_order;

 ©ISO 2000 - All rights reserved

— Project_order;

— Resource_acquisition_order;

— Shop_work_order.

4.1.7 manufacturing_process_requirement_documents

The manufacturing_process_requirement_documents UoF provides the ability to specify documents that are directly related to product data as they support the definition of the product. These documents may be specific to an operation on the part being manufactured or a property of the part at a particular stage in the manufacturing process. The documents may, but need not be, company, customer, military, national, or international standards.

The following application objects are used by the manufacturing_process_requirement_documents UoF:

— Specification;

— Specification_usage_constraint.

4.1.8 measurement_limitations

The measurement_limitations UoF contains the information necessary to identify the important sizes of the measured relationships between aspects of a part's shape or between an aspect of a part's shape and a reference shape that does not comprise the shape of the part, and the acceptable deviation from that size or relationship for the purpose of manufacturing.

The following application objects are used by the measurement_limitations UoF:

— Angular_dimension_tolerance;

— Angular_size_dimension_tolerance;

— Angularity_tolerance;

— Circular_runout_tolerance;

— Circularity_tolerance;

— Compound_datum;

— Concentricity_tolerance;

— Curved_dimension_tolerance;

— Cylindricity_tolerance;

— Datum;

©ISO 2000 - All rights reserved

— Datum_feature;

— Datum_target;

— Datum_target_set;

— Diameter_dimension_tolerance;

— Dimensional_tolerance;

— Distance_along_curve_tolerance;

— Flatness_tolerance;

— Geometric_tolerance;

— Geometric_tolerance_precedence_relationship;

— Limits_and_fits;

— Linear_profile_tolerance;

— Location_dimension_tolerance;

— Location_tolerance;

— Material_condition_modifier;

— Parallelism_tolerance;

— Perpendicularity_tolerance;

— Placed_target;

— Plus_minus_value;

— Position_tolerance;

— Projection;

— Radial_dimension_tolerance;

— Size_tolerance;

— Straightness_tolerance;

— Surface_profile_tolerance;

— Symmetry_tolerance;

 ©ISO 2000 - All rights reserved

— Target_area;

— Target_circle;

— Target_line;

— Target_point;

— Target_rectangle;

— Tolerance_limit;

— Tolerance_range;

— Tolerance_value;

— Tolerance_zone;

— Tolerance_zone_definition;

— Total_runout_tolerance.

4.1.9 part_administration_data

The part_administration_data UoF contains the information used in the management of product data.

The following application objects are used by the part_administration_data UoF:

— Approval;

— Organization;

— Person;

— Person_in_organization.

4.1.10 part_model

The part_model UoF contains the information necessary to identify the part that is to be input to the process planning function and identify the association of properties with that part. Additionally, information pertaining to feedback about the quality or necessary revisions to the product data is represented by this UoF.

The following application object is used by the part_model UoF:

— Manufactured_assembly;

— Manufactured_assembly_relationship;

©ISO 2000 - All rights reserved

— Mating_definiton;

— Mating_definition_relationship;

— Mating_relationship;

— Part;

— Single_piece_part.

4.1.11 requisitions

The requisitions UoF contains the information necessary to identify documents generated by a manufacturing organization that specify an order for obtaining necessary manufacturing resources.

The following application objects are used by the requisitions UoF:

— Cutting_tool_requisition;

— Dedicated_fixture_requisition;

— Indirect_stock_requisition;

— Machine_requisition;

— Material_requisition;

— Modular_fixture_requisition;

— Requisition.

4.1.12 shape_representation_for_machining

The shape_representation_for_machining UoF contains the physical definition of initial and final form of the part. This definition is given via a parametric method for features, and additional geometric and topological definitions.

The following application objects are used by the shape_representation_for_machining UoF:

— Base_shape;

— Block_base_shape;

— B-rep_model;

— B-rep_model_element;

— B-rep_shape_aspect_representation;

 ©ISO 2000 - All rights reserved

— B-rep_shape_representation;

— Cylindrical_base_shape;

— Direction_element;

— Explicit_base_shape_representation;

— Face_shape_element;

— Face_shape_element_relationship;

— Implicit_base_shape_representation;

— Location_element;

— Ngon_base_shape;

— Orientation;

— Part_placement;

— Path_element;

— Planar_element;

— Shape;

— Shape_aspect;

— Shape_element.

4.2 Application objects

This subclause specifies the application objects for them mechanical product definition for process planning using machining features application protocol. Each application object is an atomic element that embodies a unique application concept and contains attributes specifying the data elements of the object. The application objects and their definitions are given below.

4.2.1 Alternate_material

An Alternate_material is the identification of a secondary material that may be used when the primary material choice is not available. The data associated with an Alternate_material are the following:

— alternate_ranking;

— material_substitute.

©ISO 2000 - All rights reserved

4.2.1.1 alternate_ranking

The alternate_ranking specifies the order for selecting Alternate_material objects in the event the primary Material is not available.

4.2.1.2 material_substitute

The material_substitute specifies the material to be used as the substitute. See 4.3.1 for the application assertion.

4.2.2 Angle_taper

An Angle_taper is a constant change in shape of a feature for a part. The start of the Angle_taper is at the placement of a feature and is applied to the entire feature. The length of the taper is determined from the feature that is applying the Angle_taper.

NOTE - Figure 2 illustrates features that have an Angle_taper applied.

The data associated with an Angle_taper are the following:

— angle.

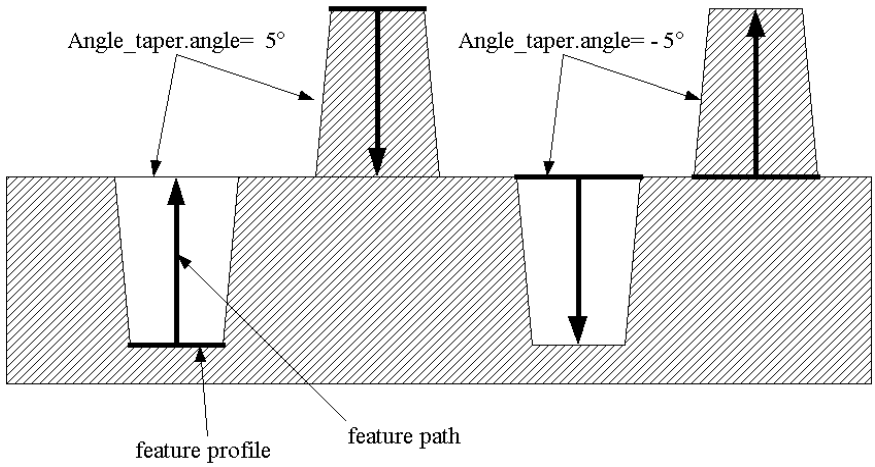


Figure 2 - Angle_taper

4.2.2.1 angle

The angle specifies the amount of slope from the start of an Angler_taper to the end of the Angle_taper. See 4.3.2 for the application assertion. An angle between 0 and 90 degrees or between -90 and -180 degrees indicates that the profile for a feature grows larger along the feature path. An angle between 0 and -90 degrees or between 90 and 180 degrees indicates that the profile for a feature grows smaller along thefeature path.

 ©ISO 2000 - All rights reserved

4.2.3 Angular_dimension_tolerance

An Angular_dimension_tolerance is a type of Location_tolerance (see 4.2.109) that defines the allowable variation in the angle between two elements of the shape of a part. Each Angular_dimension_tolerance shall have an origin shape and a termination shape.

NOTE - Figure 4 illustrates the Angular_dimension_tolerance and Figure 3 illustrates use of the attributes.

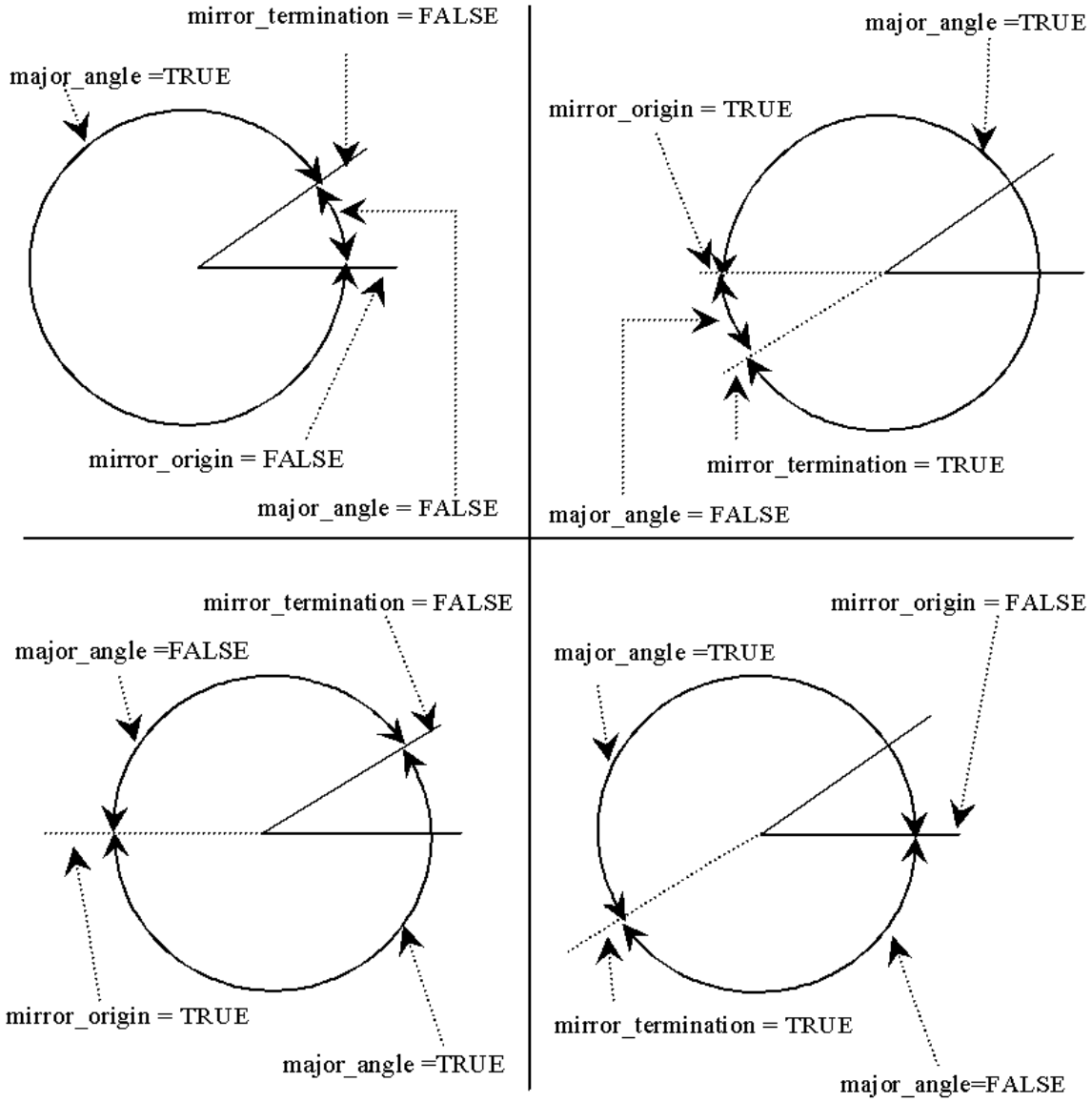


Figure 3 - Angular_dimension_tolerance attribute usage

©ISO 2000 - All rights reserved

The data associated with an Angular_dimension_tolerance are the following:

— major_angle;

— mirror_origin;

— mirror_termination.

4.2.3.1 major_angle

The major_angle specifies a boolean value that indicates the size of the angle which defines a variation. A value of true specifies the angle to be the larger of the two angles formed by the two elements of the part's shape (origin and termination) that are related to the Angular_dimension_tolerance. This attribute is used in combination with the other attributes to determine the angle that is being limited.

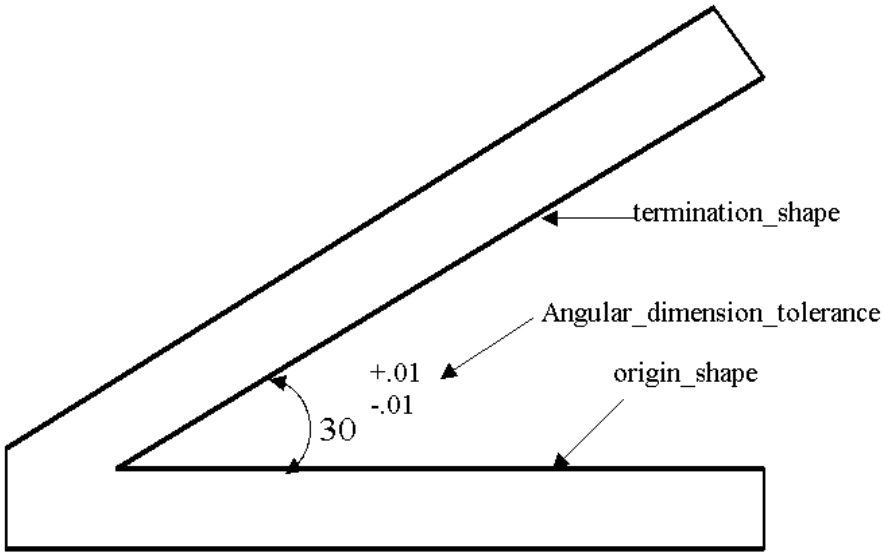


Figure 4 - Angular_dimension_tolerance

4.2.3.2 mirror_origin

The mirror_origin specifies a boolean value that indicates whether the element of the part's shape defining the origin of the limitation is to be mirrored in order to define the angle whose allowable variation is being specified. A value of true indicates that the origin element is mirrored about a plane that it is orthogonal to.

4.2.3.3 mirror_termination

The mirror_termination specifies a boolean value that indicates whether the element of the part's shape that defines the termination of the limitation is to be mirrored in order to define the angle whose

 ©ISO 2000 - All rights reserved

allowable variation is being specified. A value of true indicates that the termination element is mirrored about a plane that it is orthogonal to.

4.2.4 Angular_size_dimension_tolerance

An Angular_size_dimension_tolerance is a type of Size_tolerance (see 4.2.204) that specifies the allowable variation on the size or gap formed by two angular elements of the shape of a part.

NOTE - Figure 5 illustrates the Angular_size_dimension_tolerance.

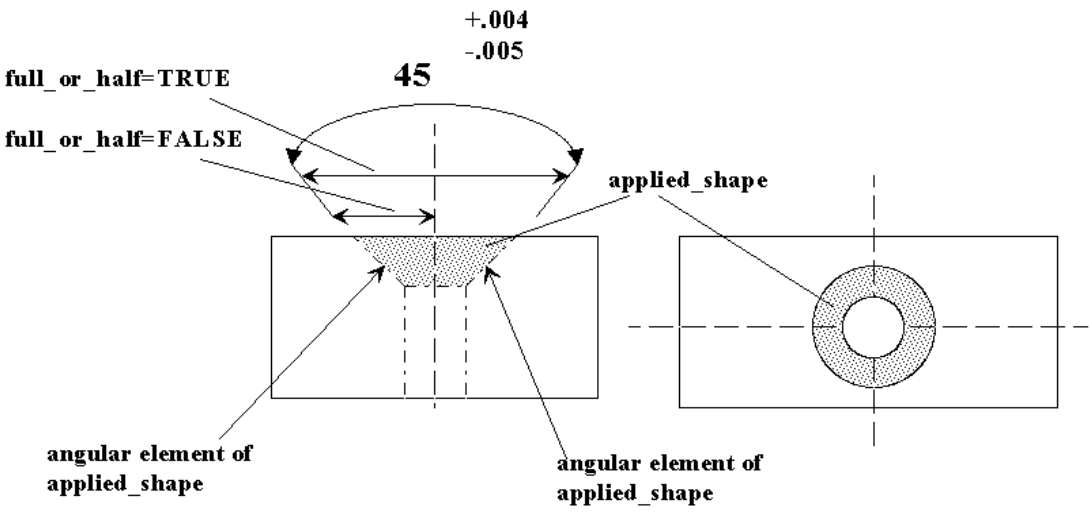


Figure 5 - Angular_size_dimension_tolerance

The data associated with an Angular_size_dimension_tolerance are the following:

— full_or_half;

— major_angle.

4.2.4.1 full_or_half

The full_or_half specifies a boolean value that indicates the method used to establish the Angular_size_-dimension_tolerance angle. A value of true specifies the angle is established between the two sides of an angular element. A value of false specifies the angle is established between a center line datum and an angular element.

4.2.4.2 major_angle

The major_angle specifies a boolean value that indicates the size of the angle for defining the variation. A value of TRUE specifies the angle to be the larger of the two angles formed by the two elements of the part's shape that are related to the Angular_size_dimension_tolerance.

©ISO 2000 - All rights reserved

4.2.5 Angularity_tolerance

An Angularity_tolerance is a type of Geometric_tolerance (see 4.2.95) that is the allowable variation of a surface or axis at a specified angle (other than 90 degrees) from a datum plane or axis. An Angularity_-tolerance constrains a shape that is one of the following:

— An allowable variation, defined by a tolerance zone, between a surface and a datum plane specified by the basic angle;

— An allowable variation, defined by a diametric tolerance zone, between an axis and a datum plane specified by the basic angle.

NOTE 1 - Figure 6 illustrates the Angularity_tolerance for a plane surface.

NOTE 2 - Figure 7 illustrates Angularity_tolerance for an axis.

NOTE 3 - The Angularity_tolerance definition is derived from paragraph 14.9 of ISO 1101.

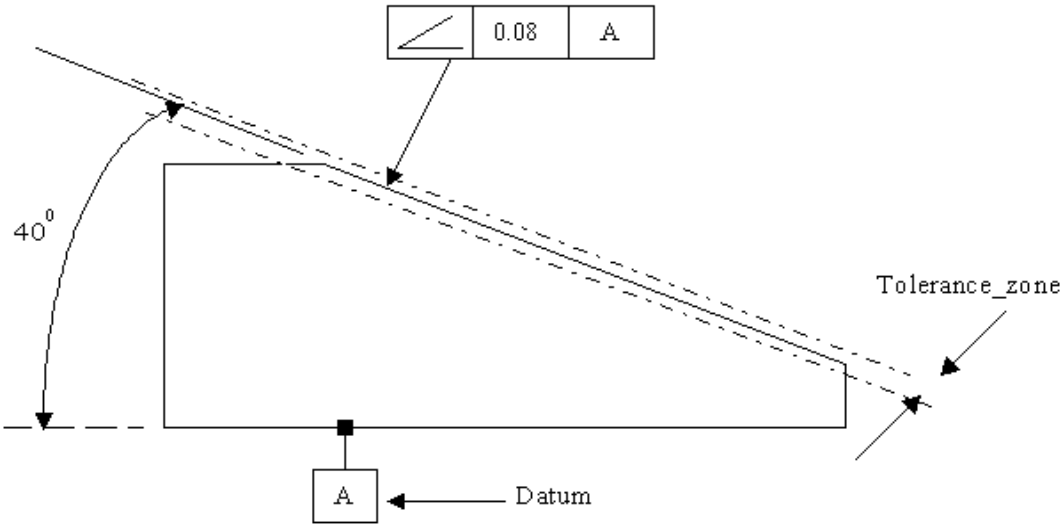


Figure 6 - Angularity_tolerance for a plane surface

The data associated with an Angularity_tolerance are the following:

— geometric_reference;

— segment_size.

4.2.5.1 geometric_reference

The geometric_reference specifies the datum to which the tolerance is related. See 4.3.3 for the application assertion.

 ©ISO 2000 - All rights reserved

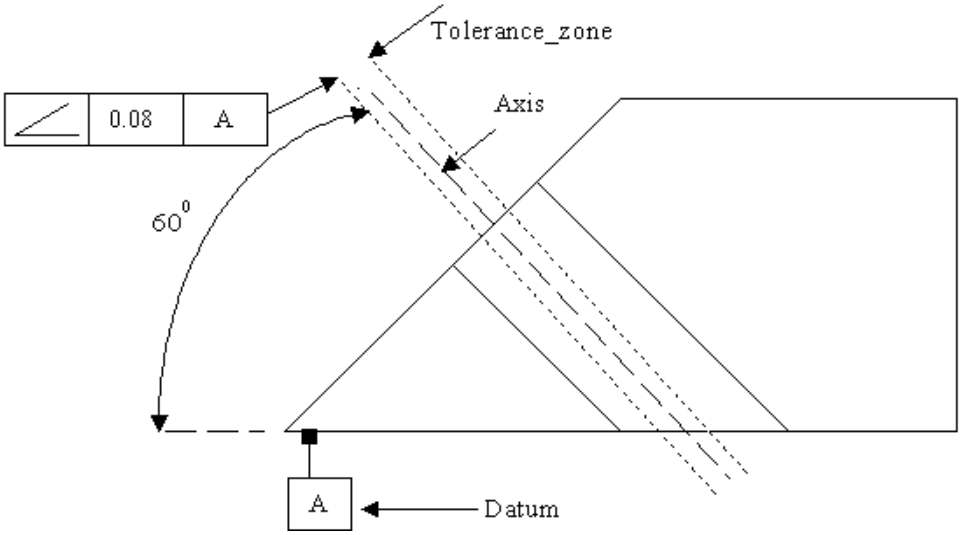


Figure 7 - Angularity_tolerance for an axis

4.2.5.2 segment_size

The segment_size specifies the length of a surface to apply a tolerance if the Angularity_tolerance is not applied to the total length. The segment_size need not be specified for a particular Angularity_tolerance.

4.2.6 Approval

An Approval is the indication within an organization of concurrence or nonconcurrence with a piece of product data. The data associated with an Approval are the following:

— approval_authority;

— approval_date;

— status.

4.2.6.1 approval_authority

The approval_authority specifies the responsible person and company for making the approval. There may be more than one approval_authority for an Approval. See 4.3.4 for the application assertion.

4.2.6.2 approval_date

The approval_date specifies a calendar date on which the approval status was set.

4.2.6.3 status

The status specifies the state of consent applied to the approval to manufacture a part.

©ISO 2000 - All rights reserved

The values of the status may be one of the following:

— approved;

— disapproved;

— not_yet_approved;

— withdrawn.

4.2.6.3.1 approved: a concurrence has been given on a specific date.

4.2.6.3.2 disapproved: a nonconcurrence has been given on a specific date.

4.2.6.3.3 not_yet_approved: a concurrence has not yet been given.

4.2.6.3.4 withdrawn: a concurrence has been removed on a specific date.

4.2.7 Base_shape

A Base_shape is the initial shape of the material before machining of the features. Each Base_shape is either an Explicit_base_shape_representation (see 4.2.70) or an Implicit_base_shape_representation (see 4.2.100).

4.2.8 Blind_bottom_condition

A Blind_bottom_condition is a selection type of Hole_bottom_condition_select that has material in the bottom of a hole and does not go through the entire part. Each Blind_bottom_condition is either a Conical_hole_bottom (see 4.2.38), Flat_hole_bottom (see 4.2.75), Flat_with_radius_hole_bottom (see 4.2.77), Flat_with_taper_hole_bottom (see 4.2.78), or a Spherical_hole_bottom (see 4.2.210).

The data associated with a Blind_bottom_condition are the following:

— start_or_end.

4.2.8.1 start_or_end

The start_or_end specifies a boolean value of TRUE if the Blind_bottom_condition is positioned at the end of a Round_hole, and a value of FALSE if it is at the start of the Round_hole.

4.2.9 Block_base_shape

A Block_base_shape is a type of Implicit_base_shape_representation (see 4.2.100) that describes the initial shape of the material as a rectangular cross section of some determined length.

NOTE - Figure 8 illustrates a Block_base_shape and the associated attributes.

 ©ISO 2000 - All rights reserved

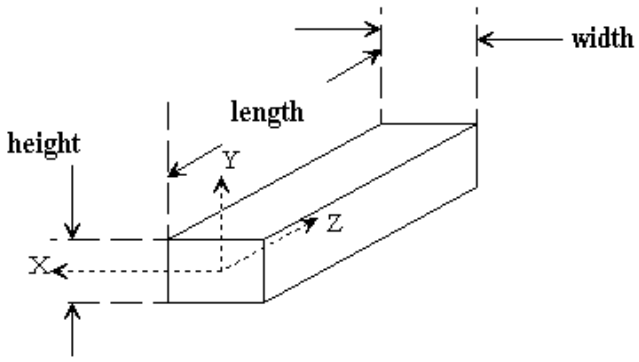


Figure 8 - Block_base_shape

The data associated with a Block_base_shape are the following:

— height;

— width.

4.2.9.1 height

The height specifies the size of the side of a Block_base_shape along the Y-axis. See 4.3.5 for the application assertion.

4.2.9.2 width

The width specifies the size of the side of a Block_base_shape along the X-axis. See 4.3.5 for the application assertion.

4.2.10 Boss

A Boss is a type of Multi_axis_feature(see 4.2.125) that is a closed shape that protrudes from the surface of the part. The intersection of the Boss and the base surface may have a radius shaped blend between them. Each Boss is either a Circular_boss (see 4.2.21), General_boss (see 4.2.80), or a Rectangular_boss (see 4.2.175). The Boss may be positioned on the face of a part with the Z-axis in the direction away from the part, or at the top of the Boss with the Z-axis in the direction toward the part face. The data associated with a Boss are the following:

— boss_height;

— fillet_radius;

©ISO 2000 - All rights reserved

— top_condition;

— top_radius.

4.2.10.1 boss_height

The boss_height specifies the maximum height of a Boss measured from the highest point on the Boss to the farthest point of intersection between the Boss and the part surface. The placement and orientation of the Linear_path shall be the same as the Boss feature. See 4.3.7 for the application assertion.

4.2.10.2 fillet_radius

The fillet_radius specifies a radius shape blend between a Boss and the surrounding surface at the base of the Boss. See 4.3.8 for the application assertion.

4.2.10.3 top_condition

The top_condition specifies the shape of the top of a Boss feature. See 4.3.6 for the application assertion.

4.2.10.4 top_radius

The top_radius specifies a radius shape blend between a Boss_top_condition and the surrounding Boss surface at the top of the Boss. See 4.3.8 for the application assertion.

NOTE - it the Boss_top_condition was a General_top_condition that defined a spherical shape, the top_radius may not apply.

4.2.11 Boss_top_condition

A Boss_top_condition is the end shape of a Boss that is the farthest distance away from the intersection of the face of the part and the Boss. A Boss top may be either flat or of any other shape. Each Boss_-top_condition is either a General_top_condition (see 4.2.94) or a Planar_top_condition (see 4.2.158). The data associated with a Boss_top are the following:

— start_or_end.

4.2.11.1 start_or_end

The start_or_end specifies a boolean value of TRUE if the Boss_top_condition is located at the start of the Boss, FALSE if it is located at the end of the Boss.

4.2.12 B-rep_model

A B-rep_model is a solid model containing complete representation of shape using manifold solid boundary representation.

 ©ISO 2000 - All rights reserved

4.2.13 B-rep_model_element

A B-rep_model_element is a portion of a boundary representation. The data associated with a B-rep_-model_element are the following:

— element.

4.2.13.1 element

The element specifies the portion of the B-rep_model that defines the B-rep_model_element. See 4.3.9 for the application assertion.

4.2.14 B-rep_shape_aspect_representation

A B-rep_shape_aspect_representation is a grouping of geometric elements with respect to the boundary representation of a part. The data associated with a B-rep_shape_aspect_representation are the following:

— shape_definition.

4.2.14.1 shape_definition

The shape_definition specifies the shape that is the boundary representation. See 4.3.10 for the application assertion.

4.2.15 B-rep_shape_representation

A B-rep_shape_representation is a grouping of boundary representations to define a shape. The data associated with a B-rep_shape_representation are the following:

— shape_definition.

4.2.15.1 shape_definition

The shape_definition specifies the shape that is the boundary representation. See 4.3.11 for the application assertion.

4.2.16 Catalogue_knurl

A Catalogue_knurl is a type of Knurl (see 4.2.102) that is a reference to a document containing the information to create a knurl on a part.

NOTE - Figure 9 illustrates a Catalogue_knurl that is a diagonal knurl with further definition found in a specification called ANSI B94.6.

The data associated with a Catalogue_knurl are the following:

— documentation.

©ISO 2000 - All rights reserved

4.2.16.1 documentation

The documentation specifies the document that defines information pertaining to a Knurl feature. See 4.3.12 for the application assertion.

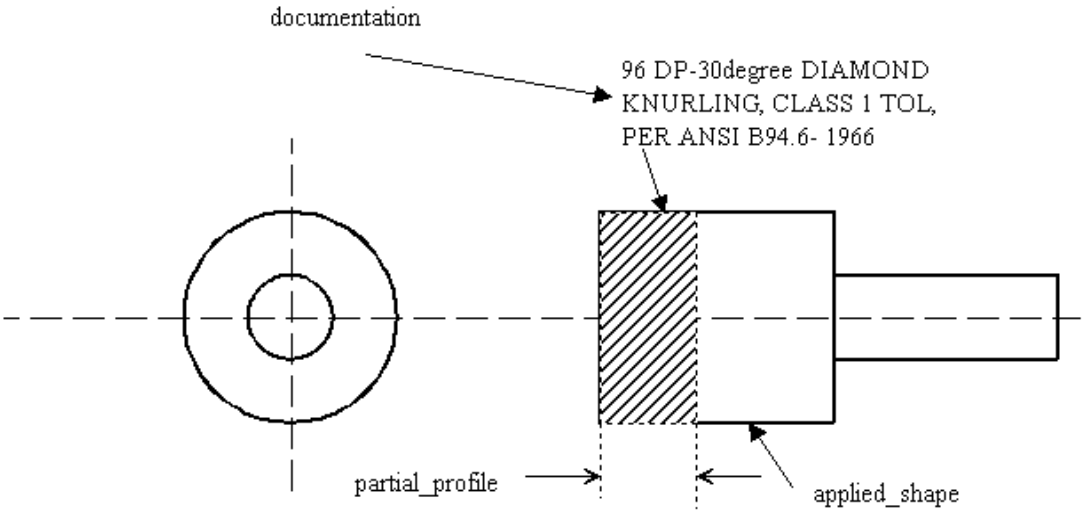


Figure 9 - Catalogue_knurl

4.2.17 Catalogue_marking

A Catalogue_marking is a type of Marking (see 4.2.116) that is a reference to a document containing the information for marking on a surface of a part.

NOTE - Figure 10 illustrates a Catalogue_marking with a text value of '1928-12345ED34' a documentation value of 'ANSI/SAE-AS478-2 ', and associated data for the applied_shape.

The data associated with a Catalogue_marking are the following:

— documentation.

4.2.17.1 documentation

The documentation specifies the document or specification that defines information pertaining to a Marking feature. See 4.3.13 for the application assertion.

4.2.18 Catalogue_thread

A Catalogue_thread is a type of Thread (see 4.2.224) that is a reference to a document containing the information to create threads on a part.

NOTE - Figure 11 illustrates a Catalogue_thread that is an outer thread with values for the documentation associated data .

 ©ISO 2000 - All rights reserved

The data associated with a Catalogue_thread are the following:

— documentation.

4.2.18.1 documentation

The documentation specifies the document that defines information pertaining to a Thread feature. See 4.3.14 for the application assertion.

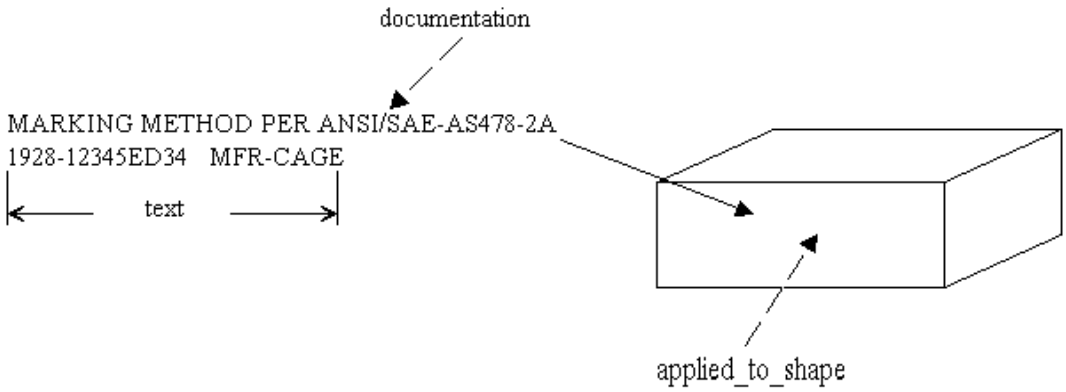


Figure 10 - Catalogue_marking

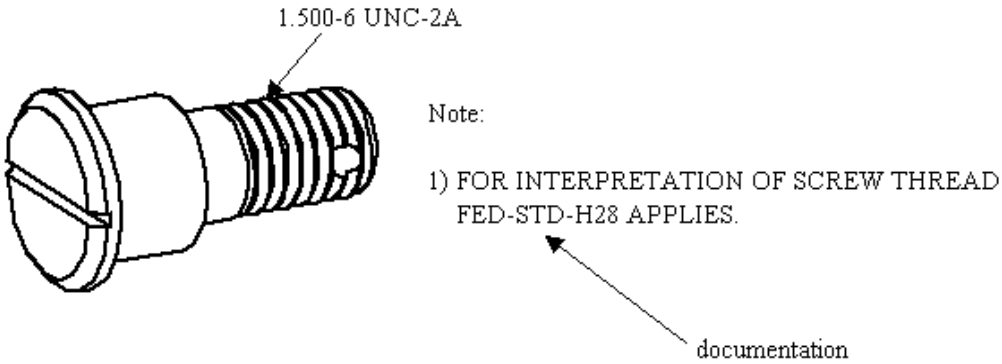


Figure 11 - Catalogue_thread

©ISO 2000 - All rights reserved

4.2.19 Chamfer

A Chamfer is a type of Transition_feature (see 4.2.234) that is a transition between corresponding edges of two joining non-coplanar surfaces, having a flat cross section. A Chamfer feature requires an offset length from one face, and either an angular amount from the same surface or an offset length from a second face.

NOTE - Figure 12 illustrates a Chamfer applied to a rectangular block and the chamfer_face associated data. The offset_amout and angle_amout attribtes apply to the First_offset, Second_offset, Second_chamfer_offset and Chamfer_angle entities.

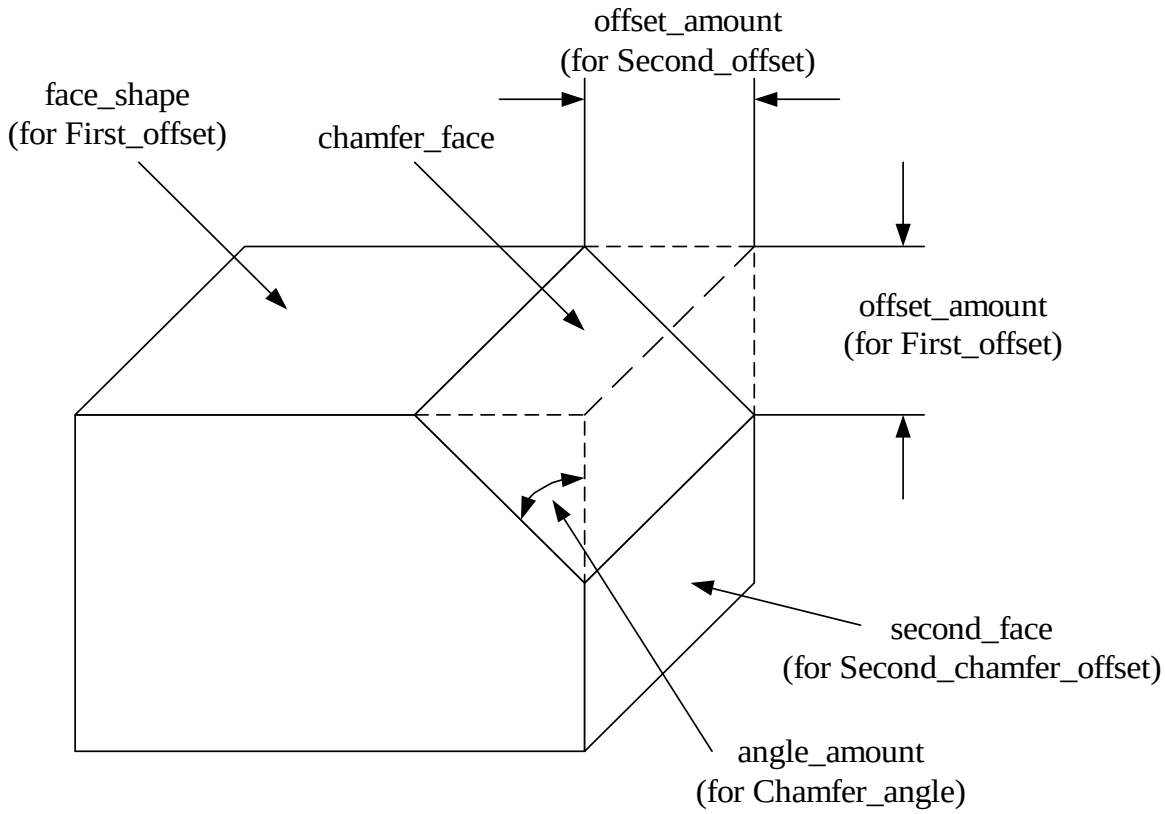


Figure 12 - Chamfer

The data associated with a Chamfer are the following:

— chamfer_face;

— first_face_offset;

— second_face_offset.

4.2.19.1 chamfer_face

The chamfer_face specifies the flat transition face between the two edges of two faces. See 4.3.15 for the application assertion.

 ©ISO 2000 - All rights reserved

4.2.19.2 first_face_offset

The first_face_offset specifies a face and offset amount for one of the edges of the Chamfer. See 4.3.16 for the application assertion.

4.2.19.3 second_face_offset

The second_face_offset specifies a face and either an offset amount or angle amount for one of the edges of the Chamfer. See 4.3.17 for the application assertion.

4.2.20 Chamfer_angle

A Chamfer_angle is a type of Second_chamfer_offset (see 4.2.196) that is the measured angle from the Second_offset face to the Chamfer face for creating a Chamfer feature.

NOTE - Figure 12 illustrates a Chamfer, a Chamfer_angle, and the angle_amount associated data.

The data associated with a Chamfer_angle are the following:

— angle_amount.

4.2.20.1 angle_amount

The Angle_amount specifies the angular measurement from a face for creating a Chamfer feature. See 4.3.18 for the application assertion.

4.2.21 Circular_boss

A Circular_boss is a type of Boss (see 4.2.10) that is a cylindrical shape. A Circular_boss may be tapered.

NOTE - Figure 13 illustrates a Circular_boss with a planar top and a fillet radius, and Figure 14 illustrates a Circular_boss with taper.

The data associated with a Circular_boss are the following:

— change_in_diameter;

— circular_profile.

4.2.21.1 change_in_diameter

The change_in_diameter specifies a taper that defines the change in shape of the Circular_boss. The change_in_diameter need not be specified for a particular Circular_boss. See 4.3.21, 4.3.22 and 4.3.20 for the application assertion.

©ISO 2000 - All rights reserved

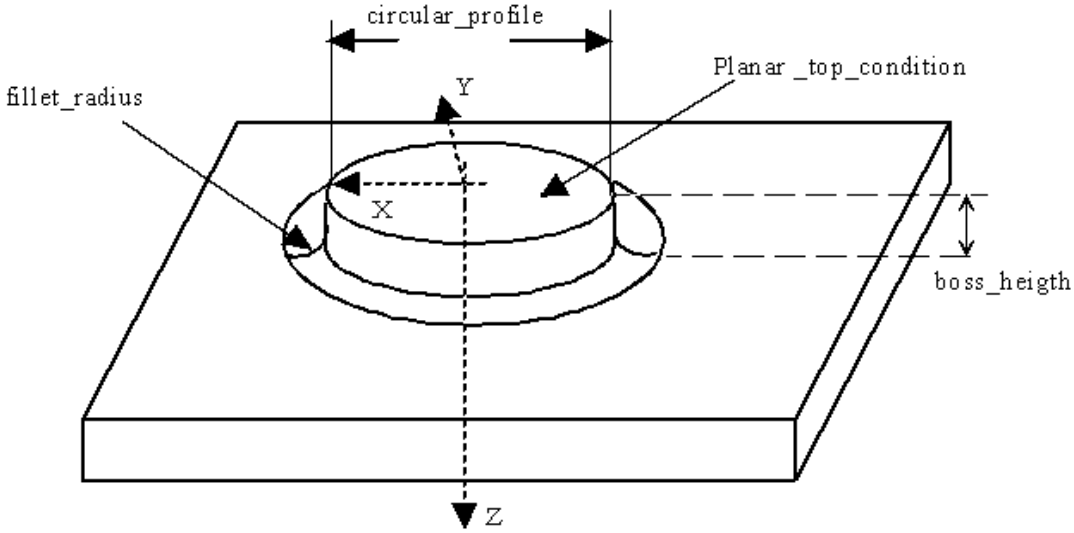


Figure 13 - Circular_boss

4.2.21.2 circular_profile

The circular_profile specifies the diameter required by a Circular_boss. The diameter is the distance across the Circular_boss. The placement of the circular_profile shall be with the origin of the Circular_-closed_profile at the origin of the Circular_boss. The X-axis and Y-axis of the profile shall be the same as the X-axis and Y-axis of the Circular_boss. See 4.3.19 for the application assertion.

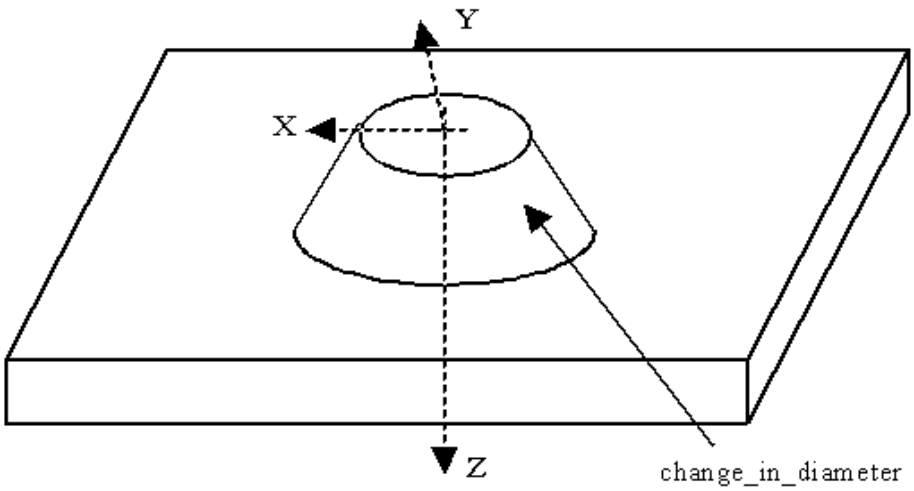


Figure 14 - Circular_boss with taper

 ©ISO 2000 - All rights reserved

4.2.22 Circular_closed_profile

A Circular_closed_profile is a type of Closed_profile (see 4.2.31) that is an enclosed area bounded by a circle. The orientation is at the center of the circle.

NOTE - Figure 15 illustrates a Circular_closed_profile.

The data associated with a Circular_closed_profile are the following:

— diameter.

4.2.22.1 diameter

The diameter specifies the distance across the Circular_closed_profile. See 4.3.23 for the application assertion.

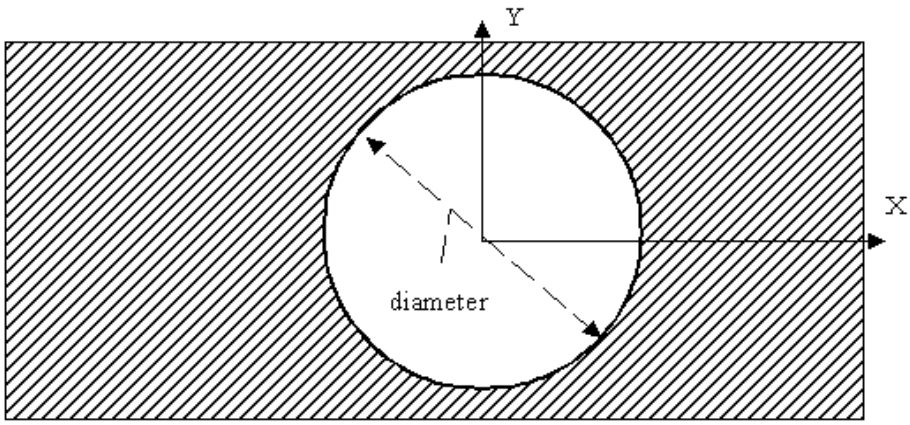


Figure 15 - Circular_closed_profile

4.2.23 Circular_closed_shape_profile

A Circular_closed_shape_profile is a type of Shape_profile (see 4.2.201) that defines a completely enclosed volume. The data associated with a Circular_closed_shape_profile are the following:

— closed_boundary.

4.2.23.1 closed_boundary

The closed_boundary specifies the outline of the Shape_profile feature. The outline defines an area that shall be enclosed and circular. The placement of the closed_profile shall be with the origin of the Path,

©ISO 2000 - All rights reserved

that defines the profile, at the origin of the Circular_closed_shape_profile. The X-axis and Y-axis of the profile shall be the same as the X-axis and Y-axis of the Circular_closed_shape_profile. See 4.3.24 for the application assertions.

4.2.24 Circular_cutout

A Circular_cutout is a type of Cutout (see 4.2.45) that is an enclosed volume of circular shape. A Circular_cutout is similar in definition to a Hole (see 4.2.99), but differ in the type of process required to manufacture. The data associated with a Circular_cutout are the following:

— circular_boundary.

4.2.24.1 circular_boundary

The circular_boundary is the distance across the Circular_cutout. The placement and orientation of the Circular_closed_profile (see 4.2.22) shall be the same as the Circular_cutout feature. See 4.3.25 for the application assertion.

4.2.25 Circular_offset_pattern

A Circular_offset_pattern is a Circular_pattern with a modification of the placement of a particular occurrence of the base feature relative to its expected placement.

NOTE - Figure 16 illustrates a Circular_offset_pattern. The rectangular base shape is offset from index number seven.

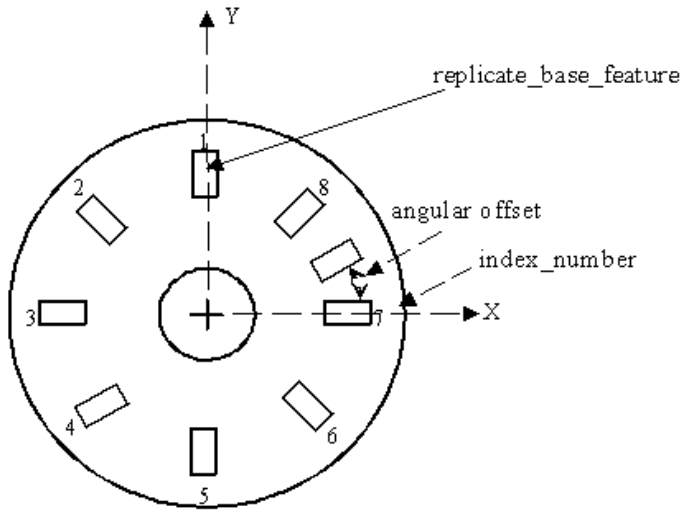


Figure 16 - Circular_offset_pattern

 ©ISO 2000 - All rights reserved

The data associated with a Circular_offset_pattern are the following:

— angular_offset;

— index_number.

4.2.25.1 angular_offset

The angular_offset is the amount of offset from the expected feature location in a Circular_pattern, along the base feature diameter for placing another feature. See 4.3.26 for the application assertion.

4.2.25.2 index_number

The index_number specifies the value for unique identification of a location of a base feature used in a Circular_pattern. See 4.3.26 for the application assertion.

4.2.26 Circular_omit_pattern

A Circular_omit_pattern is a Circular_pattern with an omission of a particular occurrence of the base feature.

NOTE - Figure 17 illustrates a Circular_omit_pattern with the omission of the rectangular base shape from index number 7.

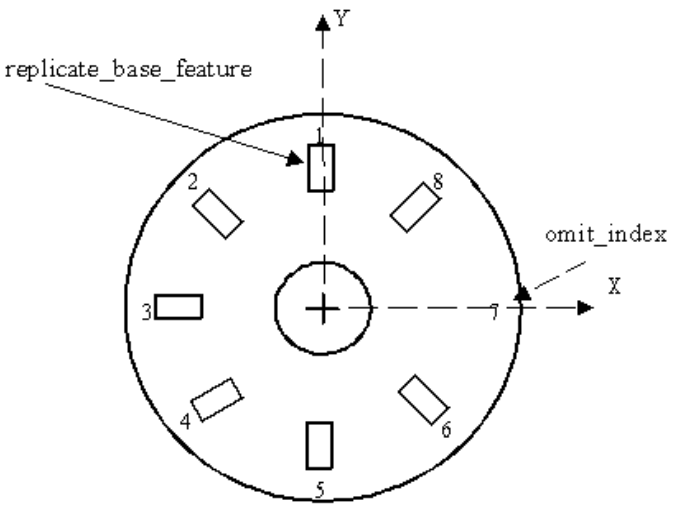


Figure 17 - Circular_omit_pattern

The data associated with a Circular_omit_pattern are the following:

— omit_index.

©ISO 2000 - All rights reserved

4.2.26.1 omit_index

The omit_index specifies the value for unique identification of the base feature to be omitted in a Circular_pattern. See 4.3.27 for the application assertion.

4.2.27 Circular_path

A Circular_path is a type of Path (see 4.2.146) that is a direction of travel along an arc of constant radius around the Z-axis of the feature. Each Circular_path is either a Complete_circular_path (see 4.2.32) or a Partial_circular_path (see 4.2.143). The data associated with a Circular_path are the following:

— radius.

4.2.27.1 radius

The radius specifies the constant distance from an axis for the Circular_path. See 4.3.28 for the application assertion.

4.2.28 Circular_pattern

A Circular_pattern is a type of Replicate_feature (see 4.2.185) that is a base feature arranged in a pattern around a circular arc, equally spaced about an axis. Instances of base features need not be rotated. When a base feature rotation is required, the first feature instance is not rotated. The second through the Nth feature instance are rotated by the same angular amount measured from the placement and orientation of the preceding feature instance. (This does not imply the use of chain dimensioning.)

NOTE - Figure 18 illustrates a Circular_pattern with rotation of a base feature, and Figure 19 illustrates a Circular_-pattern without rotation.

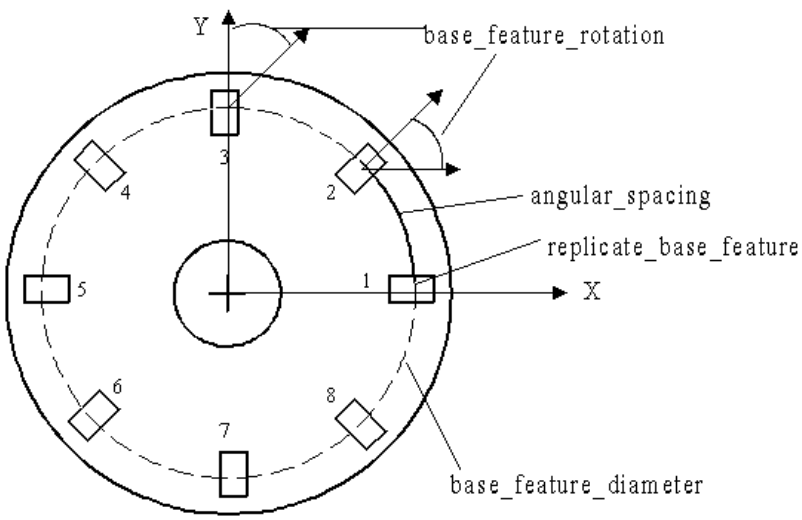


Figure 18 - Circular_pattern with rotation of base feature

 ©ISO 2000 - All rights reserved

The data associated with a Circular_pattern are the following:

— angular_spacing;

— base_feature_diameter;

— base_feature_rotation;

— missing_base_feature;

— number_of_features;

— relocated_base_feature.

4.2.28.1 angular_spacing

The angular_spacing specifies the angle amount between features in a Circular_pattern. See 4.3.31 for the application assertion.

4.2.28.2 base_feature_diameter

The base_feature_diameter specifies the size of the circle for placement of features in a Circular_pattern. See 4.3.31 for the application assertion.

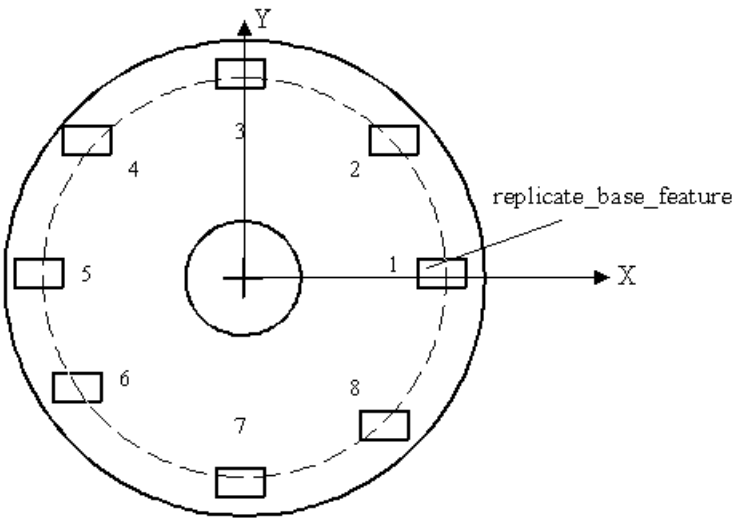


Figure 19 - Circular_pattern without rotation of base feature

©ISO 2000 - All rights reserved

4.2.28.3 base_feature_rotation

The base_feature_rotation specifies the angle to rotate one feature in regard to the orientation of a previous feature in a Circular_pattern. See 4.3.31 for the application assertion.

4.2.28.4 missing_base_feature

The missing_base_feature specifies the definition to remove any number of base features from the Circular_pattern. The missing_base_feature need not be specified for a particular Circular_pattern. There may be more than one missing_base_feature for a Circular_pattern. See 4.3.30 for the application assertion.

4.2.28.5 number_of_features

The number_of_features specifies the quantity of a base feature to be used in a Circular_pattern. See 4.3.31 for the application assertion.

4.2.28.6 relocated_base_feature

The relocated_base_feature specifies the definition to offset any number of base features from the Circular_pattern. The relocated_base_feature need not be specified for a particular Circular_pattern. There may be more than one relocated_base_feature for a Circular_pattern. See 4.3.29 for the application assertion.

4.2.29 Circular_runout_tolerance

A Circular_runout_tolerance is a type of Geometric_tolerance (see 4.2.95) that is a tolerance to control circular elements of a part to a datum axis the part is rotated about. The tolerance is applied independent of any circular position as the part is rotated 360 degrees. Where applied to surfaces around a datum axis, Circular_runout_tolerance may be used to control the cumulative variations of circularity and coaxiality. Where applied to surfaces at right angles to the datum axis, it controls circular elements of a plane surface.

NOTE - Figure 20 illustrates a Circular_runout_tolerance.

The data associated with a Circular_runout_tolerance are the following:

— geometric_reference;

— runout_angle.

4.2.29.1 geometric_reference

The geometric_reference specifies the datum to which the tolerance is related. See 4.3.32 for the application assertion.

 ©ISO 2000 - All rights reserved

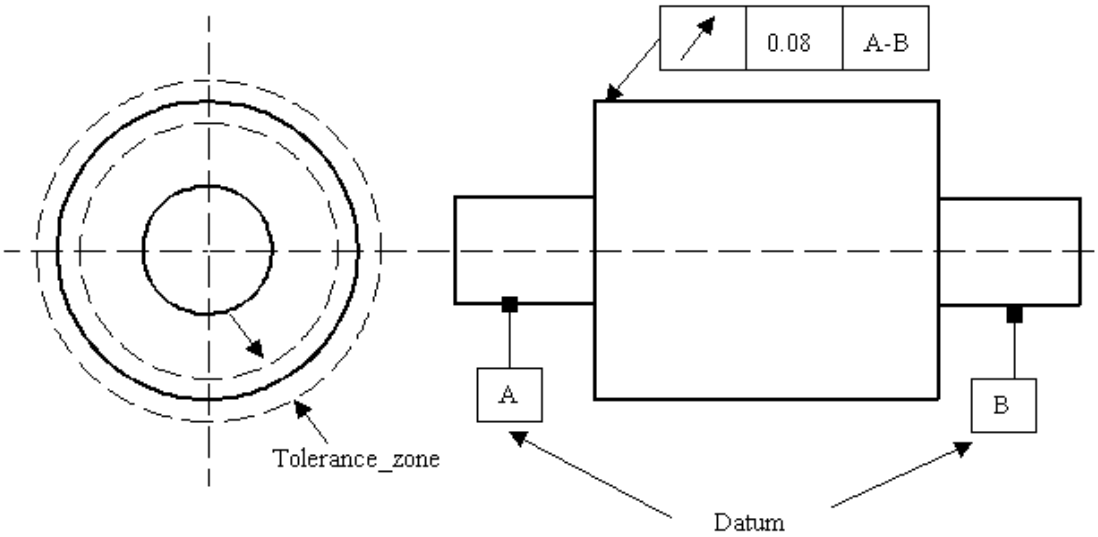


Figure 20 - Circular_runout_tolerance

4.2.29.2 runout_angle

The runout_angle specifies the direction to control a runout tolerance. If the angle is specified, the runout tolerance applies in this angle which is fixed with respect to the datum axis. The runout need not be specified for a particular Circular_runout_tolerance.

NOTE - The Circular_runout_tolerance definition is derived from paragraph 14.13 of ISO 1101.

4.2.30 Circularity_tolerance

A Circularity_tolerance is a type of Geometric_tolerance (see 4.2.95) that describes the allowable deviation of a surface from round. The actual surface shall lie within a tolerance zone defined by two concentric circles.

NOTE 1 - Figure 21 illustrates Circularity_tolerance.

NOTE 2 - The Circularity_tolerance definition is derived from paragraph 14.3 of ISO 1101.

4.2.31 Closed_profile

A Closed_profile is a type of Profile (see 4.2.164) that is an outline or shape that bounds an enclosed area with no opening. Each Closed_profile is either a Circular_closed_profile (see 4.2.22), General_closed_-profile (see 4.2.81), Ngon_profile (see 4.2.127), or a Rectangular_closed_profile (see 4.2.177).

©ISO 2000 - All rights reserved

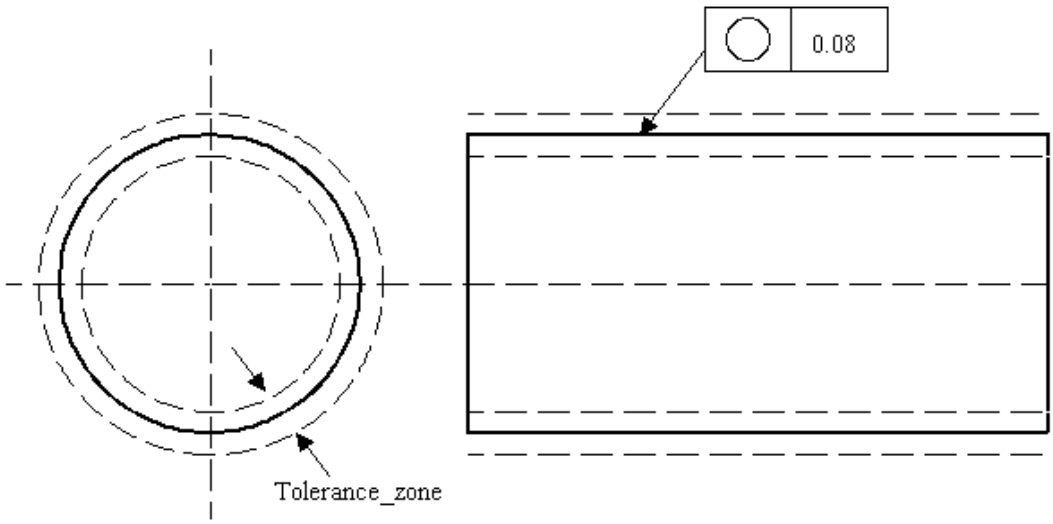


Figure 21 - Circularity_tolerance

4.2.32 Complete_circular_path

A Complete_circular_path is a type of Circular_path (see 4.2.27) that is a direction of travel that begins and ends at the same point on the arc.

NOTE - Figure 22 illustrates a Slot feature with a Square_u_profile and a Complete_circular_path.

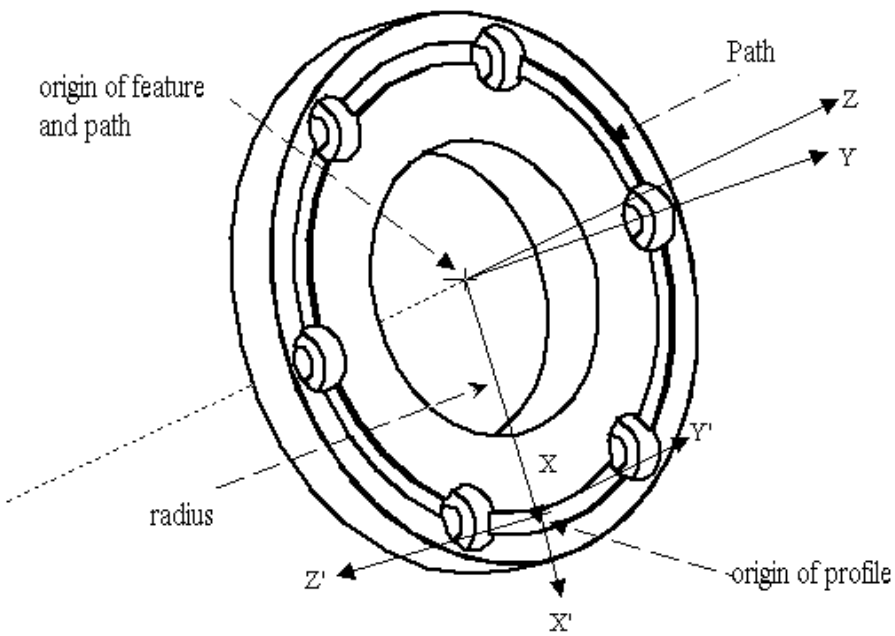


Figure 22 - Complete_circular_path

 ©ISO 2000 - All rights reserved

4.2.33 Compound_datum

A Compound_datum is a type of Datum (see 4.2.49) that is a set of two or more Datum_feature objects which are for establishing a single datum plane or axis. The data associated with a Compound_datum are the following:

— element.

4.2.33.1 element

The element specifies the list of Datum_feature objects for defining the Compound_datum. See 4.3.33 for the application assertion.

4.2.34 Compound_feature

A Compound_feature is a type of Machining_feature (see 4.2.111) that is a union of one or more Machining_feature objects to create a more complex feature definition. The placement of a Compound_feature is relative to either the part, another Compound_feature, or a Replicate_feature which uses a Compound_feature as the base feature. Features which are elements of the Compound_feature have placement defined relative to the Compound_feature placement.

NOTE - Figure 23 illustrates an example of a Compound_feature with one counterbore_hole feature and one countersunk_hole feature.

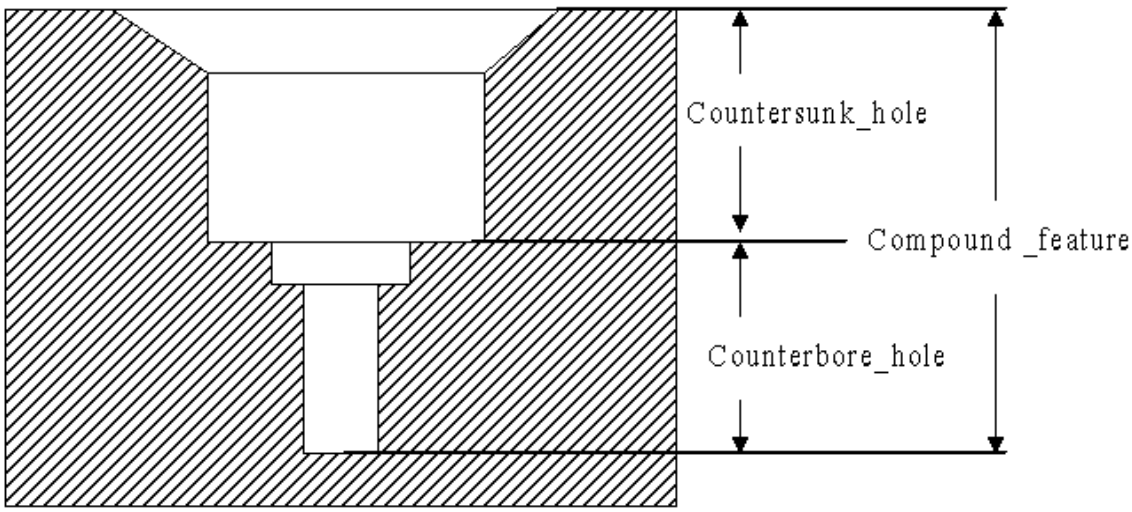


Figure 23 - Compound_feature

©ISO 2000 - All rights reserved

The data associated with a Compound_feature are the following:

— element;

— feature_description;

— feature_name.

4.2.34.1 element

The element specifies the base features that when combined defines a Compound_feature. There shall be more than one element for a Compound_feature. See 4.3.34 for the application assertion.

4.2.34.2 feature_description

The feature_description specifies a user defined explanation of the Compound_feature. A Compound_-feature may but need not require a feature_description .

4.2.34.3 feature_name

The feature_name specifies a user defined name for the Compound_feature that need not be unique.

4.2.35 Compound_feature_element

A Compound_feature_element specifies the type of Machining_feature (see 4.2.111) or Transition_feature (see 4.2.234) to be used for a particular element of a Compound_feature. A Compound_feature consists of two or more Compound_feature_element objects and need not require Compound_feature_element objects to be ordered. The data associated with a Compound_feature_element are the following:

— element.

4.2.35.1 element

The element specifies the base feature to be used as one of the components for the Compound_feature. See 4.3.35 and 4.3.36 for the application assertions.

4.2.36 Compound_feature_relationship

A Compound_feature_relationship defines the sequence in which the Compound_feature elements are applied in the Compound_feature. The Compound_feature_relationship defines which feature is the preceding feature and which is the succeeding feature.

The data associated with a Compound_feature_relationship are the following:

— predecessor;

— successor.

 ©ISO 2000 - All rights reserved

4.2.36.1 predecessor

The predecessor specifies the Compound_feature_element with the highest precedence. See 4.3.37 for the application assertion.

4.2.36.2 successor

The successor specifies Compound_feature_element with a lesser precedence. See 4.3.37 for the application assertion.

4.2.37 Concentricity_tolerance

A Concentricity_tolerance is a type of Geometric_tolerance (see 4.2.95) that is a cylindrical or conical feature of a part which shall be fundamentally concentric. When the part is rotated about the datum axis, the axis of the feature shall be within the cylindrical tolerance zone.

NOTE 1 - Figure 24 illustrates the Concentricity_tolerance

NOTE 2 - The Concentricity_tolerance definition is derived from paragraph 14.11 of ISO 1101.

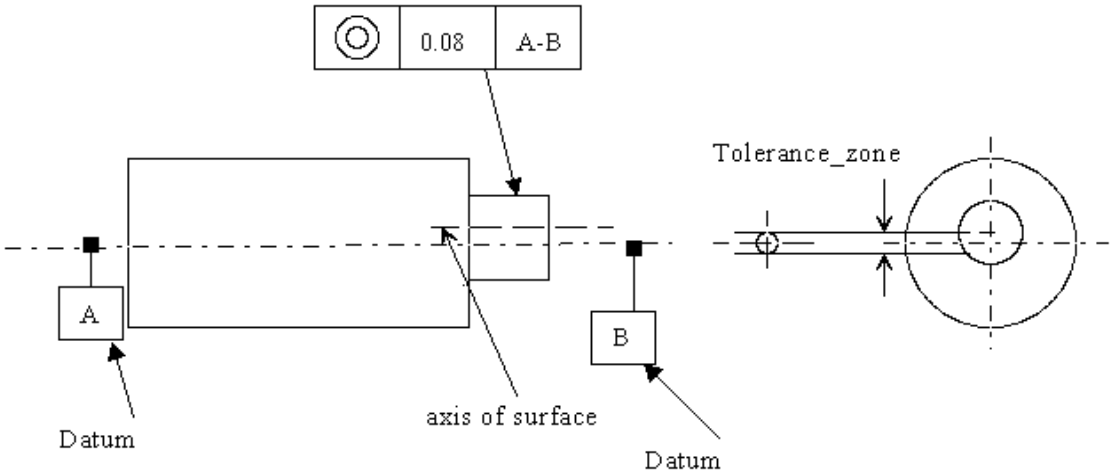


Figure 24 - Concentricity_tolerance

The data associated with a Concentricity_tolerance are the following:

— geometric_reference;

— value_qualifier.

4.2.37.1 geometric_reference

The geometric_reference specifies the datum to which the tolerance is related. See 4.3.38 for the application assertion.

©ISO 2000 - All rights reserved

4.2.37.2 value_qualifier

The value_qualifier specifies whether the Concentricity_tolerance is either a diametric tolerance or is not.

4.2.38 Conical_hole_bottom

A Conical_hole_bottom is a type of Blind_bottom_condition (see 4.2.8) that defines the bottom of a Round_hole to be conical in shape. A Conical_hole_bottom shall be a constant decrease in the hole diameter until the radius is zero. The Conical_hole_bottom may have a tip_radius defined at the smallest end of the Conical_hole_bottom.

NOTE - Figure 25 illustrates the Conical_hole_bottom for a Round_hole.

The data associated with a Conical_hole_bottom are the following:

— tip_angle;

— tip_radius.

4.2.38.1 tip_angle

The tip_angle specifies the amount of constant slope to decrease the Round_hole diameter until the diameter is zero. The tip_angle is a conical bottom for a Round_hole. The tip_angle shall be greater than 0 degrees and less than 180 degrees. See 4.3.39 for the application assertion.

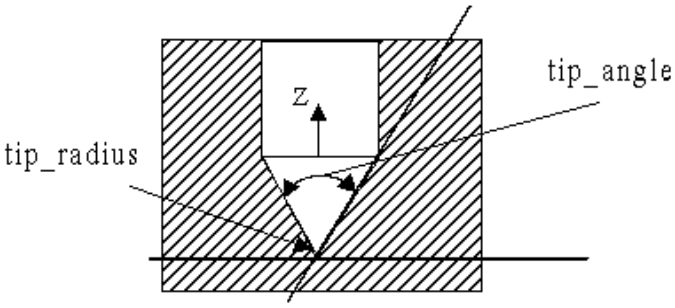


Figure 25 - Conical_hole_bottom

4.2.38.2 tip_radius

The tip_radius specifies the radius of a conical bottom for a Round_hole. A conical bottom is a constant decrease in diameter until the diameter is zero, or until it becomes tangent to a tip_radius. The tip_radius need not be specified for a particular Conical_hole_bottom. See 4.3.39 for the application assertion.

 ©ISO 2000 - All rights reserved

4.2.39 Constant_radius_edge_round

A Constant_radius_edge_round is a type of Edge_round (see 4.2.67) that is defined with a constant radius value.

NOTE - Figure 26 illustrates the Constant_radius_edge_round on a rectangular block.

The data associated with a Constant_radius_edge_round are the following:

— first_face_offset;

— second_face_offset;

— radius.

4.2.39.1 first_face_offset

The first_face_offset specifies the amount of offset from the first_face_shape (see 4.2.63.2) to the center of the edge_round_face (see 4.2.63.1). A constant_radius_edge_round may but need not require a first_-face_offset. See 4.3.40 for the application assertion.

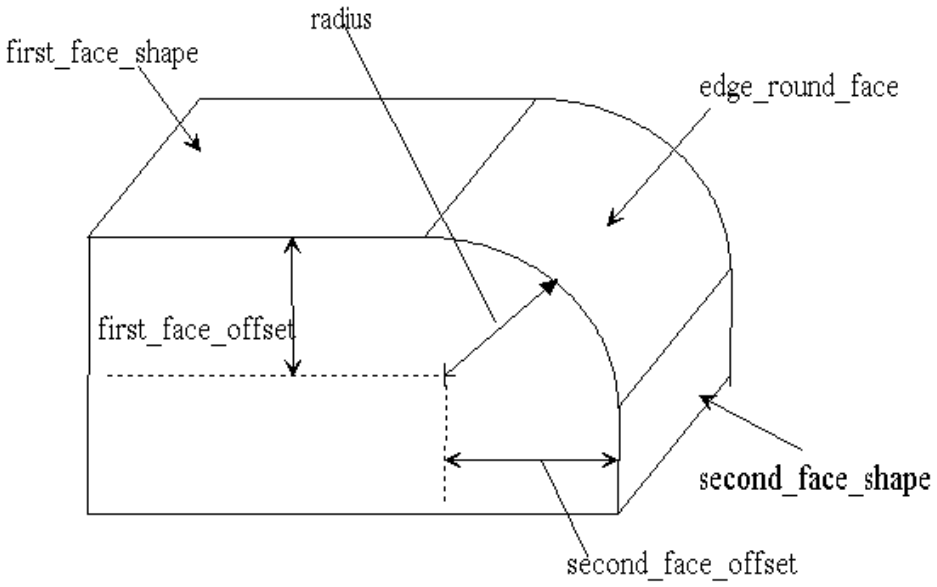


Figure 26 - Constant_radius_edge_round

©ISO 2000 - All rights reserved

4.2.39.2 second_face_offset

The second_face_offset specifies the amount of offset from the second_face_shape (see 4.2.63.3) to the center of the edge_round_face (see 4.2.63.1). A constant_radius_edge_round may but need not require a second_face_offset. See 4.3.40 for the application assertion.

4.2.39.3 radius

The radius specifies the amount of curvature for a convex transition between the two faces of a Constant_radius_edge_round. See 4.3.40 for the application assertion.

4.2.40 Constant_radius_fillet

A Constant_radius_fillet is a type of Fillet (see 4.2.73) that is defined with a constant radius value.

NOTE - Figure 27 illustrates the Constant_radius_fillet. The Fillet surface may not be tangent to the other surfaces because the offsets.

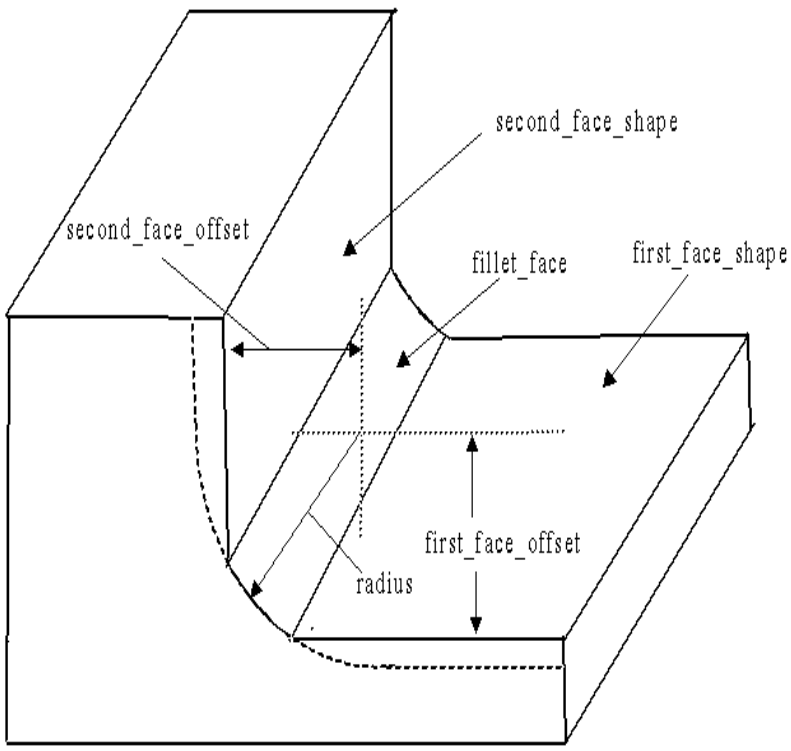


Figure 27 - Constant_radius_fillet

 ©ISO 2000 - All rights reserved

The data associated with a Constant_radius_fillet are the following:

— first_face_offset;

— second_face_offset;

— radius.

4.2.40.1 first_face_offset

The first_face_offset specifies the amount of offset from the first_face_shape (see 4.2.69.2) to the center of the fillet_face (see 4.2.69.1). A constant_radius_fillet may but need not require a first_face_offset. See 4.3.41 for the application assertion.

4.2.40.2 second_face_offset

The second_face_offset specifies the amount of offset from the second_face_shape (see 4.2.69.3) to the center of the fillet_face (4.2.69.1). A constant_radius_fillet may but need not require a first_face_offset. See 4.3.41 for the application assertion.

4.2.40.3 radius

The radius specifies the amount of curvature for a concave transition between two surfaces. See 4.3.41 for the application assertion.

4.2.41 Counterbore_hole

A Counterbore_hole is a type of Hole (see 4.2.99) that is a combination of two Round_holes. The first Round_hole shall have either a Through_bottom_condition or Blind_bottom_condition, the second shall have a Blind_bottom_condition, and a larger diameter than the first Round_hole. The top of the first Round_hole shall mate with the bottom of the second Round_hole. The Counterbore_hole orientation shall be the same as the orientation of the first Round_hole. Both Round_holes shall be co-axial.

NOTE - Figure 28 illustrates the Counterbore_hole with a Blind_bottom_condition and a Through_bottom_-condition.

The data associated with a Counterbore_hole are the following:

— larger_hole;

— smaller_hole.

4.2.41.1 larger_hole

The larger_hole specifies the Round_hole that will be used as the larger hole for the Counterbore_hole. See 4.3.42 for the application assertion.

©ISO 2000 - All rights reserved

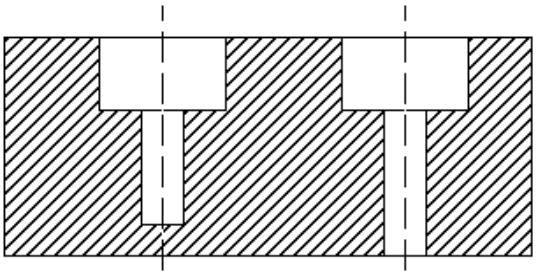


Figure 28 - Counterbore_hole

4.2.41.2 smaller_hole

The smaller_hole specifies the Round_hole that will be used as the smaller hole for the Counterbore_hole. See 4.3.42 for the application assertion.

4.2.42 Countersunk_hole

A Countersunk_hole is a type of Hole (see 4.2.99) that is a combination of two Round_holes. The first Round_hole shall have a Through_bottom_condition or Blind_bottom_condition. The second shall be a Round_hole with a Blind_bottom_condition, and a taper. The top of the first Round_hole shall mate with the bottom of the second Round_hole. The taper of the second Round_hole shall be larger than the diameter of the Round_hole, decreasing to the same diameter at the point where the two holes join. The Countersunk_hole orientation shall be the same as the orientation of the first Round_hole. Both Round_-holes shall be co-axial.

NOTE - Figure 29 illustrates the Countersunk_hole with a Blind_bottom_condition and a Through_-bottom_condition.

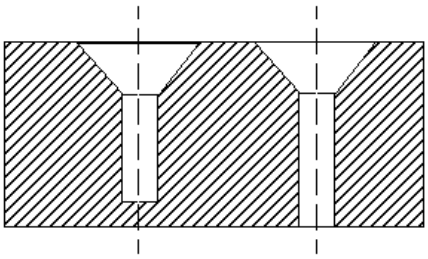


Figure 29 - Countersunk_hole

 ©ISO 2000 - All rights reserved

The data associated with a Countersunk_hole are the following:

— constant_diameter_hole;

— tapered_hole.

4.2.42.1 constant_diameter_hole

The constant_diameter specifies the Round_hole without a taper that will be used as the constant diameter hole for the Countersunk_hole. See 4.3.43 for the application assertion.

4.2.42.2 tapered_hole

The tapered_hole specifies the Round_hole with a taper that will be used as the tapered hole for the Counterbore_sunk. See 4.3.43 for the application assertion.

4.2.43 Curved_dimension_tolerance

A Curved_dimension_tolerance is a type of Size_tolerance (see 4.2.204) that is the tolerance on a dimension for a curve measured along the entire path of the curve.

NOTE - Figure 30 illustrates the Curved_dimension_tolerance.

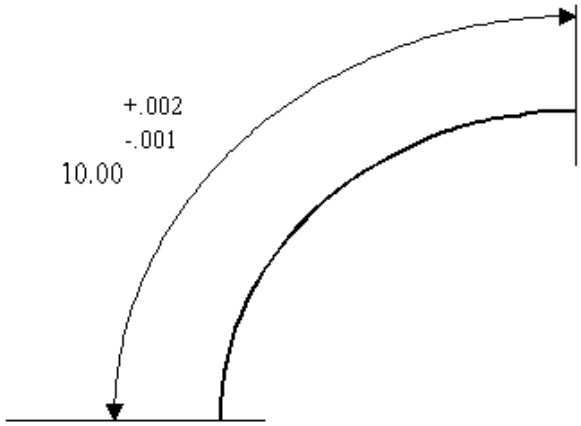


Figure 30 - Curved_dimension_tolerance

©ISO 2000 - All rights reserved

4.2.44 Customer_order

A Customer_order is a document, received from a customer, that describes data to process a request for the manufacture of a part. The data associated with a Customer_order are the following:

— customer;

— delivery_date;

— initiated_order;

— material_disposition;

— order_number;

— order_status;

— quantity_ordered;

— special_instructions.

4.2.44.1 customer

The customer specifies the person and company requesting the manufacture of the part. See 4.3.45 for the application assertion.

4.2.44.2 delivery_date

The delivery_date specifies the year, month and day when the part is to be received by the customer.

4.2.44.3 initiated_order

The initiated_order specifies Project_order that is initiated by the Customer_order. There may be more than one initiated_order for a Customer_order. See 4.3.46 for the application assertion.

4.2.44.4 material_disposition

The material_disposition specifies the instructions for handling of material that has been delivered to manufacture a part.

4.2.44.5 order_number

The order_number specifies a unique identification of the Customer_order.

4.2.44.6 order_status

The order_status specifies the disposition of the Customer_order. The disposition need not be unique.

 ©ISO 2000 - All rights reserved

4.2.44.7 quantity_ordered

The quantity_ordered specifies number of parts to be manufactured (see Ordered_part 4.2.132). There may be more than one quantity_ordered for a Customer_order. See 4.3.44 for the application assertion.

4.2.44.8 special_instructions

The special_instructions specifies directions for processing of a Customer_order that apply to manufacturing a part which are not common and need to be described and documented.

4.2.45 Cutout

A Cutout is a type of Pocket (see 4.2.160) that is a volume to be removed from the part. Cutouts shall pass through two faces of a Part. Each Cutout is either a Circular_cutout (see 4.2.24)or a General_cutout (see 4.2.82).

NOTE - Figure 31 illustrates the Cutout.

The data associated with a Cutout are the following:

— bottom_condition.

4.2.45.1 bottom_condition

The bottom_condition specifies the shape of the bottom of a Cutout feature. The bottom_condition shall pass entirely through the part, it shall be a Through_pocket_bottom_condition (see 4.2.226). See 4.3.47 for the application assertion.

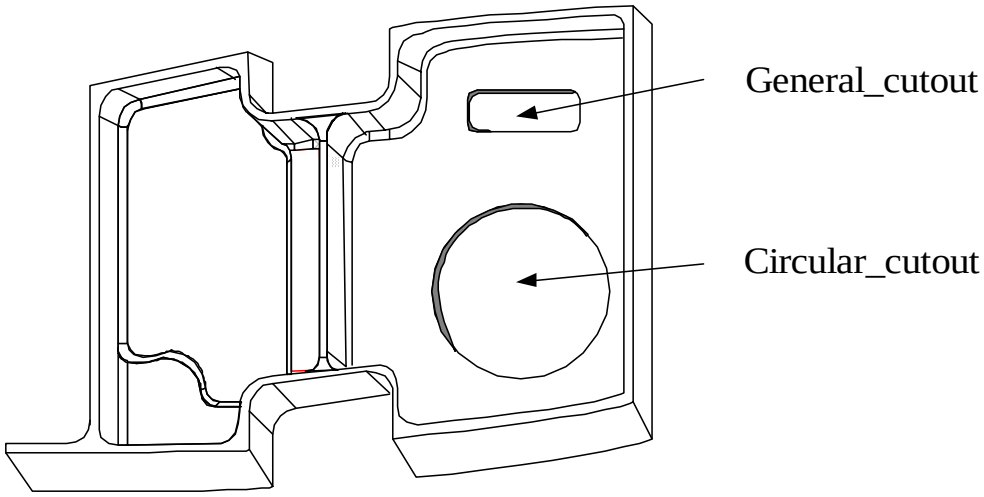


Figure 31 - Cutout

©ISO 2000 - All rights reserved

4.2.46 Cutting_tool_requisition

A Cutting_tool_requisition is a type of Requisition (see 4.2.186) that is an organization's requirements for the purchase of tools required to support part manufacturing.

4.2.47 Cylindrical_base_shape

A Cylindrical_base_shape is a type of Implicit_base_shape_representation (see 4.2.100) that is the initial shape of the material which is cylindrical.

NOTE - Figure 32 illustrates a Cylindrical_base_shape.

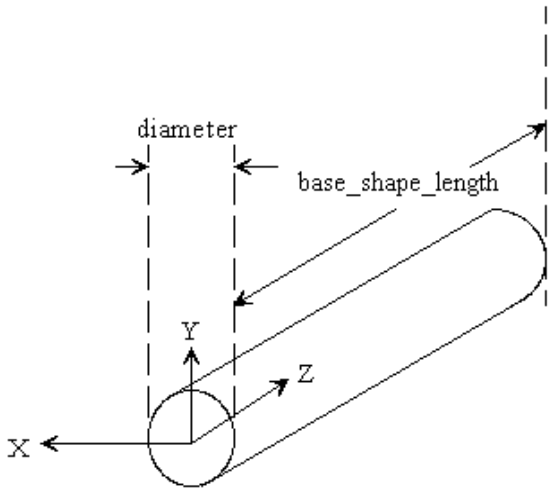


Figure 32 - Cylindrical_base_shape

The data associated with a Cylindrical_base_shape are the following:

— diameter.

4.2.47.1 diameter

The diameter specifies the distance across a Cylindrical_base_shape. See 4.3.48 for the application assertion.

 ©ISO 2000 - All rights reserved

4.2.48 Cylindricity_tolerance

A Cylindricity_tolerance is a type of Geometric_tolerance (see 4.2.95) that describes the amount of deviation a feature may have from being truly cylindrical. The feature shall be contained between two coaxial cylinders. The distance between the two coaxial cylinders defines the allowable tolerance deviation.

NOTE 1 - Figure 33 illustrates Cylindricity_tolerance.

NOTE 2 - The Cylindricity_tolerance definition is derived from paragraph 14.4 of ISO 1101.

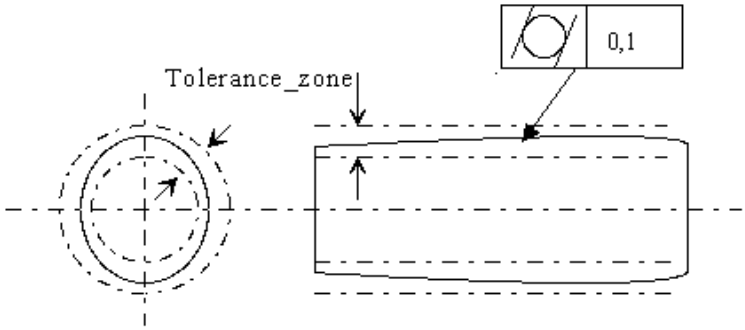


Figure 33 - Cylindricity_tolerance

4.2.49 Datum

A Datum is defined in clause 3.1 of ISO 5459. A Datum is either a Datum_feature or a Compound_-datum. The data associated with a Datum are the following:

— name;

— precedence.

4.2.49.1 name

The name specifies a word by which a Datum is commonly called. This name shall be unique within a part.

4.2.49.2 precedence

The precedence specifies a sequence that datums are applied when there are two or more datums required for the definition of a Geometric_tolerance.

4.2.50 Datum_feature

A Datum_feature is a type of Datum (see 4.2.49) that is a feature on a part used to establish a Datum.

©ISO 2000 - All rights reserved

The data associated with a Datum_feature are the following:

— datum_representation;

— modifier.

4.2.50.1 datum_representation

The datum_representation specifies the type of representation for defining a Datum_feature. The Datum_feature may be defined by either a Datum_target definition or by Part shape representation. See 4.3.50 and 4.3.51 for the application assertions.

4.2.50.2 modifier

The modifier specifies the tolerance value applied to the Datum_feature. The modifier need not be specified for a particular Datum_feature. See 4.3.49 for the application assertion.

4.2.51 Datum_target

A Datum_target is a geometric element on the surface of a part to locate a Datum for reference by a Geometric_tolerance. A Datum_target is either a Placed_target (see 4.2.152) or a Target_area (see 4.2.218). The data associated with a Datum_target are the following:

— identifier.

4.2.51.1 identifier

The identifier specifies a unique identification for the Datum_target.

4.2.52 Datum_target_set

A Datum_target_set is a set of Datum_target objects that are used to define a datum reference for a Geometric_tolerance . The data associated with a Datum_target_set are the following:

— rule_description;

— target_shape.

4.2.52.1 rule_description

The rule_description specifies the type of datum that is formed by the Datum_target_set The rule_-description need not be specified for a particular Datum_target_set.

EXAMPLE - "V-block" indicates that two Datum_target objects on a cylindrical element are to form the areas of contact on a V-shaped fixture.

 ©ISO 2000 - All rights reserved

4.2.52.2 target_shape

The target_shape specifies the set of Datum targets that define the Datum_target_set. There may be more than one target_shape for a Datum_target. See 4.3.52 for the application assertion.

4.2.53 Dedicated_fixture_requisition

A Dedicated_fixture_requisition is a type of Requisition (see 4.2.186) for dedicated fixtures required to support part manufacturing.

4.2.54 Defined_marking

A Defined_marking is a type of Marking (see 4.2.116) that is specified explicitly, all attributes for a Marking are declared and defined.

NOTE - Figure 34 illustrates the Defined_marking on a rectangular block shape and with text of 'TEXT FOR A MARKING'.

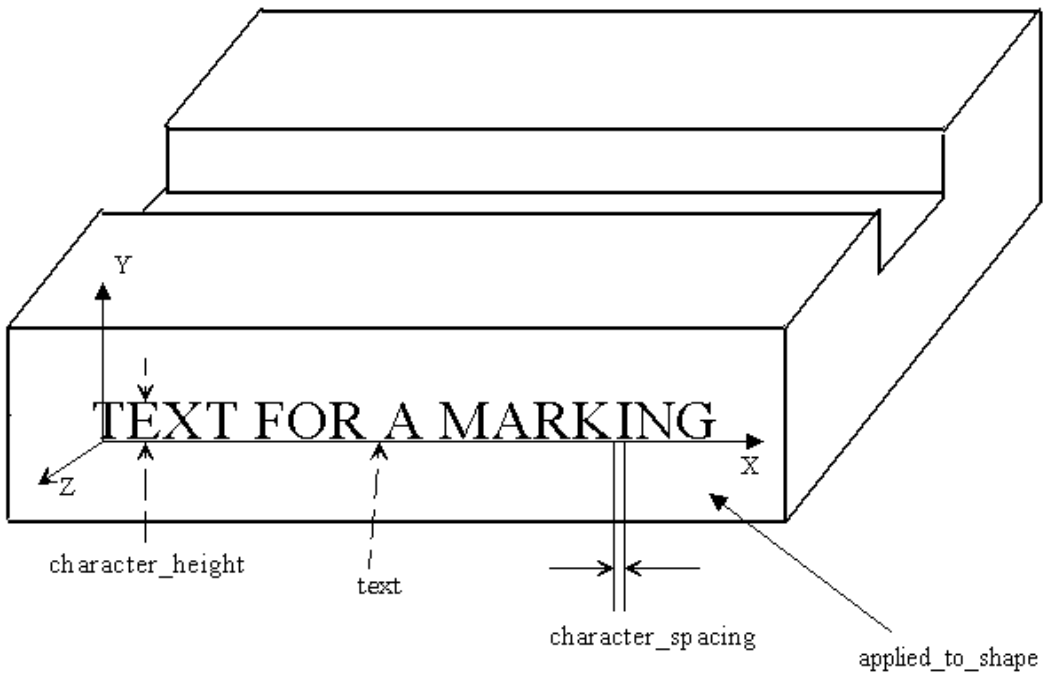


Figure 34 - Defined_marking

The data associated with a Defined_marking are the following:

— character_height;

— character_spacing;

©ISO 2000 - All rights reserved

— font_name;

— special_instructions.

4.2.54.1 character_height

The character_height specifies the size of the text used for a Defined_marking feature. See 4.3.53 for the application assertion.

4.2.54.2 character_spacing

The character_spacing specifies the amount of space between text letters used for a Defined_marking feature. See 4.3.54 for the application assertion.

4.2.54.3 font_name

The font_name specifies the appearance of the characters. A font consists of typeface, treatment, and size. See 4.3.53 for the application assertion.

EXAMPLE - Example of font_name characteristic is type face Times Roman. Example of font_name characteristic is a treatment of Bold or Italic. Example of font_name characteristic is size 10 point.

4.2.54.4 special_instructions

The special_instructions specifies a description on how to apply the text given by the Defined_marking entity. See 4.3.53 for the application assertion.

4.2.55 Defined_thread

A Defined_thread is a type of Thread (see 4.2.224) that is specified explicitly.

NOTE - Figure 35 illustrates a Defined_thread. The drawing note '1.500-6 UNC-2A' applies to the thread and is separated into several of the Defined_thread attributes.

The data associated with a Defined_thread are the following:

— crest;

— minor_diameter;

— pitch_diameter.

4.2.55.1 crest

The crest specifies the distance between the opposing points of the thread. The crest is formed by the intersection of the sides of the thread if extended, if necessary, beyond the top of the thread. The crest need not be specified for a particular Defined_thread. See 4.3.55 for the application assertion.

 ©ISO 2000 - All rights reserved

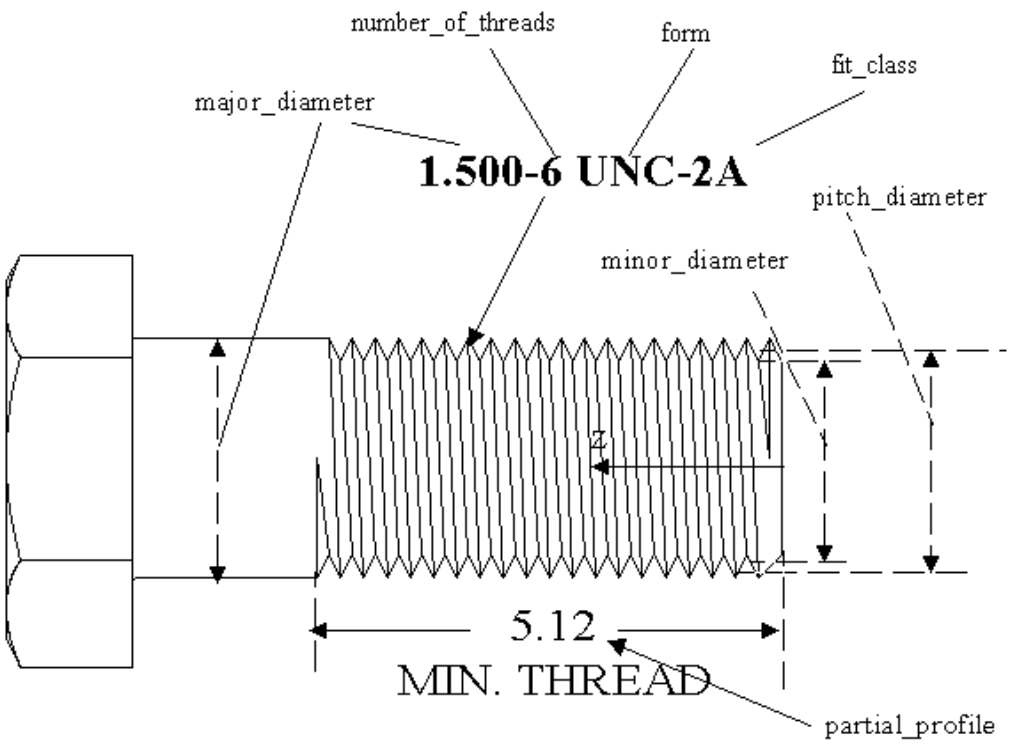


Figure 35 - Thread and Defined_thread attributes

4.2.55.2 minor_diameter

The minor_diameter specifies the dimension of the smallest diameter of the Defined_thread and is applied to both an internal and an external thread. The minor_diameter need not be specified for a particular Defined_thread. See 4.3.55 for the application assertion.

4.2.55.3 pitch_diameter

The pitch_diameter specifies the dimension of an imaginary cylinder passing through the threads so as to make equal the widths of the threads and the widths of the spaces cut by the cylinder. See 4.3.55 for the application assertion.

4.2.56 Descriptive_parameter

A Descriptive_parameter is a type of Property_parameter (see 4.2.170) that is an explanation of the property being defined by a specification. The data associated with a Descriptive_parameter are the following:

— descriptive_string.

©ISO 2000 - All rights reserved

4.2.56.1 descriptive_string

The descriptive_string specifies a word or group of words by which a Descriptive_parameter is explained.

4.2.57 Design_exception_notice

A Design_exception_notice is a notification of a design discrepancy discovered during the creation of the process plan for a given part. Process planning cannot continue until a technical recommendation is made to correct the problem. The data associated with a Design_exception_notice are the following:

— discrepant_part;

— issues;

— issuing_date;

— notice_description;

— notice_number;

— technical_recommendation.

4.2.57.1 discrepant_part

The discrepant_part specifies the set of parts that has a design discrepancy. There may be more than one discrepant_part for a Design_exception_notice. See 4.3.57 for the application assertion.

4.2.57.2 issues

The issues specifies the set of change proposals to modify a Part. The result need not be specified for a particular Design_exception_notice. There may be more than one result for a Design_exception_notice. See 4.3.56 for the application assertion.

4.2.57.3 issuing_date

The issuing_date specifies the year, month and day when the design_exception_notice was created.

4.2.57.4 notice_description

The notice_description specifies the kind of problem or non-conformance machining condition causing a rejection of a part.

EXAMPLE - If two holes were drilled simultaneously and the drill bits would run together, a recommendation would be needed to either change the depth of the hole or change the machining process.

 ©ISO 2000 - All rights reserved

4.2.57.5 notice_number

The notice_number specifies a unique identification for each Design_exception_notice.

4.2.57.6 technical_recommendation

The technical_recommendation specifies a recommended resolution to a design problem discovered during the creation of the process plan for a part.

4.2.58 Diagonal_knurl

A Diagonal_knurl is a type of Turned_knurl (see 4.2.235) with helical cuts at an angle about the axis of a surface.

NOTE - Figure 36 illustrates a Diagonal_knurl with a right hand helix.

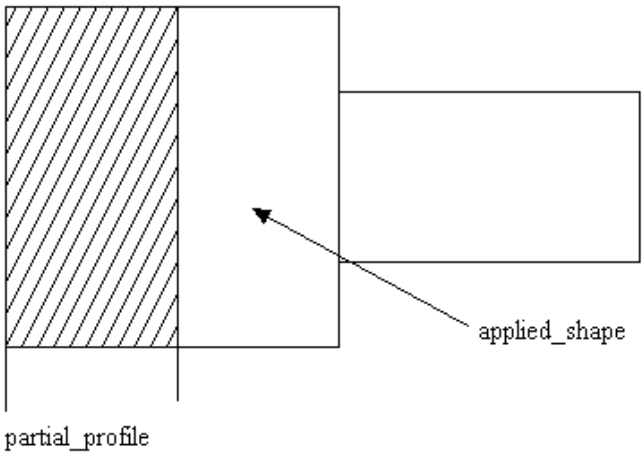


Figure 36 - Diagonal_knurl

The data associated with a Diagonal_knurl are the following:

— helix_angle;

— helix_hand.

4.2.58.1 helix_angle

The helix_angle specifies the angle the knurl pattern makes with the orientation axis of an applied to surface. See 4.3.59 for the application assertion.

©ISO 2000 - All rights reserved

4.2.58.2 helix_hand

The helix_hand specifies a description of whether the helix angle is applied to an orientation axis in a clockwise or a counterclockwise direction. See 4.3.58 for the application assertion.

4.2.59 Diameter_dimension_tolerance

A Diameter_dimension_tolerance is a type of Size_tolerance (see 4.2.204) that is the allowable variation of the size of a hole in a surface.

NOTE - Figure 37 illustrates the Diameter_dimension_tolerance.

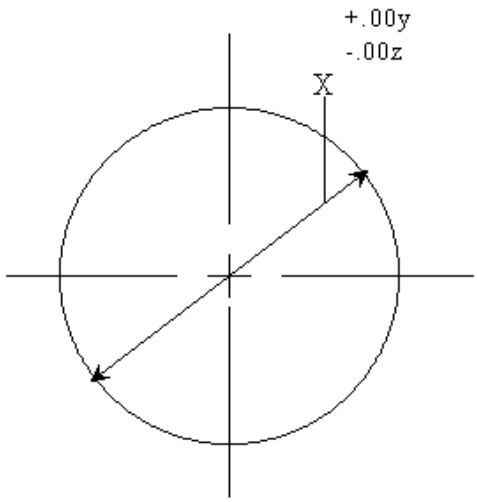


Figure 37 - Diameter_dimension_tolerance

4.2.60 Diameter_taper

A diameter_taper is a constant change in shape of a feature for a part. A Diameter_taper starts at the placement of a feature and is applied to the entire feature. The initial diameter and the length of the taper is determined from the Machining_feature that is applying the Diameter_taper. The final_diameter is specified as a diameter different than the initial diameter.

NOTE - Figure 38 illustrates the Diameter_taper.

 ©ISO 2000 - All rights reserved

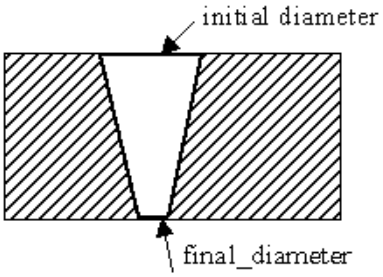


Figure 38 - Diameter_taper

The data associated with a Diameter_taper are the following:

— final_diameter.

4.2.60.1 final_diameter

The final_diameter specifies the diameter of the circle at the end of the taper. The final_diameter may be smaller or larger than the beginning diameter for a Machining_feature. See 4.3.60 for the application assertion.

NOTE - The diameter at the beginning of the Diameter_taper is the same as the referencing feature diameter.

4.2.61 Diamond_knurl

A Diamond_knurl is a type of Turned_knurl (see 4.2.235) that is a knurl ridge that is doubly helical, a left hand and a right hand helix, about the axis of a surface, with equal spacing of the two.

NOTE - Figure 39 illustrates a Diamond_knurl.

The data associated with a Diamond_knurl are the following:

— helix_angle.

4.2.61.1 helix_angle

The helix_angle specifies the angle the knurl pattern makes with the orientation axis of an applied to surface. See 4.3.61 for the application assertion.

©ISO 2000 - All rights reserved

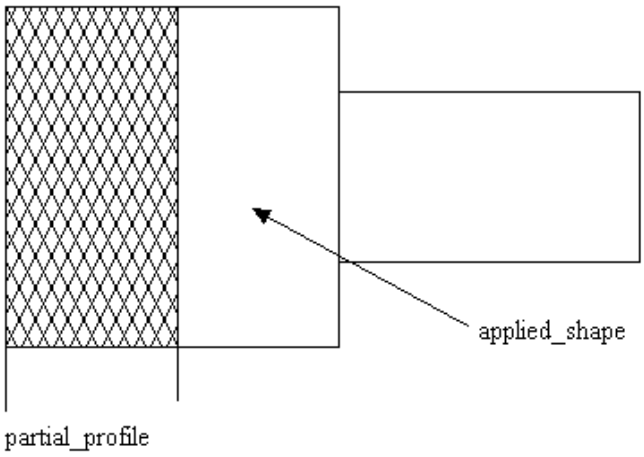


Figure 39 - Diamond_knurl

4.2.62 Digital_technical_data_package_work_order

A Digital_technical_data_package_work_order is a document that defines the need to create and track the digital product data definition for a part. The data associated with a Digital_technical_data_-package_work_order are the following:

— order_id.

4.2.62.1 order_id

The order_id specifies a unique identifier for the Digital_technical_data_package_work_order.

4.2.63 Dimensional_tolerance

A Dimensional_tolerance is the total amount a specific dimension is permitted to vary, which is the difference between maximum and minimum permitted limits of size. Each Dimensional_tolerance is either a Location_tolerance (see 4.2.109) or a Size_tolerance (see 4.2.204).

EXAMPLE - A dimension given as 1.624 +.002/-.002 means it may be 1.626 or 1.622, or anywhere between these limit dimensions.

The data associated with a Dimensional_tolerance are the following:

— dimension_description;

— dimension_value;

 ©ISO 2000 - All rights reserved

— limit;

— significant_digits;

— unit_of_measure.

4.2.63.1 dimension_description

The dimension_description specifies a textual description of any conditions which may affect the interpretation of the tolerance information that is defined. There may be more than one dimension_-description for a Dimensional_tolerance. The dimension_description need not be specified for a particular Dimensional_tolerance.

EXAMPLE - A Dimension_tolerance may apply in 2 places.

4.2.63.2 dimension_value

The dimension_value specifies the total amount by which a specific dimension is permitted to vary.

4.2.63.3 limit

The limit specifies the tolerance value applied to the Dimension_tolerance. The limit need not be specified for a particular Dimensional_tolerance. See 4.3.62 for the application assertion.

4.2.63.4 significant_digits

The significant_digits specifies the number of decimal places indicating the accuracy of dimension or tolerance. Significant_digits need not be specified for a particular Dimensional_tolerance.

4.2.63.5 unit_of_measure

The unit_of_measure specifies the unit in which the quantity is expressed.

4.2.64 Directed_taper

A Directed_taper is a constant change in shape of a feature for a part. A Directed_taper starts at the location of the placement of a feature and is applied to the entire feature. The length of the taper is determined from the Machining_feature that is applying the Directed_taper. The data associated with a Directed_taper are the following:

— angle;

— direction.

©ISO 2000 - All rights reserved

4.2.64.1 angle

The angle specifies the amount of slope from the start of the Directed_taper to the end of the Directed_taper. An angle between zero and 90 degrees or between negative 90 and negative 180 degrees indicates that the profile grows larger along the draft direction. An angle between zero and negative 90 degrees or between 90 and 180 degrees indicates that the profile grows smaller along the draft direction See 4.3.64 for the application assertion.

4.2.64.2 direction

The direction specifies the reference direction from which the Directed_taper angle is measured. The direction specifies a vector that points in the direction to apply the taper. See 4.3.63 for the application assertion.

4.2.65 Direction_element

A Direction_element is a type of Shape_element (see 4.2.200) that is a Shape_aspect definition for a direction.

4.2.66 Distance_along_curve_tolerance

A Distance_along_curve_tolerance is a type of Location_tolerance (see 4.2.109) that is the distance calculated between two elements along a path defined by a third element of geometry.

NOTE - Figure 40 illustrates the Distance_along_curve_tolerance.

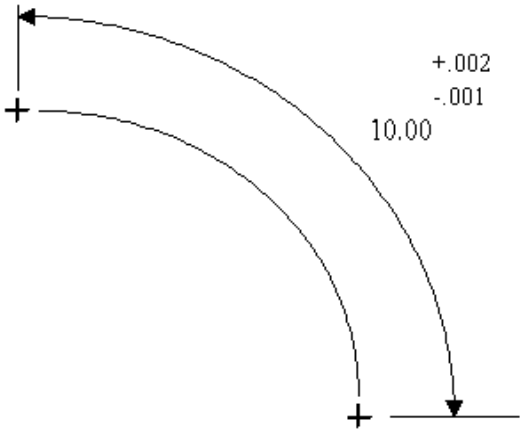


Figure 40 - Distance_along_curve_tolerance

 ©ISO 2000 - All rights reserved

The data associated with a Distance_along_curve_tolerance are the following:

— path;

— with_curve_direction.

4.2.66.1 path

The path specifies the shape that the tolerance applies to. See 4.3.65 for the application assertion.

4.2.66.2 with_curve_direction

The with_curve_direction specifies a boolean value that indicates the direction along the element to apply the tolerance. A value of true indicates the tolerance value is applied from the start point of the curve to the end point of the curve. A value of false indicates that the direction does not matter.

4.2.67 Edge_round

An Edge_round is a type of Transition_feature (see 4.2.234) that is a convex circular arc transition between two intersecting surfaces. The blend surface is tangent to both of the adjacent surface edges. An Edge_round may be a Constant_radius_edge_round (see 4.2.39).

NOTE - Figure 41 illustrates the Edge_round with an edge_round face that is not constant radius.

The data associated with an Edge_round are the following:

— edge_round_face;

— first_face_shape;

— second_face_shape.

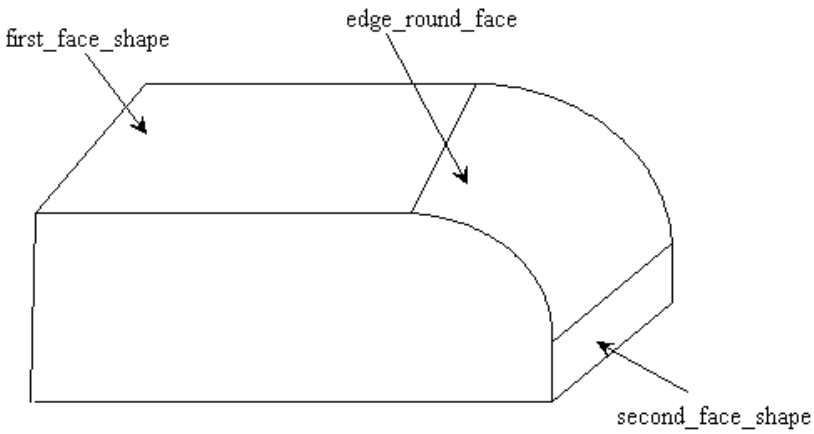


Figure 41 - Edge_round

©ISO 2000 - All rights reserved

4.2.67.1 edge_round_face

The edge_round_face specifies the circular transition surface between the two edges of two surfaces. See 4.3.66 for the application assertion.

4.2.67.2 first_face_shape

The first_face shape specifies one of two surfaces the Edge_round feature will transition between. See 4.3.66 for the application assertion.

4.2.67.3 second_face_shape

The second_face shape specifies the other surfaces the Edge_round feature will transition between. See 4.3.66 for the application assertion.

4.2.68 Engineering_change_order

An Engineering_change_order is an authorization for modification of the product data that will result in a new process plan for a part.

NOTE - These Engineering_change_orders apply only to changes that effect process planning.

The data associated with an Engineering_change_order are the following:

— change_order_number;

— new_version.

4.2.68.1 change_order_number

The change_order_number specifies a unique identification of the Engineering_change_order.

4.2.68.2 new_version

The new_version specifies the current version of a Part which has an effected change made by the Engineering_change_order. There may be more than one new_version of an Engineering_change_order. See 4.3.67 for the application assertion.

4.2.69 Engineering_change_proposal

An Engineering_change_proposal is a document that describes potential modifications to a part. The data associated with an Engineering_change_proposal are the following:

— change_proposal_number;

— incorporated_proposal.

 ©ISO 2000 - All rights reserved

4.2.69.1 change_proposal_number

The change_proposal_number specifies a unique identification of the Engineering_change_proposal.

4.2.69.2 incorporated_proposal

The incorporated_proposal specifies the change proposals that describe the modifications to the Part. There may be more than one incorporated_proposal for an Engineering_change_proposal. See 4.3.68 for the application assertion.

4.2.70 Explicit_base_shape_representation

An Explicit_base_shape_representation is a type of Base_shape (see 4.2.7) that is the geometric representation needed to define the shape of the initial material.

EXAMPLE - A B-rep model containing the geometry for a cast part may be an Explicit_base_shape_representation.

The data associated with an Explicit_base_shape_representation are the following:

— B-rep_form.

4.2.70.1 B-rep_form

The B-rep_form specifies the shape that is the representation of the initial material. See 4.3.69 for the application assertion.

4.2.71 Face_shape_element

A Face_shape_element is a type of Shape_element (see 4.2.200) that is a Shape_aspect definition for a Face.

4.2.72 Face_shape_element_relationship

A Face_shape_element_relationship defines the sequence in which face_shape_element objects are applied. The Face_shape_element_relationship defines which face is the preceding face and which face is the succeeding face. The data associated with a Face_shape_element_relationship are the following:

— predecessor;

— successor.

4.2.72.1 predecessor

The predecessor specifies the Face_shape_element (see 4.2.71) with the highest precedence. See 4.3.70 for the application assertion.

©ISO 2000 - All rights reserved

4.2.72.2 successor

The successor specifies the Face_shape_element (see 4.2.71) with the lesser precedence. See 4.3.70 for the application assertion.

4.2.73 Fillet

A Fillet is a type of Transition_feature (see 4.2.234) that is a concave circular arc transition between two intersecting surfaces. The blend surface may be tangent to both of the adjacent surface edges. A Fillet may be a Constant_radius_fillet (see 4.2.40).

NOTE - Figure 42 illustrates a Fillet with a fillet face that is not constant radius.

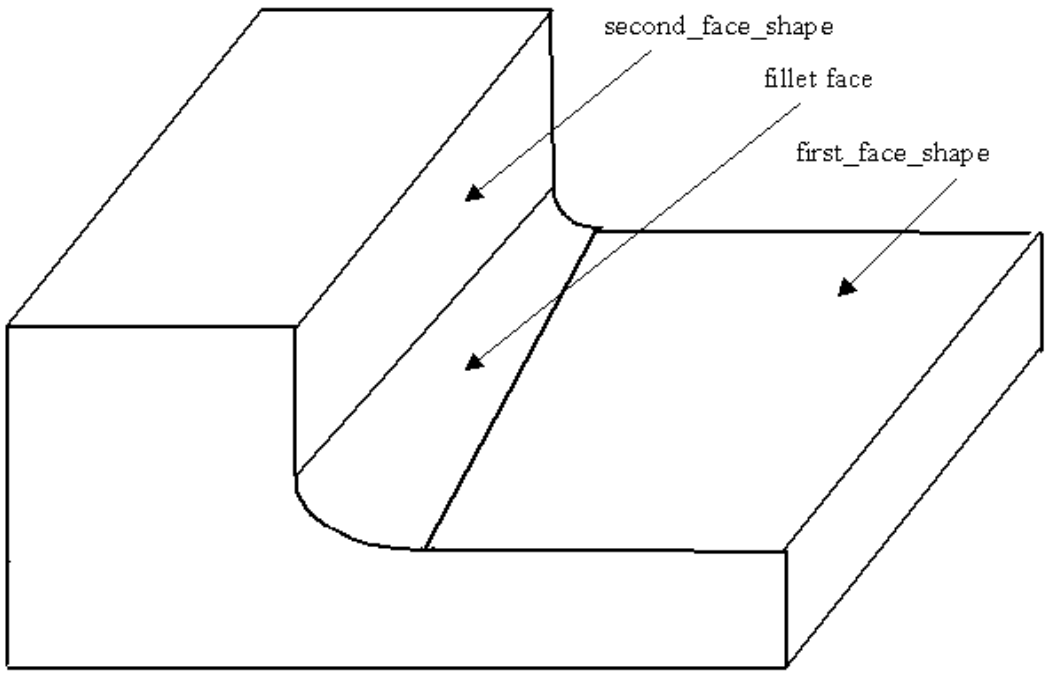


Figure 42 - Fillet

The data associated with a Fillet are the following:

— fillet_face;

— first_face_shape;

— second_face_shape.

 ©ISO 2000 - All rights reserved

4.2.73.1 fillet_face

The fillet_face specifies the circular transition surface between the two edges of two surfaces. See 4.3.71 for the application assertion.

4.2.73.2 first_face_shape

The first_face_shape specifies one of two surfaces the Fillet feature will transition between. See 4.3.71 for the application assertion.

4.2.73.3 second_face_shape

The second_face_shape specifies the second of two surfaces the Fillet feature will transition between. See 4.3.71 for the application assertion.

4.2.74 First_offset

A First_offset is the amount of length offset from a face for creating a Chamfer feature.

NOTE - Figure 12 illustrates a First_offset for a Chamfer feature.

The data associated with a First_offset are the following:

— face_shape;

— offset_amount.

4.2.74.1 face_shape

The first_face specifies a geometric shape for one of two faces the Chamfer feature will transition between. See 4.3.72 for the application assertion.

4.2.74.2 offset_amount

The offset_amount specifies a distance from the edge of a face to the start of the Chamfer. See 4.3.73 for the application assertion.

4.2.75 Flat_hole_bottom

A Flat_hole_bottom is a type of Blind_bottom_condition (see 4.2.8) that is the bottom of a Round_hole that shall be flat and have no corner radius.

NOTE - Figure 43 illustrates a Flat_hole_bottom for a Round_hole.

©ISO 2000 - All rights reserved

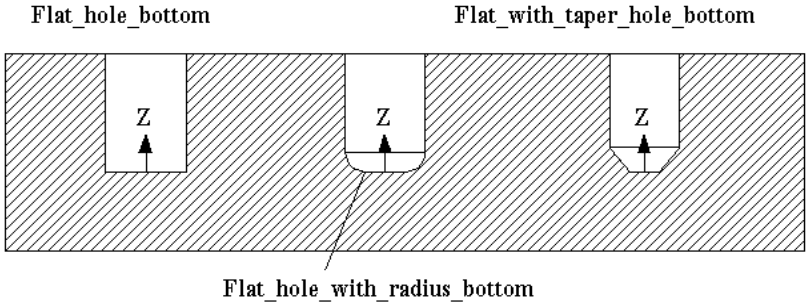


Figure 43 - Flat hole bottom types

4.2.76 Flat_slot_end_type

A Flat_slot_end_type is a type of Slot_end_type (see 4.2.206) that is an end condition of a slot that shall be a planar shape perpendicular to both of the adjacent Slot wall surfaces. The intersection of the Slot wall surfaces and the end planar shape need not be blended by a radius.

NOTE - Figure 44 illustrates a Flat_slot_end_type for a Slot.

The data associated with a Flat_slot_end_type are the following:

— first_radius;

— second_radius.

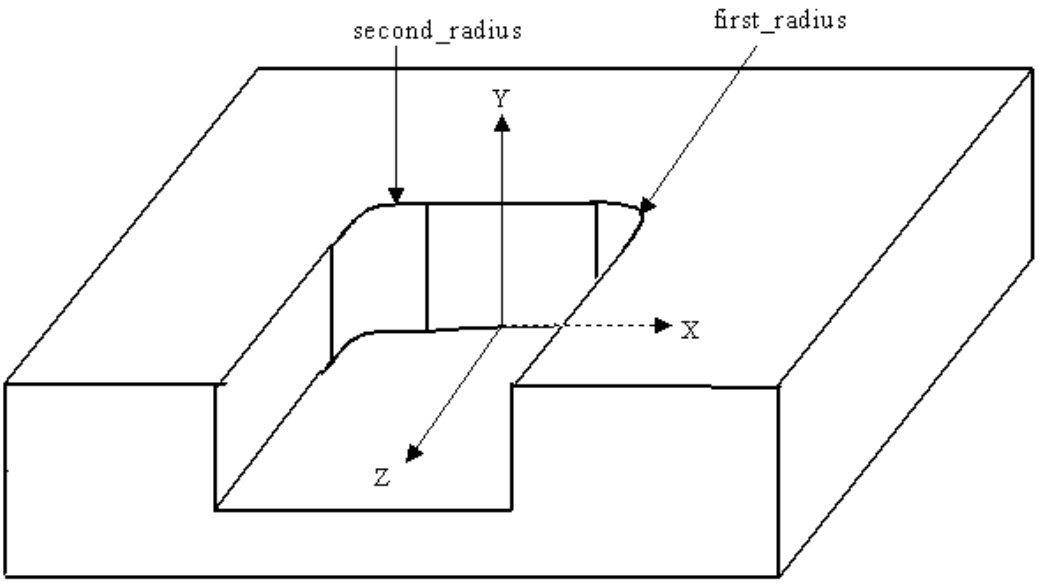


Figure 44 - Flat_slot_end_type

 ©ISO 2000 - All rights reserved

4.2.76.1 first_radius

The first radius specifies the circular arc transition between the wall of the Slot feature and the planar surface of the Flat_slot_end_type. The position of the first radius is where the Slot wall intersects the Flat_slot_end_type at a positive value along the X-axis. See 4.3.74 for the application assertion.

4.2.76.2 second_radius

The second radius specifies the circular arc transition between the wall of the Slot feature and the planar surface of the Flat_slot_end_type. The position of the second radius is where the Slot wall intersects the Flat_slot_end_type at a negative value along the X-axis. See 4.3.74 for the application assertion.

4.2.77 Flat_with_radius_hole_bottom

A Flat_with_radius_hole_bottom is a type of Blind_bottom_condition (see 4.2.8) that the bottom of a Round_hole is flat and has corner radius that are smaller than the diameter of the Round_hole.

NOTE - Figure 43 illustrates a Flat_with_radius_hole_bottom for a Round_hole.

The data associated with a Flat_with_radius_hole_bottom are the following:

— corner_radius.

4.2.77.1 corner_radius

The corner_radius specifies the radius between the side and the floor of a Round_hole (see 4.2.193). See 4.3.75 for the application assertion.

4.2.78 Flat_with_taper_hole_bottom

A Flat_with_taper_hole_bottom is a type of Blind_bottom_condition (see 4.2.8) that the bottom of a Round_hole is flat and has a planar taper from the sides of the Round_hole.

NOTE - Figure 43 illustrates a Flat_with_taper_hole_bottom for a Round_hole.

The data associated with a Flat_with_taper_hole_bottom are the following:

— final_diameter;

— taper_diameter.

4.2.78.1 final_diameter

The final_diameter specifies the diameter of the Round_hole floor which is a diameter smaller than the initial diameter. See 4.3.76 for the application assertion.

©ISO 2000 - All rights reserved

4.2.78.2 taper_diameter

The taper_diameter specifies the angle between the side and the floor of a Round_hole (see 4.2.193), measured along the axis inside the Round_hole. See 4.3.76 for the application assertion.

4.2.79 Flatness_tolerance

A Flatness_tolerance is a type of Geometric_tolerance (see 4.2.95) that is a tolerance for how much a surface is allowed to deviate from being flat. All points of the actual surface shall lie between two parallel planes that are a distance apart equal to the specified tolerance.

NOTE 1 - Figure 45 illustrates a Flatness_tolerance.

NOTE 2 - The Flatness_tolerance definition is derived from paragraph 14.2 of ISO 1101.

The data associated with a Flatness_tolerance are the following:

— segment_size.

4.2.79.1 segment_size

The segment_size specifies the length of a surface to apply a tolerance if the Flatness_tolerance is not applied to the total length. The segment_size need not be specified for a particular Flatness_tolerance.

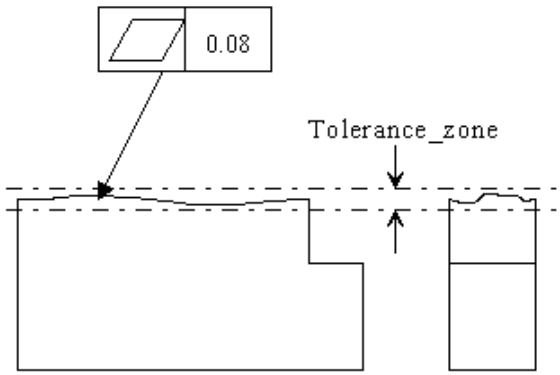


Figure 45 - Flatness_tolerance

4.2.80 General_boss

A General_boss is a type of Boss (see 4.2.10) that is an enclosed volume bounded by an arbitrary shape.

NOTE - Figure 46 illustrates the General_boss.

 ©ISO 2000 - All rights reserved

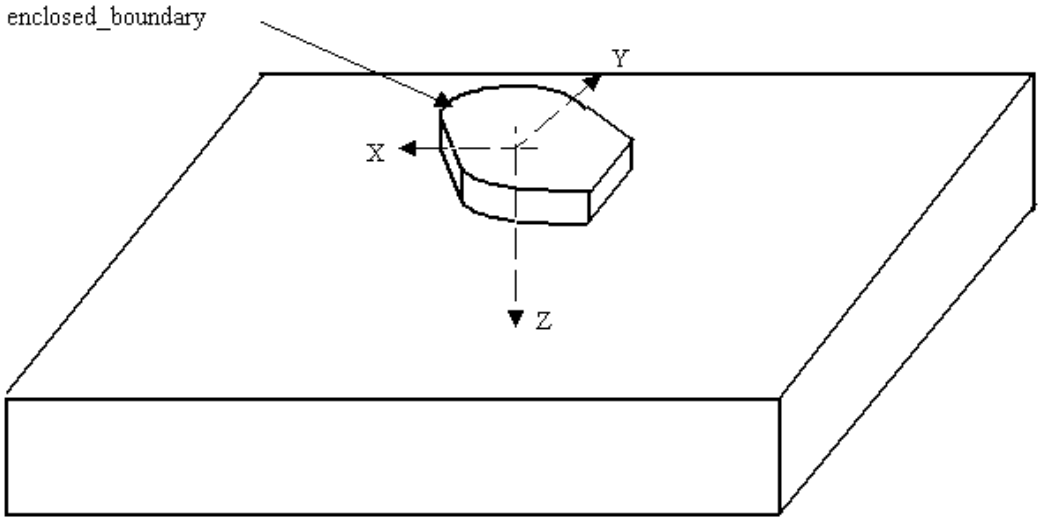


Figure 46 - General_boss

The data associated with a General_boss are the following:

— change_in_boundary;

— enclosed_boundary.

4.2.80.1 change_in_boundary

The change_in_boundary specifies a taper that defines the change in shape of the General_boss. The change_in_boundary need not be specified for a particular General_boss. See 4.3.78 for the application assertion.

4.2.80.2 enclosed_boundary

The enclosed_boundary specifies an outline or shape that bounds an enclosed area with no opening for a General_boss. The Closed_profile specifies the area required by a General_boss. The placement of the enclosed_boundary shall be with the origin of the Closed_profile at the origin of the General_boss. The X-axis and Y-axis of the profile shall be the same as the X-axis and Y-axis of the General_boss. See 4.3.77 for the application assertion.

4.2.81 General_closed_profile

A General_closed_profile is a type of Closed_profile (see 4.2.31) that is an enclosed area bounded by a arbitrary shape. The orientation is defined by the explicit geometry of the shape.

NOTE - Figure 47 illustrates a General_closed_profile.

©ISO 2000 - All rights reserved

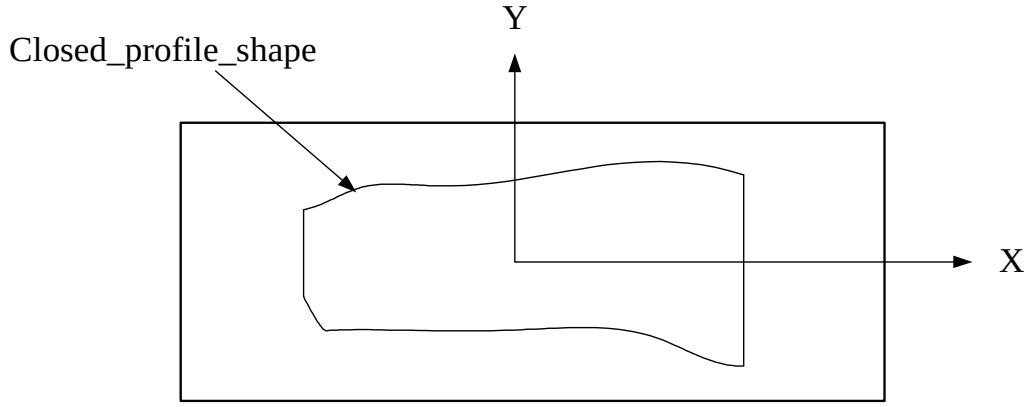


Figure 47 - General_closed_profile

The data associated with a General_closed_profile are the following:

— closed_profile_shape.

4.2.81.1 closed_profile_shape

The closed_profile_shape specifies a closed curve that defines the arbitrary shape of the profile. See 4.3.80 for the application assertion.

4.2.82 General_cutout

A General_cutout is a type of Cutout (see 4.2.45) that is a volume of arbitrary shape removed from the part and shall pass through two faces of the part. A General_cutout is similar in definition to a General_pocket, but differ in the type of process required to manufacture. The data associated with a Circular_cutout are the following:

— boundary.

4.2.82.1 boundary

The boundary specifies an outline or shape that is an area that may be closed or partially open. The profile specifies the area required by a General_cutout. The placement of the boundary shall be with the origin of the Profile at the origin of the General_cutout. The X-axis and Y-axis of the profile shall be the same as the X-axis and Y-axis of the General_cutout. See 4.3.81 for the application assertion.

4.2.83 General_open_profile

A General_open_profile is a type of Open_profile (see 4.2.130) that is specified by a shape bounded by an arbitrary planar shape. The orientation is defined by the explicit geometry of the shape.

NOTE - Figure 48 illustrates a General_open_profile.

 ©ISO 2000 - All rights reserved

The data associated with a General_open_profile are the following:

— enclosed_boundary.

4.2.83.1 enclosed_boundary

The enclosed_boundary specifies a curve with no enclosing bounds that defines the arbitrary planar shape of the profile. See 4.3.82 for the application assertion.

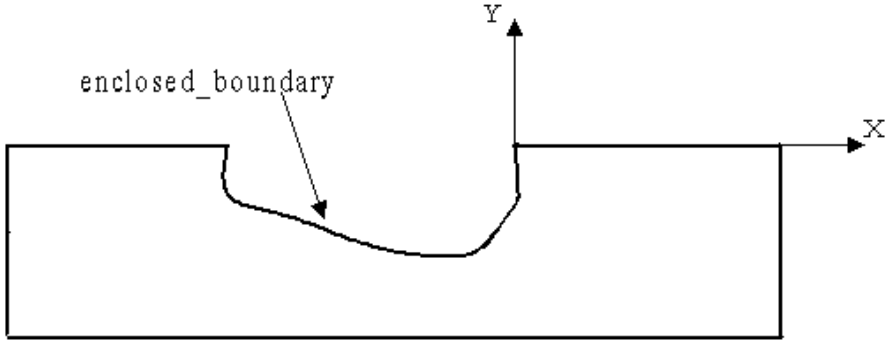


Figure 48 - General_open_profile

4.2.84 General_outside_profile

A General_outside_profile is a type of Profile_feature (see 4.2.165) that is specified the removal volume of raw stock or other excess material of arbitrary shape from the outside shape of the part. The General_outside_profile feature may remove material from the entire outside shape or some portion of the shape. A single part may have several General_outside_profile objects, with the Z-axes of the coordinate systems of the General_outside_profile objects pointing in any direction.

NOTE - Figure 49 illustrates a General_outside_profile.

The data associated with a General_outside_profile are the following:

— boundary.

4.2.84.1 boundary

The boundary specifies a profile that identifies the outside shape of the part. The placement of the boundary shall be with the origin of the profile at the origin of the General_outside_profile feature. The X-axis and Y-axis of the profile shall be the same as the X-axis and Y-axis of the feature. See 4.3.83 for the application assertion.

©ISO 2000 - All rights reserved

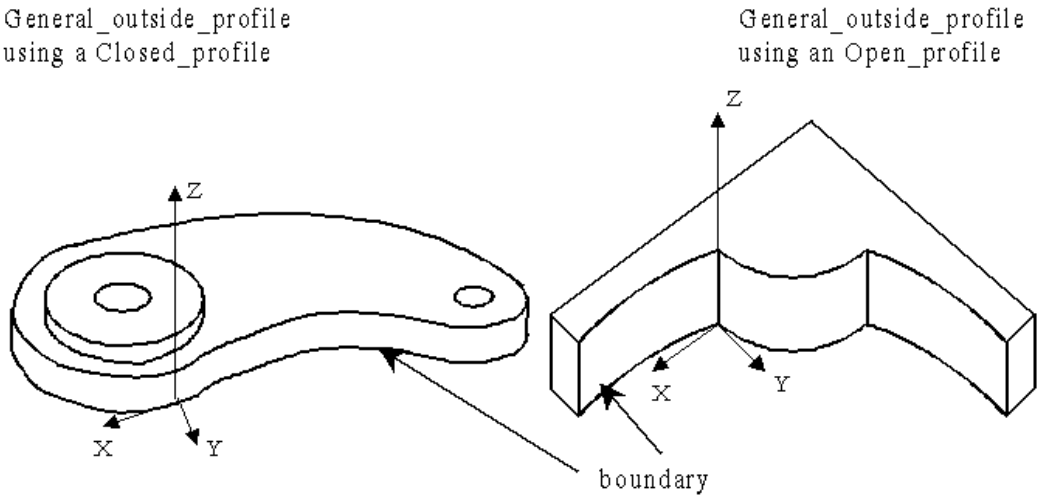


Figure 49 - General_outside_profile

4.2.85 General_path

A General_path is a type of Path (see 4.2.146) that is a direction of travel along an arbitrary curve.

NOTE - Figure 50 illustrates a Slot feature with a Square_u_profile and a General_path. The path has the same orientation as the Slot feature. The orientation of the profile positions it to the orientation of the Slot feature. The Square_U_profile has orientation to position it at the same position as the Slot feature.

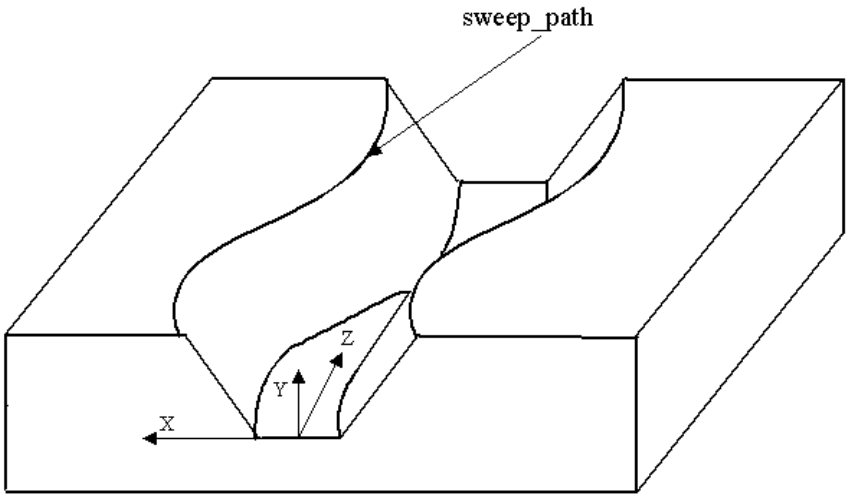


Figure 50 - General_path

 ©ISO 2000 - All rights reserved

The data associated with a General_path are the following:

— sweep_path.

4.2.85.1 sweep_path

The sweep_path specifies a continues set of curves that define an arbitrary direction of travel. See 4.3.84 for the application assertion.

4.2.86 General_pattern

A General_pattern is a type of Replicate_feature (see 4.2.185) that is a base shape component and a list of arbitrary placements to arrange identical copies of the base feature. The placement of all instances of the base feature are relative to the Replicate_feature coordinate system.

NOTE -Figure 51 illustrates a General_pattern of Round_hole features.

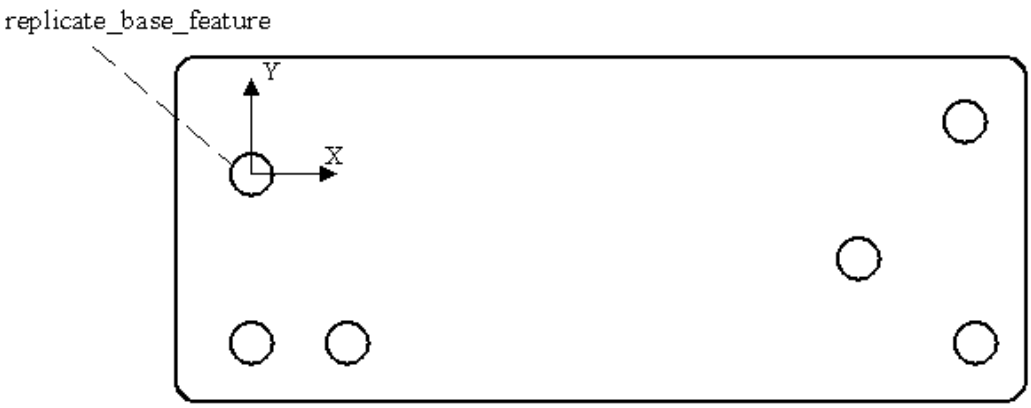


Figure 51 - General_pattern

The data associated with a General_pattern are the following:

— feature_placement.

4.2.86.1 feature_placement

The feature_placement specifies a set of axis and positions to place a base feature in the General_pattern. A General_pattern is defined as one base feature and many placements allowing the base feature to be placed several times in a random pattern. There may be more than one feature_placement for a General_pattern. See 4.3.85 for the application assertion.

©ISO 2000 - All rights reserved

4.2.87 General_pocket

A General_pocket is a type of Pocket (see 4.2.160) that is a volume of arbitrary shape removed from the part.

NOTE -Figure 52 illustrates a General_pocket with a Through_pocket_bottom_condition and a pocket depth that defines the highest point of the pocket.

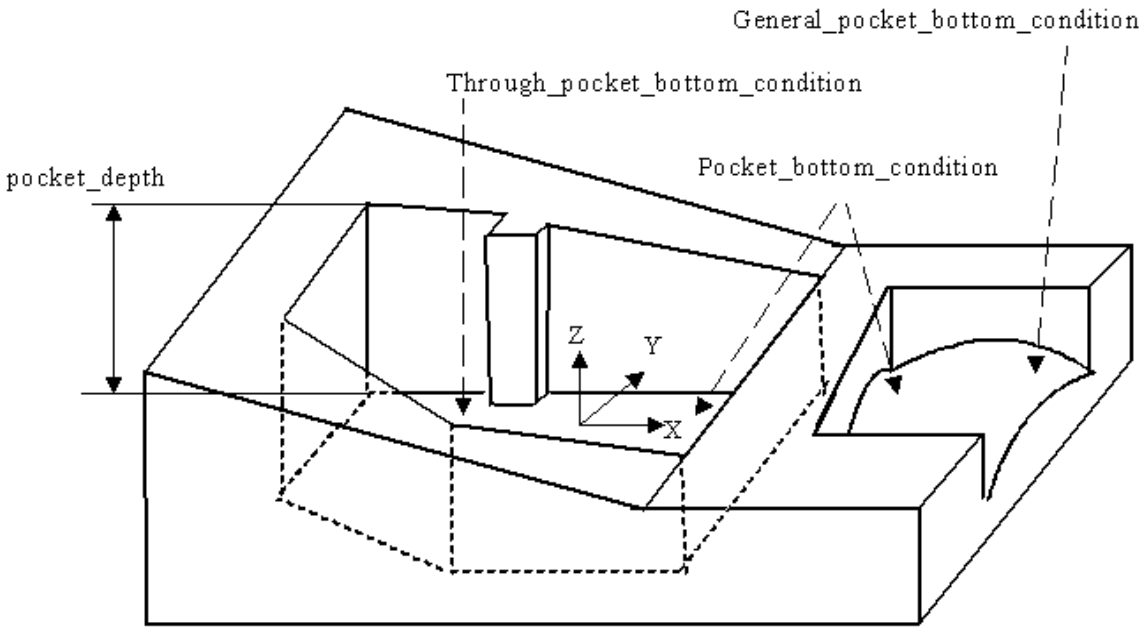


Figure 52 - General_pocket

The data associated with a General_pocket are the following:

— boundary.

4.2.87.1 boundary

The boundary specifies an outline or shape that is an area that may be closed or partially open. The profile specifies the area required by a General_pocket. The placement of the boundary shall be with the origin of the Profile at the origin of the General_pocket. The X-axis and Y-axis of the profile shall be the same as the X-axis and Y-axis of the General_pocket. See 4.3.86 for the application assertion.

 ©ISO 2000 - All rights reserved

4.2.88 General_pocket_bottom_condition

A General_pocket_bottom_condition is a type of Pocket_bottom_condition (see 4.2.161) that specifies an enclosed area bounded by an arbitrary shape.

NOTE - Figure 52 illustrates the General_pocket_bottom_condition applied to a General_pocket with an Open_profile.

The data associated with a General_pocket_bottom_condition are the following:

— floor;

— floor_radius.

4.2.88.1 floor

The floor specifies the face at the bottom of a Pocket feature, adjacent to all the pocket walls. See 4.3.87 for the application assertion.

4.2.88.2 floor_radius

The floor radius is specifies the amount of curvature for an arc between the bottom and the sides of a pocket feature. See 4.3.88 for the application assertion.

4.2.89 General_profile_floor

A General_profile_floor is a type of Profile_floor (see 4.2.166) that specifies an enclosed area bounded by an arbitrary shape.

NOTE - Figure 94 illustrates a Shape_profile with a General_profile_floor.

The data associated with a General_profile_floor are the following:

— floor.

4.2.89.1 floor

The floor specifies the face at the bottom of a General_profile_floor feature, adjacent to all the Shape_profile walls. See 4.3.89 for the application assertion.

4.2.90 General_removal_volume

A General_removal_volume is a type of Multi_axis_feature(see 4.2.125) that is an enclosed volume of arbitrary shape that shall be removed from the part. The position and orientation shall be determined from the shape defining geometry.

NOTE -Figure 53 illustrates a General_removal_volume.

©ISO 2000 - All rights reserved

The data associated with a General_removal_volume are the following:

— removal_volume.

4.2.90.1 removal_volume

The removal_volume specifies the arbitrary shape to be removed. See 4.3.90 for the application assertion.

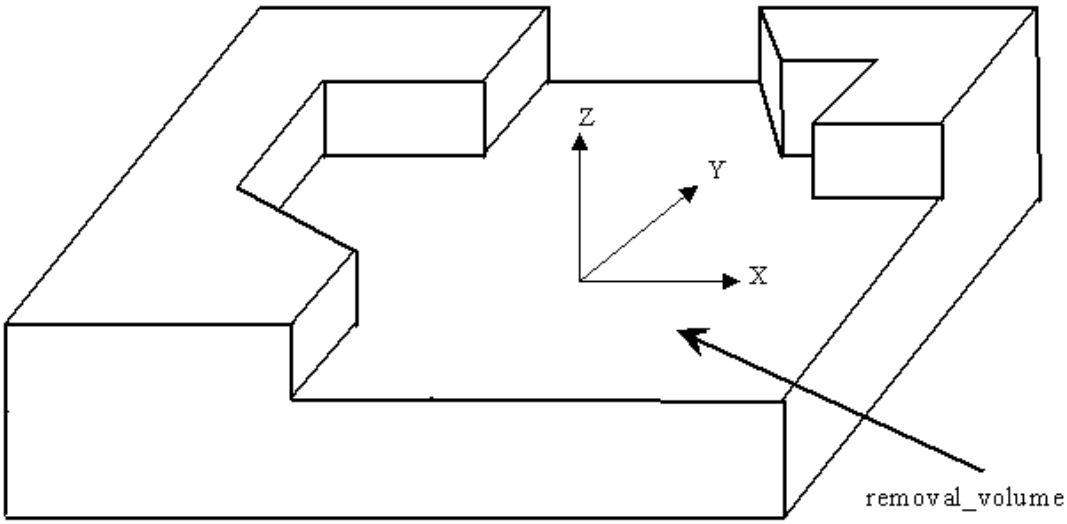


Figure 53 - General_removal_volume

4.2.91 General_revolution

A General_revolution is a type of Revolved_feature (see 4.2.188) that is an arbitrary planar shape swept one complete revolution about an axis. The arbitrary planar shape shall be finite in length, coplanar with the axis of revolution, and shall not intersect the axis of revolution. The General_revolution may be either an outer shape of a part or a volume removal, depending on the material direction.

NOTE - Figure 54 illustrates a General_revolution for an outer shape, the material direction points to the material side. Figure 55 illustrates a General_revolution for a volume removal.

The data associated with a General_revolution are the following:

— outer_edge_shape.

 ©ISO 2000 - All rights reserved

4.2.91.1 outer_edge_shape

The outer_edge_shape specifies an outline or shape that shall be revolved about an axis. The General_-open_profile specifies the outer edge shape required by a General_revolution. The placement of the outer_edge_shape shall be on the Y-axis of the General_revolution at a specified distance away from the origin of the General_revolution. The X-axis and Y-axis of the profile shall be the same as the Y-axis and Z-axis of the General_revolution. See 4.3.91 for the application assertion.

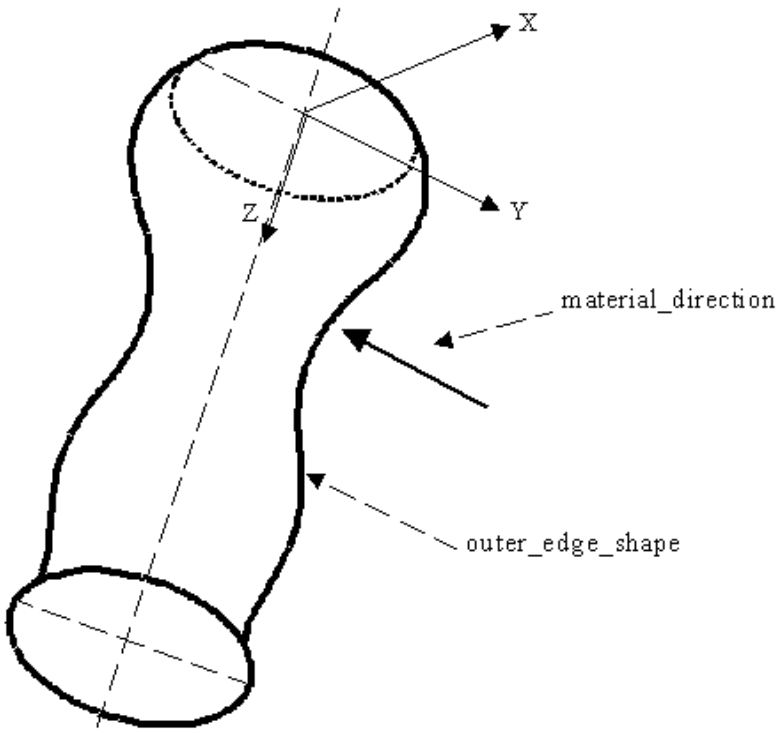


Figure 54 - General_revolution for outer shape

4.2.92 General_rib_top_floor

A General_rib_top_floor is a type of Rib_top_floor (see 4.2.192) that specifies an enclosed area bounded by an arbitrary shape. The floor defining shapes may be defined in a particular sequence. The data associated with a General_rib_top_floor are the following:

— rib_top_face.

4.2.92.1 rib_top_face

A rib_top_face specifies a set of faces at the bottom of a Rib_top (see 4.2.191). The order of the faces is achieved using a Feature_shape_element_relationship (see 4.2.72).See 4.3.92 for the application assertion.

©ISO 2000 - All rights reserved

4.2.93 General_shape_profile

A General_shape_profile is a type of Shape_profile (see 4.2.201) that is a volume of arbitrary shape which defines a portion of the part. The data associated with a General_shape_profile are the following:

— profile_boundary.

4.2.93.1 profile_boundary

The profile_boundary specifies the outline of the Shape_profile feature. The outline defines an area that may or may not be entirely enclosed. The placement of the profile_boundary shall be with the origin of the Path, that defines the profile, at the origin of the General_shape_profile. The X-axis and Y-axis of the profile shall be the same as the X-axis and Y-axis of the General_shape_profile. See 4.3.93 for the application assertions.

4.2.94 General_top_condition

A General_top_condition is a type of Boss_top_condition (see 4.2.11) that specifies an enclosed area bounded by an arbitrary shape that defines the top of a Boss feature. The data associated with a General_top_condition are the following:

— top_face.

4.2.94.1 top_face

The top_face specifies a face at the top of a Boss feature, adjacent to the Boss sides. See 4.3.94 for the application assertion.

4.2.95 Geometric_tolerance

A Geometric_tolerance is the maximum or minimum variation from true geometric form or position that may be permitted in manufacture. Geometric_tolerance should be employed only for those requirements of a part critical to its functioning or interchangability. Each Geometric_tolerance is one of the following: Angularity_tolerance (see 4.2.5), Circular_runout_tolerance (see 4.2.29), Circularity_-tolerance (see 4.2.30), Concentricity_tolerance (see 4.2.37), Cylindricity_tolerance (see 4.2.48), Flatness_tolerance (see 4.2.79), Linear_profile_tolerance (see 4.2.106), Parallelism_tolerance (see 4.2.138), Perpendicularity_tolerance (see 4.2.149), Position_tolerance (see 4.2.162), Straightness_-tolerance (see 4.2.214), Surface_profile_tolerance (see 4.2.215), Symmetry_tolerance (see 4.2.217), or a Total_runout_tolerance (see 4.2.233).

NOTE - Geometric tolerance definitions are derived from ISO 1101.

 ©ISO 2000 - All rights reserved

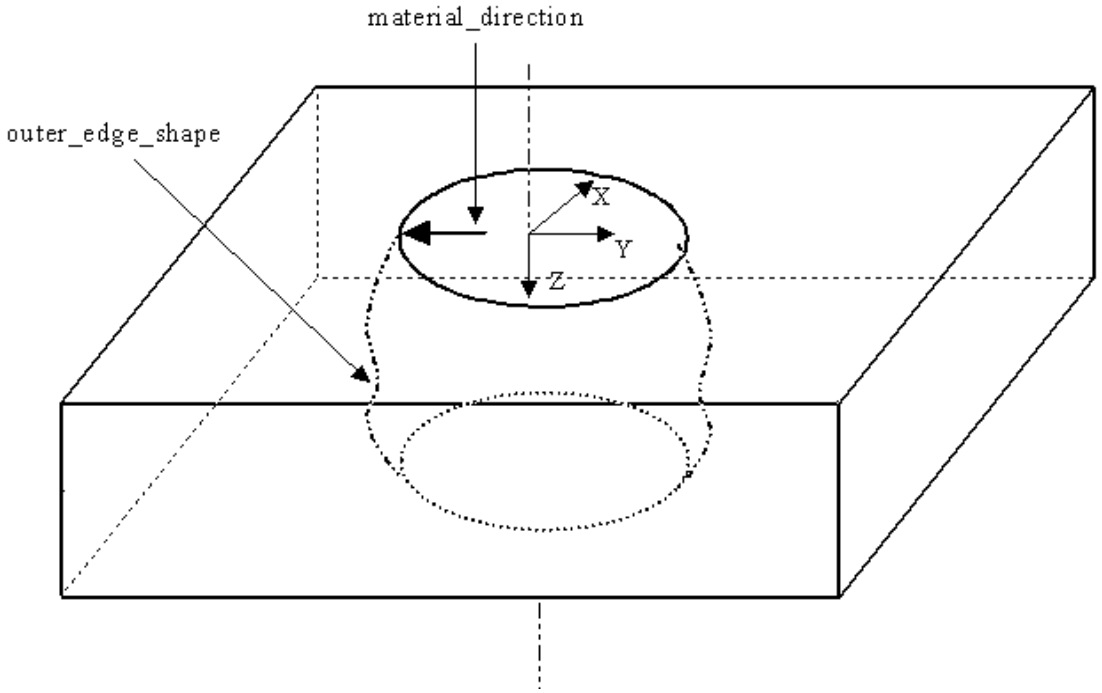


Figure 55 - General_revolution for volume removal

The data associated with a Geometric_tolerance are the following:

— applied_shape;

— geometric_tolerance_value;

— modifier_control;

— significant_digits;

— unit_of_measure;

— zone_definition.

4.2.95.1 applied_shape

The applied_shape specifies the shape on a Part that is being toleranced by a Geometric_tolerance. There may be more than one applied_shape for a Geometric_tolerance. See 4.3.96 for the application assertion.

4.2.95.2 geometric_tolerance_value

The geometric_tolerance_value specifies the tolerance amount that a part is allowed to meet the required accuracy for proper fit.

©ISO 2000 - All rights reserved

4.2.95.3 modifier_control

The modifier_control specifies the material condition which is applied to the shape being toleranced by the Geometric_tolerance. The modifier_control need not be specified for a particular Geometric_tolerance. See 4.3.95 for the application assertion.

4.2.95.4 significant_digits

The significant_digits specifies the number of decimal places indicating the accuracy of the tolerance.

4.2.95.5 unit_of_measure

The unit_of_measure specifies the quantity of measure in which the value is given.

4.2.95.6 zone_definition

The zone_definition specifies the tolerance zone that restricts the Geometric_tolerance. The zone_-definition need not be specified for a particular Geometric_tolerance. See 4.3.97 for the application assertion.

4.2.96 Geometric_tolerance_precedence_relationship

A Geometric_tolerance_precedence_relationship is a composite geometric tolerance. A part may have a shape that has a geometric tolerance and that shape is the basic shape for a pattern which may also have a geometric tolerance. The geometric tolerance for the base shape shall have precedence over the geometric tolerance for the pattern. The data associated with a Geometric_tolerance_precedence_-relationship are the following:

— base_shape_tolerance;

— pattern_shape_tolerance.

4.2.96.1 base_shape_tolerance

The base_shape_tolerance specifies the Geometric_tolerance which is applied to a shape on a Part. See 4.3.98 for the application assertion.

4.2.96.2 pattern_shape_tolerance

The pattern_shape_tolerance specifies the Geometric_tolerance which is applied to a pattern on a Part. See 4.3.98 for the application assertion.

 ©ISO 2000 - All rights reserved

4.2.97 Groove

A Groove is a type of Revolved_feature (see 4.2.188) that is a narrow channel or depression that is swept through one complete revolution about an axis.

NOTE - Figure 56 illustrates two Groove features. The face shape that has the Groove applied to it is determined by the profile orientation.

The data associated with a Groove are the following:

— sweep.

4.2.97.1 sweep

The sweep specifies an outline or shape that shall be revolved about an axis. The Open_profile specifies the sweep shape required by a Groove. The placement of the profile shall be along the X-axis of the Groove at a specified distance away from the origin. The orientation of the Open_profile is independent of the orientation of the Groove feature. The Groove feature may be defined on different faces of a part depending on the orientation of the profile. See 4.3.99 for the application assertion.

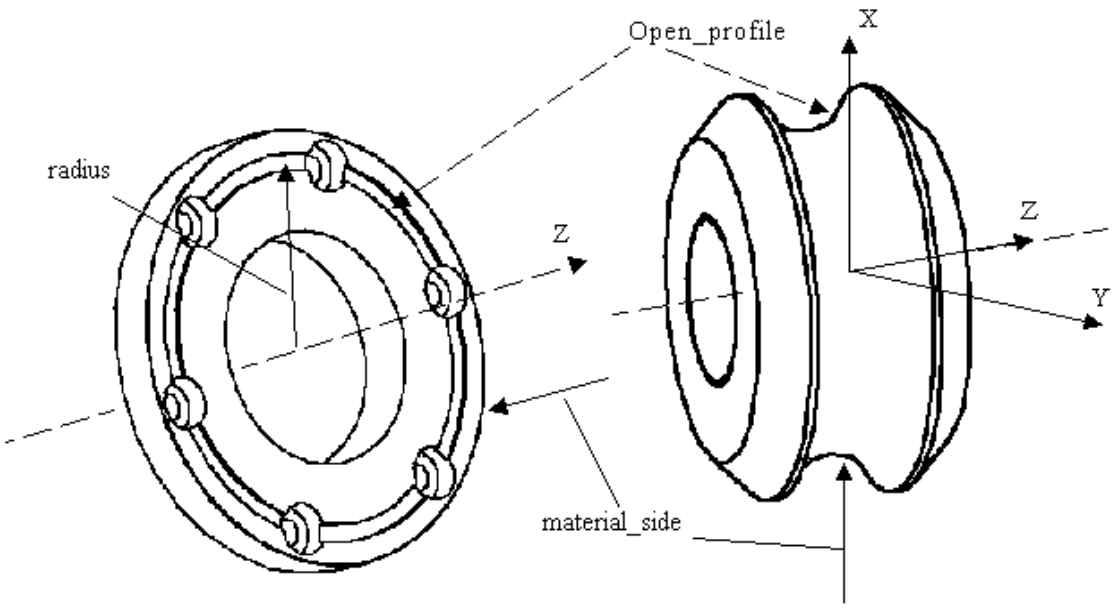


Figure 56 - Groove

©ISO 2000 - All rights reserved

4.2.98 Hardness

A Hardness is the resistance of a material surface to deformation by external forces. The data associated with a Hardness are the following:

— high_value;

— low_value;

— nominal;

— scale.

4.2.98.1 high_value

The high_value specifies the highest allowed value of hardness for a specific material type.

4.2.98.2 low_value

The low_value specifies the lowest allowed value of hardness for a specific material type.

4.2.98.3 nominal

The nominal specifies the average value of hardness for a specific material type.

4.2.98.4 scale

The scale specifies the method of determining hardness.

EXAMPLE - Rockwell and Brinell are examples of scale.

4.2.99 Hole

A Hole is a type of Multi_axis_feature(see 4.2.125) that is the removal of a cylindrical volume from a part. Each Hole is either a Counterbore_hole (see 4.2.41), Countersunk_hole (see 4.2.42), or a Round_-hole (see 4.2.193). The Hole may be positioned at its bottom with the Z-axis in the direction out of the Hole or at a position at the top of the hole with the Z-axis in the direction into the part.

4.2.100 Implicit_base_shape_representation

An Implicit_base_shape_representation is a type of Base_shape (see 4.2.7) that is the type of representation needed to define the shape of the initial material. The shape of the material may be either cylindrical, rectangular, or a polygon of any number of sides. Each Implicit_base_shape_representation is either a Block_base_shape (see 4.2.9), Cylindrical_base_shape (see 4.2.47), or a Ngon_base_shape (see 4.2.126).

NOTE - Figure 57 illustrates types of Implicit_base_shape_representations.

 ©ISO 2000 - All rights reserved

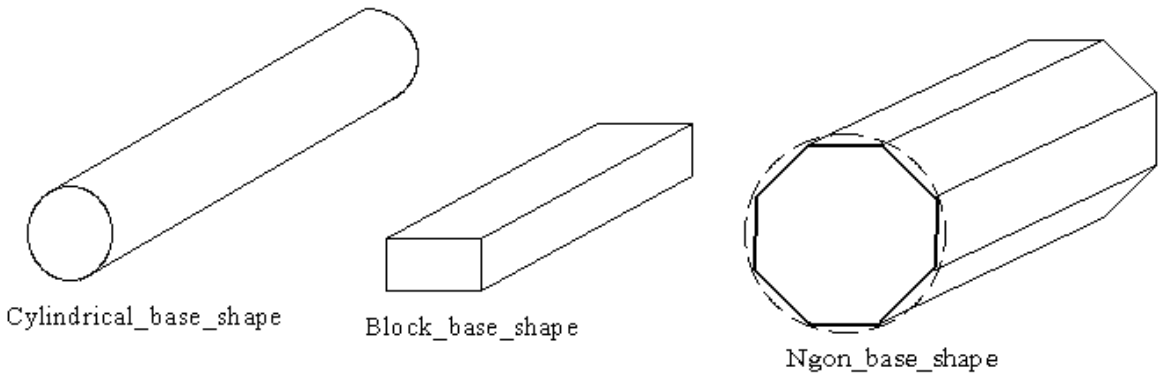


Figure 57 - Implicit_base_shape_representation

The data associated with an Implicit_base_shape_representation are the following:

— base_shape_length;

— placement.

4.2.100.1 base_shape_length

The base_shape_length is the size of the length of a Implicit_base_shape_representation. See 4.3.100 for the application assertion.

4.2.100.2 placement

A placement specifies the positioning of the part with respect to basic material stock. The positioning will be different for different types of Implicit_base_shape_representation. See 4.3.101 for the application assertion.

A Cylindrical_base_shape shall be positioned with the Z axis parallel to the length of the shape. The X and Y axis shall be orthogonal to the Z axis. The axis shall be positioned in the exact center of the circular profile of the Cylindrical_base_shape.

A Block_base_shape shall be positioned with the Z axis parallel to the length of the shape, the Y axis shall be parallel to the height of the shape, and the X axis shall be parallel to the width of the shape. The axis shall be positioned in the exact center of the rectangular profile of the Block_base_shape.

A Ngon_base_shape shall be position with the Z axis parallel to the length of the shape, the X axis shall be parallel to at least one side of the Ngon_base_shape, and the Y axis shall be orthogonal to the X and Z axis. The axis shall be positioned in the exact center of the ngon profile of the Ngon_base_shape.

©ISO 2000 - All rights reserved

4.2.101 Indirect_stock_requisition

An Indirect_stock_requisition is a type of Requisition (see 4.2.186) that defines indirect items required to support part manufacturing.

4.2.102 Knurl

A Knurl is a type of Machining_feature (see 4.2.111) that is a scoring pattern made by a series of small ridges or beads on a metal surface. Each Knurl is either a Catalogue_knurl (see 4.2.16) or a Turned_-knurl (see 4.2.235).

NOTE - Figure 58 illustrates a Knurl types.

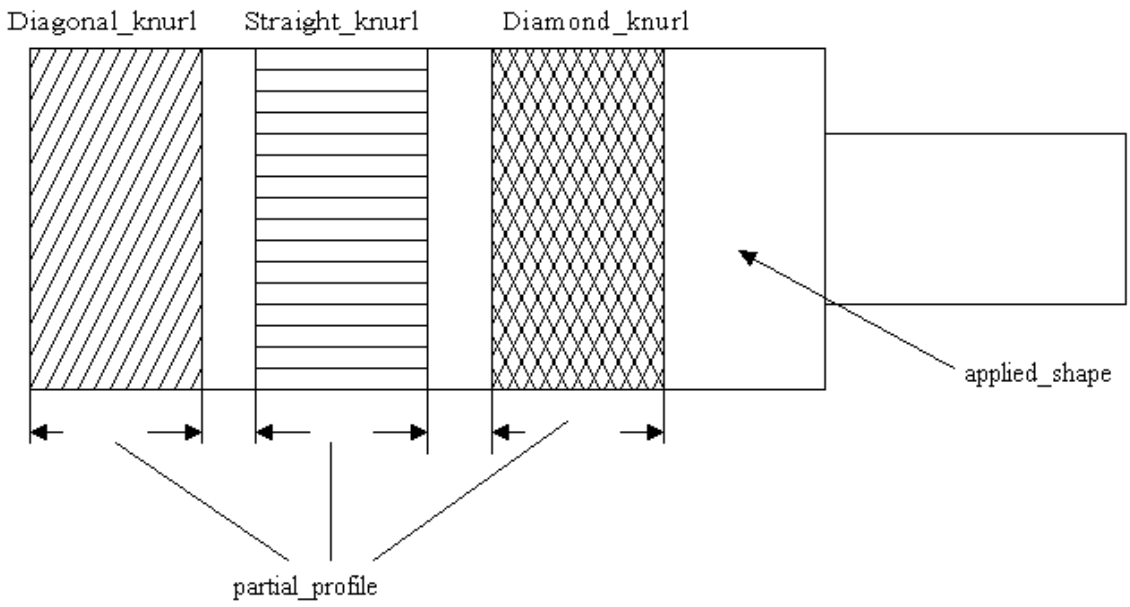


Figure 58 - Knurl

The data associated with a Knurl are the following:

— applied_shape;

— partial_profile.

4.2.102.1 applied_shape

The applied_shape specifies a base shape for applying the Knurl feature. See for 4.3.103 the application assertion.

 ©ISO 2000 - All rights reserved

4.2.102.2 partial_profile

The partial_profile specifies the placement and length of a surface to apply a Knurl feature. The partial_profile need not be specified for a particular Knurl. See 4.3.102 for the application assertion.

4.2.103 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. The data associated with a Limits_and_fits are the following:

— deviation;

— fitting type;

— grade.

4.2.103.1 deviation

The deviation specifies the class descriptor, by characters, for the designated limets and fits.

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

4.2.103.2 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.

4.2.103.3 grade

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

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

4.2.104 Linear_path

A Linear_path is a type of Path (see 4.2.146) that is a direction of travel along a line.

NOTE - Figure 59 illustrates a Machining feature with a Linear_path. This path has the same orientation as the feature.

©ISO 2000 - All rights reserved

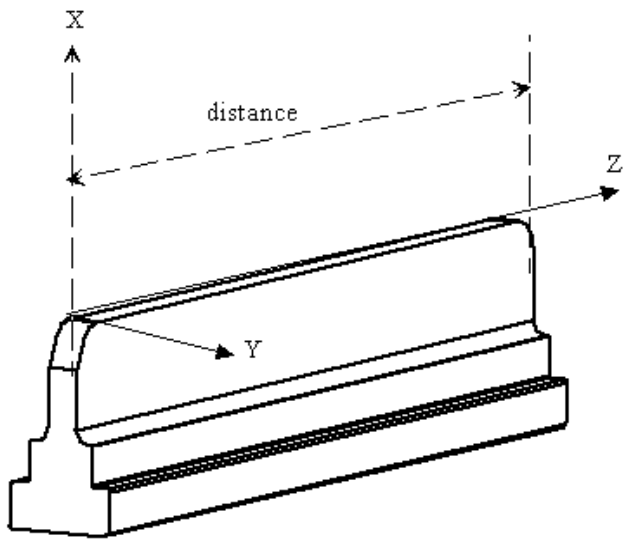


Figure 59 - Linear_path

The data associated with a Linear_path are the following:

— direction;

— distance.

4.2.104.1 direction

The direction specifies a vector which indicates the the direction of the path starting from the path placement. See 4.3.105 for the application assertion.

4.2.104.2 distance

The distance specifies the length of the path. See 4.3.104 for the application assertion.

4.2.105 Linear_profile

A Linear_profile is a type of Open_profile (see 4.2.130) that is a straight line of a specified length. The Linear_profile shall have orientation parallel to the X-axis.

NOTE - Figure 60 illustrates a Linear_profile that is being applied to the Planar_face feature.

The data associated with a Linear_profile are the following:

— profile_length.

 ©ISO 2000 - All rights reserved

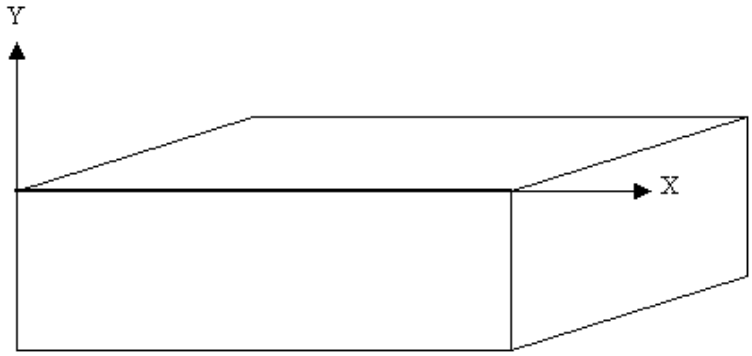


Figure 60 - Linear_profile

4.2.105.1 profile_length

The profile_length specifies the length of the profile. See 4.3.106 for the application assertion.

4.2.106 Linear_profile_tolerance

A Linear_profile_tolerance is a type of Geometric_tolerance (see 4.2.95) that is a uniform boundary or zone along the true profile within which all elements of the surface shall lie.

NOTE 1 - Figure 61 illustrates the Linear_profile_tolerance

NOTE 2 - The Linear_profile_tolerance definition is derived from paragraph 14.5 of ISO 1101.

The data associated with a Linear_profile_tolerance are the following:

— affected_plane;

— geometric_reference.

4.2.106.1 affected_plane

The affected_plane specifies the plane to apply the Linear_profile_tolerance. The affected_plane need not be specified for a particular Linear_profile_tolerance. See 4.3.108 for the application assertion.

4.2.106.2 geometric_reference

The geometric_reference specifies the datum to which the tolerance is related. See 4.3.107 for the application assertion.

©ISO 2000 - All rights reserved

4.2.107 Location_dimension_tolerance

A Location_dimension_tolerance is a type of Location_tolerance (see 4.2.109) that is the allowable variation in locating one feature of a part with respect to another.

NOTE - Figure 62 illustrates a Location_dimension_tolerance.

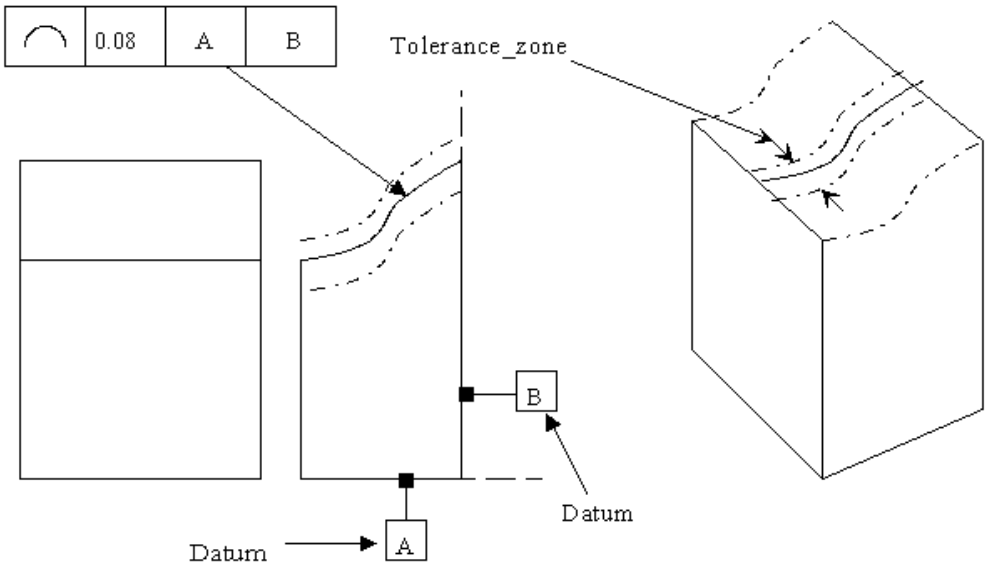


Figure 61 - Linear_profile_tolerance

The data associated with a Location_dimension_tolerance are the following:

— directed;

— plane_and_direction.

4.2.107.1 directed

The directed specifies a logical value designating the importance of direction for measuring a location_-dimension_tolerance. If value is TRUE, location_dimension_tolerance is measured from point of origin to point of termination, if FALSE, an in tolerance result shall occur regardless of direction of measurement.

4.2.107.2 plane_and_direction

The plane_and_direction specifies a plane that contains the geometry for the Location_dimension_-tolerance and a direction that is the location of the plane that contains the Location_dimension_tolerance. The plane_and_direction need not be specified for a particular Location_dimension_tolerance. See 4.3.109 for the application assertion.

EXAMPLE - A part might be viewed in a front view for defining a location_dimension_tolerance.

 ©ISO 2000 - All rights reserved

4.2.108 Location_element

A Location_element is a type of Shape_element (see 4.2.200) that is a Shape_aspect representation for a reference point.

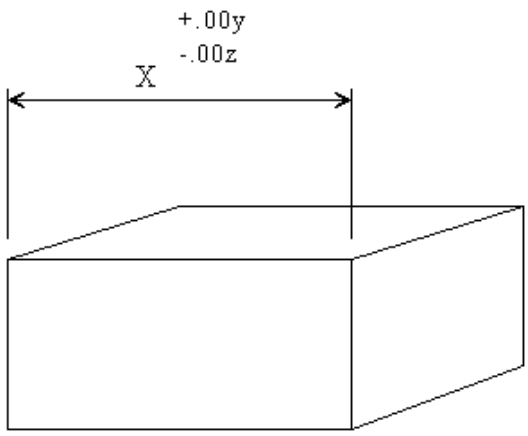


Figure 62 - Location_dimension_tolerance

4.2.109 Location_tolerance

A Location_tolerance is a type of Dimensional_tolerance (see 4.2.63) that defines tolerances that are an allowable variation in location between an origin shape and a termination shape. Each Location_-tolerance is either an Angular_dimensional_tolerance (see 4.2.2), Location_dimension_tolerance (see 4.2.107), or Distance_along_curve_tolerance (see 4.2.66). The data associated with a Location_tolerance are the following:

— origin_shape;

— termination_shape.

4.2.109.1 origin_shape

The origin_shape specifies the shape on the Part that defines the starting position for a Location _tolerance. See 4.3.110 for the application assertion.

4.2.109.2 termination_shape

The termination_shape specifies the shape on the Part that defines the ending position for a Location_-tolerance. See 4.3.110 for the application assertion.

©ISO 2000 - All rights reserved

4.2.110 Machine_requisition

A Machine_requisition is a type of Requisition (see 4.2.186) that describes machines required to support part manufacturing.

4.2.111 Machining_feature

A Machining_feature is a type of Manufacturing_feature (see 4.2.114) that identifies a volume of material that shall be removed to obtain the final part geometry from the initial stock. Machining_-features requires both direction and location in placing them on a part. Each Machining_feature may be one of the following: Knurl (see 4.2.102), Multi_axis_feature(see 4.2.125),Outer_round (see 4.2.137), Revolved_feature (see 4.2.188), Thread (see 4.2.224), Marking (see 4.2.116), Spherical_cap (see 4.2.209), or a Compound_feature (see 4.2.34). The data associated with a Machining_feature are the following:

— placement;

— usage_name.

4.2.111.1 placement

The placement specifies the position and orientation of a Machining_feature relative to the base shape for a part. See 4.3.111 for the application assertion.

4.2.111.2 usage_name

The usage_name specifies a user defined name that is additional information about the use of a feature. The usage_name need not be specified for a particular Machining_feature.

4.2.112 Manufactured_assembly

A Manufactured_assembly is a type of Part (see 4.2.139) that specifies a collection of individual parts or sub-assembly of parts, with orientation. A Manufactured_assembly is considered a sub-assembly when it is a component in another Manufactured_assembly.

NOTE - Figure 63 illustrates the Manufactured_assembly and Mating_definition

4.2.113 Manufactured_assembly_relationship

A Manufactured_assembly_relationship identifies the Manufactured_assembly, and a component of the assembly. A component is either Single_piece_part or another Manufactured_assembly.

 ©ISO 2000 - All rights reserved

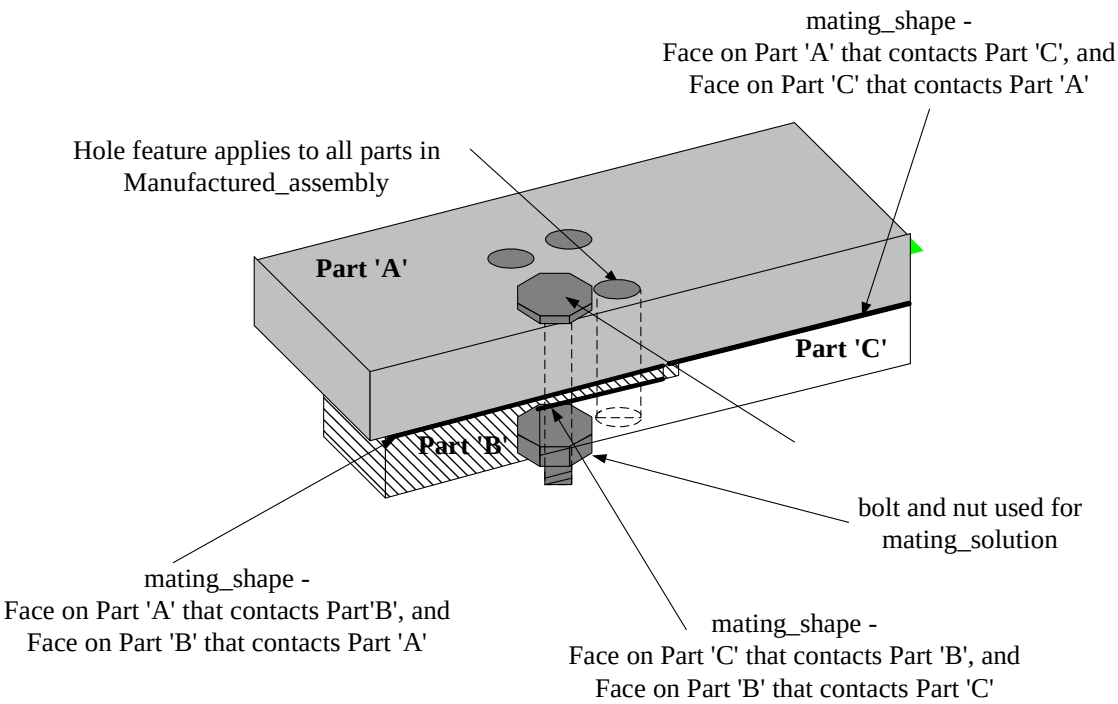


Figure 63 - Manufactured_assembly and Mating_defintion

The data associated with a Manufactured_assembly_relationship are the following:

— assembly;

— component;

— orientation.

4.2.113.1 assembly

The assembly specifies the Manufactured_assembly that shall have a component of Single_piece_part or another Manufactured_assembly. See 4.3.112 for the application assertion.

4.2.113.2 component

The component specifies either a Single_piece_part or another Manufactured_assembly used to define an assembly. See 4.3.113 for the application assertion.

©ISO 2000 - All rights reserved

4.2.113.3 orientation

The orientation specifies the transformation of a Part to define its placement in a manufacturing assembly, or sub-assembly. See 4.3.114 for the application assertion.

4.2.114 Manufacturing_feature

A Manufacturing_feature is a type of Shape_element (see 4.2.200) that identifies the types of features necessary to manufacture a machined part. Each Manufacturing_feature is either a Machining_feature (see 4.2.111), a Replicate_feature (see 4.2.185), or a Transition_feature (see 4.2.234).

4.2.115 Manufacturing_feature_group

A Manufacturing_feature_group specifies the collection of manufacturing features with a usage identification. A Manufacturing_feature_group shall allow for the collection of collections.

EXAMPLE - A Manufacturing_feature_group may be used to group together all of the hold down holes for a part.

The data associated with a Manufacturing_feature_group are the following:

— feature_groups;

— group_description;

— group_name.

4.2.115.1 feature_groups

The feature_group specifies the list of Manufacturing_feature (see 4.2.114) or the Manufacturing_feature_-group (see 4.2.115) to be grouped. See 4.3.115 and 4.3.116 for the application assertion.

4.2.115.2 group_description

The group_description specifes additional text information about the Manufacturing_feature_group.

4.2.115.3 group_name

The group_name specifies identification for the Manufacturing_feature_group.

4.2.116 Marking

A Marking is a type of Machining_feature (see 4.2.111) that is one or more text characters on a surface of a part. Each Marking is either a Defined_marking (see 4.2.54) or a Catalogue_marking (see 4.2.17).

 ©ISO 2000 - All rights reserved

The data associated with a Marking are the following:

— applied_to_shape;

— text.

4.2.116.1 applied_to_shape

The applied_to_shape specifies a base shape for applying the Marking feature. See 4.3.118 for the application assertion.

4.2.116.2 text

The text specifies the characters that will be applied to the part. See 4.3.117 for the application assertion.

4.2.117 Material

A Material is the identification of the raw stock from which a part is produced. Material identifies primary and substitution material. The data associated with a Material are the following:

— material_characteristics;

— material_description;

— material_id;

— material_specification;

— stock_size.

4.2.117.1 material_characteristics

The material_characteristics specifies the properties which define the Material. The material_-characteristics need not be specified for a particular Material. There may be more than one material_-characteristics for a Material. See 4.3.119 for the application assertion.

4.2.117.2 material_description

The material_description specifies a user defined explanation of the material required for the part.

4.2.117.3 material_id

The material_id specifies a word or group of words that make up the unique designation of the material.

©ISO 2000 - All rights reserved

4.2.117.4 material_specification

The material_specification specifies the documentation which contains additional information about Material. The material_specification need not be specified for a particular Material. There may be more than one material_specification for a Material. See 4.3.120 for the application assertion.

4.2.117.5 stock_size

The stock_size specifies the dimensions of the raw material required to make the part. The stock_size need not be specified for a particular Material.

4.2.118 Material_condition_modifier

A Material_condition_modifier is a description of a condition for a part surface, it is an indication whether the maximum material principle is to be applied respectively to the toleranced feature, the datum feature, or both. Tolerances of position are determined by a material condition of a mating surface. The data associated with a Material_condition_modifier are the following:

— material_type.

4.2.118.1 material_type

The material_type specifies the type of condition allowed for the Material_condition_modifier.

EXAMPLE - maximum material condition (MMC), least material condition (LMC) and regardless of feature size (RFS).

4.2.119 Material_property

A Material_property is the information that describes the properties of the material used to produce a part. The data associated with a Material_property are the following:

— material_hardness;

— property_characteristic.

4.2.119.1 material_hardness

The material_hardness specifies additional information to define hardness properties. The material_-hardness need not be specified for a particular Material_property. There may be more than one material_hardness for a Material_property. See 4.3.121 for the application assertion.

4.2.119.2 property_characteristic

The property_characteristic specifies the parameter for the description of Material_property. The property_characteristics need not be specified for a particular Material_property. There may be more than one property_characteristics for a Material_property. See 4.3.122 for the application assertion.

 ©ISO 2000 - All rights reserved

4.2.120 Material_requisition

A Material_requisition is a type of Requisition (see 4.2.186) which describes an order for purchase of raw materials by the manufacturing organization.

4.2.121 Mating_definition

A Mating_definition is a view of a Manufactured_assembly, defining the physical connection of two or more Single_piece_part objects. It includes technical information about the kind of connection. This information is independent from the hierarchical assembly structure.

NOTE - Figure 63 illustrates the Manufactured_assembly and Mating_definition

The data associated with a Mating_definition are the following:

— applied_assembly;

— mating_shape;

— mating_solution;

— mating_type.

4.2.121.1 applied_assembly

An applied_assembly specifies the Manufactured_assembly that contains the Single_piece_parts that have a mating definition. See 4.3.123 for the application assertion.

4.2.121.2 mating_shape

The mating_shape specifies the shape that form the area of mating contact between two single_piece_part objects. See 4.3.124 for the application assertion.

4.2.121.3 mating_solution

The mating_solution specifies additional Single_piece_parts that participate in the Mating_definition. See 4.3.125 for the application assertion.

EXAMPLE - Two parts may be mated together and a nut and bolt used to hold the parts together. The nut and bolt are additional parts used in the mating_solution.

4.2.121.4 mating_type

The mating_type specifies the kind of mating, or how the items shall be mated together.

©ISO 2000 - All rights reserved

4.2.122 Mating_definition_relationship

A Mating_definition_relationship specifies additional information about the mating of two particular Single_piece_parts that go into a Mating_definition. Two Single_piece_part objects that are referenced by the same Mating_relationship object shall refer to the same Mating_definition . The data associated with a Mating_definition_relationship are the following:

— mated_part;

— mating_part_definition;

— orientation.

4.2.122.1 mated_part

The mated_part specifies the single_piece_part that shall have a mating definition with another single_-piece_part. See 4.3.128 for the application assertion.

4.2.122.2 mating_part_definition

The mating_part_definition specifies the mating definition for two single_piece_parts that contact each other in an assembly. See 4.3.126 for the application assertion.

4.2.122.3 orientation

The orientation specifies the transformation of a part to define its placement in a mating definition. A Mating_definition_relationship may but need not require an orientation to be specified. See 4.3.127 for the application assertion.

4.2.123 Mating_relationship

A Mating_relationship defines two single_piece_parts that are in the same Manufactured_assembly and are in contact with each other. The data associated with a Mating_relationship are the following:

— predecessor;

— successor.

4.2.123.1 predecessor

The predecessor specifies the Single_piece_part with the highest precedence. See 4.3.129 for the application assertion.

4.2.123.2 successor

The predecessor specifies the Single_piece_part with the lesser precedence. See 4.3.129 for the application assertion.

 ©ISO 2000 - All rights reserved

4.2.124 Modular_fixture_requisition

A Modular_fixture_requisition is a type of Requisition (see 4.2.186) which describes an order for purchase of modular fixtures by the manufacturing organization.

4.2.125 Multi_axis_feature

A Multi_axis_feature is a type of Machining_feature (see 4.2.111) that identifies milling features for a part, and not turned features. Each Multi_axis_feature may be one of the following: Boss (see 4.2.10), General_removal_volume (see 4.2.90), Hole (see 4.2.99), Rounded_end (see 4.2.194), Planar_face (see 4.2.154), Pocket (see 4.2.160), Profile_feature (see 4.2.165), Protrusion (see 4.2.171), Rib_top (see 4.2.191),Slot (see 4.2.205), and Step (see 4.2.212). The data associated with Multi_axis_feature are the following:

— maximum_feature_limit.

4.2.125.1 maximum_feature_limit

The maximum_feature_limit specifies a planar limitation for a feature. No portion of the feature shall exist beyond this planar definition. The normal to the plane shall be in the direction away from the Machining_feature. See 4.3.130 for the application assertion.

4.2.126 Ngon_base_shape

An Ngon_base_shape is a type of Implicit_base_shape_representation (see 4.2.100) that specifies the initial shape of the material is a polygon with any number of sides.

NOTE - Figure 64 illustrates a Ngon_base_shape.

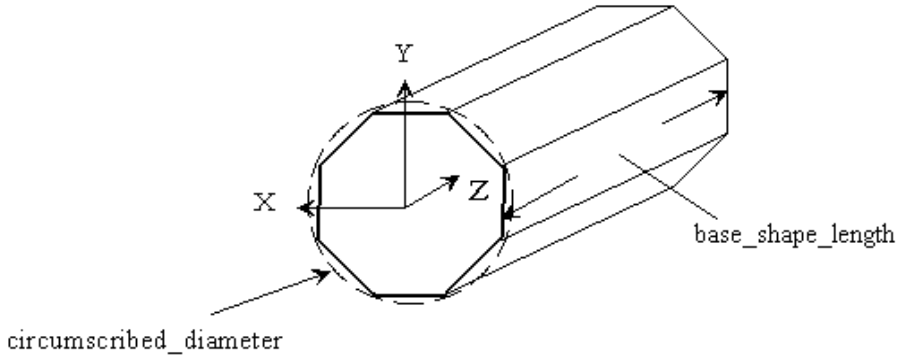


Figure 64 - Ngon_base_shape

©ISO 2000 - All rights reserved

The data associated with a Ngon_base_shape are the following:

— circumscribed_or_across_flats;

— corner_radius;

— diameter;

— number_of_sides.

4.2.126.1 circumscribed_or_across_flats

The circumscribed_or_across_flats specifies the type of diameter being used to define the Ngon_profile. Circumscribed is the diameter that the Ngon_base_shape fits inside of, with the corners on the circle that defines the diameter. Across flats, is the diameter that fits inside of the Ngon_profile with the sides of the shape being tangent to the circle that defines the diameter.

4.2.126.2 corner_radius

The corner_radius specifies the size of an arc blend between two sides of the ngon. See 4.3.131 for the application assertion.

4.2.126.3 diameter

The diameter specifies specifies the size of either the circumscribed diameter, or the diameter across the flats. See 4.3.131 for the application assertion.

4.2.126.4 number_of_sides

The number_of_sides specifies how many sides are needed for the Ngon. See 4.3.131 for the application assertion.

4.2.127 Ngon_profile

An Ngon_profile is a type of Closed_profile (see 4.2.31) that is an enclosed area bounded by three or more connected straight line sides. The orientation is at the center of the profile with one side of the ngon parallel to the X-axis crossing the Y-axis at a negative value.

NOTE - Figure 65 illustrates the Ngon_profile.

 ©ISO 2000 - All rights reserved

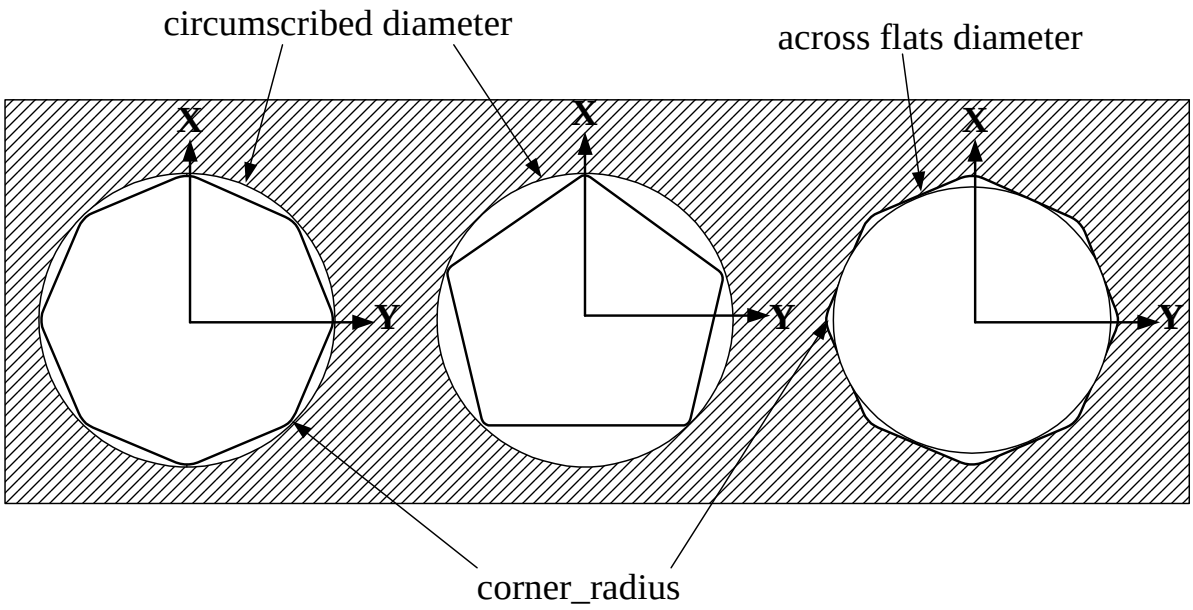


Figure 65 - Ngon_profile

The data associated with a Ngon_profile are the following:

— circumscribed_or_across_flats;

— corner_radius;

— diameter;

— number_of_sides.

4.2.127.1 circumscribed_or_across_flats

The circumscribed_or_across_flats specifies the type of diameter being used to define the Ngon_profile. Circumscribed is the diameter that the Ngon_base_shape fits inside of, with the corners on the circle that defines the diameter. Across flats, is the diameter that fits inside of the Ngon_profile with the sides of the shape being tangent to the circle that defines the diameter.

4.2.127.2 corner_radius

The corner_radius specifies the size of an arc blend between two sides of the ngon. See 4.3.132 for the application assertion.

©ISO 2000 - All rights reserved

4.2.127.3 diameter

The diameter specifies specifies the size of either the circumscribed diameter, or the diameter across the flats. See 4.3.132 for the application assertion.

4.2.127.4 number_of_sides

The number_of_sides specifies how many sides are needed for the Ngon. See 4.3.132 for the application assertion.

4.2.128 Numeric_parameter

A Numeric_parameter is a type of Property_parameter (see 4.2.170) that a numeric value with units of the property being defined. A Numeric_parameter is either a Numeric_parameter or a Numeric_-parameter_with_tolerance (see 4.2.129). The data associated with a Numeric_parameter are the following:

— parameter_units;

— parameter_value.

4.2.128.1 parameter_units

The parameter_units specifies the quantity of measure in which the value is given.

EXAMPLE - watt, meters, degrees, etc.

4.2.128.2 parameter_value

The parameter_value specifies the numeric amount associated with the units of a specific characteristic of interest.

4.2.129 Numeric_parameter_with_tolerance

A Numeric_parameter_with_tolerance is a type of Numeric_parameter (see 4.2.128) with an implied tolerance value.

NOTE - A thread has a implicit definition for the minor_diameter attribute. This attribute has no explicit geometry definition, so the dimensional tolerance of this attribute is represented with Numeric_parameter_with_tolerance.

The data associated with a Numeric_parameter_with_tolerance are the following:

— implicit_tolerance.

4.2.129.1 implicit_tolerance

The implicit_tolerance specifies the type of tolerance to apply to a numeric parameter value. See 4.3.133 and 4.3.134 for the application assertion.

 ©ISO 2000 - All rights reserved

4.2.130 Open_profile

An Open_profile is a type of Profile (see 4.2.164) that is an outline or shape with no enclosing or confining bounds. The open ends of the profile may extend infinitely. Each Open_profile is either a General_open_profile (see 4.2.83), Linear_profile (see 4.2.105), Partial_circular_profile (see 4.2.144), Rounded_U_profile (see 4.2.195), Square_U_profile (see 4.2.211), Tee_profile (see 4.2.223), or a Vee_-profile (see 4.2.236). The data associated with an Open_profile are the following:

— profile_limit.

4.2.130.1 profile_limit

The profile_limit specifies a planar limitation for a feature. No portion of the profile shall exist beyond this planar definition. The normal to the plane shall be in the direction away from the Open_profile. See 4.3.135 for the application assertion.

4.2.131 Open_slot_end_type

An Open_slot_end_type is a type of Slot_end_type (see 4.2.206) that is an end condition of a slot that shall pass through the end of the part.

NOTE - Figure 66 illustrates a slot with two Open_slot_end_type objects.

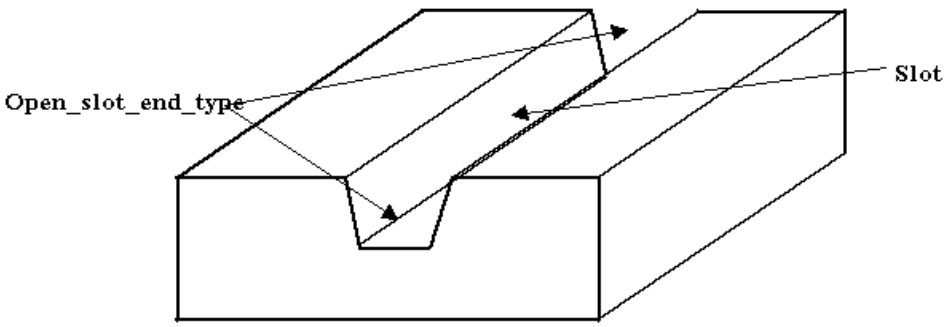


Figure 66 - Open_slot_end_type

4.2.132 Ordered_part

An Ordered_part specifies the number of machined parts, of a particular order number, a Customer_order is requesting. The data associated with an Ordered_part are the following:

— quantity_required;

— quantity_unit_of_measure.

©ISO 2000 - All rights reserved

4.2.132.1 quantity_required

The quantity_required specifies the number of items to be manufactured.

4.2.132.2 quantity_unit_of_measure

The quantity_unit_of_measure specifies the units in terms of which the part is expressed.

EXAMPLE - Examples of units of measure are:each, meter, watt, etc.

4.2.133 Organization

An Organization is a functional group that is involved in the production of a part. The data associated with an Organization are the following:

— organization_address;

— organization_id;

— organization_name.

4.2.133.1 organization_address

The organization_address specifies a location at which an organization exists and may be contacted. This location may be physical or electronic.

4.2.133.2 organization_id

The organization_id specifies the unique identification of an organization.

4.2.133.3 organization_name

The organization_name specifies a word or group of words by which an organization is commonly referred. This name need not be unique within an organization.

4.2.134 Orientation

An Orientation is the direction and location of the basic shape of a part, feature on the part, or of the component of a feature which are Profile objects and Path objects. The data associated with an Orientation are the following:

— axis;

— location.

 ©ISO 2000 - All rights reserved

4.2.134.1 axis

The axis specifies a line in 3D space about which the part or portions of the part are arranged.

4.2.134.2 location

The location specifies a point in 3D space used to position the part or portions of the part.

4.2.135 Outer_diameter

An Outer_diameter is a type of Outer_round (see 4.2.137) that is a sweeping of an outline specified by a line segment one complete revolution about an axis. The line is finite in length, coplanar with the axis. An Outer_diameter may have a constant diameter around the axis of rotation, or it may be tapered.

NOTE - Figure 67 illustrates the Outer_diameter with and without a taper.

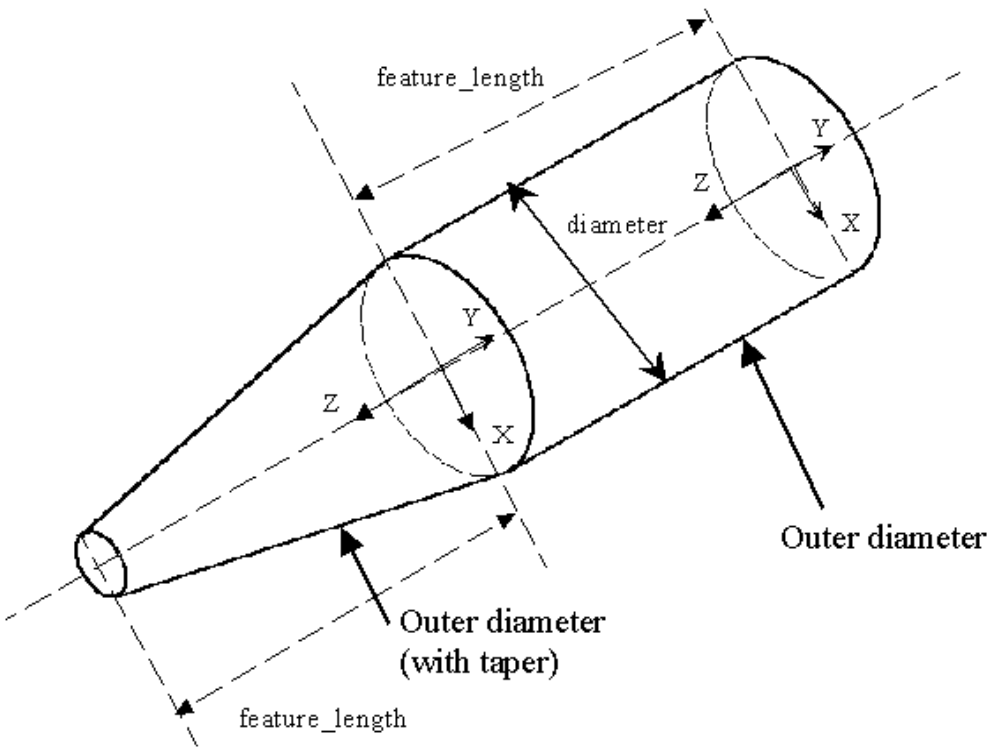


Figure 67 - Outer_diameter

©ISO 2000 - All rights reserved

The data associated with an Outer_diameter are the following:

— diameter;

— feature_length;

— reduced_size.

4.2.135.1 diameter

The diameter specifies the maximum diametric size of an Outer_diameter feature. See 4.3.136 for the application assertion.

4.2.135.2 feature_length

The length specifies the size of a Outer_diameter feature, measured along the feature's axis.
See 4.3.136 for the application assertion.

4.2.135.3 reduced_size

The reduced_size specifies the constant change in the Outer_diameter along the feature length. See 4.3.137 and 4.3.138, 4.3.139 for the application assertion.

4.2.136 Outer_diameter_to_shoulder

An Outer_diameter_to_shoulder is a type of Outer_round (see 4.2.137) that is a sweeping of a shape one complete revolution about an axis. The shape shall be specified by two lines that connect at a point and extend infinitely. The enclosed angle shall be smaller than a straight angle. The intersection of the two lines need not be blended with a radius.

NOTE - Figure 68 illustrates the Outer_diameter_to_shoulder.

The data associated with an Outer_diameter_to_shoulder are the following:

— diameter;

— v_shape_boundary.

4.2.136.1 diameter

The diameter specifies the size of the part at the point of the Vee, or where the two sides come together, swept about an axis of rotation. See 4.3.140 for the application assertion.

 ©ISO 2000 - All rights reserved

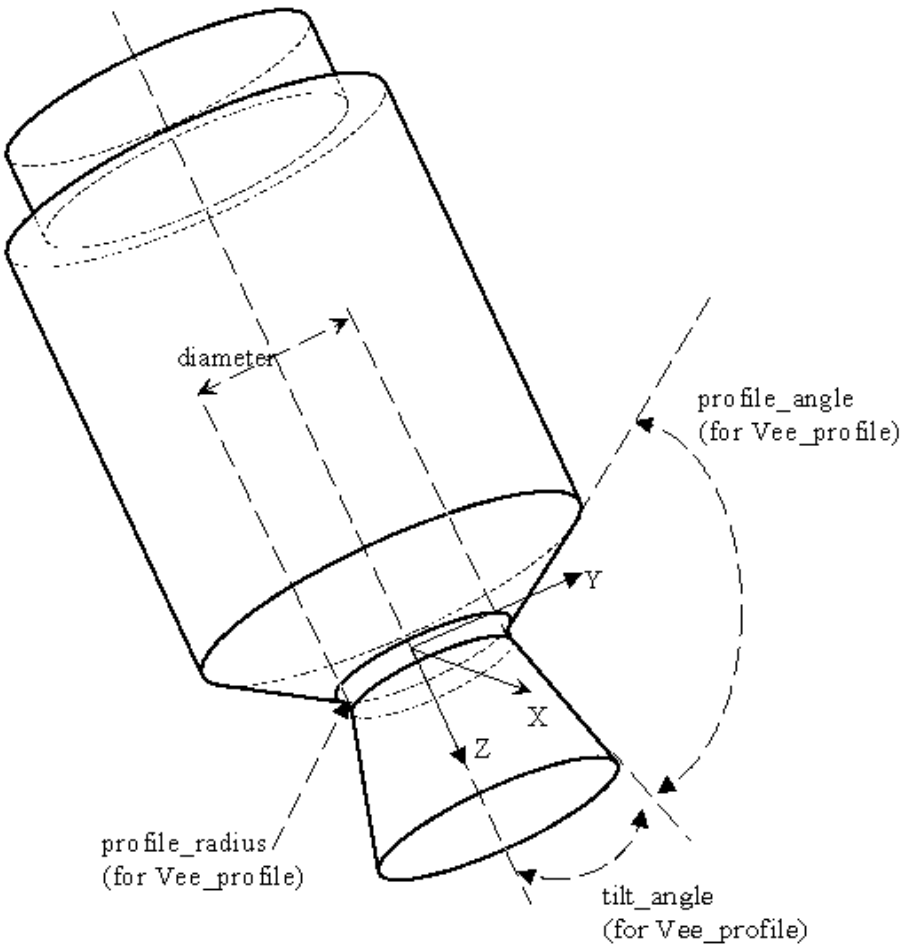


Figure 68 - Outer_diameter_to_shoulder

4.2.136.2 v_shape_boundary

The v_shape_boundary specifies an outline or shape that shall be revolved about an axis. The Vee_profile specifies the revolved shape required by an Outer_diameter_to_shoulder. The placement of the profile shall be along the X-axis of the Outer_diameter_to_shoulder at a specified distance away from the origin. The orientation of the Y-axis of the Vee_profile shall be the same as the Y-axis of the Outer_diameter_to_shoulder and the X-axis of the Vee_profile shall be the same as the Z-axis of the Outer_diameter_to_shoulder. See 4.3.141 for the application assertion.

4.2.137 Outer_round

An Outer_round is a type of Machining_feature (see 4.2.111) that is an outline or significant shape that is swept through a complete revolution about an axis. Each Outer_round is either an Outer_diameter (see 4.2.135) or an Outer_diameter_to_shoulder (see 4.2.136). The axis of revolution shall be the same as the Z-axis of the feature.

©ISO 2000 - All rights reserved

4.2.138 Parallelism_tolerance

A Parallelism_tolerance is a type of Geometric_tolerance (see 4.2.95) that is the condition of a surface equidistant at all points from a datum plane or an axis equidistant along its length to a datum axis or plane. A Parallelism_tolerance specifies one of the following:

— A tolerance zone defined by two planes or lines parallel to a datum plane, or axis, within which the line elements of the surface or axis shall lie.

— A cylindrical tolerance zone whose axis is parallel to a datum axis within which the axis shall lie

NOTE 1- Figure 69 illustrates Parallelism_tolerance for a plane.

NOTE 2 - Figure 70 illustrates Parallelism_tolerance for an axis.

NOTE 3 - The Parallelism_tolerance definition is derived from paragraph 14.7 of ISO 1101.

The data associated with a Parallelism_tolerance are the following:

— affected_plane;

— geometric_reference;

— segment_size.

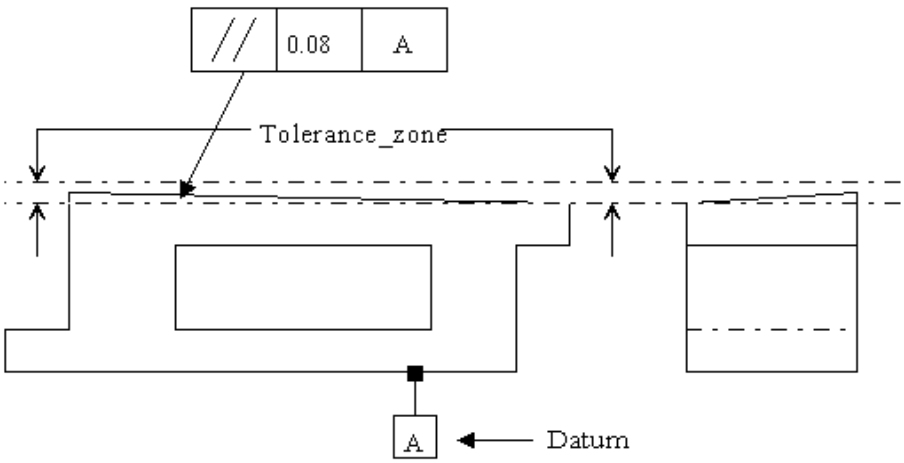


Figure 69 - Parallelism_tolerance for a plane

 ©ISO 2000 - All rights reserved

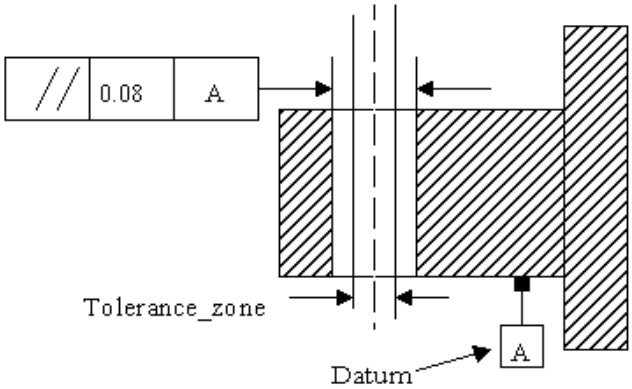


Figure 70 - Parallelism_tolerance for an axis

4.2.138.1 affected_plane

The affected_plane specifies the plane to apply the tolerance value. The affected_plane is equivalent to a 2D drawing view. The affected_plane need not be specified for a particular Parallelism_tolerance. See 4.3.143 for the application assertion.

4.2.138.2 geometric_reference

The geometric_reference specifies the datum to which the tolerance is related. See 4.3.142 for the application assertion.

4.2.138.3 segment_size

The segment_size specifies the length of a surface to apply a tolerance if the Parallelism_tolerance is not applied to the total length. A segment_size need not be specified for a particular Parallelism_tolerance.

4.2.139 Part

A Part is the physical item which is intended to be produced through the manufacturing process. Each Part may be one of the following: Manufactured_assembly (see 4.2.112), or Single_piece_part (see 4.2.203). The data associated with a Part are the following:

— manufacture_authorization;

— manufactured_by_organization;

— manufactured_by_person;

— owned_by_organization;

©ISO 2000 - All rights reserved

— owned_by_person;

— part_description;

— part_id;

— part_name;

— part_revision_id;

— physical_form;

— property_characteristics;

— quantity_ordered;

— security_classification.

4.2.139.1 manufacture_authorization

The manufacture_authorization specifies the approval description for the manufacture of a Part. The manufacture_authorization need not be specified for a particular Part. There may be more than one manufacture_authorization for a Part. See 4.3.144 for the application assertion.

4.2.139.2 manufactured_by_organization

The manufactured_by_organization specifies the organization responsible for the manufacture of the Part. There may be more than one manufactured_by_organization for a Part. See 4.3.146 for the application assertion.

4.2.139.3 manufactured_by_person

The manufactured_by_person specifies the person responsible for the manufacture of the Part. There may be more than one manufactured_by_person for a Part. See 4.3.147 for the application assertion.

4.2.139.4 owned_by_organization

The owned_by_organization specifies the organization that will own the completed manufactured Part There may be more than one owned_by_organization for a Part. See 4.3.146 for the application assertion.

4.2.139.5 owned_by_person

The owned_by_person specifies the person that will own the completed manufactured Part. There may be more than one owned_by_person for a Part. See 4.3.147 for the application assertion.

 ©ISO 2000 - All rights reserved

4.2.139.6 part_description

The part_description specifies in a textural format, human interpretable summary of the Part's characteristics which is appropriate for the manufacturing of the part. Any comments or requirements on part characteristics may be given as a portion of the part_description.

4.2.139.7 part_id

The part_id specifies a unique identifier for a Part within an organization.

4.2.139.8 part_name

The part_name specifies a word or group of words by which a Part commonly is called within an organization. This name need not be unique within an organization.

4.2.139.9 part_revision_id

The part_revision_id specifies a unique identifier that defines the appropriate level of change that is incorporated into the design of the Part for the manufacturing function. The part_revision_id need not specify the latest version of a Part's design to be manufactured.

4.2.139.10 physical_form

The physical_form specifies the shape of the Part. See 4.3.149 for the application assertion.

4.2.139.11 property_characteristic

The property_characteristic specifies the associated data about the physical structure (see 4.2.169). The property characteristic need not be specified for a particular Part. There may be more than one property characteristic for a Part. See 4.3.148 for the application assertion.

4.2.139.12 quantity_ordered

The quantity_ordered specifies the number of Parts to be manufactured. There may be more than one quantity_ordered for a Part. See 4.3.145 for the application assertion.

4.2.139.13 security_classification

The security_classification specifies an organizational, national or international code that defines the availability of the Part or information about the Part to a particular individual, group of individuals, organization or group of organizations.

©ISO 2000 - All rights reserved

4.2.140 Part_placement

The Part_placement is the transformation of part shape from the originating orientation to define manufacturing assembly and sub-assembly. The data associated with a Part_placement are the following:

— originating_orientation;

— oriented_physical_form;

— resulting_orientation.

4.2.140.1 originating_orientation

The originating_orientation specifies the orientation of a part or a sub-assembly prior to being positioned in an assembly or another sub_assembly. See 4.3.150 for the application assertion.

4.2.140.2 oriented_physical_form

The oriented_physical_form specifies the shape of a part or sub-assembly that is being re-positioned. See 4.3.151 for the application assertion.

4.2.140.3 resulting_orientation

The resulting_orientation specifies the orientation of a part or sub-assembly in an assembly or another sub_assembly. See 4.3.150 for the application assertion.

4.2.141 Part_property

A Part_property is a specific characteristic about the form, fit, or function of a part. The data associated with a Part_property are the following:

— property_characteristic.

4.2.141.1 property_characteristic

The property_characteristic specifies the parameter to describe the Part_property. The property_-characteristic need not be specified for a particular Part_property. There may be more than one property_characteristic for a Part_property. See 4.3.152 for the application assertion.

4.2.142 Partial_area_definition

A Partial_area_definition is the limitations of a surface for applying a Machining_feature. Thread and Knurl features are applied to cylindrical shapes. Partial_area_definition places a limitation on how much and where to apply the feature on the cylindrical shape.

NOTE - Figure 35 illustrates a Defined_thread with a Partial_area_definition. The drawing call out '5.12 MIN.THREAD' defines the amount of cylindrical shape that has the thread applied.

 ©ISO 2000 - All rights reserved

The data associated with a Partial_area_definition are the following:

— effective_length;

— maximum_length;

— placement.

4.2.142.1 effective_length

The effective_length specifies the length of the thread which is usable by the feature. When applied to a knurl the effective_length defines the overall length of the knurl. The effective_length is not required for a Knurl feature. See 4.3.153 for the application assertion.

4.2.142.2 maximum_length

The maximum_length specifies the dimension along a surface to apply a feature. A Thread is a type of Machining_features that is applied to a surface. The dimensional distance limits the length along the surface axis for defining these feature objects. A knurl does not require maximum_length. The maximum_length need not be specified for a particular Partial_area_definition. See 4.3.153 for the application assertion.

4.2.142.3 placement

The placement specifies where to locate the Partial_area_definition. See 4.3.154 for the application assertion.

4.2.143 Partial_circular_path

A Partial_circular_path is a type of Circular_path (see 4.2.27) that is a direction of travel along an arc of constant radius around an axis. The path shall begin and end at different points on the arc.

NOTE - Figure 71 illustrates two Slot feature with a Square_U_profile and a Partial_circular_paths.

The data associated with a Partial_circular_path are the following:

— sweep_angle.

4.2.143.1 sweep_angle

The sweep_angle specifies the size of the angle to define an arc shaped path. See 4.3.155 for the application assertion.

©ISO 2000 - All rights reserved

4.2.144 Partial_circular_profile

A Partial_circular_profile is a type of Open_profile (see 4.2.130) that is specified by an arc. The arc shall be a constant radius swept about a point. The orientation of the profile shall be positioned at the origin of the arc, with one end point of the arc on the X-axis.

NOTE - Figure 72 illustrates a Partial_circular_profile.

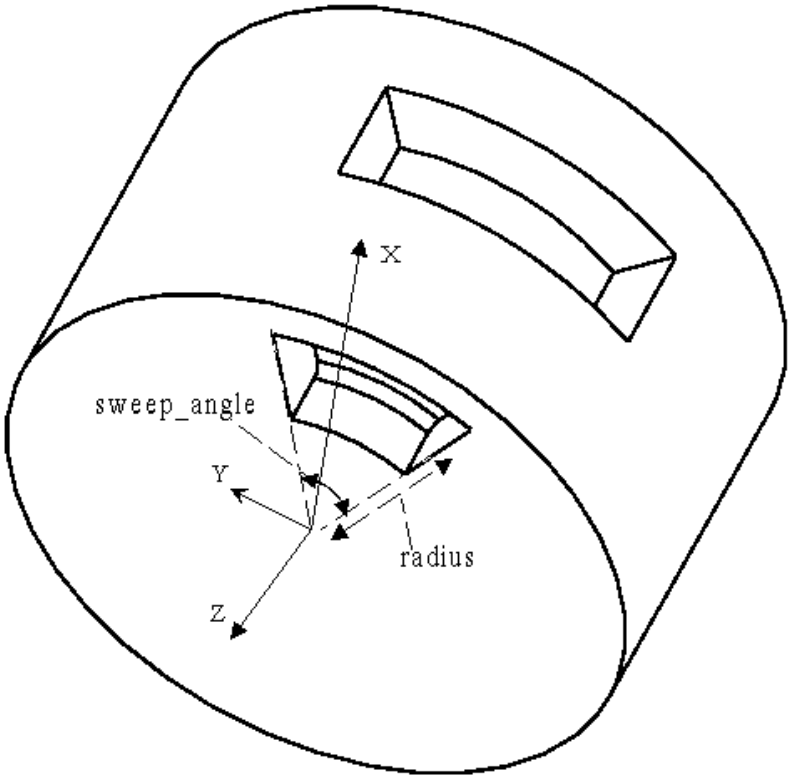


Figure 71 - Partial_circular_path

 ©ISO 2000 - All rights reserved

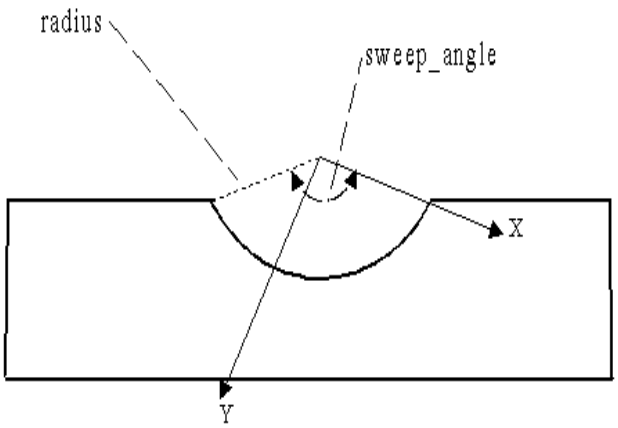


Figure 72 - Partial_circular_profile

The data associated with a Partial_circular_profile are the following:

— radius;

— sweep_angle.

4.2.144.1 radius

The radius specifies the size of the arc to define a Partial_circular_profile. See 4.3.156 for the application assertion.

4.2.144.2 sweep_angle

The sweep_angle specifies the size of the angle to define an circular shaped profile. See 4.3.156 for the application assertion.

4.2.145 Partial_circular_shape_profile

A Partial_circular_shape_profile is a type of Shape_profile (see 4.2.201) that defines a volume that is not enclosed on all sides. The data associated with a Partial_circular_shape_profile are the following:

— open_boundary.

©ISO 2000 - All rights reserved

4.2.145.1 open_boundary

The open_boundary specifies the outline of the Shape_profile feature. The outline defines an area that shall be circular and shall not be enclosed. The placement of the open_boundary shall be with the origin of the Path, that defines the profile, at the origin of the Partial_circular_shape_profile The X-axis and Y-axis of the profile shall be the same as the X-axis and Y-axis of the Partial_circular_shape_profile. See 4.3.157 for the application assertions.

4.2.146 Path

A Path is a continuous set of curves that define a direction of travel. These curves do not intersect or duplicate themselves. A Path shall have its own orientation which may be the same orientation as the Machining_feature which requires it as a part of a feature definition. The orientation of a Path may be the same orientation as the Machining_feature. Each Path is either a Circular_path (see 4.2.27), General_path (see 4.2.85), or a Linear_path (see 4.2.104).The data associated with a Path are the following:

— placement.

4.2.146.1 placement

The placement specifies where to locate the Path. See 4.3.158 for the application assertion.

4.2.147 Path_element

A Path_element is a type of Shape_element (see 4.2.200) that is a continous set of geometric curve that represent the path for a particular Machining_feature.

4.2.148 Pedigree_creation_order

A Pedigree_creation_order is a document that defines the requirements for data to be gathered, documented and archived about a particular instance of a Part or batch of Parts throughout the manufacturing process. The data associated with a Pedigree_creation_order are the following:

— order_id.

4.2.148.1 order_id

The order_id specifies a unique identifier for the Pedigree_creation_order.

4.2.149 Perpendicularity_tolerance

A Perpendicularity_tolerance is a type of Geometric_tolerance (see 4.2.95) that is the amount of deviation of a surface to a perpendicular to a datum. The actual surface shall lie within two planes a tolerance apart and perpendicular to a datum surface.

NOTE 1 - Figure 73 illustrates a Perpendicularity_tolerance.

 ©ISO 2000 - All rights reserved

NOTE 2 - The Perpendicularity_tolerance definition is derived from paragraph 14.8 of ISO 1101.

The data associated with a Perpendicularity_tolerance are the following:

— affected_plane;

— geometric_reference;

— segment_size.

4.2.149.1 affected_plane

The affected_plane specifies the plane to apply the tolerance value. The affected_plane is equivalent to a 2D drawing view. The affected_plane need not be specified for a particular Perpendicularity_-tolerance. See 4.3.160 for the application assertion.

4.2.149.2 geometric_reference

The geometric_reference specifies the datum to which the tolerance is related. See 4.3.159 for the application assertion.

4.2.149.3 segment_size

The segment_size specifies the length of a surface to apply a tolerance if the Perpendicularity_tolerance is not applied to the total length. The segment_size need not be specified for a particular Perpendicularity_tolerance.

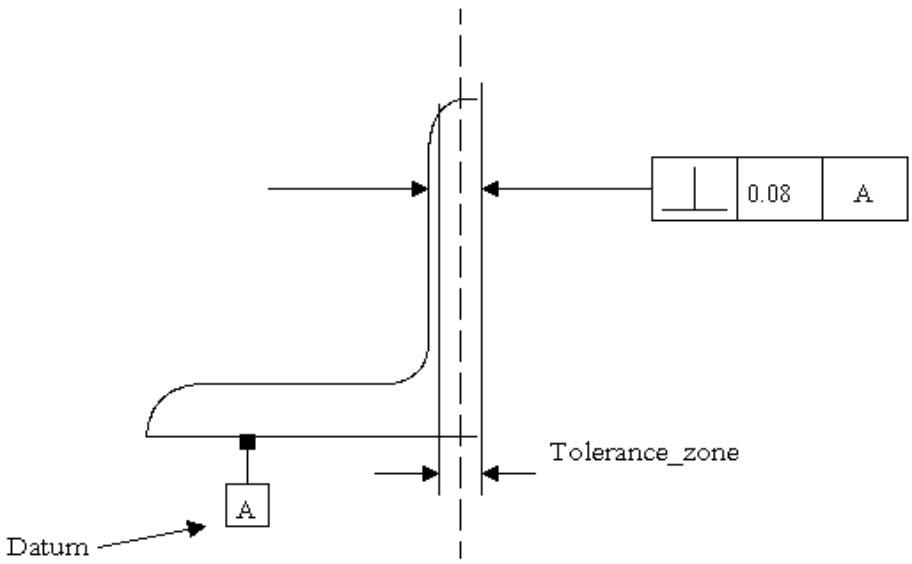


Figure 73 - Perpendicularity_tolerance

©ISO 2000 - All rights reserved

4.2.150 Person

A Person is an individual human being. The data associated with a Person are the following:

— person_address;

— person_id;

— person_name;

— person_phone_number.

4.2.150.1 person_address

The person_address specifies a location at which a Person is contacted. This location may be physical or electronic.

4.2.150.2 person_id

The person_id specifies the unique identification of an individual within the context to which that individual is being referenced.

EXAMPLE - Within the United States individuals have unique social security numbers. Within a particular organization employing individuals, each individual may have a unique employee id.

4.2.150.3 person_name

The person_name specifies a word or group of words by which a Person is commonly referred. The person_name may be comprised of compound elements that are commonly referred to as first name, last name, middle names and titles or suffixes.

4.2.150.4 person_phone_number

The person_phone_number specifies the telephone number at which the Person can be reached.

4.2.151 Person_in_organization

A Person_in_organization is a person within the context of an organization. The data associated with a Person_in_organization are the following:

— company;

— employee;

— role.

 ©ISO 2000 - All rights reserved

4.2.151.1 company

The company specifies the functional group that employ's the Person. See 4.3.161 for the application assertion.

4.2.151.2 employee

The employee specifies Person employed by the Organization. See 4.3.162 for the application assertion.

4.2.151.3 role

The role specifies the purpose that the person is fulfilling in an organization.

4.2.152 Placed_target

A Placed_target is a type of Datum_target (see 4.2.51) that is the implicit definition of a Datum_target. A Placed_target is either a Target_point (see 4.2.221), Target_line (see 4.2.220), Target_rectangle (see 4.2.222), or a Target_circle (see 4.2.219). The data associated with Placed_target are the following:

— placement.

4.2.152.1 placement

The placement specifies location and orientation for the implicit definitions of the types of Placed_target. See 4.3.163 for the application assertion.

4.2.153 Planar_element

A Planar_element is a type of Shape_element (see 4.2.200) that is a flat surface. The data associated with a Planar_element are the following:

— location;

— normal.

4.2.153.1 location

The location specifies the position of the planar surface. See 4.3.165 for the application assertion.

4.2.153.2 normal

The normal specifies the vector which indicates the normal of a plane being defined for the planar surface. See 4.3.164 for the application assertion.

©ISO 2000 - All rights reserved

4.2.154 Planar_face

A Planar_face is a type of Multi_axis_feature(see 4.2.125) that is an unbounded planar cut of a part. The Planar_face shall have an orientation such that the Z-axis is the direction away from the part.

NOTE - Figure 74 illustrates the Planar_face.

The data associated with a Planar_face are the following:

— course_of_travel;

— face_boundary;

— removal_boundary;

— removal_direction.

4.2.154.1 course_of_travel

The course_of_travel specifies a straight line with magnitude and direction. The placement and orientation of the Linear_path shall be the same as the Planar_face feature. See 4.3.168 for the application assertion.

4.2.154.2 face_boundary

The face_boundary specifies the complete or partial outside final shape of the part after the planar cut has been applied. A Planar_face may but need not require face_boundary to be defined. See 4.3.166 for the application assertion.

4.2.154.3 removal_boundary

The removal boundary specifies a line with direction and magnitude that when swept along a path defines the area on a part for volume removal. The orientation and placement of the Linear_profile shall be the same as the Planar_face. See 4.3.169 for the application assertion.

4.2.154.4 removal_direction

The removal_direction specifies the direction of material removal from the Planar_face feature. See 4.3.167 for the application assertion.

 ©ISO 2000 - All rights reserved

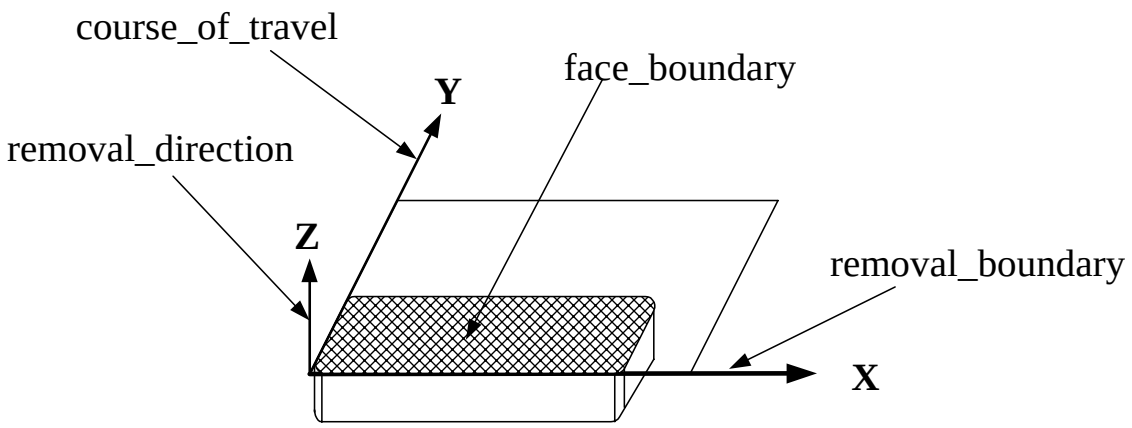


Figure 74 - Planar_face

4.2.155 Planar_pocket_bottom_condition

A Planar_pocket_bottom_condition is a type of Pocket_bottom_condition (see 4.2.161) that characterizes the bottom of a pocket which is flat.

NOTE - Figure 86 illustrates the Rectangular_open_pocket with a Planar_pocket_bottom_condition. Figure 82 illustrates a Rectangular_closed_pocket with a Planar_pocket_bottom_condition.

The data associated with a Planar_pocket_bottom_condition are the following:

— floor_location;

— floor_normal;

— floor_radius.

4.2.155.1 floor_location

The floor_location specifies the position of the bottom of a pocket feature for a planar floor. See 4.3.171 for the application assertion.

NOTE - Figure 86 illustrates the Rectangular_open_pocket with a Planar_pocket_bottom_condition and a floor_-location.

4.2.155.2 floor_normal

The floor_normal specifies the vector which indicates the normal of a plane being defined for the bottom of a pocket. See 4.3.170 for the application assertion.

NOTE - Figure 86 illustrates the Rectangular_open_pocket with a Planar_pocket_bottom_condition and a floor_-normal.

©ISO 2000 - All rights reserved

4.2.155.3 floor_radius

The floor_radius specifies the amount of curvature for an arc between the bottom and the sides of a Pocket feature. See 4.3.172 for the application assertion.

4.2.156 Planar_profile_floor

A Planar_profile_floor is a type of Profile_floor (see 4.2.166) that characterizes the bottom of a Shape_profile feature which is flat.

NOTE - Figure 94 illustrates a Shape_profile with a Planar_profile_floor.

The data associated with a Planar_profile_floor are the following:

— floor.

4.2.156.1 floor

A floor specifies a planar floor for the Shape_profile (see 4.2.201) feature. The normal to the plane shall be in the direction away from the Planar_profile_floor feature. See 4.3.173 for the application assertion.

4.2.157 Planar_rib_top_floor

A Planar_rib_top_floor is a type of Rib_top_floor (see 4.2.192) that is the bottom of a Rib_top (see 4.2.191) which is flat. The data associated with a Planar_rib_top_floor are the following:

— boundary;

— floor_face.

4.2.157.1 boundary

The boundary specifies the complete or partial outside final shape of the Rib_top feature. See 4.3.174 for the application assertion.

4.2.157.2 floor_face

A floor_face specifies a planar floor for the Rib_top (see 4.2.191) feature. The normal to the plane shall be in the direction away from the Rib_top feature. See 4.3.175 for the application assertion.

4.2.158 Planar_top_condition

A Planar_top_condition is a type of Boss_top_condition (see 4.2.11) for a Boss that is flat.

NOTE - Figure 13 illustrates a Circular_boss with a Planar_top_condition.

 ©ISO 2000 - All rights reserved

The data associated with a Planar_top_condition are the following:

— top_location;

— top_normal.

4.2.158.1 top_location

The top_location specifies the position of the top of a boss feature for a planar top. See 4.3.177 for the application assertion.

4.2.158.2 top_normal

The top_normal specifies the vector that indicates the normal of a plane being defined for the top of a boss. See 4.3.176 for the application assertion.

4.2.159 Plus_minus_value

The Plus_minus_values is the upper and lower limits or tolerance value applied directly to a dimension. When applied to a Dimensional_tolerance, the dimensional_value is the tolerance value. When applied to a Numeric_parameter_with_tolerance, the parameter_value is the tolerance value.

EXAMPLE - An illustration of Dimensional_tolerance with a Plus_minus_value is 10 +.005 / -.002.

The data associated with a Plus_minus_value are the following:

— lower_limit;

— significant_digits;

— upper_limit.

4.2.159.1 lower_limit

The lower_limit specifies the low limit value.

4.2.159.2 significant_digits

The significant_digits specifies the number of decimal places indicating the accuracy of a dimension or tolerance.

4.2.159.3 upper_limit

The upper_limit specifies the high limit value.

©ISO 2000 - All rights reserved

4.2.160 Pocket

A Pocket is a type of Multi_axis_feature(see 4.2.125) that is a volume with a specific shape, removed from the part. The sides of a pocket may be parallel to the pocket's orientation vector coming out of the pocket or the sides may be tapered. The placement may be at the bottom of the Pocket with the Z-axis in the direction out of the pocket or at the top of the pocket with the Z-axis in the direction into the pocket. Each Pocket is either a Cutout (see 4.2.45), General_pocket (see 4.2.87), Recess (see 4.2.174), Rectangular_closed_pocket (see 4.2.176), or a Rectangular_open_pocket (see 4.2.181).

The data associated with a Pocket are the following:

— base_radius;

— bottom_condition;

— change_in_boundary;

— pocket_depth.

4.2.160.1 base_radius

The base_radius specifies a radius shape blend between a Pocket and the surrounding Part surface at the top of the Pocket. See 4.3.181 for the application assertion.

4.2.160.2 bottom_condition

The bottom_condition specifies the shape of the bottom of a Pocket feature. See 4.3.182 and 4.3.183 for the application assertions.

4.2.160.3 change_in_boundary

The change_in_boundary specifies a taper that defines the change in shape of the Pocket. The change_in_boundary need not be specified for a particular Pocket. See 4.3.178 and 4.3.79, 4.3.179, 4.3.208 for the application assertion.

4.2.160.4 pocket_depth

The pocket_depth specifies an measured distance from the bottom of a pocket to a point that is outside of the pocket feature. Pocket_depth places a limitation on the Pocket definition so not to interfere with other features that might be nearby. The placement and orientation of the Linear_path that defines pocket_depth shall be the same as the Pocket feature. See 4.3.180 for the application assertion.

EXAMPLE - If a portion of the part should extend over the top of the pocket feature, the depth value would not interfere with it.

 ©ISO 2000 - All rights reserved

4.2.161 Pocket_bottom_condition

A Pocket_bottom_condition specifies the bottom state for a pocket. The Pocket bottom may be flat, or any arbitrary shape, or the pocket may pass through the part. Each Pocket_bottom_condition is either a General_pocket_bottom_condition (see 4.2.88), Planar_pocket_bottom_condition (see 4.2.155), or a Through_pocket_bottom_condition (see 4.2.226).

The data associated with a Pocket_bottom_condition are the following:

— start_or_end.

4.2.161.1 start_or_end

The start_or_end specifies a boolean value of TRUE if the Pocket_bottom_condition is positioned at the start of the of the defining Linear_path (see 4.2.104) for the Pocket, and a value of FALSE if it is at the end of the of the defining Linear_path for the Pocket.

NOTE - Figure 75 illustrates the start_or_end attribute for a Rectangular_open_pocket

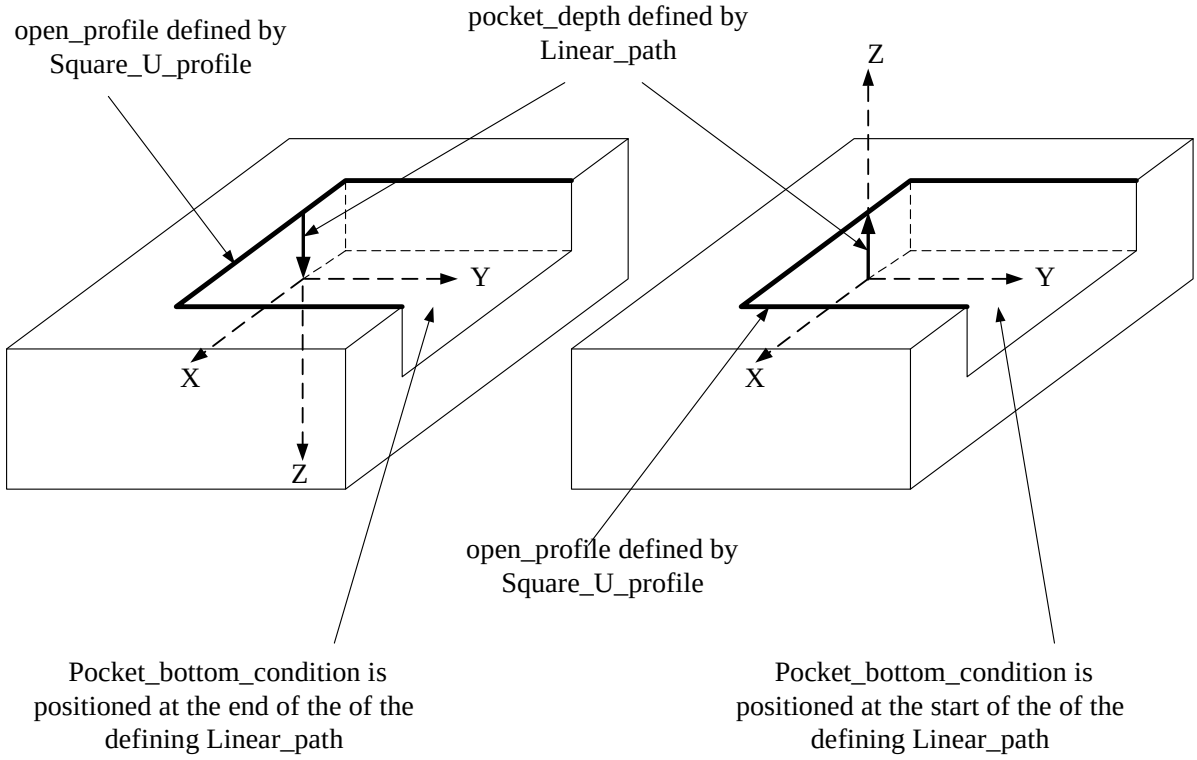


Figure 75 - Pocket_bottom_condition with start_or_end attribute

©ISO 2000 - All rights reserved

4.2.162 Position_tolerance

A Position_tolerance is a type of Geometric_tolerance (see 4.2.95) that denotes a tolerance zone for a theoretically exact position_tolerance of a surface, and is established with respect to a datum.

NOTE 1 - Figure 76 illustrates the Position_tolerance.

NOTE 2 - The Position_tolerance definition is derived from paragraph 14.10 of ISO 1101.

The data associated with a Position_tolerance are the following:

— affected_plane;

— geometric_reference;

— value_qualifier.

4.2.162.1 affected_plane

The affected_plane specifies the plane to apply the tolerance value. The affected_plane is equivalent to a 2D drawing view. The affected_plane need not be specified for a particular Position_tolerance. See 4.3.185 for the application assertion.

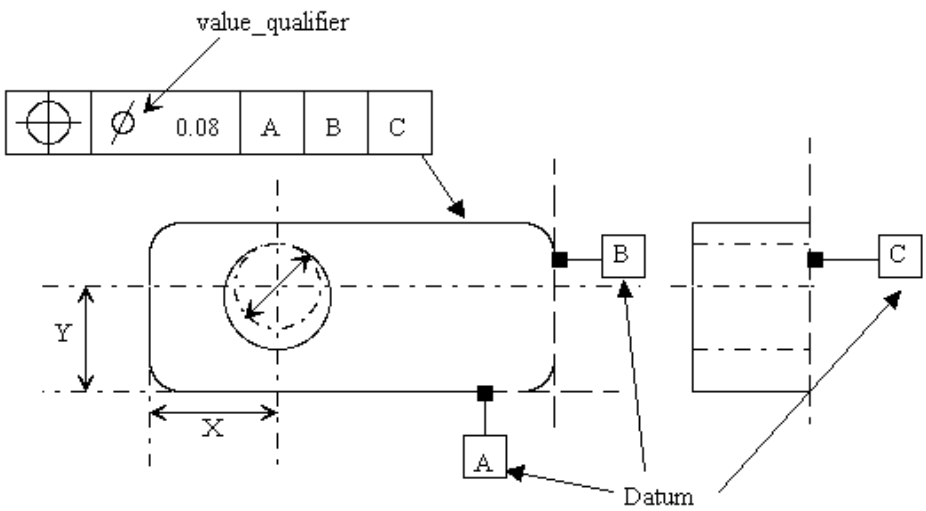


Figure 76 - Position_tolerance

4.2.162.2 geometric_reference

The geometric_reference specifies the datum to which the tolerance is related. See 4.3.184 for the application assertion.

 ©ISO 2000 - All rights reserved

4.2.162.3 value_qualifier

The value_qualifier specifies the type of tolerance. A Position_tolerance of TRUE is a diametric tolerance. A Position_tolerance of FALSE is not a diametric tolerance.

4.2.163 Process_property

A Process_property is the characteristics of a series of actions or operations directed toward changing the part.

EXAMPLE - Examples of Process_property may be paint or coat.

The data associated with a Process_property are the following:

— process_name;

— property_characteristic.

4.2.163.1 process_name

The process_name specifies a word or group of words by which a Process_property is commonly referred.

4.2.163.2 property_characteristic

The property_characteristic specifies the parameter to describe the Process_property. The property_-characteristic need not be specified for a particular Process_property. There may be more than one property_characteristic for a Process_property. See 4.3.186 for the application assertion.

4.2.164 Profile

A Profile is a planar outline used in the definition of a feature. A Profile may be either open or closed. A Profile shall be in the X-Y plane and have an orientation that will position it in reference to a Machining_feature, which may require a profile as a part of its definition. Each Profile is either a Closed_profile (see 4.2.31) or an Open_profile (see 4.2.130). The data associated with a Profile are the following:

— placement.

4.2.164.1 placement

The placement specifies where to locate the Profile in reference to the Part orientation. See 4.3.187 for the application assertion.

©ISO 2000 - All rights reserved

4.2.165 Profile_feature

A Profile_feature is a type of Multi_axis_feature (see 4.2.125) that is the removal of excess material from the boundary shape of a part. Each Profile_feature is either a General_outside_profile (see 4.2.84), or a Shape_profile(see 4.2.201). The data associated with a Profile_feature are the following:

— profile_swept_shape.

4.2.165.1 profile_swept_shape

The profile_swept_shape specifies an implicit 2D line (see 4.2.105) definition that, when combined with a Profile, creates the shape of the Profile_feature. The profile_swept_shape places a limitation on the Profile_feature definition so not to interfere with other features that might be nearby. The placement of the Linear_path that defines profile_sweep_shape shall be the same as the Profile_feature feature. The orientation shall be with the Z-axis toward the direction of travel of the profile boundary, and the Y-axis in the direction away from the part material. See 4.3.188 for the application assertion.

EXAMPLE - If a portion of the part should extend over the top of the Profile_feature, the depth value would not interfere with it.

4.2.166 Profile_floor

A Profile_floor is the bottom condition for a Shape_profile (see 4.2.201). The Profile_floor may be flat, or any arbitrary shape. Each Profile_floor is either a General_profile_floor (see 4.2.89),or a Planar_profile_floor (see 4.2.156). The data associated with a Profile_floor are the following:

— floor_radius;

— start_or_end.

4.2.166.1 floor_radius

The floor_radius specifies the radius of curvature for an arc between the bottom and the sides of a Pocket feature. See 4.3.189 for the application assertion.

4.2.166.2 start_or_end

The start_or_end specifies a boolean value of TRUE if the Profile_floor is positioned at the end of a Shape_profile, and a value of FALSE if it is at the start of the Shape_profile.

4.2.167 Project_order

A Project_order is a document that is used to initiate and track a project within the manufacturing organization. A Project_order may track a single part or a part batch.

 ©ISO 2000 - All rights reserved

The data associated with a Project_order are the following:

— ordered_resource;

— part_status;

— pedigree_creation_status;

— project_order_id;

— release_authorization;

— resource_acquisition_status;

— shop_work_status;

— technical_data_package_status.

4.2.167.1 ordered_resource

The ordered_resource specifies the resources required by the Project_order. There may be more than one ordered_resource for a Project_order. See 4.3.194 for the application assertion.

4.2.167.2 part_status

The part_status specifies Part that is tracked by the Project_order. There may be more than one part_-status for a Project_order. See 4.3.192 for the application assertion.

4.2.167.3 pedigree_creation_status

The pedigree_creation_status specifies the Pedigree_creation_order that is tracked by the Project_order. The pedigree_creation_status need not be specified for a particular Project_order. See 4.3.193 for the application assertion.

4.2.167.4 project_order_id

The project_order_id specifies a unique identifier for the Project_order.

4.2.167.5 release_authorization

The release_authorization specifies the approval to manufacture a Part. See 4.3.190 for the application assertion.

4.2.167.6 resource_acquisition_status

The resource_acquisition_status specifies Resource_acquisition_order that is tracked by the Project_order. The resource_acquisition_status need not be specified for a particular Project_order. See 4.3.195 for the application assertion.

©ISO 2000 - All rights reserved

4.2.167.7 shop_work_status

The shop_work_status specifies the shop_work_order that is tracked by the Project_order. The shop_-work_status need not be specified for a particular Project_order. See 4.3.196 for the application assertion.

4.2.167.8 technical_data_package_status

The technical_data_package_status specifies the Digital_technical_data_package_work_order that is tracked by the Project_order. The technical_data_package_status need not be specified for a particular Project_order. See 4.3.191 for the application assertion.

4.2.168 Projection

A Projection is an extension of a feature from the part so that a tolerance zone can be created. A feature is extended from one end of a feature for a specified length. The data associated with a Projection are the following:

— projection_end;

— projection_length.

4.2.168.1 projection_end

The projection_end specifies the physical shape that is the extension of a feature. See 4.3.197 for the application assertion.

4.2.168.2 projection_length

The projection_length specifies the amount to extend the end of a feature.

4.2.169 Property

A Property is a characteristic associated with the physical structure or integrity of an element of a part. The data associated with a Property are the following:

— material_characteristic;

— property_name;

— part_property_characteristic;

— process_characteristic;

— property_characteristic;

 ©ISO 2000 - All rights reserved

— property_description;

— surface_characteristic.

4.2.169.1 material_characteristic

The material_characteristic specifies the information that describe material for manufacturing the Part. The material_characteristic need not be specified for a particular Property. There may be more than one material_characteristic for a Property. See 4.3.198 for the application assertion.

4.2.169.2 property_name

The property_name specifies a word or group of words by which a property is commonly referred.

4.2.169.3 part_property_characteristic

The part_property_characteristic specifies the information that describe properties of the Part. The part_property_characteristic need not be specified for a particular Property. There may be more than one part_property_characteristic for a Property. See 4.3.199 for the application assertion.

4.2.169.4 process_characteristic

The process_characteristic specifies information that describe processes for manufacturing the part. The process_characteristic need not be specified for a particular Property. There may be more than one process_characteristic for a Property. See 4.3.200 for the application assertion.

4.2.169.5 property_characteristic

The property_characteristic specifies information that describe properties of the Part. The property_-characteristic need not be specified for a particular Property. There may be more than one property_-characteristic for a Property. See 4.3.201 for the application assertion.

4.2.169.6 property_description

The property_description specifies the Specification that has additional information about the properties of the Part. The property_description need not be specified for a particular Property. There may be more than one property_description for a Property. See 4.3.202 for the application assertion.

4.2.169.7 surface_characteristic

The surface_characteristic specifies information that describe surface conditions of the Part. The surface_characteristic need not be specified for a particular Property. There may be more than one surface_characteristic for a Property. See 4.3.203 for the application assertion.

©ISO 2000 - All rights reserved

4.2.170 Property_parameter

A Property_parameter is an element of information that describes a characteristic that comprises the property. Each Property_parameter may be one of the following: Descriptive_parameter (see 4.2.56) or a Numeric_parameter (see 4.2.128). The data associated with a Property_parameter are the following:

— parameter_name.

4.2.170.1 parameter_name

The parameter_name specifies a word or group of words that identify a characteristic of interest for a Property_parameter.

4.2.171 Protrusion

A Protrusion is a type of Multi_axis_feature(see 4.2.125) that is an arbitrary shape that extends out from a surrounding surface.

NOTE - Figure 77 illustrates a Protrusion.

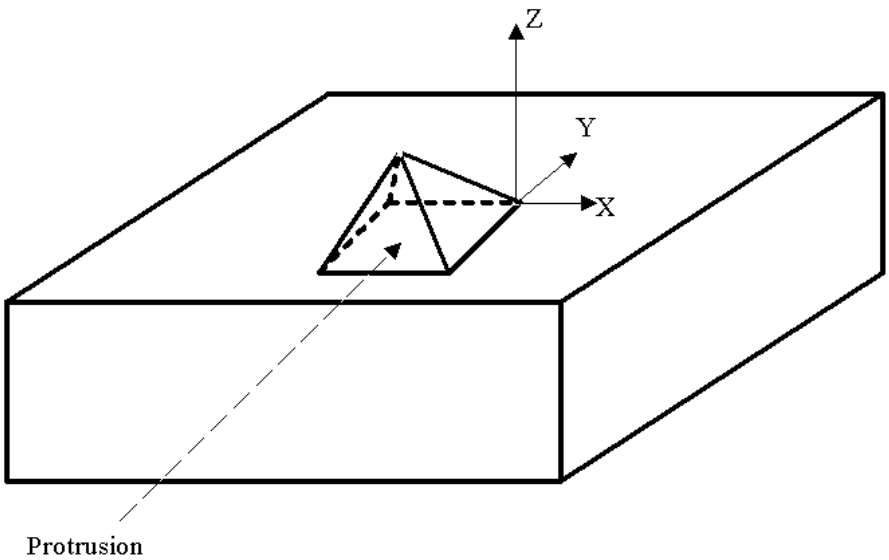


Figure 77 - Protrusion

The data associated with a Protrusion are the following:

— shape_volume.

 ©ISO 2000 - All rights reserved

4.2.171.1 shape_volume

The shape_volume specifies the arbitrary shape that defines the shape of a Protrusion. See 4.3.204 for the application assertion.

4.2.172 Radial_dimension_tolerance

A Radial_dimension_tolerance is a type of Size_tolerance (see 4.2.204) that is the allowable variation for the radial distance from the center of a circular curve to a point on the curve.

NOTE - Figure 78 illustrates the Radial_dimension_tolerance.

Figure 78 - Radial_dimension_tolerance

4.2.173 Radiused_slot_end_type

A Radiused_slot_end_type is a type of Slot_end_type (see 4.2.206) that is an end condition of a slot that shall be a cylindrical shape tangent to both of the adjacent Slot wall surfaces.

NOTE - Figure 79 illustrates the Radiused_slot_end_type.

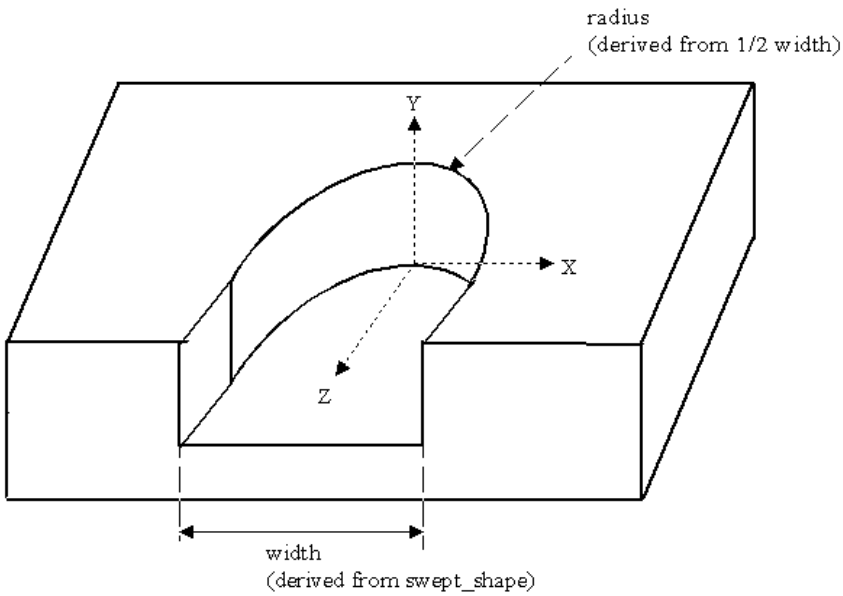


Figure 79 - Radiused_slot_end_type

©ISO 2000 - All rights reserved

4.2.174 Recess

A Recess is a type of a Pocket (see 4.2.160) that has a bottom, and a floor radius, but the sides of the pocket shall be no higher that the floor radius. The Recess shall not pass entirely through the part but shall have a floor bottom. The type of bottom condition shall be a Pocket bottom_condition (see 4.2.161).

EXAMPLE - Figure 80 illustrates the Recess. A Recess is commonly found in the bottom of a Pocket feature.

The data associated with a Recess are the following:

— bottom_condition;

— fillet_boundary.

4.2.174.1 bottom_condition

The bottom_condition specifies the shape of the bottom of a Recess feature. The bottom_condition shall not pass entirely through the part. See 4.3.206 for the application assertion.

4.2.174.2 fillet_boundary

The fillet_boundary specifies an outline or shape that is an enclosed area that shall be a close profile. The profile specifies the area required by a Recess. The placement of the fillet_boundary shall be with the origin of the Profile at the origin of the Recess. The X-axis and Y-axis of the profile shall be the same as the X-axis and Y-axis of the Recess. See 4.3.205 for the application assertion.

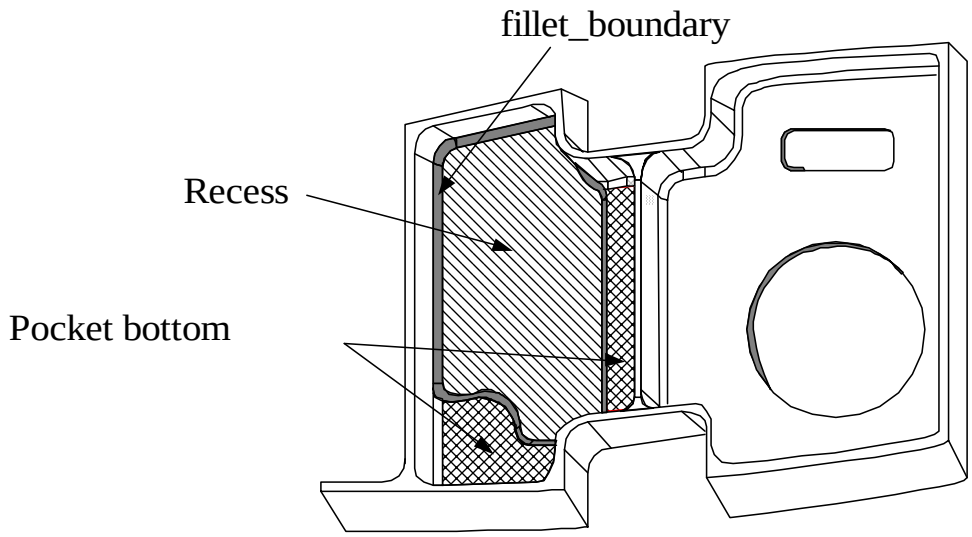


Figure 80 - Recess

 ©ISO 2000 - All rights reserved

4.2.175 Rectangular_boss

A Rectangular_boss is a type of Boss (see 4.2.10) that is an enclosed volume with opposite sides that are equal in length.

NOTE - Figure 81 illustrates the Rectangular_boss. The lengths are determined in the profile view of the Boss.

The data associated with a Rectangular_boss are the following:

— change_in_boundary;

— rectangular_profile.

4.2.175.1 change_in_boundary

The change_in_boundary specifies a taper that defines the change in shape of the Rectangular_boss. The change_in_boundary need not be specified for a particular Rectangular_boss. See 4.3.207 for the application assertion.

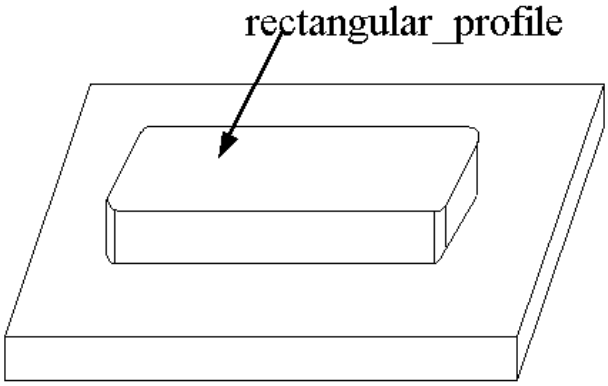


Figure 81 - Rectangular_boss

4.2.175.2 rectangular_profile

The rectangular_profile specifies an enclosed area bounded by four sides with opposite sides equal in length and corners at 90 degrees. The orientation is at the center of the rectangle, the X-axis is parallel to the length of the rectangle and the Y-axis is parallel to the width. See 4.3.209 for the application assertion.

4.2.176 Rectangular_closed_pocket

A Rectangular_closed_pocket is a type of Pocket (see 4.2.160) that is an enclosed volume with opposite sides that are equal in length.

NOTE - Figure 82 illustrates the Rectangular_closed_pocket.

©ISO 2000 - All rights reserved

The data associated with a Rectangular_closed_pocket are the following:

— closed_boundary.

4.2.176.1 closed_boundary

The closed_boundary specifies the outline or shape that is an enclosed area that has a completely closed profile. The profile specifies the area required by a Rectangular_closed_pocket. The placement of the closed_boundary shall be with the origin of the profile at the origin of the pocket. The X-axis and Y-axis of the profile shall be the same as the X-axis and Y-axis of the Rectangular_closed_pocket. See 4.3.210 for the application assertion.

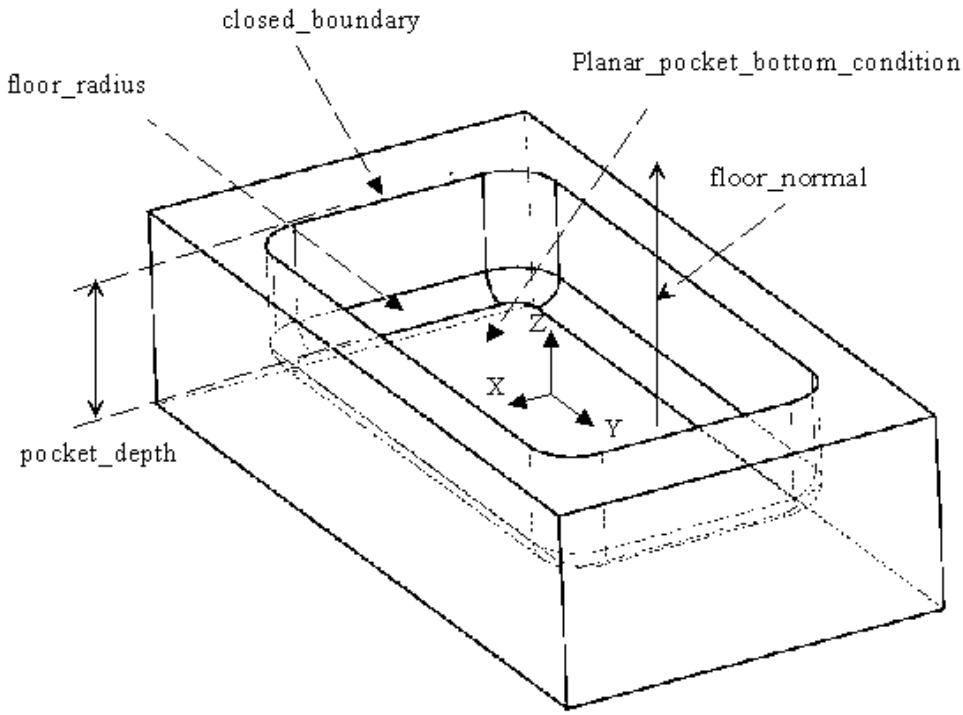


Figure 82 - Rectangular_closed_pocket

4.2.177 Rectangular_closed_profile

A Rectangular_closed_profile is a type of Closed_profile (see 4.2.31) that is an enclosed area bounded by four sides with opposite sides equal in length and corners at 90 degrees. The orientation is at the center of the rectangle, the X-axis is parallel to the length of the rectangle and the Y-axis is parallel to the width.

NOTE - Figure 83 illustrates the Rectangular_closed_profile.

 ©ISO 2000 - All rights reserved

The data associated with a Rectangular_closed_profile are the following:

— corner_radius;

— profile_length;

— profile_width.

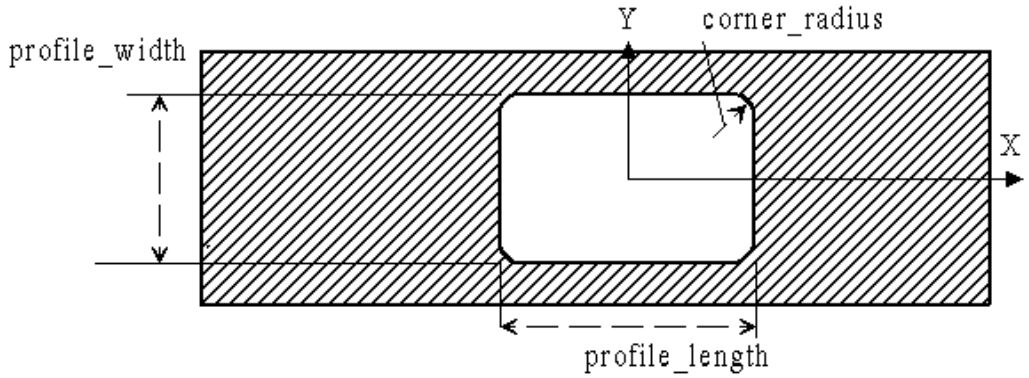


Figure 83 - Rectangular_closed_profile

4.2.177.1 corner_radius

The corner_radius specifies the size of the arc in all four corners of the rectangular profile. See 4.3.211 for the application assertion.

4.2.177.2 profile_length

The profile_length specifies the length of the side, along the X-axis, of the rectangular profile. See 4.3.211 for the application assertion.

4.2.177.3 profile_width

The profile_width specifies the length of the side, along the Y-axis, of the rectangular profile. See 4.3.211 for the application assertion.

4.2.178 Rectangular_closed_shape_profile

The Rectangular_closed_shape_profile is a type of Shape_profile (see 4.2.201) that is an enclosed volume with opposite sides that are equal in length. The data associated with a Rectangular_closed_shape_profile are the following:

— closed_boundary.

©ISO 2000 - All rights reserved

4.2.178.1 closed_boundary

The closed_boundary specifies the outline or shape that is an enclosed area that has a completely closed profile. The profile specifies the area required by a Rectangular_closed_shape_profile. The placement of the closed_boundary shall be with the origin of the profile at the origin of the profile. The X-axis and Y-axis of the profile shall be the same as the X-axis and Y-axis of the Rectangular_closed_shape_profile. See 4.3.212 for the application assertion.

4.2.179 Rectangular_offset_pattern

A Rectangular_offset_pattern is a modification of the placement of a particular occurrence of the base feature in a Rectangular_pattern relative to its expected placement.

NOTE - Figure 84 illustrates the Rectangular_offset_pattern.

The data associated with a Rectangular_offset_pattern are the following:

— column_index;

— offset_direction;

— offset_distance;

— row_index.

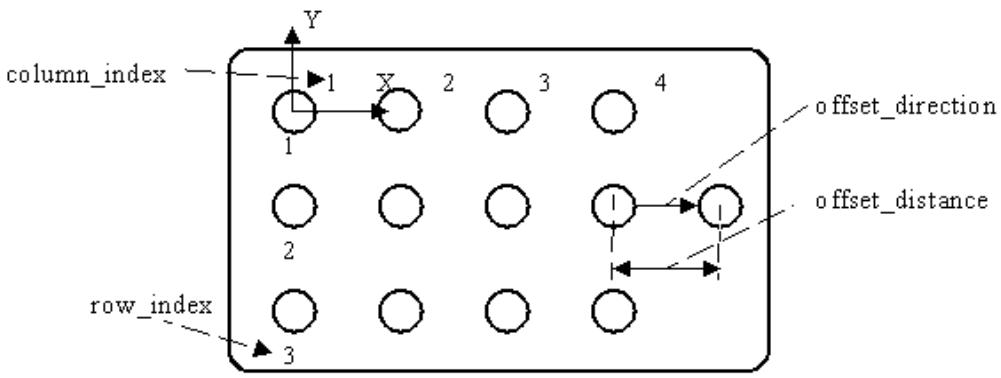


Figure 84 - Rectangular_offset_pattern

4.2.179.1 column_index

The column_index specifies the unique identification for a feature in a column of multiple features. See 4.3.214 for the application assertion.

 ©ISO 2000 - All rights reserved

4.2.179.2 offset_direction

The offset_direction specifies the direction to offset a base feature from it's original position in the rectangular pattern. See 4.3.213 for the application assertion.

4.2.179.3 offset_distance

The offset_distance specifies the amount of offset from a feature location in a Rectangular_pattern for placing another feature. See 4.3.214 for the application assertion.

4.2.179.4 row_index

The row_index specifies the unique identification for a feature in a row of multiple features. See 4.3.214 for the application assertion.

4.2.180 Rectangular_omit_pattern

A Rectangular_omit_pattern is an omission of a particular occurrence of the base feature in an Rectangular_pattern.

NOTE - Figure 85 illustrates Rectangular_omit_pattern.

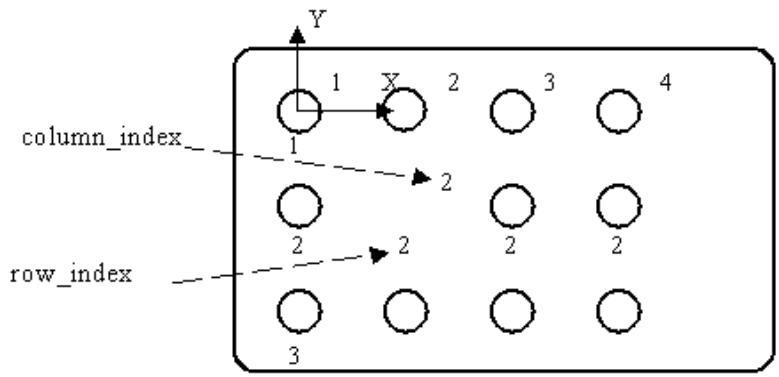


Figure 85 - Rectangular_omit_pattern

The data associated with a Rectangular_omit_pattern are the following:

— column_index;

— row_index.

4.2.180.1 column_index

The column_index specifies the unique identification for a feature in a column of multiple features. See 4.3.215 for the application assertion.

©ISO 2000 - All rights reserved

4.2.180.2 row_index

The row_index specifies the unique identification for a feature in a row of multiple features. See 4.3.215 for the application assertion.

4.2.181 Rectangular_open_pocket

A Rectangular_open_pocket is a type of Pocket (see 4.2.160) that is an open profile with opposite sides that are of equal length and with one side that does not make contact with the part. The open boundary is defined by a Square_U_profile (see 4.2.211) such that, when swept along a path, the profile defines the area on a part for volume removal. The orientation and placement of the Square_U_profile shall be the same as the Rectangular_open_pocket.

NOTE - Figure 86 illustrates the Rectangular_open_pocket.

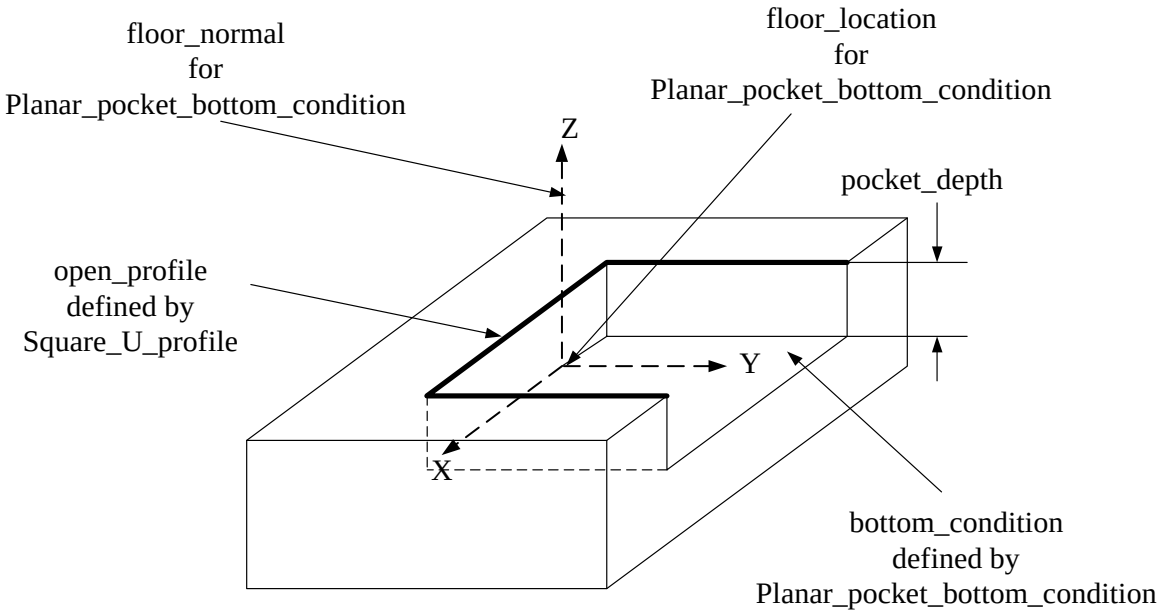


Figure 86 - Rectangular_open_pocket

The data associated with a Rectangular_open_pocket are the following:

— open_boundary.

4.2.181.1 open_boundary

The open_boundary specifies the outline or shape that is an enclosed area that is open on one side. The profile specifies the area required by a Rectangular_open_pocket. The placement of the open boundary shall be with the origin of the profile at the origin of the pocket. The X-axis and Y-axis of the profile shall be the same as the X-axis and Y-axis of the Rectangular_open_pocket. See 4.3.216 for the application assertion.

 ©ISO 2000 - All rights reserved

4.2.182 Rectangular_open_shape_profile

The Rectangular_open_shape_profile is a type of Shape_profile (see 4.2.201) that is an open profile with opposite sides that are of equal length and with one side that does not make contact with the part. The data associated with a Rectangular_open_shape_profile are the following:

— open_boundary.

4.2.182.1 open_boundary

The open_boundary specifies the outline or shape that is an enclosed area that is open on one side. The profile specifies the area required by a Rectangular_open_shape_profile. The placement of the open_-boundary shall be with the origin of the profile at the origin of the feature. The X-axis and Y-axis of the profile shall be the same as the X-axis and Y-axis of the Rectangular_open_shape_profile. See 4.3.217 for the application assertion.

4.2.183 Rectangular_pattern

A Rectangular_pattern is a type of Replicate_feature (see 4.2.185) that is a shape component arranged in a pattern of rows and columns.

NOTE - Figure 87 illustrates the Rectangular_pattern.

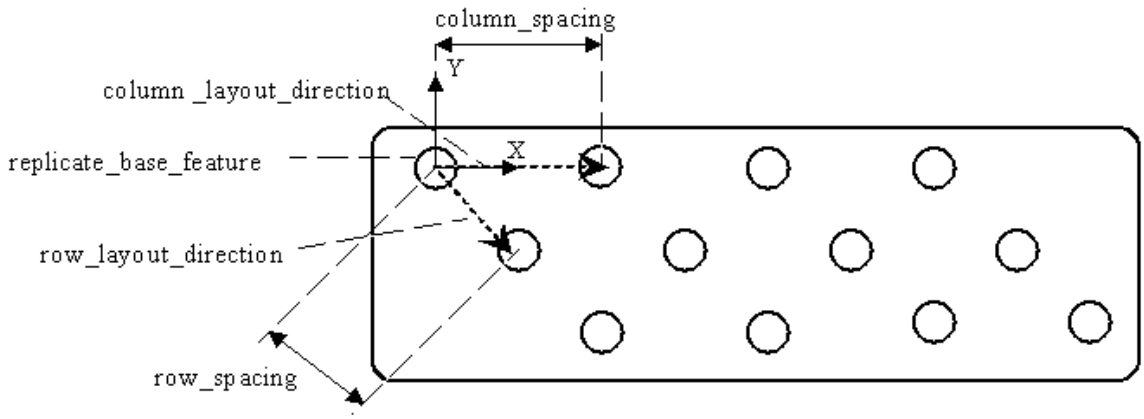


Figure 87 - Rectangular_pattern

The data associated with a Rectangular_pattern are the following:

— column_layout_direction;

— column_spacing;

— columns;

— missing_base_feature;

©ISO 2000 - All rights reserved

— relocated_base_feature;

— row_layout_direction;

— row_spacing;

— rows.

4.2.183.1 column_layout_direction

The column_layout_direction specifies the linear direction for defining the columns of the pattern. See 4.3.218 for the application assertion.

4.2.183.2 column_spacing

The column_spacing specifies the amount of space between features in a Rectangular_pattern column. See 4.3.219 for the application assertion.

4.2.183.3 columns

The columns specifies the number of columns for placing features in the Rectangular_pattern. See 4.3.219 for the application assertion.

4.2.183.4 missing_base_feature

The missing_base_feature specifies the definition to remove any number of base features from the Rectangular_pattern. The missing_base_feature need not be specified for a particular Rectangular_pattern. There may be more than one missing_base_feature for a Rectangular_pattern. See 4.3.221 for the application assertion.

4.2.183.5 relocated_base_feature

The relocated_base_feature specifies the definition to offset any number of base features from the Rectangular_pattern. The relocated_base_feature need not be specified for a particular Rectangular_-pattern. There may be more than one relocated_base_feature for a Rectangular_pattern. See 4.3.220 for the application assertion.

4.2.183.6 row_layout_direction

The row_layout_direction specifies the linear direction for defining the rows of the pattern. See 4.3.218 for the application assertion.

4.2.183.7 row_spacing

The row_spacing specifies the amount of space between features in a Rectangular_pattern row. See 4.3.219 for the application assertion.

 ©ISO 2000 - All rights reserved

4.2.183.8 rows

The rows specifies the number of rows for placing features in the Rectangular_pattern. See 4.3.219 for the application assertion.

4.2.184 Replicate_base

A Replicate_base is the type of feature to be used as a base feature for reproduction. The Replicate_base shall have a base defined by either a Machining_feature (see 4.2.111) or a Replicate_feature (see 4.2.185). The data associated with a Replicate_base are the following:

— base_feature.

4.2.184.1 base_feature

The base_feature specifies the feature that will be reproduced by the Replicate_feature. The base_feature may be either a Machining_feature (see 4.2.111) or a Replicate_feature (see 4.2.185). See 4.3.222 and 4.3.223 for the application assertion.

4.2.185 Replicate_feature

A Replicate_feature is a type of Machining_feature (see 4.2.111) that is a basis shape, and the arrangement of identical copies of that base shape. Each base shape is a Machining_feature oriented to the first defined position of a pattern. The patterns describe how to replicate that feature to different placements on the part. Each Replicate_feature is either a Circular_pattern (see 4.2.28), General_pattern (see 4.2.86), or a Rectangular_pattern (see 4.2.183). The data associated with a Replicate_feature are the following:

— placement;

— replicate_base_feature.

4.2.185.1 placement

The placement specifies the position and orientation of a Replicate_feature relative to the base shape for a part. See 4.3.224 for the application assertion.

4.2.185.2 replicate_base_feature

The replicate_base_feature specifies the feature that will be replicated by the Replicate_feature. See 4.3.225 for the application assertion.

4.2.186 Requisition

A Requisition is a document that contains an order for equipment or materials to perform or support the manufacturing process. Each Requisition may be one of the following: Cutting_tool_requisition (see 4.2.46), Dedicated_fixture_requisition (see 4.2.53), Machine_requisition (see 4.2.110), Material_-requisition (see 4.2.120) Modular_fixture_requisition (see 4.2.124), or an Indirect_stock_requisition (see

©ISO 2000 - All rights reserved

4.2.101). The data associated with a Requisition are the following:

— quantity_ordered;

— required_delivery_date;

— requisition_date;

— requisition_description;

— requisition_number.

4.2.186.1 quantity_ordered

The quantity_ordered specifies the number of items that are being purchased by an organization using the Requisition.

4.2.186.2 required_delivery_date

The required_delivery_date specifies the year, month and day on which the item or items specified in the Requisition are required at the site that ordered them.

4.2.186.3 requisition_date

The requisition_date specifies the year, month, and day on which the Requisition was issued.

4.2.186.4 requisition_description

The requisition_description specifies human interpretable prose that describes the item being ordered, and provides any special instructions that may pertain to the order.

4.2.186.5 requisition_number

The requisition_number specifies a unique identification of the Requisition.

4.2.187 Resource_acquisition_order

A Resource_acquisition_order is a document required for process planning of a part. This order defines the requirements for material needed to produce a part.

EXAMPLE - The Resource_acquisition_order may contain customer order information, part identification, quantity, or schedules.

The data associated with a Resource_acquisition_order are the following:

— order_id.

 ©ISO 2000 - All rights reserved

4.2.187.1 order_id

The order_id specifies a unique identifier for the Resource_acquisition_order.

4.2.188 Revolved_feature

A Revolved_feature is a type of Machining_feature (see 4.2.111) that is a sweeping of a planar shape one complete revolution about an axis. The planar shape shall be finite in length, coplanar with the axis of revolution, and shall not intersect the axis of revolution. The axis of revolution shall be the same as the Z-axis of the feature. The Revolved_feature may be either an outer shape of a part or a volume removal, depending on the material direction. Each Revolved_feature is either a General_revolution (see 4.2.91), Groove (see 4.2.97), Revolved_flat (see 4.2.189), or a Revolved_round (see 4.2.190).

The data associated with a Revolved_feature are the following:

— material_side;

— radius.

4.2.188.1 material_side

The material_side specifies the material direction. The direction of removal indicates the direction the material will be removed from the part. See 4.3.227 for the application assertion.

4.2.188.2 radius

The radius specifies the distance from the axis of rotation to define placement of the profile that will be swept about the axis. See 4.3.226 for the application assertion.

4.2.189 Revolved_flat

A Revolved_flat is a type of Revolved_feature (see 4.2.188) that is the sweeping of a straight line about an axis.

NOTE - Figure 88 illustrates the Revolved_flat.

The data associated with a Revolved_flat are the following:

— flat_edge_shape.

4.2.189.1 flat_edge_shape

The flat_edge_shape specifies the line with direction and magnitude that when revolved about an axis defines the area on a part for volume removal. The placement of the profile shall be along the X-axis of the Revolved_flat at a specified distance away from the origin. The Y-axis orientation of the Linear_profile shall be the same as the Y-axis of the Revolved_flat, the X-axis and Z-axis are independent of the orientation of the Revolved_flat feature. See 4.3.228 for the application assertion.

©ISO 2000 - All rights reserved

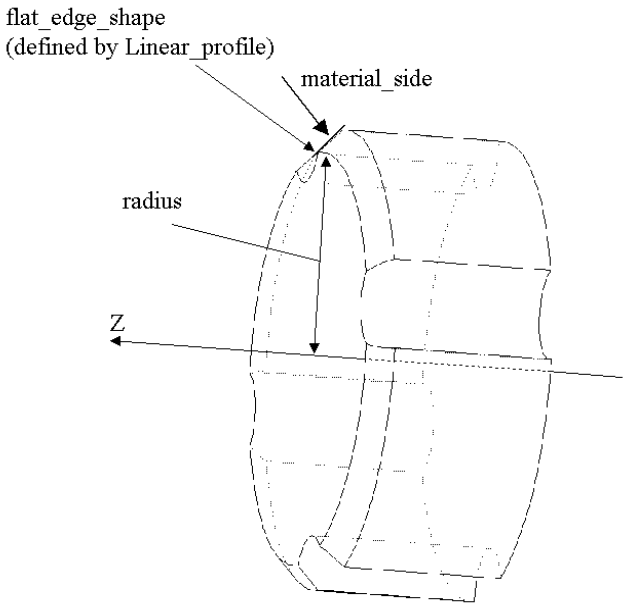


Figure 88 - Revolved_flat

4.2.190 Revolved_round

A Revolved_round is a type of Revolved_feature (see 4.2.188) that is the sweeping of an arc about an axis.

NOTE - Figure 89 illustrates the Revolved_round.

The data associated with a Revolved_round are the following:

— rounded_edge_shape.

4.2.190.1 rounded_edge_shape

The rounded_edge_shape specifies the arc that when revolved about an axis defines the area on a part for volume removal. The placement of the profile shall be along the X-axis of the Revolved_round at a specified distance away from the origin. The Z-axis orientation of the Partial_circular_profile shall be the same as the Y-axis of the Revolved_round, the X-axis and Y-axis are independent of the orientation of the Revolved_round feature. See 4.3.229 for the application assertion.

 ©ISO 2000 - All rights reserved

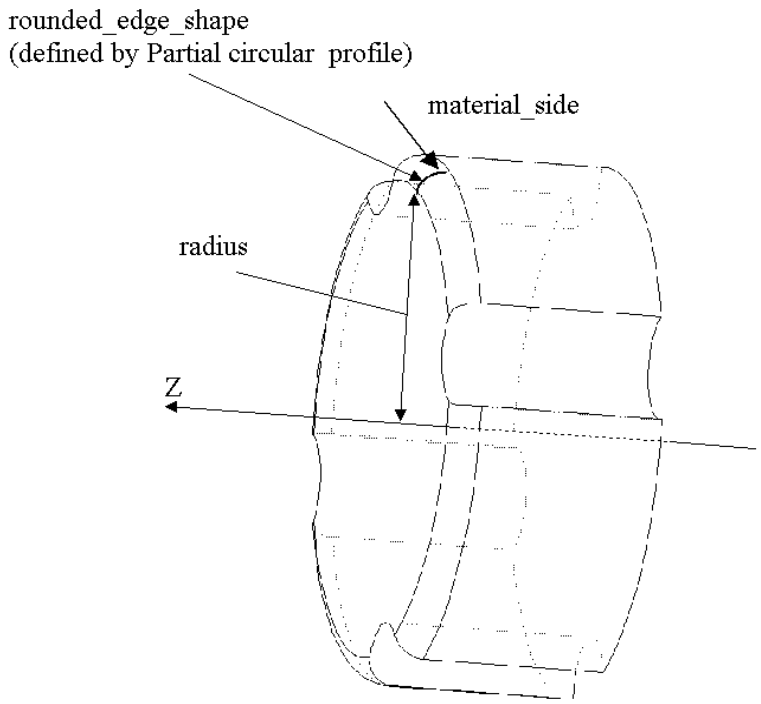


Figure 89 - Revolved_round

4.2.191 Rib_top

The Rib_top is a type of Multi_axis_feature (see 4.2.125) that is the removal of a volume to a floor with no sides, or floor radius. Rib_top features may adjoin with another Rib_top feature.

EXAMPLE - The material that seperates two pockets on a part is an example of a rib. The top surface of that rib would be an example of a Rib_top.

NOTE - Figure 90 illustrates the Rib_top feature.

The data associated with a Rib_top are the following:

— floor_condition;

— removal_direction.

©ISO 2000 - All rights reserved

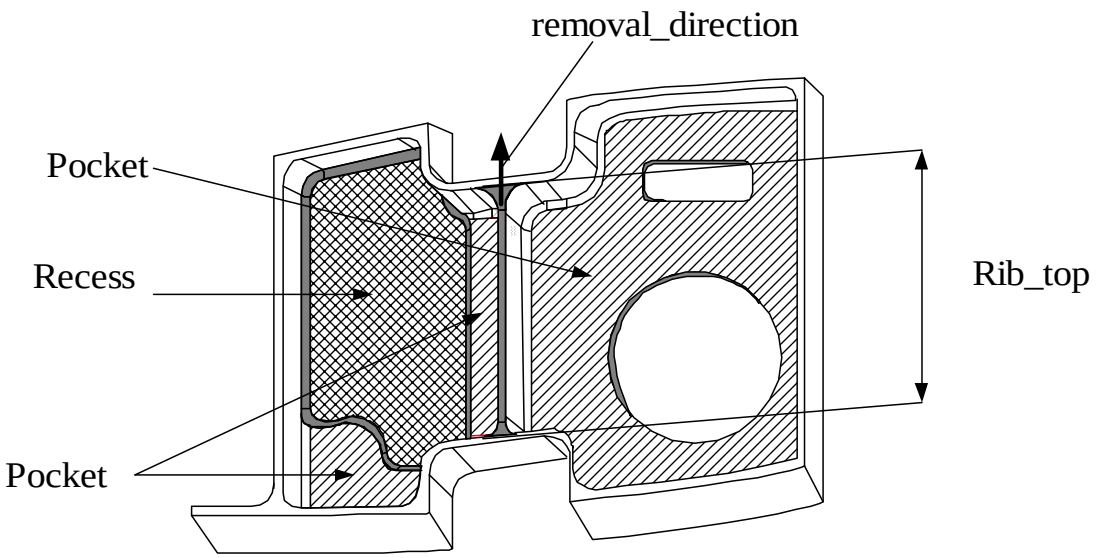


Figure 90 - Rib_top

4.2.191.1 floor_condition

A floor_condition specifies the bottom state of a Rib_top. The floor may be flat or any arbitrary shape. See 4.3.231 for the application assertion.

4.2.191.2 removal_direction

A removal_direction specifies a vector that points in the general direction away from the material for a Rib_top. See 4.3.230 for the application assertion.

4.2.192 Rib_top_floor

A Rib_top_floor is the bottom restriction for a Rib_top (see 4.2.191). The bottom may be flat, or any arbitrary shape. Each Rib_top_floor is either a General_rib_top_floor (see 4.2.92), or a Planar_rib_top_floor (see 4.2.157).

4.2.193 Round_hole

A Round_hole is a type of Hole (see 4.2.99) that is a removal of a volume of cylindrical shape from a part. A Round_hole need not be tapered. The orientation is at a point in the bottom of the hole. The Z-axis is along the centerline with the direction out of the hole.

NOTE - Figure 91 illustrates the Round_hole with and without a taper.

 ©ISO 2000 - All rights reserved

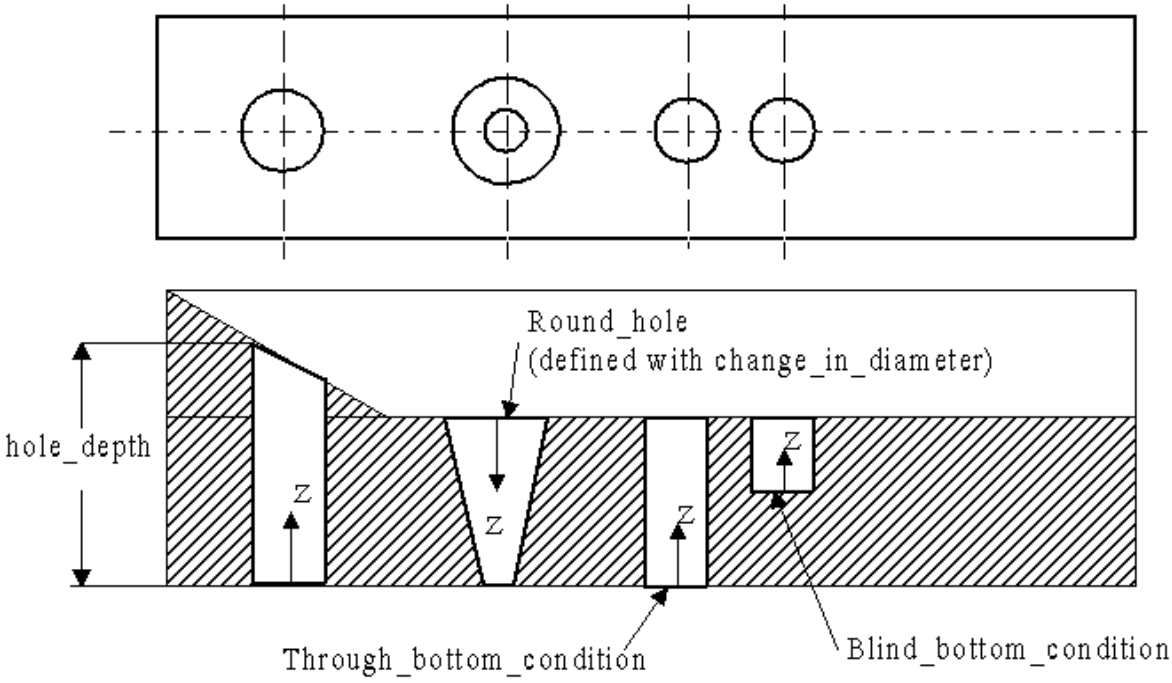


Figure 91 - Round_hole

The data associated with a Round_hole are the following:

— bottom_condition;

— change_in_diameter;

— diameter;

— hole_depth.

4.2.193.1 bottom_condition

The bottom_condition specifies the shape of the bottom of a Round_hole feature. Each bottom_condition may be one of the following: Blind_bottom_condition (see 4.2.8), or Through_bottom_condition (see 4.2.225). See 4.3.232and 4.3.233 for the application assertion.

4.2.193.2 change_in_diameter

The change_in_diameter specifies the taper that defines the change in shape of the Round_hole. The change_in_diameter need not be specified for a particular Round_hole. See 4.3.236 and 4.3.237, 4.3.238 for the application assertion.

©ISO 2000 - All rights reserved

4.2.193.3 diameter

The diameter is the distance across a Round_hole. The placement and orientation of the Circular_-closed_profile shall be the same as the Round_hole feature. See 4.3.234 for the application assertion.

4.2.193.4 hole_depth

The hole depth is some amount of distance from the bottom of a Round_hole to a point that is outside of the Round_hole feature. Hole depth places a limitation on the Round_hole depth definition so not to interfere with other features that might be nearby. The placement and orientation of the Linear_path shall be the same as the Round_hole feature. See 4.3.235 for the application assertion.

4.2.194 Rounded_end

A Rounded_end is a type of Multi_axis_feature(see 4.2.125) that is a partially circular shape passed along a linear path.

NOTE - Figure 92 illustrates the Rounded_end.

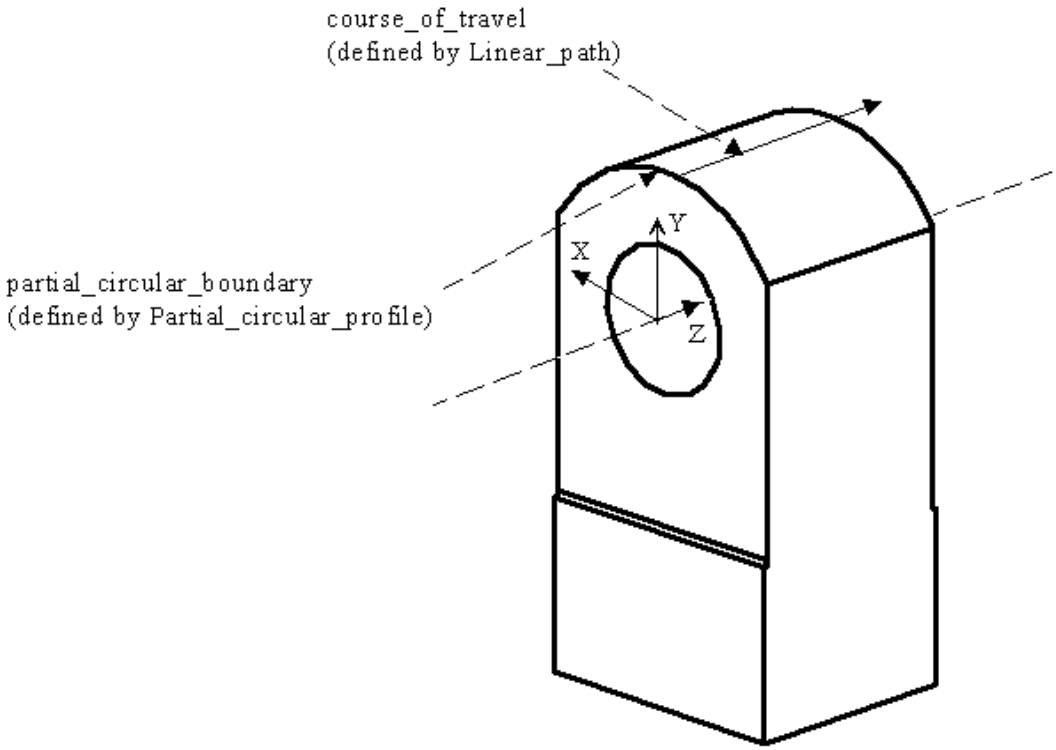


Figure 92 - Rounded_end

 ©ISO 2000 - All rights reserved

The data associated with a Rounded_end are the following:

— course_of_travel;

— partial_circular_boundary.

4.2.194.1 course_of_travel

The course_of_travel specifies a straight line with magnitude and direction. The placement and orientation of the Linear_path that defines the course_of_travel shall be the same as the Rounded_end feature. See 4.3.239 for the application assertion.

4.2.194.2 partial_circular_boundary

The partial_circular_boundary specifies the arc that when swept along a path defines the area on a part for volume removal. The placement and orientation of the Partial_circular_profile that defines the partial_circular_boundary shall be the same as the Rounded_end feature. See 4.3.240 for the application assertion.

4.2.195 Rounded_U_profile

A Rounded_U_profile is a type of Open_profile (see 4.2.130) that is a shape bounded by two parallel lines and a semicircle. Each line begins at the end point of the semicircle. The lines are tangent to the circle and extend infinitely. The profile is positioned with the opening in the direction of the Y-axis. The orientation is at a point on the profile the farthest distance from the opening measured along the Y-axis. The X-axis is tangent to the semicircle.

NOTE - Figure 93 illustrates the Rounded_end.

The data associated with a Rounded_U_profile are the following:

— width.

4.2.195.1 width

The width specifies the distance across a Round_U_profile. See 4.3.241 for the application assertion.

NOTE - Figure 93 illustrates the Rounded_U_profile.

4.2.196 Second_chamfer_offset

A Second_chamfer_offset is a choice of methods for creating a Chamfer feature. A Chamfer requires an offset and the choice between a second offset or an angle. Each Second_chamfer_offset is either a Chamfer_angle (see 4.2.20) or a Second_offset (see 4.2.197). The data associated with a Second_-chamfer_offset are the following:

— second_face.

©ISO 2000 - All rights reserved

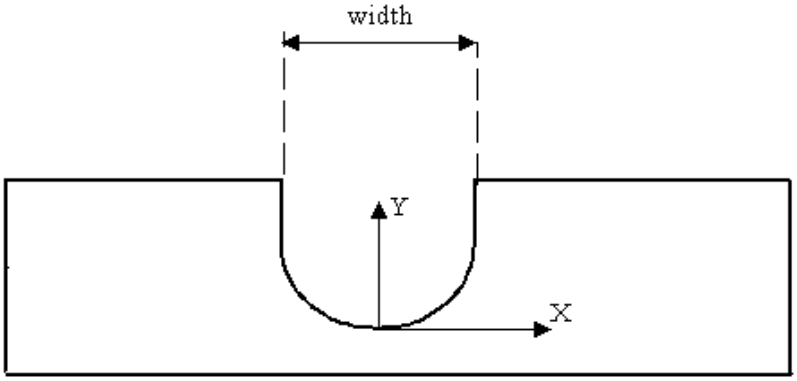


Figure 93 - Rounded_U_profile

4.2.196.1 second_face

The second_face is one of two faces the Chamfer feature will transition between. The first face is specified by First_offset (see 4.2.74). See 4.3.242 for the application assertion.

4.2.197 Second_offset

A Second_offset is a type of Second_chamfer_offset (see 4.2.196) that is the amount of length offset from a face for creating a Chamfer feature.

NOTE - Figure 12 illustrates a Second_offset for a Chamfer feature.

The data associated with a Second_offset are the following:

— offset_amount.

4.2.197.1 offset_amount

The offset_amount specifies the offset value from the edge of a face to the Chamfer face. See 4.3.243 for the application assertion.

4.2.198 Shape

A Shape is the physical form of the part that is being machined. The data associated with a Shape are the following:

— base_shape_definition;

— B-rep_form;

— element.

 ©ISO 2000 - All rights reserved

4.2.198.1 base_shape_definition

The base_shape_definition specifies either the implicit or the explicit definition of the Part. See 4.3.244 for the application assertion.

4.2.198.2 B-rep_form

The B-rep_form specifies the boundary representation shape of the Part. The B-rep_form need not be specified for a particular Shape. There may be more than one B-rep_form for a Shape. See 4.3.245 for the application assertion.

4.2.198.3 element

The element specifies the components of the shape of the Part. The element need not be specified for a particular Shape. There may be more than one element for a Shape. See 4.3.246 for the application assertion.

4.2.199 Shape_aspect

A Shape_aspect is a region of interest with respect to the shape of a part. A Shape_aspect may be an element of the shape of the part or a reference shape that does not lie on the shape of the part, but is used to specify a characteristic of the shape of the part. The data associated with a Shape_aspect are the following:

— B-rep_form;

— B-rep_shape;

— element.

4.2.199.1 B-rep_form

The B-rep_form specifies aspects of the boundary representation of the Part. There may be more than one B-rep_form for a Shape_aspect. See 4.3.248 for the application assertion.

4.2.199.2 B-rep_shape

The B-rep_shape specifies the boundary representation of the shape of the Part. The B-rep_shape need not be specified for a particular Shape_aspect. There may be more than one B-rep_shape for a Shape_aspect. See 4.3.247 for the application assertion.

4.2.199.3 element

The element specifies components of the shape of the Part. The element need not be specified for a particular Shape_aspect. See 4.3.249 for the application assertion.

©ISO 2000 - All rights reserved

4.2.200 Shape_element

A Shape_element is a specific kind of Shape_aspect that identifies a portion of a shape and its representation. Each Shape_element is either a Direction_element (see 4.2.66), Face_shape_element (see 4.2.71), Location_element (see 4.2.108), Manufacturing_feature (see 4.2.114), Path_element (see 4.2.147) or a Planar_element (see 4.2.153).

4.2.201 Shape_profile

A Shape_profile is a type of Profile_feature (see 4.2.165) that is the removal volume of raw stock or other excess material from the boundary shape of a part. The sides of a profile may be parallel to the profile's orientation vector, or may be parallel with an offset amount.. The placement may be at the bottom of the profile with the Z-axis in the direction toward the top of the profile, or at the top of the profile with the Z-axis in the direction toward the bottom of the profile. The bottom of the boundary shape is limited by a floor condition. Each Shape_profile is either a Circular_closed_shape_profile (see 4.2.23), Partial_circular_shape_profile (see 4.2.145), General_shape_profile (see 4.2.93), Rectangular_-closed_shape_profile (see 4.2.178), or a Rectangular_open_shape_profile (see 4.2.182).

NOTE - Figure 94 illustrates a Shape_profile.

The data associated with a Shape_profile are the following:

— floor_condition;

— removal_direction.

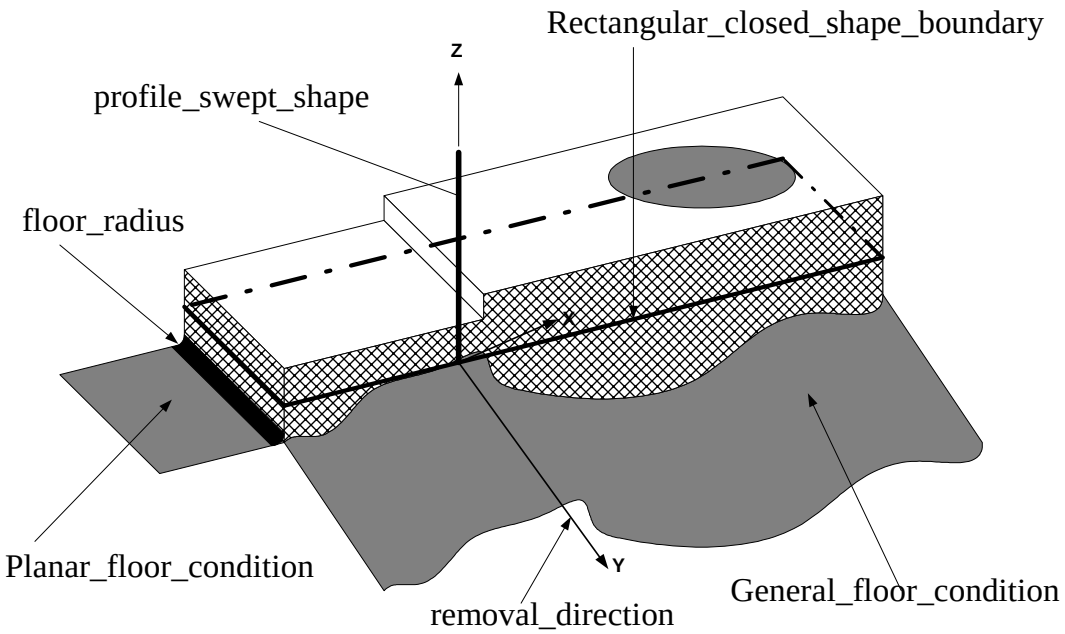


Figure 94 - Shape_profile

 ©ISO 2000 - All rights reserved

4.2.201.1 floor_condition

The floor_condition specifies the shape of the bottom of a Shape_profile feature. The floor_condition is either a Profile_floor (see 4.2.166) or a Through_profile_floor (see 4.2.227). See 4.3.251 and 4.3.252 for the application assertions.

4.2.201.2 removal_direction

A removal_direction specifies a vector that points in the general direction away from the material for a Shape_profile. See 4.3.250 for the application assertion.

4.2.202 Shop_work_order

A Shop_work_order is a document that contains the information to process a part on the shop floor, and track the part's progress throughout its manufacture. The data associated with a Shop_work_order are the following:

— order_id.

4.2.202.1 order_id

The order_id specifies a unique identifier for the Shop_work_order.

4.2.203 Single_piece_part

A Single_piece_part is a type of Part (see 4.2.139) that is the physical item which is intended to be produced through the manufacturing process. The Single_piece_part may be an Manufactured_assembly (see 4.3.112). The data associated with a Single_piece_part are the following:

— alternate_material_definition;

— material_definition.

4.2.203.1 alternate_material_definition

The alternate_material_definition specifies the secondary material choices of raw stock for producing the Part. There may be more than one alternate_material_definition for a part. See 4.3.253 for the application assertion.

4.2.203.2 material_definition

The material_definition specifies primary material choice of raw stock for producing the Part. There may be more than one material_definition for a part. See 4.3.254 for the application assertion.

©ISO 2000 - All rights reserved

4.2.204 Size_tolerance

A Size_tolerance is a type of Dimensional_tolerance (see 4.2.63) that is the size dimension tolerance characteristic for a geometric element. Each Size_tolerance is either an Angular_size_dimension_-tolerance (see 4.2.4), Curved_dimension_tolerance (see 4.2.43), Diameter_dimension_tolerance (see 4.2.59), or a Radial_dimension_tolerance (see 4.2.172). The data associated with a Size_tolerance are the following:

— applied_shape.

4.2.204.1 applied_shape

The applied_shape specifies the physical shape of the Part that is being toleranced. See 4.3.255 for the application assertion.

4.2.205 Slot

A Slot is a type of Multi_axis_feature(see 4.2.125) that is a channel or depression with continuous direction of travel. The Slot origin shall be located at one end of the slot, the Z-axis shall indicate the direction of the slot, and the Y-axis shall indicate the direction away from the part.

NOTE - Figure 95 illustrates types of Slots.

The data associated with a Slot are the following:

— course_of_travel;

— end_conditions;

— sweep_shape.

4.2.205.1 course_of_travel

The course_of_travel specifies a 3D space curve, that when combine with a Profile, creates the shape of the Slot. See 4.3.257 for the application assertion.

 ©ISO 2000 - All rights reserved

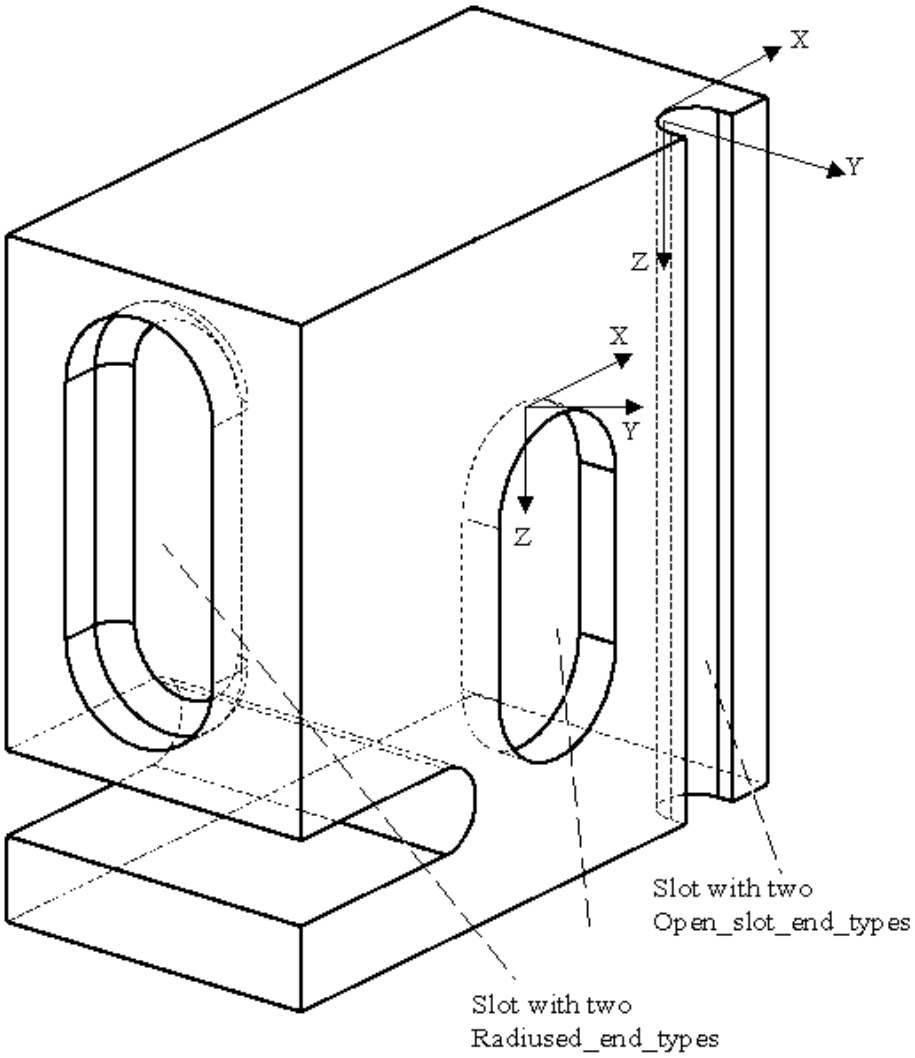


Figure 95 - Slot

4.2.205.2 end_conditions

The end_conditions specifies the type of implicit shape at the ends of the Slot. See 4.3.258 for the application assertion.

4.2.205.3 sweep_shape

The sweep_shape specifies the implicit 2D profile definition that, when combine with a Path, creates the shape of the Slot. See 4.3.256 for the application assertion.

©ISO 2000 - All rights reserved

4.2.206 Slot_end_type

A Slot_end_type is the end conditions of a slot. A slot shall have two ends, each end shall be open or closed. Each Slot_end_type is either a Flat_slot_end_type (see 4.2.76), Open_slot_end_type (see 4.2.131), Radiused_slot_end_type (see 4.2.173), or a Woodruff_slot_end_type (see 4.2.237). The data associated with a Slot_end_type are the following:

— first_or_second.

4.2.206.1 first_or_second

The first_or_second specifies a value of 'FIRST' if the Slot_end_type is closest to the positioning point of a Slot or 'SECOND' if it is the farthest away.

4.2.207 Specification

A Specification is a document that defines information pertaining to properties or processes for a part or an aspect of a part. The data associated with a Specification are the following:

— constraint;

— specification_class;

— specification_description;

— specification_id.

4.2.207.1 constraint

The constraint specifies the restriction on the Specification. The constraint need not be specified for a particular Specification. There may be more than one constraint for a Specification. See 4.3.259 for the application assertion.

4.2.207.2 specification_class

The specification_class specifies a section within a Specification that is divided into classes. A Specification may but need not require a specification_class.

4.2.207.3 specification_description

The specification_description specifies in human interpretable prose a description of the contents of the specification and any notes with respect to the Specification. A Specification may but need not require a specification_description.

4.2.207.4 specification_id

The specification_id specifies a unique identifier of the document.

 ©ISO 2000 - All rights reserved

4.2.208 Specification_usage_constraint

A Specification_usage_constraint is a restriction on the application of information defined within a Specification. The data associated with a Specification_usage_constraint are the following:

— class_id;

— element.

4.2.208.1 class_id

The class_id specifies the data or range of data with respect to the element that defines the restriction imposed on the usage of the Specification.

4.2.208.2 element

The element specifies the particular piece or area of information that is being restricted within the Specification.

4.2.209 Spherical_cap

A Spherical_cap is a type of Machining_feature (see 4.2.111) that is circular about an axis of rotation. A Spherical_cap consists of all points a given distance from a point constituting its center. The Z-axis shall be in the direction away from the material.

NOTE - Figure 96 illustrates the Spherical_cap.

The data associated with a Spherical_cap are the following:

— internal_angle;

— radius.

4.2.209.1 internal_angle

The internal_angle specifies the size of an angle from an axis for defining a portion of a sphere to use as a spherical_cap feature. See 4.3.260 for the application assertion. The X-axis defines the start of the spherical_cap and the internal_angle is measured from this axis.

4.2.209.2 radius

The radius specifies the constant distance from a point for defining a sphere. See 4.3.260 for the application assertion.

©ISO 2000 - All rights reserved

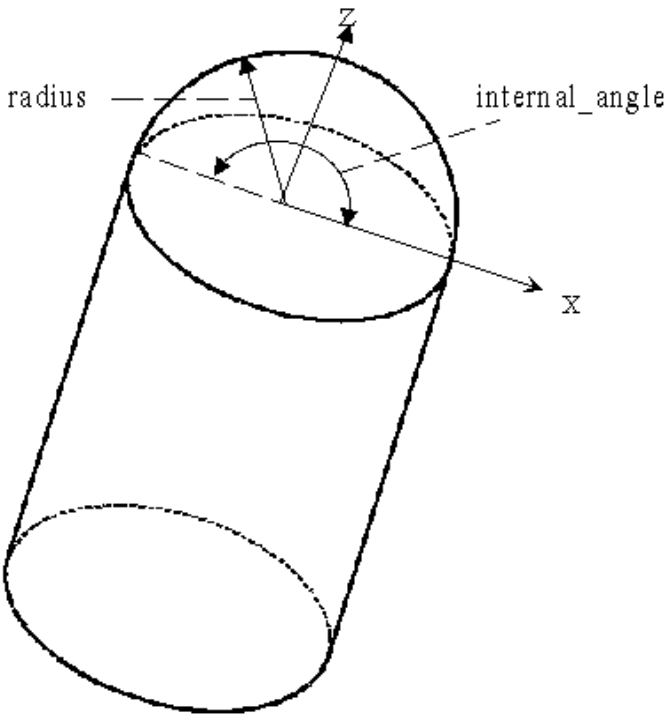


Figure 96 - Spherical_cap

4.2.210 Spherical_hole_bottom

A Spherical_hole_bottom is a type of Blind_bottom_condition (see 4.2.8) that is a bottom of a Round_-hole which is concentric about an axis and defined by a radius. The radius is the same as the radius of the hole.

NOTE - Figure 97 illustrates the Spherical_hole_bottom.

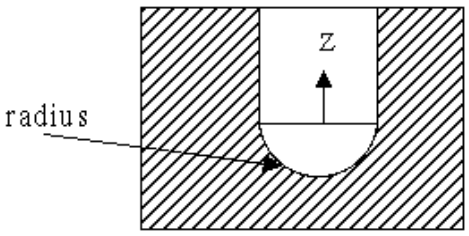


Figure 97 - Spherical_hole_bottom

 ©ISO 2000 - All rights reserved

The data associated with Spherical_hole_bottom are the following:

— radius.

4.2.210.1 radius

The radius specifies the radius at the bottom of the Round_hole. See 4.3.261 for the application assertion.

4.2.211 Square_U_profile

A Square_U_profile is a type of Open_profile (see 4.2.130) that is a shape bounded by three lines. One is the base line and has a defined length. The other two lines begin at the ends of the base line, and extend infinitely at any angle to the base line greater than zero degrees and less than 180 degrees. The two lines may also be at right angle to the base line. The corners of the Square_U_profile need not be blended by a radius.

NOTE - Figure 98 illustrates the Square_U_profile.

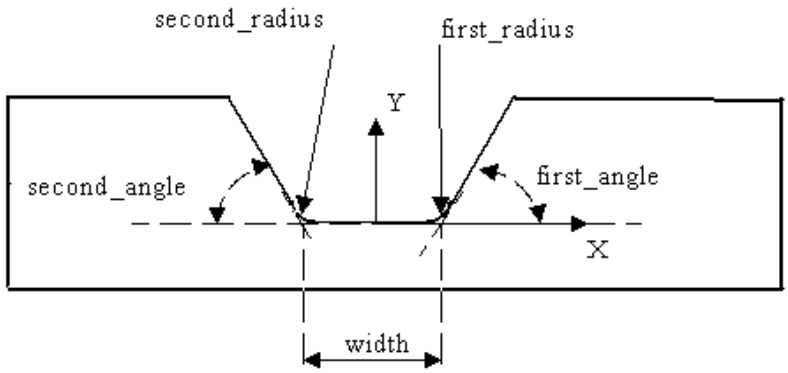


Figure 98 - Square_U_profile

The data associated with Square_U_profile are the following:

— first_angle;

— first_radius;

— second_angle;

— second_radius;

— width.

©ISO 2000 - All rights reserved

4.2.211.1 first_angle

The first_angle specifies the size of an angle between one side of the profile and the base. See 4.3.262 for the application assertion.

4.2.211.2 first_radius

The first_radius specifies the radius shape blend between one side of the profile and the base. See 4.3.262 for the application assertion.

4.2.211.3 second_angle

The second_angle specifies the size of an angle between the second side of the profile and the base. See 4.3.262 for the application assertion.

4.2.211.4 second_radius

The second_radius specifies the radius shape blend between the second side of the profile and the base. See 4.3.262 for the application assertion.

4.2.211.5 width

The width specifies the size of the base line for a Square_U_profile. See 4.3.262 for the application assertion.

4.2.212 Step

A Step is a type of Multi_axis_feature(see 4.2.125) that is a linear sweep of a shape. The shape shall be specified by two lines that connect at a point and extend infinitely. The enclosed angle shall be smaller than 180 degrees. The intersection of the two lines need not be blended with a radius.

NOTE - Figure 99 illustrates the Step.

The data associated with a Step are the following:

— course_of_travel;

— removal_boundary.

4.2.212.1 course_of_travel

The course_of_travel specifies the straight line with magnitude and direction. The placement and orientation of the Linear_path shall be the same as the Step feature. See 4.3.264 for the application assertion.

 ©ISO 2000 - All rights reserved

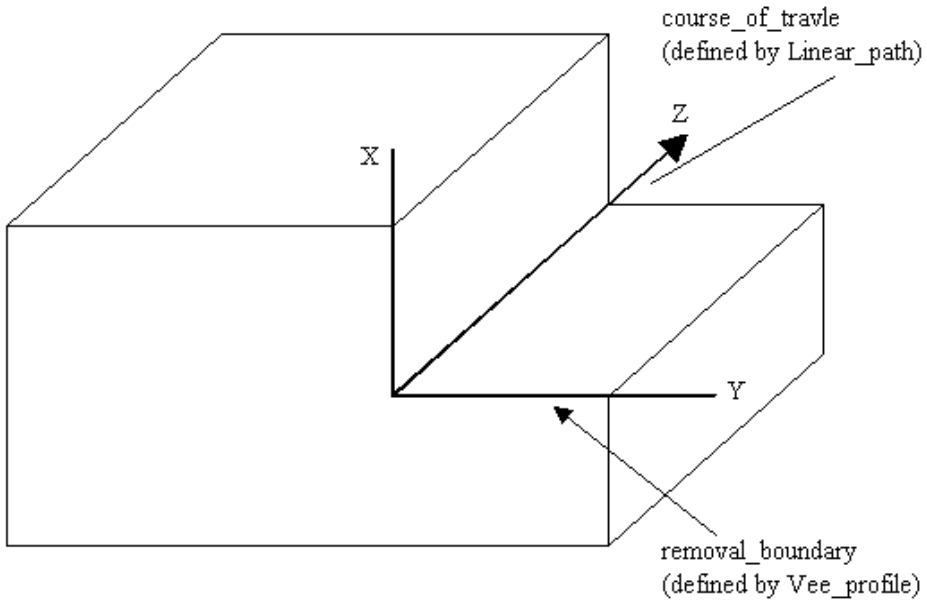


Figure 99 - Step

4.2.212.2 removal_boundary

The removal_boundary specifies the a Vee_profile (see 4.2.236) that when swept along a path defines the area on a part for volume removal. The placement and orientation of the Vee_profile shall be the same as the Step feature. See 4.3.263 for the application assertion.

4.2.213 Straight_knurl

A Straight_knurl is a type of Turned_knurl (see 4.2.235) that is typified by knurl scoring that is parallel to the axis of the scored surface.

NOTE - Figure 100 illustrates the Straight_knurl.

4.2.214 Straightness_tolerance

A Straightness_tolerance is a type of Geometric_tolerance (see 4.2.95) that is the amount of deviation a surface or line shall have from being straight. No element of the surface or line deviate more than a specified tolerance amount from a straight line.

NOTE 1 - Figure 101 illustrates the Straightness_tolerance.

NOTE 2 - The Straightness_tolerance definition is derived from paragraph 14.1 of ISO 1101.

©ISO 2000 - All rights reserved

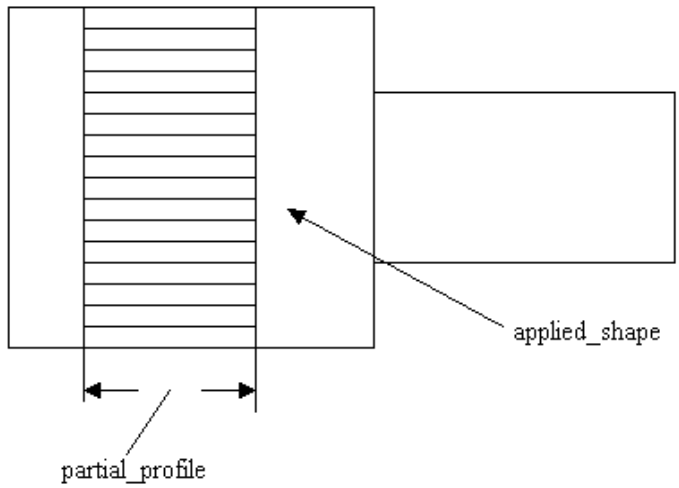


Figure 100 - Straight_knurl

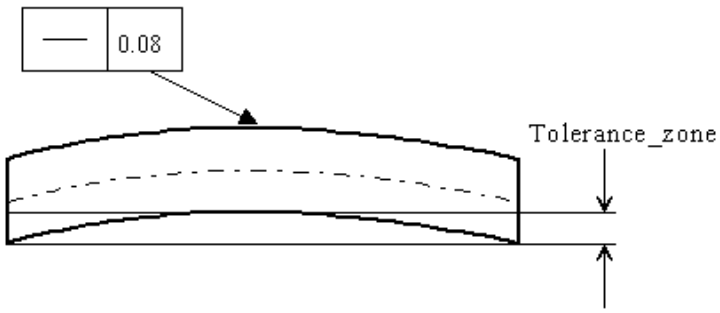


Figure 101 - Straightness_tolerance

The data associated with a Straightness_tolerance are the following:

— affected_plane;

— segment_size.

4.2.214.1 affected_plane

The affected_plane specifies the plane to apply the tolerance value. The affected_plane is equivalent to a 2D drawing view. The affected_plane need not be specified for a particular Straightness_tolerance. See 4.3.265 for the application assertion.

 ©ISO 2000 - All rights reserved

4.2.214.2 segment_size

The segment_size specifies the length of a surface to apply a tolerance if the Straightness_tolerance is not applied to the total length. The segment_size need not be specified for a particular Straightness_tolerance.

4.2.215 Surface_profile_tolerance

A Surface_profile_tolerance is a type of Geometric_tolerance (see 4.2.95) that is a uniform boundary or zone along the true profile within which all elements of the surface shall lie.

NOTE 1 - Figure 102 illustrates Surface_profile_tolerance.

NOTE 2 - The Surface_profile_tolerance definition is derived from paragraph 14.6 of ISO 1101.

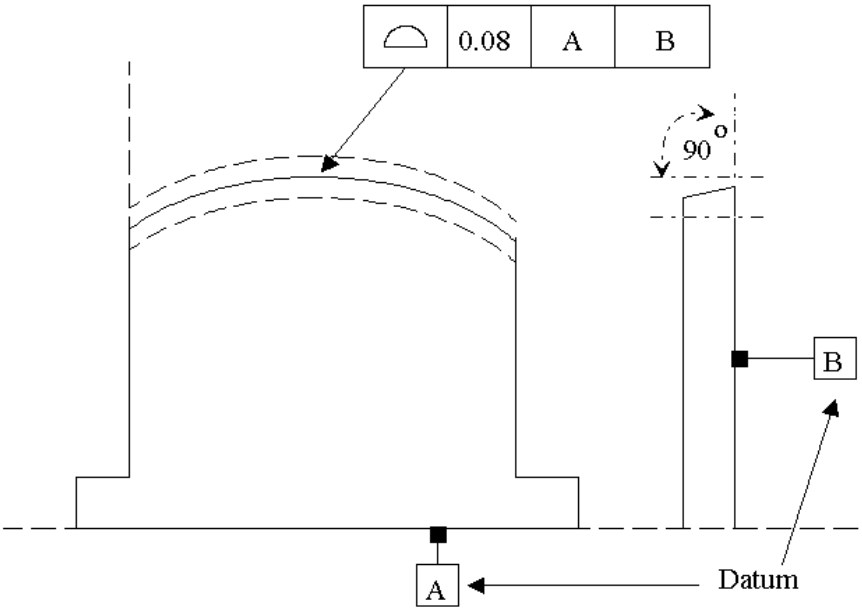


Figure 102 - Surface_profile_tolerance

The data associated with a Surface_profile_tolerance are the following:

— geometric_reference.

4.2.215.1 geometric_reference

The geometric_reference specifies the datum to which the tolerance is related. See 4.3.266 for the application assertion.

©ISO 2000 - All rights reserved

4.2.216 Surface_property

A Surface_property specifies characteristics of a surface that are elements of the shape of a part. The data associated with a Surface_property are the following:

— property_characteristics;

— surface_finish.

4.2.216.1 property_characteristics

The property_characteristics specifies the parameter to describe the Surface_property. See 4.3.267 for the application assertion.

4.2.216.2 surface_finish

The surface_finish specifies a boolean value that indicates a type of Surface_property is a surface finish. A value of true specifies the Surface_property is a surface finish property, a value of false specifies the Surface_property is for other surface properties.

4.2.217 Symmetry_tolerance

A Symmetry_tolerance is a type of Geometric_tolerance (see 4.2.95) that defines a tolerance zone limited by two parallel planes a distance apart and disposed symmetrically to the median plane with respect to a datum axis or datum plane; or when projected in a plane, limited by two parallel straight lines a distance apart and disposed symmetrically with respect to a datum axis or datum plane.

NOTE 1 - Figure 103 illustrates the Symmetry_tolerance.

NOTE 2 - The Symmetry_tolerance definition is derived from paragraph 14.12 of ISO 1101.

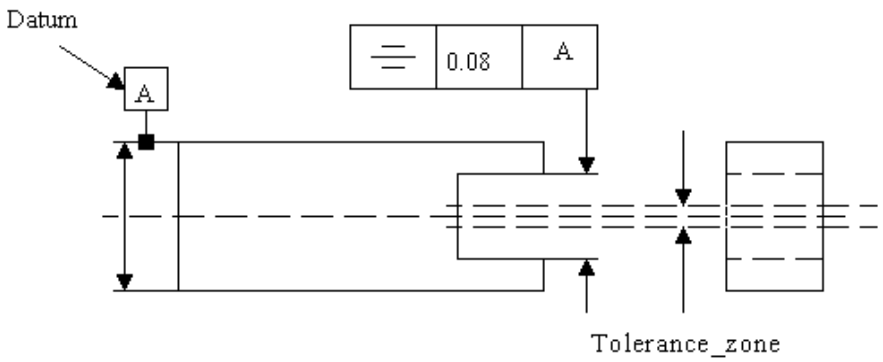


Figure 103 - Symmetry_tolerance

 ©ISO 2000 - All rights reserved

The data associated with a Symmetry_tolerance are the following:

— affected_plane;

— geometric_reference.

4.2.217.1 affected_plane

The affected_plane specifies the plane to apply the tolerance value. The affected_plane is equivalent to a 2D drawing view. The affected_plane need not be specified for a particular Symmetry_tolerance. See 4.3.269 for the application assertion.

4.2.217.2 geometric_reference

The geometric_reference specifies the datum to which the tolerance is related. See 4.3.268 for the application assertion.

4.2.218 Target_area

A Target_area is a type of Datum_target (see 4.2.51) that is an enclosed area bounded by an arbitrary shape required to define a Datum_target. The shape of the Target_area is described explicitly by a set of curves. The data associated with Target_area are the following:

— area_shape.

4.2.218.1 area_shape

The area_shape specifies the physical form of the Target_area shape. See 4.3.270 for the application assertion.

4.2.219 Target_circle

A Target_circle is a type of Placed_target (see 4.2.152) that is an enclosed area bounded by a circle required to define a Datum_target. The origin of the Datum_target is the center of the circle, and the orientation is the x-y plane. The data associated with a Target_circle are the following:

— target_diameter.

4.2.219.1 target_diameter

The target_diameter specifies the diameter value of the Target_circle.

©ISO 2000 - All rights reserved

4.2.220 Target_line

A Target_line is a type of Placed_target (see 4.2.152) that is a straight curve. The origin shall be the first end point of the Target_line, the second end point shall be located on the Z-axis at a specified length. The data associated with a Target_line are the following:

— target_length.

4.2.220.1 target_length

The target_length specifies the length value of the Target_line.

4.2.221 Target_point

A Target_point is a type of Placed_target (see 4.2.152) that is a single point. The origin shall be at the Target_point.

4.2.222 Target_rectangle

A Target_rectangle is a type of Placed_target (see 4.2.152) that is an area bounded by four sides with opposite sides equal in length. The center of the rectangle is at the origin. The orientation of the rectangle is with the length along the X-axis and the width along the Y-axis. The data associated a with Target_rectangle are the following:

— target_length;

— target_width.

4.2.222.1 target_length

The target_length specifies the length value of the Target_rectangle.

4.2.222.2 target_width

The target_width specifies the width value of the Target_rectangle.

4.2.223 Tee_profile

A Tee_profile is a type of Open_profile (see 4.2.130) the cross-section of which has the shape of the twentieth letter of the English alphabet in capital form. The first line has a defined length. The second line begins at the midpoint of the first line and is perpendicular to it. The second line extends infinitely. The corners of the Tee_profile need not be blended by a radius. The profile is positioned with the opening in the direction of the Y-axis. The orientation is at a point on the profile the farthest distance from the opening measured along the Y-axis. The X-axis is tangent to the bottom of the profile.

NOTE - Figure 104 illustrates the Tee_profile.

 ©ISO 2000 - All rights reserved

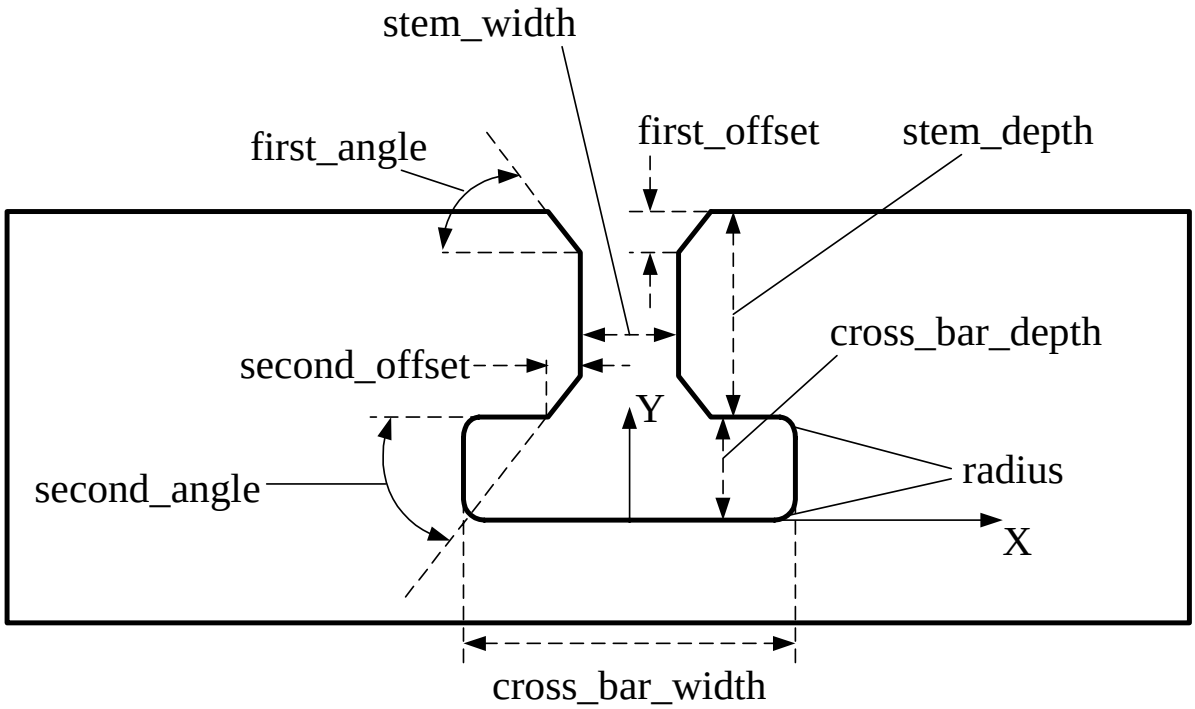


Figure 104 - Tee_profile

The data associated with a Tee_profile are the following:

— cross_bar_depth;

— cross_bar_width;

— depth;

— first_angle;

— first_offset;

— radius;

— second_angle;

— second_offset;

— width.

4.2.223.1 cross_bar_depth

The cross_bar_depth specifies the depth dimension of the Tee cross bar size. See 4.3.271 for the application assertion.

©ISO 2000 - All rights reserved

4.2.223.2 cross_bar_width

The cross_bar_width specifies the width dimension of the Tee cross bar size. See 4.3.271 for the application assertion.

4.2.223.3 depth

The depth specifies the depth dimension of the Tee stem. See 4.3.271 for the application assertion.

4.2.223.4 first_angle

The first_angle specifies the angular measurement for creating a chamfer on the open end of a Tee_-profile. See 4.3.271 for the application assertion.

4.2.223.5 first_offset

The first_offset specifies the distance from the edge of the Tee stem to create a chamfer on the open end of a Tee_profile. See 4.3.271 for the application assertion.

4.2.223.6 radius

The radius specifies the arc size for blending the sides of a Tee_profile cross bar. See 4.3.271 for the application assertion.

4.2.223.7 second_angle

The second_angle specifies the angular measurement for creating a chamfer between the stem and the cross bar parts of a Tee_profile. See 4.3.271 for the application assertion.

4.2.223.8 second_offset

The second_offset specifies a distance from the edge of the Tee stem to create a chamfer a distance from the edge of a surface to the finish of a chamfer. See 4.3.271 for the application assertion.

4.2.223.9 width

The width specifies the width dimension of the Tee stem. See 4.3.271 for the application assertion.

 ©ISO 2000 - All rights reserved

4.2.224 Thread

A Thread is a type of Machining_feature (see 4.2.111) that is a ridge of uniform section on the form of a helix on the external or internal surface of a cylinder. Each Thread is either a Catalogue_thread (see 4.2.18) or a Defined_thread (see 4.2.55).

NOTE 1 - Figure 105 illustrates the Thread and Figure 35 illustrates Thread attributes.

NOTE 2 - Threads may be used to screw parts together.

NOTE 3 - An outside thread might be on a Outer_round, an inside thread might be in a Round_hole.

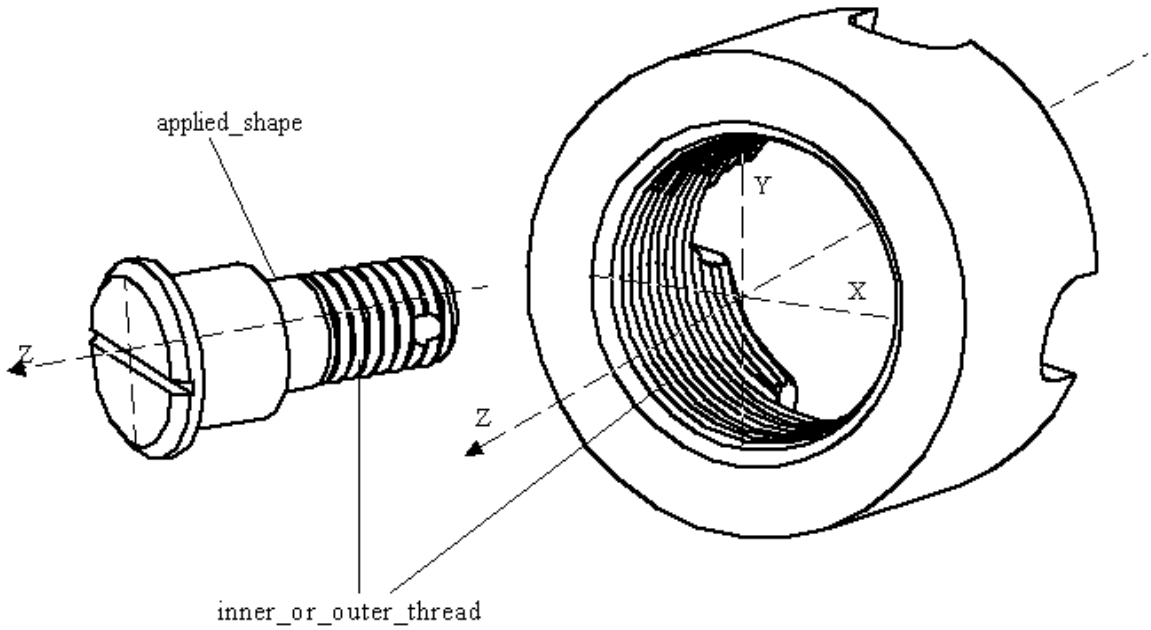


Figure 105 - Thread

The data associated with a Thread are the following:

— applied_shape;

— fit_class;

— form;

— inner_or_outer_thread;

— major_diameter;

— number_of_threads;

©ISO 2000 - All rights reserved

— partial_profile;

— qualifier;

— thread_hand.

4.2.224.1 applied_shape

The applied_shape specifies the physical shape of the Part that will define where the Thread feature will be applied. See 4.3.275 for the application assertion.

4.2.224.2 fit_class

The fit_class specifies the value for the type of fit specification for the thread. These types are distinguished from each other by the amount of tolerance and allowance. The fit_class need not be specified for a particularThread. See 4.3.272 for the application assertion.

EXAMPLE - Examples of ANSI fit class are: 1A, 2A, and 3A which apply to external threads only, and 1B, 2B, and 3B which apply to internal threads only.

4.2.224.3 form

The form specifies the definition of the shape of the thread. Various forms of threads are used to hold parts together, to adjust parts with reference to each other, or to transmit power. See 4.3.272 for the application assertion.

EXAMPLE - Examples of form are: metric, square, unified, sharp V, buttress, standard worm, and knuckle.

4.2.224.4 inner_or_outer_thread

An inner_or_outer_thread specifies whether or not the thread is applied as an internal thread or an external thread.

4.2.224.5 major_diameter

The major_diameter specifies the dimension of the largest diameter of the Thread and is applied to both an internal and an external thread. See 4.3.273 for the application assertion.

4.2.224.6 number_of_threads

The number_of_threads specifies the density of threads per inch when used with English unit of measure and is the thread pitch when used with metric unit of measure. See 4.3.273 for the application assertion.

4.2.224.7 partial_profile

The partial_profile specifies the limitations to be applied on the Thread feature. See 4.3.274 for the application assertion.

 ©ISO 2000 - All rights reserved

4.2.224.8 qualifier

The qualifier specifies additional text information that describes a Thread. The qualifier need not be specified for a particular Thread. See 4.3.272 for the application assertion.

4.2.224.9 thread_hand

The thread_hand specifies a description of whether the thread is right or left handed. When viewed toward an end, a right hand winds in a clockwise direction and a left hand winds in a counterclockwise direction. See 4.3.272 for the application assertion.

4.2.225 Through_bottom_condition

A Through_bottom_condition is a selection type of Hole_bottom_condition_select that shall pass through two faces of a part; the depth is specified by the feature. The Through_bottom_condition length is specified by the size of the Hole feature.

NOTE - Figure 106 illustrates the Through_bottom_condition.

4.2.226 Through_pocket_bottom_condition

A Through_pocket_bottom_condition is a pocket that passes through two faces of a part; the depth is defined by the pocket_depth (see 4.2.160.4) of the Pocket feature.

NOTE - Figure 52 illustrates Through_pocket_bottom_condition.

4.2.227 Through_profile_floor

A Through_profile_floor is a Shape_profile which passes through two faces of a part; the depth is specified by the size of the feature.

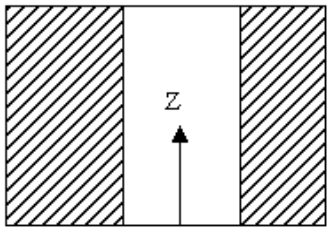


Figure 106 - Through_bottom_condition

4.2.228 Tolerance_limit

A Tolerance_limit is an upper or lower tolerance value applied directly to a dimension. When applied to a Dimensional_tolerance, the dimensional_value (see 4.2.63.2) shall be a tolerance value. When

©ISO 2000 - All rights reserved

applied to a Numeric_parameter_with_tolerance, the parameter_value (see 4.2.128.2) shall be a tolerance value. There shall be a qualifier that describes the tolerance context. The data associated with a Tolerance_limit are the following:

— limit_qualifier.

4.2.228.1 limit_qualifier

The limit_qualifier specifies a description of the Tolerance_limit context.

EXAMPLE - '30.5 MAX' or '5 MIN' are Tolerance_limit examples and the words 'MAX' or 'MIN' are limit_qualifier examples.

4.2.229 Tolerance_range

A Tolerance_range is the upper and lower tolerance range applied directly to a dimension. When applied to a Dimensional_tolerance, the dimensional_value (see 4.2.63.2) may be a nominal tolerance value. When applied to a Numeric_parameter_with_tolerance, the parameter_value (see 4.2.128.2) may be a nominal tolerance value. The data associated with a Tolerance_range are the following:

— lower_range;

— significant_digits;

— upper_range.

4.2.229.1 lower_range

The lower_range specifies the lowest allowable value for a dimensional tolerance.

4.2.229.2 significant_digits

The significant_digits specifies the number of decimal places indicating the accuracy of the tolerance.

4.2.229.3 upper_range

The upper_range specifies the highest allowable value for a dimensional tolerance.

4.2.230 Tolerance_value

A Tolerance_value is the representation of the magnitude of the allowable deviation required for dimensions. These tolerance values may be explicitly defined or may require a specification for definition. The data associated with a Tolerance_value are the following:

— defined_value;

— envelope.

 ©ISO 2000 - All rights reserved

4.2.230.1 defined_value

The defined_value specifies the tolerance deviation value. See 4.3.276, 4.3.277, 4.3.278, and 4.3.279 for the application assertion.

4.2.230.2 envelope

The envelope specifies that each geometric constraint has to be fulfilled in itself. The envelope of the perfect shape corresponding to the maximum material shall not be larger than the specified dimension and tolerance. The envelope attribute shall be a boolean value, if TRUE the envelope is required for the Tolerance_value.

4.2.231 Tolerance_zone

A Tolerance_zone is an area where all points of the geometric element that have tolerances shall be contained.

EXAMPLE - A point, line, surface, or plane are examples for geometric elements that have tolerances zones.

The data associated with a Tolerance_zone are the following:

— common_zone;

— extended_shape;

— form_type;

— zone_definition.

4.2.231.1 common_zone

The common_zone specifies a boolean value that indicates if a Tolerance_zone is applied to more than one geometric tolerance. A TRUE value would indicate a common zone.

4.2.231.2 extended_shape

The extended_shape specifies the extension of a feature for the purpose of creating the Tolerance_zone. The extended_shape need not be specified for a particular Tolerance_zone. See 4.3.281 for the application assertion.

4.2.231.3 form_type

The form_type specifies the shape of the Tolerance_zone.

EXAMPLE - 'Cylindrical', 'parallelepiped', 'spherical' are examples for form_type.

©ISO 2000 - All rights reserved

4.2.231.4 zone_definition

The zone_definition specifies the defining boundaries for a Tolerance_zone. See 4.3.280 for the application assertion.

4.2.232 Tolerance_zone_definition

A Tolerance_zone_definition is the boundaries of a Tolerance_zone. Each Tolerance_zone shall be defined by at least one shape and may be defined with two shapes.

The data associated with a Tolerance_zone_definition are the following:

— first_element;

— second_element.

4.2.232.1 first_element

The first_element specifies a geometric shape for defining the boundary for the Tolerance_zone_-definition. A second element specifies a second geometric shape. See 4.3.282 for the application assertion.

4.2.232.2 second_element

The second_element specifies the second of two shapes for defining the boundary for the Tolerance_-zone_definition. The second_element need not be specified for a particular Tolerance_zone_definition. See 4.3.282 for the application assertion.

4.2.233 Total_runout_tolerance

A Total_runout_tolerance is a type of Geometric_tolerance (see 4.2.95) that is a compound tolerance used to control the functional relationship of one or more surfaces of a part to a datum axis. The types of surfaces controlled by Total_runout_tolerance tolerances include those surfaces constructed around a datum axis and those constructed at right angles to a datum axis. Surfaces shall be within the tolerance when the part is rotated about the datum axis.

NOTE 1 - Figure 107 illustrates the Total_runout_tolerance.

NOTE 2 - The Total_runout_tolerance definition is derived from paragraph 14.13 of ISO 1101.

The data associated with a Total_runout_tolerance are the following:

— geometric_reference;

— runout.

 ©ISO 2000 - All rights reserved

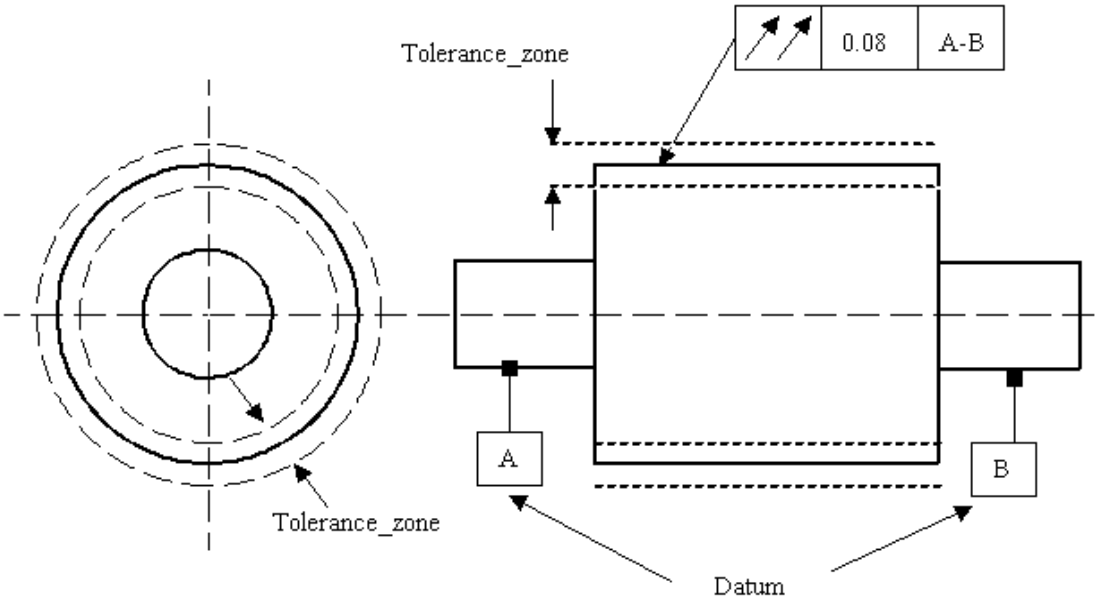


Figure 107 - Total_runout_tolerance

4.2.233.1 geometric_reference

The geometric_reference specifies the datum to which the tolerance is related. See 4.3.283 for the application assertion.

4.2.233.2 runout

The runout specifies the direction to control a runout tolerance. If the angle is specified the runout tolerance applies in this angle which is fixed with respect to the datum axis. The runout need not be specified for a particular Total_runout_tolerance.

4.2.234 Transition_feature

A Transition_feature is a type of Manufacturing_feature (see 4.2.114) that is a transition area between two surfaces. This feature differs from Machining_feature objects in that it requires no orientation for placement. Each Transition_feature is either a Chamfer (see 4.2.19), Edge_round (see 4.2.67), or a Fillet (see 4.2.73).

©ISO 2000 - All rights reserved

4.2.235 Turned_knurl

A Turned_knurl is a type of Knurl (see 4.2.102) that is a scoring pattern consisting of a series of shallow cuts on a cylindrical surface. Each Turned_knurl is either a Diagonal_knurl (see 4.2.58), Diamond_knurl (see 4.2.61), or a Straight_knurl (see 4.2.213).

NOTE 1 - Figure 108 illustrates the Turned_knurl.

NOTE 2 - A knurl may be used to aid in gripping a part.

The data associated with a Turned_knurl are the following:

— diametral_pitch;

— major_diameter;

— nominal_diameter;

— number_of_teeth;

— root_fillet;

— tooth_depth.

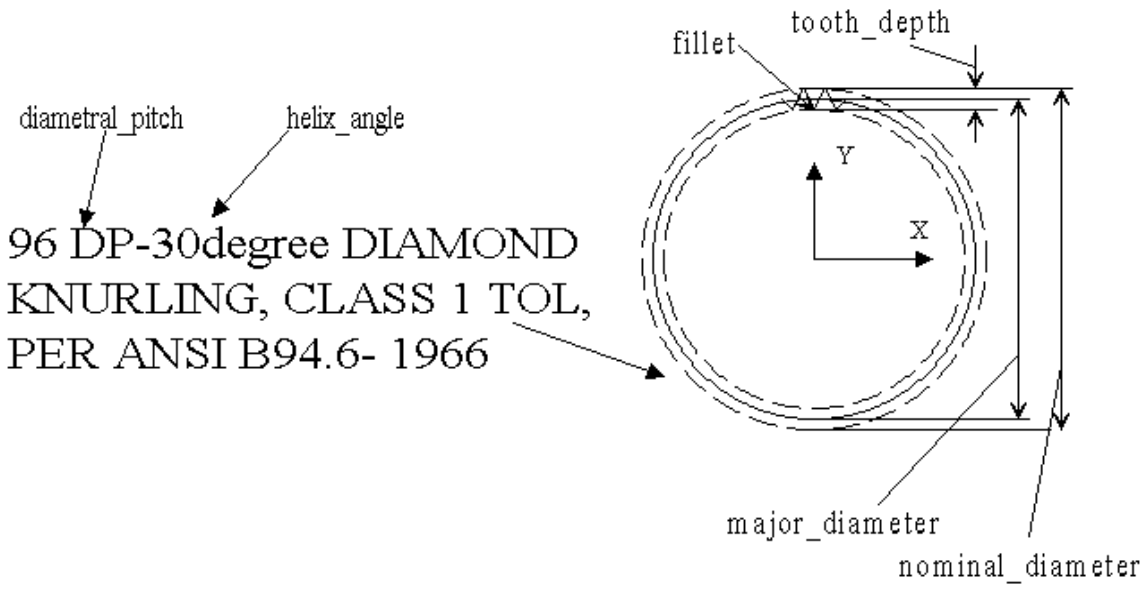


Figure 108 - Turned_knurl

 ©ISO 2000 - All rights reserved

4.2.235.1 diametral_pitch

The diametral_pitch specifies the ratio of the number of teeth in the circumference to the nominal diameter. See 4.3.284 for the application assertion.

4.2.235.2 major_diameter

The major_diameter specifies size of the part before a knurl is applied to it. See 4.3.284 for the application assertion.

4.2.235.3 nominal_diameter

The nominal_diameter specifies the size of the part after a knurl has been applied. See 4.3.284 for the application assertion.

4.2.235.4 number_of_teeth

The number_of_teeth specifies the number of teeth in the circumference produced on the part surface. The number_of_teeth need not be specified for a particular Turned_knurl. See 4.3.284 for the application assertion.

4.2.235.5 root_fillet

The root _fillet specifies the dimension of a radius between teeth on a knurling tool. See 4.3.284 for the application assertion.

4.2.235.6 tooth_depth

The tooth_depth specifies the depth from the crest of a tooth to the point where two teeth intersect. See 4.3.284 for the application assertion.

4.2.236 Vee_profile

A Vee_profile is a type of Open_profile (see 4.2.130) that is a shape bounded by two lines that connect at a point and extends infinitely. The enclosed angle is less than 180 degrees. The intersection of the two lines need not be blended with a radius. The profile is positioned with the opening in the direction of the Y axis. The Y-axis intersects the angle between the two sides.

NOTE - Figure 109 illustrates the Vee_profile.

The data associated with a Vee_profile are the following:

— profile_angle;

— profile_radius;

— tilt_angle.

©ISO 2000 - All rights reserved

4.2.236.1 profile_angle

The profile_angle specifies the size of the angle between the two sides of the Vee_profile. The angle shall be greater than 0 and not more than 180 degrees. See 4.3.285 for the application assertion.

4.2.236.2 profile_radius

The profile_radius specifies the size of the blend radius at the point of the V, or where the two sides come together. See 4.3.285 for the application assertion.

4.2.236.3 tilt_angle

The tilt_angle specifies the size of the angle between one side of the Vee_profile and the x-axis of the local coordinate system that defines the Vee_profile orientation on the part. See 4.3.285 for the application assertion.

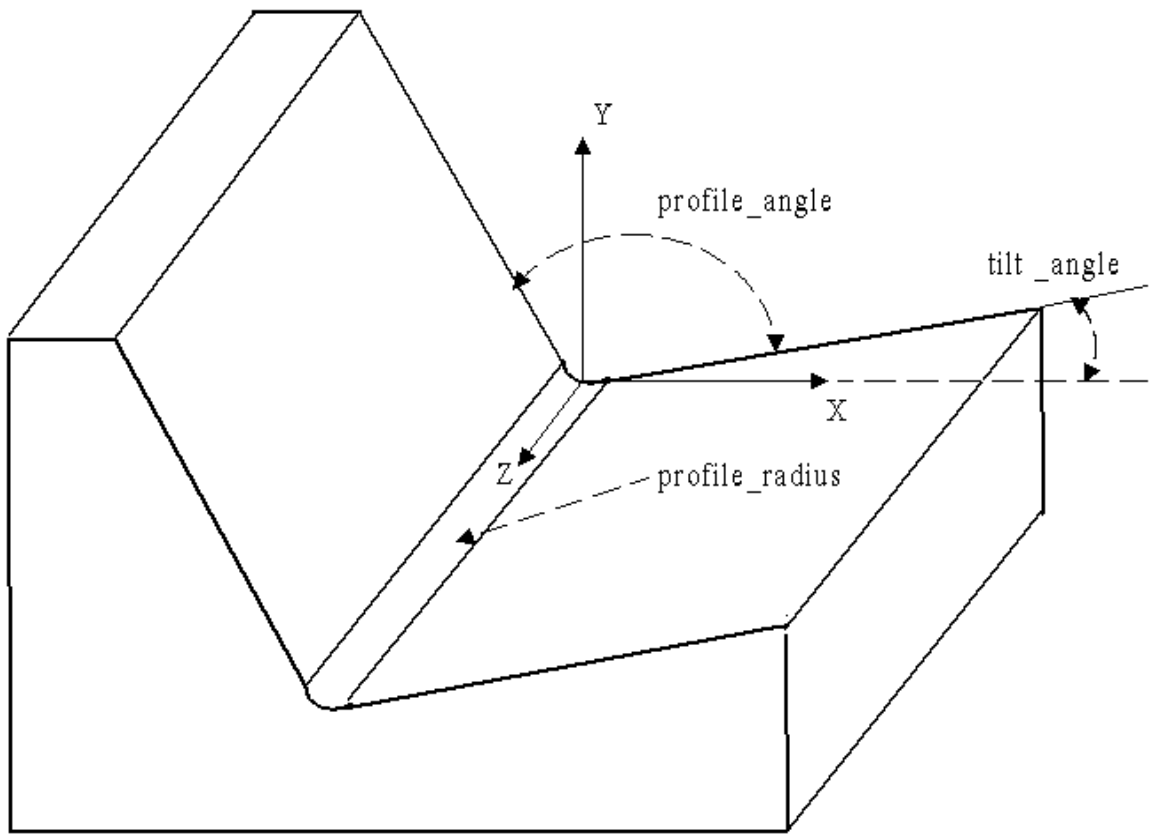


Figure 109 - Vee_profile

 ©ISO 2000 - All rights reserved

4.2.237 Woodruff_slot_end_type

A Woodruff_slot_end_type is a type of Slot_end_type (see 4.2.206) that is an end condition of a slot that shall be a radius tangent to the Slot bottom, and curved upward about an axis.

NOTE - Figure 110 illustrates the Woodruff_slot_end_type.

The data associated with a Woodruff_slot_end_type are the following:

— radius.

4.2.237.1 radius

The radius specifies the size of the radius swept about an axis, tangent to the Slot bottom and the end of a Slot. See 4.3.286 for the application assertion.

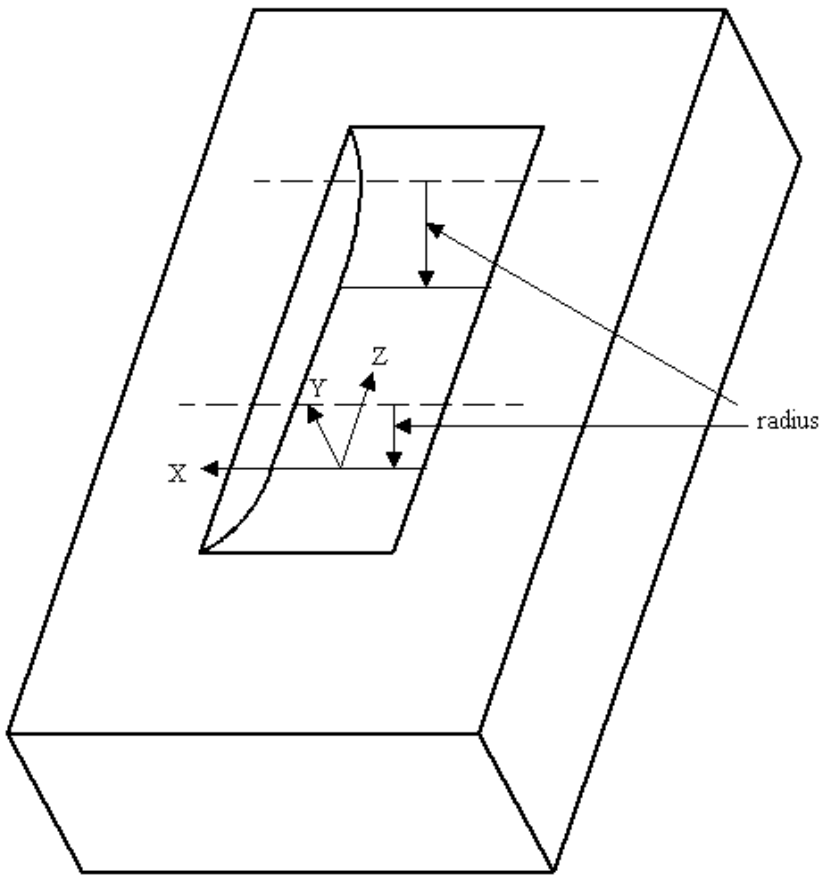


Figure 110 - Slot with Woodruff_slot_end_type at each end

©ISO 2000 - All rights reserved

4.3 Application assertions

This subclause specifies the application assertions for the Mechanical product definition for process planning using machining features application protocol. Application assertions specify the relationships between application objects, the cardinality of the relationships, and the rules required for the integrity and validity of the application objects and UoFs. The application assertions and their definitions are given below.

4.3.1 Alternate_material to Material

Each Alternate_material has the material_substitute defined by exactly one Material. Each Material is the material_substitute for zero, one, or many Alternate_material objects.

4.3.2 Angle_taper to Numeric_parameter

Each Angle_taper has the angle defined by exactly one Numeric_parameter. Each Numeric_parameter defines the angle for zero, one, or many Angle_taper objects.

4.3.3 Angularity_tolerance to Datum

Each Angularity_tolerance has the geometric_reference defined by exactly one Datum. Each Datum is the geometric_reference for zero, one, or many Angularity_tolerance objects.

4.3.4 Approval to Person_in_organization

Each Approval has the approval_authority defined by one or many Person_in_organization objects. Each Person_in_organization is the approval_authority for zero, one, or many Approval objects.

4.3.5 Block_base_shape to Numeric_parameter

Each Block_base_shape has the width defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the width for zero, one, or many Block_base_shape objects.

Each Block_base_shape has the height defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the height for zero, one, or many Block_base_shape objects.

4.3.6 Boss to Boss_top_condition

Each Boss has the top_condition defined by exactly one Boss_top_condition. Each Boss_top_condition defines the top_condition for zero, one or many Boss objects.

4.3.7 Boss to Linear_path

Each Boss has the boss_height defined by exactly one Linear_path. Each Linear_path defines the boss_-height for zero, one, or many Boss objects.

 ©ISO 2000 - All rights reserved

4.3.8 Boss to Numeric_parameter

Each Boss has the fillet_radius defined by zero or one Numeric_parameter. Each Numeric_parameter defines the fillet_radius for zero, one, or many Boss objects.

Each Boss has the top_radius defined by zero or one Numeric_parameter. Each Numeric_parameter defines the top_radius for zero, one, or many Boss objects.

4.3.9 B-rep_model_element to B-rep_model

Each B-rep_model_element defines an element of exactly one B-rep_model. Each B-rep_model defines the element for zero, one, or many B-rep_model_element objects.

4.3.10 B-rep_shape_aspect_representation to B-rep_model

Each B-rep_shape_aspect_representation has the shape_definition defined by exactly one B-rep_model. Each B-rep_model defines the shape_definition for zero, one, or many B-rep_shape_aspect_-representation objects.

4.3.11 B-rep_shape_representation to B-rep_model

Each B-rep_shape_representation has the shape_definition defined by exactly one B-rep_model. Each B-rep_model defines the shape_definition for zero, one, or many B-rep_shape_representation objects.

4.3.12 Catalogue_knurl to Specification

Each Catalogue_knurl has documentation defined by exactly one Specification. Each Specification defines the documentation for zero, one, or many Catalogue_knurl objects.

4.3.13 Catalogue_marking to Specification

Each Catalogue_marking has documentation defined by exactly one Specification. Each Specification defines the documentation for zero, one, or many Catalogue_marking objects.

4.3.14 Catalogue_thread to Specification

Each Catalogue_thread has documentation defined by exactly one Specification. Each Specification defines the documentation for zero, one, or many Catalogue_thread objects.

4.3.15 Chamfer to Face_shape_element

Each Chamfer has the chamfer_face identified by exactly one Face_shape_element. Each Face_shape_-element is the chamfer_face shape for zero, one, or many Chamfer objects.

©ISO 2000 - All rights reserved

4.3.16 Chamfer to First_offset

Each Chamfer has the first_face_offset defined by exactly one First_offset. Each First_offset defines the first_face_offset for zero, one, or many Chamfer objects.

4.3.17 Chamfer to Second_chamfer_offset

Each Chamfer has the second_face_offset defined by exactly one Second_chamfer_offset. Each Second_chamfer_offset defines the second_face_offset for zero, one, or many Chamfer objects.

4.3.18 Chamfer_angle to Numeric_parameter

Each Chamfer_angle has the angle_amount defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the angle_amount for zero, one, or many Chamfer_angle objects.

4.3.19 Circular_boss to Circular_closed_profile

Each Circular_boss has the circular_profile defined by exactly one Circular_closed_profile. Each Circular_closed_profile defines the circular_profile for zero, one, or many Circular_boss objects.

4.3.20 Circular_boss to Angle_taper

Each Circular_boss has the change_in_diameter defined by zero or one Angle_taper. Each Angle_taper defines the change_in_diameter for zero, one, or many Circular_boss objects.

NOTE - This assertion is established through Taper_select.

4.3.21 Circular_boss to Diameter_taper

Each Circular_boss has the change_in_diameter defined by zero or one Diameter_taper. Each Diameter_taper defines the change_in_diameter for zero, one, or many Circular_boss objects.

NOTE - This assertion is established through Taper_select.

4.3.22 Circular_boss to Directed_taper

Each Circular_boss has the change_in_diameter defined by zero or one Directed_taper. Each Directed_-taper defines the change_in_diameter for zero, one, or many Circular_boss objects.

NOTE - This assertion is established through Taper_select.

4.3.23 Circular_closed_profile to Numeric_parameter

Each Circular_closed_profile has the diameter defined by exactly one Numeric_parameter. Each Numeric_parameter defines the diameter for zero, one, or many Circular_closed_profile objects.

 ©ISO 2000 - All rights reserved

4.3.24 Circular_closed_shape_profile to Circular_closed_profile

Each Circular_closed_shape_profile has the closed_boundary defined by exactly one Circular_closed_-profile. Each Circular_closed_profile defines the closed_boundary for zero, one, or many Circular_-closed_shape_profile objects.

4.3.25 Circular_cutout to Circular_closed_profile

Each Circular_cutout has the closed_boundary defined by exactly one Circular_closed_profile. Each Circular_closed_profile defines the closed_boundary for zero, one, or many Circular_cutout objects.

4.3.26 Circular_offset_pattern to Numeric_parameter

Each Circular_offset_pattern has the angular_offset defined by exactly one Numeric_parameter. Each Numeric_parameter defines the angular_offset for zero, one, or many Circular_offset_pattern objects.

Each Circular_offset_pattern has the index_number defined by exactly one Numeric_parameter. Each Numeric_parameter defines the index_number for zero, one, or many Circular_offset_pattern objects.

4.3.27 Circular_omit_pattern to Numeric_parameter

Each Circular_omit_pattern has the omit_index defined by exactly one Numeric_parameter. Each Numeric_parameter defines the omit_index for zero, one, or many Circular_omit_pattern objects.

4.3.28 Circular_path to Numeric_parameter

Each Circular_path has the radius defined by exactly one Numeric_parameter. Each Numeric_parameter defines the radius for zero, one, or many Circular_path objects.

4.3.29 Circular_pattern to Circular_offset_pattern

Each Circular_pattern has the relocated_base_feature defined by zero, one, or many Circular_offset_-pattern objects. Each Circular_offset_pattern defines the relocated_base_feature for one or more Circular_pattern objects.

4.3.30 Circular_pattern to Circular_omit_pattern

Each Circular_pattern has the missing_base_feature defined by zero, one, or many Circular_omit_pattern objects. Each Circular_omit_pattern defines the missing_base_feature for one or more Circular_pattern objects.

4.3.31 Circular_pattern to Numeric_parameter

Each Circular_pattern has the angular_spacing defined by exactly one Numeric_parameter. Each Numeric_parameter defines the angular_spacing for zero, one, or many Circular_pattern objects.

©ISO 2000 - All rights reserved

Each Circular_pattern has the base_feature_diameter defined by zero or. one Numeric_parameter. Each Numeric_parameter defines the base_feature_diameter for zero, one, or many Circular_pattern objects.

Each Circular_pattern has the base_feature_rotation defined by exactly one Numeric_parameter. Each Numeric_parameter defines the base_feature_rotation for zero, one, or many Circular_pattern objects.

Each Circular_pattern has the number_of_features defined by exactly one Numeric_parameter. Each Numeric_parameter defines the number_of_features for zero, one, or many Circular_pattern objects.

4.3.32 Circular_runout_tolerance to Datum

Each Circular_runout_tolerance has the geometric_reference defined by one or two Datum objects. Each Datum defines the geometric_reference for zero, one, or many Circular_runout_tolerance objects.

4.3.33 Compound_datum to Datum_feature

Each Compound_datum has the element defined by two or more Datum_feature objects. Each Datum_-feature defines the element for zero, one, or many Compound_datum objects.

4.3.34 Compound_feature to Compound_feature_element

Each Compound_feature has the element defined by two or more Compound_feature_element objects. Each Compound_feature_element defines the element for zero, one, or many Compound_feature objects.

4.3.35 Compound_feature_element to Machining_feature

Each Compound_feature_element has the element defined by exactly one Machining_feature. Each Machining_feature defines the element for zero, one, or many Compound_feature_element objects.

NOTE - This assertion is established through Compound_feature_select.

4.3.36 Compound_feature_element to Transition_feature

Each Compound_feature_element has the element defined by exactly one Transition_feature. Each Transition_feature defines the element for zero, one, or many Compound_feature_element objects.

NOTE - This assertion is established through Compound_feature_select.

4.3.37 Compound_feature_relationship to Compound_feature_element

Each Compound_feature_relationship has the successor defined by exactly one Compound_feature_-element. Each Compound_feature_element defines the successor for zero, one, or many Compound_-feature_relationship objects.

 ©ISO 2000 - All rights reserved

Each Compound_feature_relationship has the predecessor defined by exactly one Compound_feature_-element. Each Compound_feature_element defines the predecessor for zero, one, or many Compound_-feature_relationship objects.

4.3.38 Concentricity_tolerance to Datum

Each Concentricity_tolerance has the geometric_reference defined by exactly one Datum. Each Datum is the geometric_reference for zero, one, or many Concentricity_tolerance objects.

4.3.39 Conical_hole_bottom to Numeric_parameter

Each Conical_hole_bottom has the tip_angle defined by exactly one Numeric_parameter. Each Numeric_parameter defines the tip_angle for zero, one, or many Conical_hole_bottom objects.

Each Conical_hole_bottom has the tip_radius defined by zero or one Numeric_parameter. Each Numeric_parameter defines the tip_radius for zero, one, or many Conical_hole_bottom objects.

4.3.40 Constant_radius_edge_round to Numeric_parameter

Each Constant_radius_edge_round has the first_face_offset defined by zero or one Numeric_parameter. Each Numeric_parameter defines the first_face_offset for zero, one, or many Constant_radius_edge_-round objects.

Each Constant_radius_edge_round has the second_face_offset defined by zero or one Numeric_-parameter. Each Numeric_parameter defines the second_face_offset for zero, one, or many Constant_-radius_edge_round objects.

Each Constant_radius_edge_round has the radius defined by exactly one Numeric_parameter. Each Numeric_parameter defines the radius for zero, one, or many Constant_radius_edge_round objects.

4.3.41 Constant_radius_fillet to Numeric_parameter

Each Constant_radius_fillet has the first_face_offset defined by zero or one Numeric_parameter. Each Numeric_parameter defines the first_face_offset for zero, one, or many Constant_radius_fillet objects.

Each Constant_radius_fillet has the second_face_offset defined by zero or one Numeric_parameter. Each Numeric_parameter defines the second_face_offset for zero, one, or many Constant_radius_fillet objects.
Each Constant_radius_fillet has the radius defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the radius for zero, one, or many Constant_radius_fillet objects.

4.3.42 Counterbore_hole to Round_hole

Each Counterbore_hole has the smaller_hole defined by exactly one Round_hole. Each Round_hole defines the smaller_hole for zero, one, or many Counterbore_hole objects.

©ISO 2000 - All rights reserved

Each Counterbore_hole has the larger_hole defined by exactly one Round_hole. Each Round_hole defines the larger_hole for zero, one, or many Counterbore_hole objects.

4.3.43 Countersunk_hole to Round_hole

Each Countersunk_hole has the constant_diameter_hole defined by exactly one Round_hole. Each Round_hole defines the constant_diameter_hole for zero, one, or many Countersunk_hole objects.

Each Countersunk_hole has the tapered_hole defined by exactly one Round_hole. Each Round_hole defines the taper hole for zero, one, or many Countersunk_hole objects.

4.3.44 Customer_order to Ordered_part

Each Customer_order has the quantity_ordered defined by one or many Ordered_part objects. Each Ordered_part defines the quantity_ordered for zero or one Customer_order.

4.3.45 Customer_order to Person_in_organization

Each Customer_order has the customer defined by exactly one Person_in_organization. Each Person_-in_organization defines the customer for zero, one, or many Customer_order objects.

4.3.46 Customer_order to Project_order

Each Customer_order has the initiated_order defined by one or many Project_order objects. Each Project_order defines the initiated_order for zero or one Customer_order.

4.3.47 Cutout to Through_pocket_bottom_condition

Each Cutout has the bottom_condition defined by exactly one Through_pocket_bottom_condition. Each Through_pocket_bottom_condition defines the bottom_condition for zero, one, or many Cutout objects.

4.3.48 Cylindrical_base_shape to Numeric_parameter

Each Cylindrical_base_shape has the diameter defined by exactly one Numeric_parameter. Each Numeric_parameter defines the diameter for zero, one, or many Cylindrical_base_shape objects.

4.3.49 Datum_feature to Material_condition_modifier

Each Datum has the modifier defined by zero or one Material_condition_modifier. Each Material_-condition_modifier defines the modifier for zero, one, or many Datum objects.

4.3.50 Datum_feature to Datum_target_set

Each Datum_feature has the datum_representation defined by exactly one Datum_target_set. Each Datum_target_set defines the datum_representation for zero, one, or many Datum_feature objects.

NOTE - This assertion is established through Datum_representation_select.

 ©ISO 2000 - All rights reserved

4.3.51 Datum_feature to Shape_element

Each Datum_feature has the datum_representation defined by exactly one Shape_element. Each Shape_-element defines the datum_representation for zero, one, or many Datum_feature objects.

NOTE - This assertion is established through Datum_representation_select.

4.3.52 Datum_target_set to Datum_target

Each Datum_target_set has the target_shape defined by one or many Datum_target objects. Each Datum_target defines the target_shape for zero, one, or many Datum_target_set objects.

4.3.53 Defined_marking to Descriptive_parameter

Each Defined_marking has the special_instructions defined by zero or one Descriptive_parameter. Each Descriptive_parameter defines special_instructions for zero, one, or many Defined_marking objects.

Each Defined_marking has the font_name defined by zero or one Descriptive_parameter. Each Descriptive_parameter defines font_name for zero, one, or many Defined_marking objects.

4.3.54 Defined_marking to Numeric_parameter

Each Defined_marking has the character_height defined by zero or one Numeric_parameter. Each Numeric_parameter defines the character_height for zero, one, or many Defined_marking objects.

Each Defined_marking has the character_spacing defined by zero or one Numeric_parameter. Each Numeric_parameter defines the character_spacing for zero, one, or many Defined_marking objects.

4.3.55 Defined_thread to Numeric_parameter

Each Defined_thread has the crest defined by zero or one Numeric_parameter. Each Numeric_parameter defines the crest for zero, one, or many Defined_thread objects.

Each Defined_thread has the minor_diameter defined by zero or one Numeric_parameter. Each Numeric_parameter defines the minor_diameter for zero, one, or many Defined_thread objects.

Each Defined_thread has the pitch_diameter defined exactly one Numeric_parameter. Each Numeric_-parameter defines the pitch_diameter for zero, one, or many Defined_thread objects.

4.3.56 Design_exception_notice to Engineering_change_proposal

Each Design_exception_notice issues zero, one, or many Engineering_change_proposal objects. Each Engineering_change_proposal formalized exactly one Design_exception_notice.

4.3.57 Design_exception_notice to Part

Each Design_exception_notice has the discrepant_part defined by one or many Part objects. Each Part defines the discrepant_part for zero, one, or many Design_exception_notice objects.

©ISO 2000 - All rights reserved

4.3.58 Diagonal_knurl to Descriptive_parameter

Each Diagonal_knurl has the helix_hand defined by exactly one Descriptive_parameter. Each Descriptive_parameter defines the helix_hand for zero, one, or many Diagonal_knurl objects.

4.3.59 Diagonal_knurl to Numeric_parameter

Each Diagonal_knurl has the helix_angle defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the helix_angle for zero, one, or many Diagonal_knurl objects.

4.3.60 Diameter_taper to Numeric_parameter

Each Diameter_taper has the final_diameter defined by exactly one Numeric_parameter. Each Numeric_parameter defines the final_diameter for zero, one, or many Diameter_taper objects.

4.3.61 Diamond_knurl to Numeric_parameter

Each Diamond_knurl has the helix_angle defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the helix_angle for zero, one, or many Diamond_knurl objects.

4.3.62 Dimensional_tolerance to Tolerance_value

Each Dimensional_tolerance has the limit defined by zero or one Tolerance_value. Each Tolerance_-value defines the limit for zero, one, or many Dimensional_tolerance objects.

4.3.63 Directed_taper to Direction_element

Each Directed_taper has the direction defined by exactly one Direction_element. Each Direction_-element defines the direction for zero, one, or many Directed_taper objects.

4.3.64 Directed_taper to Numeric_parameter

Each Directed_taper has the angle defined by exactly one Numeric_parameter. Each Numeric_parameter defines the angle for zero, one, or many Directed_taper objects.

4.3.65 Distance_along_curve_tolerance to Shape_aspect

Each Distance_along_curve_tolerance_tolerance has a path defined by exactly one Shape_aspect. Each Shape_aspect defines the path for zero, one, or many Distance_along_curve_tolerance objects.

4.3.66 Edge_round to Face_shape_element

Each Edge_round has the edge_round_face defined by exactly one Face_shape_element. Each Face_-shape_element defines the edge_round_face for zero, one, or many Edge_round objects.

Each Edge_round has the first_face_shape defined by exactly one Face_shape_element. Each Face_-shape_element defines the first_face_shape for zero, one, or many Edge_round objects.

 ©ISO 2000 - All rights reserved

Each Edge_round has the second_face_shape defined by exactly one Face_shape_element. Each Face_-shape_element defines the second_face_shape for zero, one, or many Edge_round objects.

4.3.67 Engineering_change_order to Part

Each Engineering_change_order has the new_version defined by one or many Part objects. Each Part defines the new_version_changes by zero or one Engineering_change_order.

4.3.68 Engineering_change_proposal to Engineering_change_order

Each Engineering_change_proposal has the incorporated_proposal defined by one or more Engineering_-change_order objects. Each Engineering_change_order defines the approved_proposal for exactly one Engineering_change_proposal.

4.3.69 Explicit_base_shape_representation to B-rep_shape_representation

Each Explicit_base_shape_representation has the B-rep_form defined by exactly one B-rep_shape_-representation. Each B-rep_shape_representation defines the B-rep_form for zero, one, or many Explicit_base_shape_representation objects.

4.3.70 Face_shape_element_relationship to Face_shape_element

Each Face_shape_element_relationship has the predecessor defined by exactly one Face_shape_element. Each Face_shape_element defines the predecessor for zero, one, or many Face_shape_element_relationship objects.

Each Face_shape_element_relationship has the successor defined by exactly one Face_shape_element. Each Face_shape_element defines the successor for zero, one, or many Face_shape_element_relationship objects.

4.3.71 Fillet to Face_shape_element

Each Fillet has the fillet_face defined by exactly one Face_shape_element. Each Face_shape_element defines the fillet_face for zero, one, or many Fillet objects.

Each Fillet has the first_face_shape defined by exactly one Face_shape_element. Each Face_shape_-element defines the first_face_shape for zero, one, or many Fillet objects.

Each Fillet has the second_face_shape defined by exactly one Face_shape_element. Each Face_shape_-element defines the second_face_shape for zero, one, or many Fillet objects.

4.3.72 First_offset to Face_shape_element

Each First_offset has the face_shape defined by exactly one Face_shape_element. Each Face_shape_-element defines the face_shape for zero, one, or many First_offset objects.

©ISO 2000 - All rights reserved

4.3.73 First_offset to Numeric_parameter

Each First_offset has the offset_amount defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the offset_amount for zero, one, or many First_offset objects.

4.3.74 Flat_slot_end_type to Numeric_parameter

Each Flat_slot_end_type has the first_radius defined by exactly one Numeric_parameter. Each Numeric_parameter defines the first_radius for zero, one, or many Flat_slot_end_type objects.

Each Flat_slot_end_type has the second_radius defined by exactly one Numeric_parameter. Each Numeric_parameter defines the second_radius for zero, one, or many Flat_slot_end_type objects.

4.3.75 Flat_with_radius_hole_bottom to Numeric_parameter

Each Flat_with_radius_hole_bottom has the corner_radius defined by exactly one Numeric_parameter. Each Numeric_parameter defines the corner_radius for zero, one, or many Flat_with_radius_hole_bottom objects.

4.3.76 Flat_with_taper_hole_bottom to Numeric_parameter

Each Flat_with_taper_hole_bottom has the taper_diameter defined by exactly one Numeric_parameter. Each Numeric_parameter defines the taper_diameter for zero, one, or many Flat_with_taper_hole_bottom objects.

Each Flat_with_taper_hole_bottom has the final_diameter defined by exactly one Numeric_parameter. Each Numeric_parameter defines the final_diameter for zero, one, or many Flat_with_taper_hole_bottom objects.

4.3.77 General_boss to Closed_profile

Each General_boss has enclosed_boundary defined by exactly one Closed_profile. Each Closed_profile defines the enclosed_boundary for zero, one, or many General_boss objects.

4.3.78 General_boss to Angle_taper

Each General_boss has the change_in_boundary defined by zero or one Angle_taper. Each Angle_taper defines the change_in_boundary for zero, one, or many General_boss objects.

NOTE - This assertion is established through Angle_or_directed_taper_select.

4.3.79 General_boss to Directed_taper

Each General_boss has the change_in_boundary defined by zero or one Directed_taper. Each Directed_taper defines the change_in_boundary for zero, one, or many General_boss objects.

NOTE - This assertion is established through Angle_or_directed_taper_select.

 ©ISO 2000 - All rights reserved

4.3.80 General_closed_profile to Path_element

Each General_closed_profile has the closed_profile_shape defined by exactly one Path_element. Each Path_element defines the closed_profile_shape for zero, one, or many General_closed_profile objects.

4.3.81 General_cutout to Profile

Each General_cutout has the boundary defined by exactly one Profile. Each Profile defines the boundary for zero, one, or many General_cutout objects.

4.3.82 General_open_profile to Path_element

Each General_open_profile has enclosed_boundary defined by exactly one Path_element. Each Path_-element identifies the enclosed_boundary for zero, one, or many General_open_profile objects.

4.3.83 General_outside_profile to Profile

Each General_outside_profile has boundary defined by exactly one Profile. Each Profile defines outside boundary for zero, one, or many General_outside_profile objects.

4.3.84 General_path to Path_element

Each General_path has the sweep_path defined by exactly one Path_element. Each Path_element defines the sweep_path for zero, one, or many General_path objects.

4.3.85 General_pattern to Orientation

Each General_pattern has the feature_placement defined by one or more Orientation. Each Orientation defines the feature_placement for zero, one, or many General_pattern objects.

4.3.86 General_pocket to Profile

Each General_pocket has the boundary defined by exactly one Profile. Each Profile defines the boundary for zero, one, or many General_pocket objects.

4.3.87 General_pocket_bottom_condition to Face_shape_element

Each General_pocket_bottom_condition has the floor defined by exactly one Face_shape_element. Each Face_shape_element defines the floor for zero, one, or many General_pocket_bottom_condition objects.

4.3.88 General_pocket_bottom_condition to Numeric_parameter

Each General_pocket_bottom_condition has the floor_radius defined by zero or one Numeric_parameter. Each Numeric_parameter defines the floor_radius for zero, one, or many General_pocket_bottom_-condition objects.

©ISO 2000 - All rights reserved

4.3.89 General_profile_floor to Face_shape_element

Each General_profile_floor has the floor defined by exactly one Face_shape_element. Each Face_-shape_element defines the floor for zero, one, or many General_profile_floor objects.

4.3.90 General_removal_volume to Shape_element

Each General_removal_volume has the removal_volume defined by one or more Shape_element objects. Each Shape_element defines the removal_volume for zero, one, or many General_removal_volume objects.

4.3.91 General_revolution to General_open_profile

Each General_revolution has the outer_edge_shape defined by exactly one General_open_profile. Each General_open_profile defines outer_edge_shape for zero, one, or many General_revolution objects.

4.3.92 General_rib_top_floor to Face_shape_element

Each General_rib_top_floor has the rip_top_face defined by one or more Ordered_face_element objects. Each Ordered_face_element defines the rib_top_floor for zero, one, or many General_rib_top_floor objects.

4.3.93 General_shape_profile to Profile

Each General_shape_profile has the profile_boundary defined by exactly one Profile. Each Profile defines the profile_boundary for zero, one, or many General_shape_profile objects.

4.3.94 General_top_condition to Face_shape_element

Each General_top_condition has the top_face defined by exactly one Face_shape_element. Each Face_-shape_element defines the top_face for zero, one, or many General_top_condition objects.

4.3.95 Geometric_tolerance to Material_condition_modifier

Each Geometric_tolerance has the modifier_control defined by zero or one Material_condition_modifier. Each Material_condition_modifier defines the modifier_control for zero, one, or many Geometric_-tolerance objects.

4.3.96 Geometric_tolerance to Shape_aspect

Each Geometric_tolerance has the applied_shape defined by one or more Shape_aspect objects. Each Shape_aspect defines the applied_shape for zero, one, or many Geometric_tolerance objects.

4.3.97 Geometric_tolerance to Tolerance_zone

Each Geometric_tolerance has the zone_definition defined by zero or one Tolerance_zone. Each Tolerance_zone defines the zone_definition for zero, one, or many Geometric_tolerance objects.

 ©ISO 2000 - All rights reserved

4.3.98 Geometric_tolerance_precedence_relationship to Geometric_tolerance

Each Geometric_tolerance_precedence_relationship has the base_shape_tolerance defined by exactly one Geometric_tolerance. Each Geometric_tolerance defines the base_shape_tolerance for zero, one, or many Geometric_tolerance_precedence_relationship objects.

Each Geometric_tolerance_precedence_relationship has the pattern_shape_tolerance defined by exactly one Geometric_tolerance. Each Geometric_tolerance defines the pattern_shape_tolerance for zero, one, or many Geometric_tolerance_precedence_relationship objects.

4.3.99 Groove to Open_profile

Each Groove has a sweep defined by exactly one Open_profile. Each Open_profile defines the sweep for zero, one, or many Groove objects.

4.3.100 Implicit_base_shape_representation to Numeric_parameter

Each Implicit_base_shape_representation has the base_shape_length defined by exactly one Numeric_-parameter. Each Numeric_parameter defines the base_shape_length for zero, one, or many Implicit_-base_shape_representation objects.

4.3.101 Implicit_base_shape_representation to Orientation

Each Implicit_base_shape_representation has placement defined by exactly one orientation. Each Orientation defines the placement for zero, one, or many Implicit_base_shape_representation objects.

4.3.102 Knurl to Partial_area_definition

Each Knurl has the partial_profile defined by zero or one Partial_area_definition. Each Partial_area_-definition defines the partial_profile for zero, one, or many Knurl objects.

4.3.103 Knurl to Shape_element

Each Knurl has the applied_shape defined by exactly one Shape_element. Each Shape_element defines the applied_shape for zero, one, or many Knurl objects.

4.3.104 Linear_path to Numeric_parameter

Each Linear_path has the distance defined by exactly one Numeric_parameter. Each Numeric_parameter defines the distance for zero, one, or many Linear_path objects.

4.3.105 Linear_path to Direction_element

Each Linear_path has the direction defined by exactly one Direction_element. Each Direction_element defines the direction for zero, one, or many Linear_path objects.

©ISO 2000 - All rights reserved

4.3.106 Linear_profile to Numeric_parameter

Each Linear_profile has the profile_length defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the profile_length for zero, one, or many Linear_profile objects.

4.3.107 Linear_profile_tolerance to Datum

Each Linear_profile_tolerance has the geometric_reference defined by zero, one, two, or three Datum objects. Each Datum defines the geometric_reference for zero, one, or many Linear_profile_tolerance objects.

4.3.108 Linear_profile_tolerance to Orientation

Each Linear_profile_tolerance has an affected_plane defined by zero or one Orientation. Each Orientation defines an affected plane for zero, one, or many Linear_profile_tolerance objects.

4.3.109 Location_dimension_tolerance to Orientation

Each Location_dimension_tolerance has the plane and direction defined by zero or one Shape_aspect. Each Orientation defines the plane and direction for zero, one, or many Location_dimension_tolerance objects.

4.3.110 Location_tolerance to Shape_element

Each Location_tolerance has the termination_shape defined by exactly one Shape_element. Each Shape_element defines the termination_shape for zero, one, or many Location_dimension objects.

Each Location_dimension has the origin_shape defined by exactly one Shape_element. Each Shape_-element defines the origin_shape for zero, one, or many Location_dimension objects.

4.3.111 Machining_feature to Orientation

Each Machining_feature has placement defined by exactly one Orientation. Each Orientation defines the placement for zero, one, or many Machining_feature objects.

4.3.112 Manufacured_assembly_relationship to Manufactured_assembly

Each Manufactured_assembly_relationship has the assembly defined by exactly one Manufactured_-assembly. Each Manufactured_assembly has components defined by two or more Manufactured_-assembly_relationship objects.

4.3.113 Manufactured_assembly_relationship to Part

Each Manufactured_assembly_relationship has the component defined by exactly one Part. Each Part defines the component for zero, one, or many Manufactured_assembly_relationship objects.

 ©ISO 2000 - All rights reserved

4.3.114 Manufactured_assembly_relationship to Part_placement

Each Manufactured_assembly_relationship has the orientation defined by exactly one Part_placement. Each Part_placement defines the orientation for zero, one, or many Manufactured_assembly_relationship objects.

4.3.115 Manufacturing_feature_group to Manufacturing_feature

Each Manufacturing_feature_group has the feature_groups defined by two or more Manufacturing_feature objects. Each Manufacturing_feature defines the feature_groups for zero, one, or many Manufacturing_-feature_group objects.

NOTE - This assertion is established through Manufacturing_group_select.

4.3.116 Manufacturing_feature_group to Manufacturing_feature_group

Each Manufacturing_feature_group has the feature_groups defined by two or more Manufacturing_-feature_group objects. Each Manufacturing_feature_group defines the feature_groups for zero, one, or many Manufacturing_feature_group objects.

NOTE - This assertion is established through Manufacturing_group_select.

4.3.117 Marking to Descriptive_parameter

Each Marking has text defined by exactly one Descriptive_parameter. Each Descriptive_parameter defines text for zero, one, or many Marking objects.

4.3.118 Marking to Shape_element

Each Marking has an applied_to_shape defined by exactly one Shape_element. Each Shape_element defines the applied_to_shape for zero, one, or many Marking objects.

4.3.119 Material to Material_property

Each Material has the material_characteristics defined by zero, one, or many Material_property objects. Each Material_property defines the material_characteristics for exactly one Material.

4.3.120 Material to Specification

Each Material has the material_specification defined by zero, one, or many Specification objects. A Specification defines the material_specification for exactly one Material.

4.3.121 Material_property to Hardness

Each Material_property has the material_hardness defined by zero, one, or many Hardness objects. A Hardness defines material_hardness for exactly one Material_property.

©ISO 2000 - All rights reserved

4.3.122 Material_property to Property_parameter

Each Material_property has the property_characteristic defined by zero, one, or many Property_-parameter objects. Each Property_parameters defines the property_characteristic for exactly one Material_property.

4.3.123 Mating_definition to Manufactured_assembly

Each Mating_definition has the applied_assembly defined by exactly one Manufactured_assembly. Each Manufactured_assembly defines the applied_assembly for zero, one, or many Mating_definition objects.

4.3.124 Mating_definition to Shape_element

Each Mating_definition has the mating_shape defined by zero, two, or more Shape_element objects. Each Shape_element object defines the mating_shape for zero, one, or many Mating_definition objects.

4.3.125 Mating_definition to Single_piece_part

Each Mating_definition has the mating_solution defined by zero, one, or many Single_piece_part objects. Each Single_piece_part object defines the mating_solution for zero, one, or many Mating_definition objects.

4.3.126 Mating_definition_relationship to Mating_definition

Each Mating_definition_relationship has the mating_part_definiton defined by exactly one Mating_-definition. Each Mating_definition object defines the mating_part_definition for two, or more Mating_-definition_relationship objects.

4.3.127 Mating_definition_relationship to Part_placement

Each Mating_definition_relationship has the orientation defined by zero or one Part_placement object. Each Part_placement object defines the orientation for zero, one, or many Mating_definition_relationship objects.

4.3.128 Mating_definition_relationship to Single_piece_part

Each Mating_definition_relationship has the mated_part defined by exactly one Single_piece_part. Each Single_piece_part object defines the mated_part for zero, one, or many Mating_definition_relationship objects.

4.3.129 Mating_relationship to Single_piece_part

Each Mating_relationship has the predecessor defined by exactly one single_piece_part. Each Single_piece_part object defines the predecessor for zero, one, or many Mating_relationship objects.

Each Mating_relationship has the successor defined by exactly one single_piece_part. Each Single_piece_part object defines the successor for zero, one, or many Mating_relationship objects.

 ©ISO 2000 - All rights reserved

4.3.130 Multi_axis_feature to Planar_element

Each Multi_axis_feature has maximum_feature_limit defined by zero or one Planar_element. Each Planar_element defines the maximum_feature_limit for zero, one, or many Multi_axis_feature objects.

4.3.131 Ngon_base_shape to Numeric_parameter

Each Ngon_base_shape has the number_of_sides defined by exactly one Numeric_parameter. Each Numeric_parameter defines the number_of_sides for zero, one, or many Ngon_base_shape objects.

Each Ngon_base_shape has the diameter defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the diameter for zero, one, or many Ngon_base_shape objects.

Each Ngon_base_shape has the corner_radius defined by exactly one Numeric_parameter. Each Numeric_parameter defines the corner_radius for zero, one, or many Ngon_base_shape objects.

4.3.132 Ngon_profile to Numeric_parameter

Each Ngon_profile has the number_of_sides defined by exactly one Numeric_parameter. Each Numeric_parameter defines the number_of_sides for zero, one, or many Ngon_profile objects.

Each Ngon_profile has the diameter defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the diameter for zero, one, or many Ngon_profile objects.

Each Ngon_profile has the corner_radius defined by zero or one Numeric_parameter. Each Numeric_-parameter defines the corner_radius for zero, one, or many Ngon_profile objects.

4.3.133 Numeric_parameter_with_tolerance to Plus_minus_value

Each Numeric_parameter_with_tolerance has the implicit_tolerance defined by exactly one Plus_-minus_value. Each Plus_minus_value defines the implicit_tolerance for zero, one, or many Numeric_-parameter_with_tolerance objects.

NOTE - This assertion is established through the Numeric_parameter_tolerance_select.

4.3.134 Numeric_parameter_with_tolerance to Tolerance_limit

Each Numeric_parameter_with_tolerance has the implicit_tolerance defined by exactly one Tolerance_-limit. Each Tolerance_limit defines the implicit_tolerance for zero, one, or many Numeric_parameter_-with_tolerance objects.

NOTE - This assertion is established through the Numeric_parameter_tolerance_select.

4.3.135 Open_profile to Planar_element

Each Open_profile has the profile_limit defined by zero or one Planar_element. Each Planar_element defines the profile_limit for zero, one, or many Open_profile objects.

©ISO 2000 - All rights reserved

4.3.136 Outer_diameter to Numeric_parameter

Each Outer_diameter has the diameter defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the diameter for zero, one, or many Outer_diameter objects.

Each Outer_diameter has the feature_length defined by exactly one Numeric_parameter. Each Numeric_parameter defines the feature_length for zero, one, or many Outer_diameter objects.

4.3.137 Outer_diameter to Angle_taper

Each Outer_diameter has the reduced_size defined by zero or one Angle_taper. Each Angle_taper defines the reduced_size for zero, one, or many Outer_diameter objects.

NOTE - This assertion is established through Taper_select.

4.3.138 Outer_diameter to Diameter_taper

Each Outer_diameter has the reduced_size defined by zero or one Diameter_taper. Each Diameter_taper defines the reduced_size for zero, one, or many Outer_diameter objects.

NOTE - This assertion is established through Taper_select.

4.3.139 Outer_diameter to Directed_taper

Each Outer_diameter has the reduced_size defined by zero or one Directed_taper. Each Directed_taper defines the reduced_size for zero, one, or many Outer_diameter objects.

NOTE - This assertion is established through Taper_select.

4.3.140 Outer_diameter_to_shoulder to Numeric_parameter

Each Outer_diameter_to_shoulder has the diameter defined by exactly one Numeric_parameter. Each Numeric_parameter defines the diameter for zero, one, or many Outer_diameter_to_shoulder objects.

4.3.141 Outer_diameter_to_shoulder to Vee_profile

Each Outer_diameter_to_shoulder has the v_shape_boundary defined by exactly one Vee_profile. Each Vee_profile defines the v_shape_boundary for zero, one, or many Outer_diameter_to_shoulder objects.

4.3.142 Parallelism_tolerance to Datum

Each Parallelism_tolerance has the geometric_reference defined by exactly one Datum. Each Datum defines the geometric_reference for zero, one, or many Parallelism_tolerance objects.

4.3.143 Parallelism_tolerance to Orientation

Each Parallelism_tolerance has an affected_plane defined by zero or one Orientation. Each Orientation defines the affected_plane for zero, one, or many Parallelism_tolerance objects.

 ©ISO 2000 - All rights reserved

4.3.144 Part to Approval

Each Part has the manufacture_authorization defined by zero, one, or many Approval objects. Each Approval defines the manufacture_authorization for zero, one, or many Part objects.

4.3.145 Part to Ordered_part

Each Part has the quantity_ordered defined by one or more Ordered_part objects. Each Ordered_part defines quantity_ordered for exactly one Part.

4.3.146 Part to Organization

Each Part has the manufactured_by_organization defined by one or more Organization objects. Each Organization defines the manufactured_by_organization for zero, one or many Part objects.

Each Part has the owned_by_organization defined by one or more Organization objects. Each Organization defines the owned_by_organization for zero, one, or many Part objects.

4.3.147 Part to Person_in_organization

Each Part has the manufactured_by_person defined by one or more Person_in_organization objects. Each Person_in_organization defines the manufactured_by_person for zero, one, or many Part objects.

Each Part has the owned_by_person defined by one or more Person_in_organization objects. Each Person_in_organization defines the owned_by_person for zero, one, or many Part objects.

4.3.148 Part to Property

Each Part has the property_characteristics defined by zero, one, or many Property objects. Each Property defines the applied_to for exactly one Part.

4.3.149 Part to Shape

Each Part has the physical_form defined by exactly one Shape. Each Shape defines the physical_form for zero, one, or many Part objects.

4.3.150 Part_placement to Orientation

Each Part_placement has the resulting_orientation defined by exactly one Orientation. Each Orientation defines the resulting_orientation for zero, one, or many Part_placement objects.

Each Part_placement has the originating_orientation defined by exactly one Orientation. Each Orientation defines the originating_orientation for zero, one, or many Part_placement objects.

©ISO 2000 - All rights reserved

4.3.151 Part_placement to Shape

Each Part_placement has the oriented_physical_form defined by exactly one Shape. Each Shape defines the oriented_physical_form for zero, one, or many Part_placement objects.

4.3.152 Part_property to Property_parameter

Each Part_property has the property_characteristic defined by zero, one, or many Property_parameter objects. Each Property_parameter defines the property_characteristic for exactly one Part_property.

4.3.153 Partial_area_definition to Numeric_parameter

Each Partial_area_definition has the effective_length defined by exactly one Numeric_parameter. Each Numeric_parameter defines the effective_length for zero, one, or many Partial_area_definition objects.

Each Partial_area_definition has the maximum_length defined by zero or one Numeric_parameter. Each Numeric_parameter defines the maximum_length for zero, one, or many Partial_area_definition objects.

4.3.154 Partial_area_definition to Orientation

Each Partial_area_definition has the placement defined by exactly one Orientation. Each Orientation defines the placement for zero, one, or many Partial_area_definition objects.

4.3.155 Partial_circular_path to Numeric_parameter

Each Partial_circular_path has the sweep_angle defined by exactly one Numeric_parameter. Each Numeric_parameter defines the sweep_angle for zero, one, or many Partial_circular_path objects.

4.3.156 Partial_circular_profile to Numeric_parameter

Each Partial_circular_profile has the radius defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the radius for zero, one, or many Partial_circular_profile objects.

Each Partial_circular_profile has the sweep_angle defined by exactly one Numeric_parameter. Each Numeric_parameter defines the sweep_angle for zero, one, or many Partial_circular_profile objects.

4.3.157 Partial_circular_shape_profile to Partial_circular_profile

Each Partial_circular_shape_profile has the open_boundary defined by exactly one Partial_circular_-profile. Each Partial_circular_profile defines the open_boundary for zero, one, or many Partial_-circular_-shape_profile objects.

4.3.158 Path to Orientation

Each Circular_path has the placement defined by exactly one Orientation. Each Orientation defines the placement for zero, one, or many Circular_path objects.

 ©ISO 2000 - All rights reserved

4.3.159 Perpendicularity_tolerance to Datum

Each Perpendicularity_tolerance has the geometric_reference defined by exactly one Datum. Each Datum defines the geometric_reference for zero, one, or many Perpendicularity_tolerance objects.

4.3.160 Perpendicularity_tolerance to Orientation

Each Perpendicularity_tolerance has the affected_plane defined by zero or one Orientation. Each Orientation defines the affected_plane for zero, one, or many Perpendicularity_tolerance objects.

4.3.161 Person_in_organization to Organization

Each Person_in_organization has the company defined by exactly one Organization. Each Organization defines the company for zero, one, or many Person_in_organization objects.

4.3.162 Person_in_organization to Person

Each Person_in_organization has the employee defined by exactly one Person. Each Person defines the employee for zero, one, or many Person_in_organization objects.

4.3.163 Placed_target to Orientation

Each Placed_target has the placement defined by exactly one Orientation. Each Orientation defines placement for zero, one, or many Placed_target objects.

4.3.164 Planar_element to Direction_element

Each Planar_element has the normal defined by exactly one Direction_element. Each Direction_element defines the normal for zero, one, or many Planar_element objects.

4.3.165 Planar_element to Location_element

Each Planar_element has the location defined by exactly one Location_element. Each Location_element defines the location for zero, one, or many Planar_element objects.

4.3.166 Planar_face to Closed_profile

Each Planar_face has the face_boundary defined by zero or one Closed_profile object. Each Closed_profile defines the face_boundary for zero, one, or many Planar_face objects.

4.3.167 Planar_face to Direction_element

Each Planar_face has the removal_direction defined by exactly one Direction_element. Each Direction_-element defines the removal_direction for zero, one, or many Planar_face objects.

©ISO 2000 - All rights reserved

4.3.168 Planar_face to Linear_path

Each Planar_face has the course_of_travel defined by exactly one Linear_path. Each Linear_path defines the course_of_travel for zero, one, or many Planar_face objects.

4.3.169 Planar_face to Linear_profile

Each Planar_face has the removal_boundary defined by exactly one Linear_profile. Each Linear_profile defines the removal_boundary for zero, one, or many Planar_face objects.

4.3.170 Planar_pocket_bottom_condition to Direction_element

Each Planar_pocket_bottom_condition has the floor_normal defined by exactly one Direction_element. Each Direction_element defines floor_normal for zero, one, or many Planar_pocket_bottom_condition objects.

4.3.171 Planar_pocket_bottom_condition to Location_element

Each Planar_pocket_bottom_condition has the floor_location defined by exactly one Location_element. Each Location_element defines the floor_location for zero, one, or many Planar_pocket_bottom_-condition objects.

4.3.172 Planar_pocket_bottom_condition to Numeric_parameter

Each Planar_pocket_bottom_condition has the floor_radius defined by zero or one Numeric_parameter. Each Numeric_parameter defines the floor_radius for zero, one, or many Planar_pocket_bottom_-condition objects.

4.3.173 Planar_profile_floor to Planar_element

Each Planar_profile_floor has the floor defined by exactly one Planar_element. Each Planar_element defines the floor for zero, one, or many Planar_profile_floor objects.

4.3.174 Planar_rib_top_floor to Closed_profile

Each Planar_rib_top_floor has the boundary defined by exactly one Closed_profile. Each Closed_profile defines the boundary for zero, one, or many Planar_rib_top_floor elements.

4.3.175 Planar_rib_top_floor to Planar_element

Each Planar_rib_top_floor has the floor_face defined by exactly one Planar_element. Each Planar_element defines the floor_face for zero, one, or many Planar_rib_top_floor elements.

4.3.176 Planar_top_condition to Direction_element

Each Planar_top_condition has the top_normal defined by exactly one Direction_element. Each Direction_element defines the top_normal direction for zero, one, or many Planar_top_condition objects.

 ©ISO 2000 - All rights reserved

4.3.177 Planar_top_condition to Location_element

Each Planar_top_condition has the top_location defined by exactly one Location_element. Each Location_element defines the top_location for zero, one, or many Planar_top_condition objects.

4.3.178 Pocket to Angle_taper

Each Pocket has the change_in_boundary defined by zero or one Angle_taper. Each Angle_taper defines the change_in_boundary for zero, one, or many Pocket objects.

NOTE - This assertion is established through Angle_or_directed_taper_select.

4.3.179 Pocket to Directed_taper

Each Pocket has the change_in_boundary defined by zero or one Directed_taper. Each Directed_taper defines the change_in_boundary for zero, one, or many Pocket objects.

NOTE - This assertion is established through Angle_or_directed_taper_select.

4.3.180 Pocket to Linear_path

Each Pocket has the pocket_depth defined by exactly one Linear_path. Each Linear_path defines the pocket_depth for zero, one, or many Pocket objects.

4.3.181 Pocket to Numeric_parameter

Each Pocket has the base_radius defined by zero or one Numeric_parameter. Each Numeric_parameter defines the base_radius for zero, one, or many Pocket objects.

4.3.182 Pocket to Pocket_bottom_condition

Each Pocket has the bottom_condition defined by exactly one Pocket_bottom_condition. Each Pocket_-bottom_condition defines the bottom_condition for zero, one, or many Pocket objects.

4.3.183 Pocket to Through_pocket_bottom_condition

Each Pocket has the bottom_condition defined by exactly one Through_pocket_bottom_condition. Each Through_pocket_bottom_condition defines the bottom_condition for zero, one, or many Pocket objects.

4.3.184 Position_tolerance to Datum

Each Position_tolerance has the geometric_reference defined by one, two, or three Datum objects. Each Datum defines the geometric_reference for zero, one, or many Position_tolerance objects.

©ISO 2000 - All rights reserved

4.3.185 Position_tolerance to Orientation

Each Position_tolerance has the affected_plane defined by zero or one Orientation. Each Orientation defines the affected_plane for zero, one, or many Position_tolerance objects.

4.3.186 Process_property to Property_parameter

Each Process_property has the property_characteristic defined by zero, one, or many Property_-parameters objects. Each Property_parameter defines the property characteristic for exactly one Process_property.

4.3.187 Profile to Orientation

Each Profile has the placement defined by exactly one orientation. Each Orientation defines the placement for zero, one, or many Profile objects.

4.3.188 Profile_feature to Linear_path

Each Profile_feature has the profile_swept_shape defined by exactly one Linear_path. Each Linear_path defines the profile_swept_shape for zero, one, or many Profile_feature objects.

4.3.189 Profile_floor to Numeric_parameter

Each Profile_floor has the floor_radius defined by zero or one Numeric_parameter. Each Numeric_-parameter defines the floor_radius for zero, one, or many Planar_floor objects.

4.3.190 Project_order to Approval

Each Project_Order has the release_authorization defined by exactly one Approval. Each Approval defines the release_authorization for zero or one Project_Order.

4.3.191 Project_order to Digital_technical_data_package_work_order

Each Project_order has the technical_data_package_status defined by zero or one Digital_technical_-data_package_work_order objects. Each Digital_technical_data_package_work_order defines the technical_data_package_status for exactly one Project_order.

4.3.192 Project_order to Part

Each Project_order has the part_status defined by one or more Part objects. Each Part defines the part_status for exactly one Project_order.

4.3.193 Project_order to Pedigree_creation_order

Each Project_order has the pedigree_creation_status of zero or one Pedigree_creation_order. Each Pedigree_creation_order defines the pedigree_creation_status for exactly one Project_order.

 ©ISO 2000 - All rights reserved

4.3.194 Project_order to Requisition

Each Project_order has the ordered_resource defined by zero, one, or many Requisition objects. Each Requisition defines the ordered_resource for exactly one Project_order.

4.3.195 Project_order to Resource_acquisition_order

Each Project_order has the resource_acquisition_status defined by zero or one Resource_acquisition_-order. Each Resource_acquisition_order defines the resource_acquisition_order for exactly one Project_-order.

4.3.196 Project_order to Shop_work_order

Each Project_order has the shop_work_status defined by of zero or one Shop_work_order. Each Shop_-work_order defines the shop_work_status for exactly one Project_order.

4.3.197 Projection to Shape_element

Each Projection has the projection_end defined by exactly one Shape_element. Each Shape_element defines the projection_end for zero, one, or many Projection objects.

4.3.198 Property to Material_property

Each Property has the material_characteristic defined by zero, one, or many Material_property objects. Each Material_property defines the material_characteristic for one or more Property objects.

4.3.199 Property to Part_property

Each Property has the part_property_characteristic defined by zero, one, or many Part_property objects. Each Part_property defines the part_property_characteristic for one or more Property objects.

4.3.200 Property to Process_property

Each Property has the process_characteristic defined by zero, one, or many Process_property objects. Each Process_property defines the process_characteristic for one or more Property objects.

4.3.201 Property to Shape_aspect

Each Property has the property_characteristic defined by zero, one, or many Shape_aspect objects. Each Shape_aspect defines the property_characteristic for zero, one, or many Property objects.

4.3.202 Property to Specification

Each Property has the property_description defined by zero, one, or many Specification objects. Each Specification defines the property_description for exactly one Property.

©ISO 2000 - All rights reserved

4.3.203 Property to Surface_property

Each Property has the surface_characteristic defined by zero, one, or many Surface_property objects. Each Surface_property defines the surface_characteristic for one or more Property objects.

4.3.204 Protrusion to Shape_element

Each Protrusion has the shape_volume defined by one or more Shape_element objects. Each Shape_-element defines the shape_volume for zero, one, or many Protrusion objects.

4.3.205 Recess to Profile

Each Recess has the fillet_boundary defined by exactly one Profile. Each Profile defines the fillet_boundary for zero, one, or many Recess objects.

4.3.206 Recess to Pocket_bottom_condition

Each Recess has the bottom_condition defined by exactly one Pocket_bottom_condition. Each Pocket_bottom_condition defines the bottom_condition for zero, one, or many Recess objects.

4.3.207 Rectangular_boss to Angle_taper

Each Rectangular_boss has the change_in_boundary defined by zero or one Angle_taper. Each Angle_taper defines the change_in_boundary for zero, one, or many Rectangular_boss objects.

NOTE - This assertion is established through Angle_or_directed_taper_select.

4.3.208 Rectangular_boss to Directed_taper

Each Rectangular_boss t has the change_in_boundary defined by zero or one Directed_taper. Each Directed_taper defines the change_in_boundary for zero, one, or many Rectangular_boss objects.

NOTE - This assertion is established through Angle_or_directed_taper_select.

4.3.209 Rectangular_boss to Rectangular_closed_profile

Each Rectangular_boss has the rectangular_profile defined by exactly one Rectangular_closed_profile. Each Rectangular_closed_profile defines the rectangular_profile for zero, one, or many Rectangular_boss objects.

4.3.210 Rectangular_closed_pocket to Rectangular_closed_profile

Each Rectangular_closed_pocket has a closed_boundary defined by exactly one Rectangular_closed_-profile. Each Rectangular_closed_profile identifies a closed_boundary for zero, one, or many Rectangular_closed_pocket objects.

 ©ISO 2000 - All rights reserved

4.3.211 Rectangular_closed_profile to Numeric_parameter

Each Rectangular_closed_profile has the profile_width defined by exactly one Numeric_parameter. Each Numeric_parameter defines the profile_width for zero, one, or many Rectangular_closed_profile objects.

Each Rectangular_closed_profile has the profile_length defined by exactly one Numeric_parameter. Each Numeric_parameter defines the profile_length for zero, one, or many Rectangular_closed_profile objects.

Each Rectangular_closed_profile has the corner_radius defined by zero or one Numeric_parameter. Each Numeric_parameter defines the corner_radius for zero, one, or many Rectangular_closed_profile objects.

4.3.212 Rectangular_closed_shape_profile to Rectangular_closed_profile

Each Rectangular_closed_shape_profile has the closed_boundary defined by exactly one Rectangular_-closed_profile. Each Rectangular_closed_profile defines the closed_boundary for zero, one, or many Rectangular_closed_shape_profile objects.

4.3.213 Rectangular_offset_pattern to Direction_element

Each Rectangular_offset_pattern has the offset_direction defined by exactly one Direction_element. Each Direction_element defines the offset_direction for zero, one, or many Rectangular_offset_pattern objects.

4.3.214 Rectangular_offset_pattern to Numeric_parameter

Each Rectangular_offset_pattern has the column_index defined by exactly one Numeric_parameter. Each Numeric_parameter defines the column_index for zero, one, or many Rectangular_offset_pattern objects.

Each Rectangular_offset_pattern has the offset_distance defined by exactly one Numeric_parameter. Each Numeric_parameter defines the offset_distance for zero, one, or many Rectangular_offset_pattern objects.

Each Rectangular_offset_pattern has the row_index defined by exactly one Numeric_parameter. Each Numeric_parameter defines the row_index for zero, one, or many Rectangular_offset_pattern objects.

4.3.215 Rectangular_omit_pattern to Numeric_parameter

Each Rectangular_omit_pattern has the row_index defined by exactly one Numeric_parameter. Each Numeric_parameter defines the row_index for zero, one, or many Rectangular_omit_pattern objects.

Each Rectangular_omit_pattern has the column_index defined by exactly one Numeric_parameter. Each Numeric_parameter defines the column_index for zero, one, or many Rectangular_omit_pattern objects.

©ISO 2000 - All rights reserved

4.3.216 Rectangular_open_pocket to Square_U_profile

Each Rectangular_open_pocket has the open_boundary defined by exactly one Square_U__profile. Each Square_U_profile defines the open_boundary for zero, one, or many Rectangular_open_pocket objects.

4.3.217 Rectangular_open_shape_profile to Square_U_profile

Each Rectangular_open_shape_profile has the open_boundary defined by exactly one Square_U_profile. Each Square_U_profile defines the open_boundary for zero, one, or many Rectangular_open_shape_profile objects.

4.3.218 Rectangular_pattern to Direction_element

Each Rectangular_pattern has the row_layout_direction defined by exactly one Direction_element. Each Direction_element defines the row_layout_direction for zero, one, or many Rectangular_pattern objects.

Each Rectangular_pattern has the column_layout_direction defined by exactly one Direction_element. Each Direction_element defines the column_layout_direction for zero, one, or many Rectangular_pattern objects.

4.3.219 Rectangular_pattern to Numeric_parameter

Each Rectangular_pattern has the rows defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the rows for zero, one, or many Rectangular_pattern objects.

Each Rectangular_pattern has the columns defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the columns for zero, one, or many Rectangular_pattern objects.

Each Rectangular_pattern has the row_spacing defined by exactly one Numeric_parameter. Each Numeric_parameter defines the row_spacing for zero, one, or many Rectangular_pattern objects.

Each Rectangular_pattern has the column_spacing defined by exactly one Numeric_parameter. Each Numeric_parameter defines the column_spacing for zero, one, or many Rectangular_pattern objects.

4.3.220 Rectangular_pattern to Rectangular_offset_pattern

Each Rectangular_pattern has the relocated_base_feature defined by zero, one, or many Rectangular_-offset_pattern objects. Each Rectangular_offset_pattern defines the relocated_base_feature for one or more Rectangular_pattern objects.

4.3.221 Rectangular_pattern to Rectangular_omit_pattern

Each Rectangular_pattern has the missing_base_feature defined by zero, one, or many Rectangular_-omit_pattern objects. Each Rectangular_omit_pattern defines the missing_base_feature for one or more Rectangular_pattern objects.

 ©ISO 2000 - All rights reserved

4.3.222 Replicate_base to Machining_feature

Each Replicate_base has the base_feature defined by exactly one Machining_feature. Each Machining_-feature defines the base_feature for zero, one, or many Replicate_base objects.

NOTE - This assertion is established through Replicate_base_select.

4.3.223 Replicate_base to Replicate_feature

Each Replicate_base has the base_feature defined by exactly one Replicate_feature. Each Replicate_-feature defines the base_feature for zero, one, many Replicate_base objects.

NOTE - This assertion is established through Replicate_base_select.

4.3.224 Replicate_feature to Orientation

Each Replicate_feature has placement defined by exactly one Orientation. Each Orientation defines the placement for zero, one, or many Replicate_feature objects.

4.3.225 Replicate_feature to Replicate_base

Each Replicate_feature has the replicate_base_feature defined by exactly one Replicate_base. Each Replicate_base defines the replicate_base_feature for one or more Replicate_feature objects.

4.3.226 Revolved_feature to Numeric_parameter

Each Revolved_feature has the radius defined by exactly one Numeric_parameter. Each Numeric_parameter defines the radius for zero, one, or many Revolved_feature objects.

4.3.227 Revolved_feature to Direction_element

Each Revolved_feature has the material_side defined by exactly one Direction_element. Each Direction_element defines the material_side for zero, one, or many Revolved_feature objects.

4.3.228 Revolved_flat to Linear_profile

Each Revolved_flat has the flat_edge_shape defined by exactly one Linear_profile. Each Linear_profile defines the flat_edge_shape for zero, one, or many Revolved_flat objects.

4.3.229 Revolved_round to Partial_circular_profile

Each Revolved_round has the rounded_edge_shape defined by exactly one Partial_circular_profile. Each Partial_circular_profile defines the rounded_edge_shape for zero, one, or many Revolved_round objects.

©ISO 2000 - All rights reserved

4.3.230 Rib_top to Direction_element

Each Rib_top has the removal_direction defined by exactly one Direction_element. Each Direction_element defines the removal_direction for zero, one or many Rib_top objects.

4.3.231 Rib_top to Rib_top_floor

Each Rib_top has the floor_condition defined by exactly one Rib_top_floor. Each Rib_top_floor defines the floor_condition for zero, one, or many Rib_top objects.

4.3.232 Round_hole to Blind_bottom_condition

Each Round_hole has the bottom_condition defined by exactly one Blind_bottom_condition. Each Blind_bottom_condition defines the bottom_condition for zero, one, or many Round_hole objects.

NOTE - This assertion is established through Hole_bottom_condition_select.

4.3.233 Round_hole to Through_bottom_condition

Each Round_hole has the bottom_condition defined by exactly one Through_bottom_condition. Each Through_bottom_condition defines the bottom_condition for zero, one, or many Round_hole objects.

NOTE - This assertion is established through Hole_bottom_condition_select.

4.3.234 Round_hole to Circular_closed_profile

Each Round_hole has the diameter defined by exactly one Circular_closed_profile. Each Circular_-closed_profile defines the diameter for zero, one, or many Round_hole objects.

4.3.235 Round_hole to Linear_path

Each Round_hole has the hole_depth defined by exactly one Linear_path. Each Linear_path defines the hole_depth for zero, one, or many Round_hole objects.

4.3.236 Round_hole to Angle_taper

Each Round_hole has the change_in_diameter defined by zero or one Angle_taper. Each Angle_taper defines change_in_diameter for zero, one, or many Round_hole objects.

NOTE - This assertion is established through Taper_select.

4.3.237 Round_hole to Diameter_taper

Each Round_hole has the change_in_diameter defined by zero or one Diameter_taper. Each Diameter_taper defines change_in_diameter for zero, one, or many Round_hole objects.

NOTE - This assertion is established through Taper_select.

 ©ISO 2000 - All rights reserved

4.3.238 Round_hole to Directed_taper

Each Round_hole has the change_in_directed defined by zero or one Diameter_taper. Each Directed_taper defines change_in_diameter for zero, one, or many Round_hole objects.

NOTE - This assertion is established through Taper_select.

4.3.239 Rounded_end to Linear_path

Each Rounded_end has the course_of_travel defined by exactly one Linear_path. Each Linear_path defines the course_of_travel for zero, one, or many Rounded_end objects.

4.3.240 Rounded_end to Partial_circular_profile

Each Rounded_end has the partial_circular_boundary defined by exactly one Partial_circular_profile. Each Partial_circular_profile defines a partial_circular_boundary for zero, one, or many Rounded_end objects.

4.3.241 Rounded_U_profile to Numeric_parameter

Each Rounded_U_profile has the width defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the width for zero, one, or many Rounded_U_profile objects.

4.3.242 Second_chamfer_offset to Face_shape_element

Each Second_chamfer_offset has the second_face defined by exactly one Face_shape_element. Each Face_shape_element defines the second_face for zero, one, or many Second_chamfer_offset objects.

4.3.243 Second_offset to Numeric_parameter

Each Second_offset has the offset_amount defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the offset_amount for zero, one, or many Second_offset objects.

4.3.244 Shape to Base_shape

Each Shape has the base_shape_definition defined by zero or one Base_shape. Each Base_shape defines the base_shape_definition for exactly one Shape.

4.3.245 Shape to B-rep_shape_representation

Each Shape has the B-rep_form defined by zero, one, or many B-rep_shape_representation objects. Each B-rep_shape_representation defines the B-rep_form for exactly one Shape.

4.3.246 Shape to Shape_aspect

Each Shape has the element defined by zero, one, or many Shape_aspect objects. Each Shape_aspect defines the element for zero, one, or many Shape objects.

©ISO 2000 - All rights reserved

4.3.247 Shape_aspect to B-rep_model_element

Each Shape_aspect has the B-rep_shape defined by zero, one, or many B-rep_model_element objects. Each B-rep_model_element defines the B-rep_shape for exactly one Shape_aspect.

4.3.248 Shape_aspect to B-rep_shape_aspect_representation

Each Shape_aspect has the B-rep_form defined by zero, one, or many B-rep_shape_aspect_representation objects. Each B-rep_shape_aspect_representation defines the B-rep_form for exactly one Shape_aspect.

4.3.249 Shape_aspect to Shape_element

Each Shape_aspect has the element defined by zero or one Shape_element. Each Shape_element defines the element for exactly one Shape_aspect.

4.3.250 Shape_profile to Direction_element

Each Shape_profile has the removal_direction defined by exactly one Direction_element. Each Direction_element defines the removal_direction for zero, one or many Shape_profile objects.

4.3.251 Shape_profile to Profile_floor

Each Shape_profile has the floor_condition defined by exactly one Profile_floor. Each Profile_floor defines the floor_condition for zero, one, or many Shape_profile objects.

NOTE - This assertion is established through Profile_select.

4.3.252 Shape_profile to Through_profile_floor

Each Shape_profile has the floor_condition defined by exactly one Profile_floor. Each Profile_floor defines the floor_condition for zero, one, or many Shape_profile objects.

NOTE - This assertion is established through Profile_select.

4.3.253 Single_piece_part to Alternate_material

Each Single_piece_part has the alternate_material_definition defined by zero, one, or many Alternate_-material objects. Each Alternate_material defines the alternate_material_definition for one or more Single_piece_part.

4.3.254 Single_piece_part to Material

Each Single_piece_part has the material_definition defined by one or more Material objects. Each Material defines the material_definition for one or more Single_piece_part objects.

 ©ISO 2000 - All rights reserved

4.3.255 Size_tolerance to Shape_element

Each Size_tolerance has the applied_shape defined exactly one Shape_element. Each Shape_element defines the applied_shape for zero, one, or many Size_dimension objects.

4.3.256 Slot to Open_profile

Each Slot has the sweep_shape defined by exactly one Open_profile. Each Open_profile defines the sweep_shape for zero, one, or many Slot objects.

4.3.257 Slot to Path

Each Slot has the course_of_travel defined by exactly one Path. Each Path defines the course_of_travel for zero, one, or many Slot objects.

4.3.258 Slot to Slot_end_type

Each Slot has the end_conditions defined by exactly two Slot_end_type objects. Each Slot_end_type defines the end_condition for zero, one, or many Slot objects.

4.3.259 Specification to Specification_usage_constraint

Each Specification has the constraint defined by zero, one, or many Specification_usage_constraint objects. Each Specification_usage_constraint defines the constraint for exactly one Specification.

4.3.260 Spherical_cap to Numeric_parameter

Each Spherical_cap has the internal_angle defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the internal_angle for zero, one, or many Spherical_cap objects.

Each Spherical_cap has the radius defined by exactly one Numeric_parameter. Each Numeric_parameter defines the radius for zero, one, or many Spherical_cap objects.

4.3.261 Spherical_hole_bottom to Numeric_parameter

Each Spherical_hole_bottom has the radius defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the radius for zero, one, or many Spherical_hole_bottom objects.

4.3.262 Square_U_profile to Numeric_parameter

Each Square_U_profile has the first_angle defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the first_angle for zero, one, or many Square_U_profile objects.

Each Square_U_profile has the first_radius defined by zero or one Numeric_parameter. Each Numeric_-parameter defines the first_radius for zero, one, or many Square_U_profile objects.

©ISO 2000 - All rights reserved

Each Square_U_profile has the second_angle defined by exactly one Numeric_parameter. Each Numeric_parameter defines the second_angle for zero, one, or many Square_U_profile objects.

Each Square_U_profile has the second_radius defined by zero or one Numeric_parameter. Each Numeric_parameter defines the second_radius for zero, one, or many Square_U_profile objects.

Each Square_U_profile has the width defined by exactly one Numeric_parameter Each Numeric_-parameter defines the width for zero, one, or many Square_U_profile objects.

4.3.263 Step to Vee_profile

Each Step has the removal_boundary defined by exactly one Vee_profile. Each Vee_profile defines the removal_boundary for zero, one, or many Step objects.

4.3.264 Step to Linear_path

Each Step has the course_of_travel defined by exactly one Linear_path. Each Linear_path defines the course_of_travel for zero, one, or many Step objects.

4.3.265 Straightness_tolerance to Orientation

Each Straightness_tolerance has the affected_plane defined by zero or one Orientation. Each Orientation defines the affected_plane for zero, one, or many Straightness_tolerance objects.

4.3.266 Surface_profile_tolerance to Datum

Each Surface_profile has the geometric_reference defined by zero, one, two, or three Datum objects. Each Datum defines the geometric_reference for zero, one, or many Surface_profile objects.

4.3.267 Surface_property to Property_parameter

Each Surface_property has the property_characteristic defined by zero, one, or many Property_parameter objects. A Property_parameter defines the property_characteristic for exactly one Surface_property.

4.3.268 Symmetry_tolerance to Datum

Each Symmetry_tolerance has the geometric_reference defined by one, two, or three Datum objects. Each Datum defines the geometric_reference for zero, one, or many Symmetry_tolerance objects.

4.3.269 Symmetry_tolerance to Orientation

Each Symmetry_tolerance has the affected_plane defined by zero or one Orientation. Each Orientation defines the affected_plane for zero, one, or many Symmetry_tolerance objects.

 ©ISO 2000 - All rights reserved

4.3.270 Target_area to Shape_element

Each Target_area has the area_shape defined by exactly one Shape_element. Each Shape_element defines the area_shape by zero, one, or many Shape_element objects.

4.3.271 Tee_profile to Numeric_parameter

Each Tee_profile has the cross_bar_depth defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the cross_bar_depth for zero, one, or many Tee_profile objects.

Each Tee_profile has the cross_bar_width defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the cross_bar_width for zero, one, or many Tee_profile objects.

Each Tee_profile has the depth defined by exactly one Numeric_parameter. Each Numeric_parameter defines the depth for zero, one, or many Tee_profile objects.

Each Tee_profile has the first angle defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the first angle for zero, one, or many Tee_profile objects.

Each Tee_profile has the first offset defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the first offset for zero, one, or many Tee_profile objects.

Each Tee_profile has the second angle defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the second angle for zero, one, or many Tee_profile objects.

Each Tee_profile has the second offset defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the second offset for zero, one, or many Tee_profile objects.

Each Tee_profile has the radius defined by zero or one Numeric_parameter. Each Numeric_parameter defines the radius for zero, one, or many Tee_profile objects.

Each Tee_profile has the width defined by exactly one Numeric_parameter. Each Numeric_parameter defines the width for zero, one, or many Tee_profile objects.

4.3.272 Thread to Descriptive_parameter

Each Thread has the fit_class defined by exactly one Descriptive_parameter. Each Descriptive_-parameter defines the fit_class for zero, one, or many Thread objects.

Each Thread has the form defined by exactly one Descriptive_parameter. Each Descriptive_parameter defines the form for zero, one, or many Thread objects.

Each Thread has the qualifier defined by zero or one Descriptive_parameter. Each Descriptive_parameter defines the qualifier for zero, one, or many Thread objects.

Each Thread has the thread_hand defined by exactly one Descriptive_parameter. Each Descriptive_-parameter defines the thread_hand for zero, one, or many Thread objects.

©ISO 2000 - All rights reserved

4.3.273 Thread to Numeric_parameter

Each Thread has the major_diameter defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the major_diameter for zero, one, or many Thread objects.

Each Thread has the number_of_threads defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the number_of_threads for zero, one, or many Thread objects.

4.3.274 Thread to Partial_area_definition

Each Thread has the partial_profile defined by exactly one Partial_area_definition. Each Partial_area_-definition defines the partial_profile for zero, one, or many Thread objects.

4.3.275 Thread to Shape_element

Each Thread has the applied_shape defined by exactly one Shape_element. Each Shape_element defines the applied_shape for zero, one, or many Thread objects.

4.3.276 Tolerance_value to Limits_and_fits

Each Tolerance_value has the defined_value by zero or one Limit_and_fits. Each Limits_and_fits defines the defined_value for zero, one, or many Tolerance_value objects.

NOTE - This assertion is established through Tolerance_definition_select.

4.3.277 Tolerance_value to Plus_minus_value

Each Tolerance_value has the defined_value by zero or one Plus_minus_value. Each Plus_minus_value defines the defined_values for zero, one, or many Tolerance_value objects.

NOTE - This assertion is established through Tolerance_definition_select.

4.3.278 Tolerance_value to Tolerance_limit

Each Tolerance_value has the defined_value by zero or one tolerance_limit. Each tolerance_limit defines the defined_value for zero, one, or many Tolerance_value objects.

NOTE - This assertion is established through Tolerance_definition_select.

4.3.279 Tolerance_value to Tolerance_range

Each Tolerance_value has the defined_value by zero or one tolerance_range. Each tolerance_range defines the defined_value for zero, one, or many Tolerance_value objects.

NOTE - This assertion is established through Tolerance_definition_select.

 ©ISO 2000 - All rights reserved

4.3.280 Tolerance_zone to Tolerance_zone_definition

Each Tolerance_zone has zone_definition defined by exactly one Tolerance_zone_definition. Each Tolerance_zone_definition defines zone_definition for zero, one, or many Tolerance_zone objects.

4.3.281 Tolerance_zone to Projection

Each Tolerance_zone has extended_shape defined by zero, one, two, or three Projection objects. Each Projection defines extended_shape for zero, one, or many Tolerance_zone objects.

4.3.282 Tolerance_zone_definition to Shape_element

Each Tolerance_zone_definition has the first_element defined by exactly one Shape_element. Each Shape_element defines the first_element for zero, one, or many Tolerance_zone_definition objects.

Each Tolerance_zone_definition has the second_element defined by zero or one Shape_element. Each Shape_element defines the second_element for zero, one, or many Tolerance_zone_definition objects.

4.3.283 Total_runout_tolerance to Datum

Each Total_runout_tolerance has the geometric_reference defined by one or two Datum objects. Each Datum defines the geometric_reference for zero, one, or many Total_runout_tolerance objects.

4.3.284 Turned_knurl to Numeric_parameter

Each Turned_knurl has the tooth_depth defined by zero or one Numeric_parameter. Each Numeric_-parameter defines the tooth_depth for zero, one, or many Turned_knurl objects.

Each Turned_knurl has the root_fillet defined by zero or one Numeric_parameter. Each Numeric_-parameter defines the root_fillet for zero, one, or many Turned_knurl objects.

Each Turned_knurl has the major_diameter defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the major_diameter for zero, one, or many Turned_knurl objects.

Each Turned_knurl has the nominal_diameter defined by exactly one Numeric_parameter. Each Numeric_parameter defines the nominal_diameter for zero, one, or many Turned_knurl objects.

Each Turned_knurl has the number_of_teeth defined by zero or one Numeric_parameter. Each Numeric_parameter defines the number_of_teeth for zero, one, or many Turned_knurl objects.

Each Turned_knurl has the diametral_pitch defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the diametral_pitch for zero, one, or many Turned_knurl objects.

4.3.285 Vee_profile to Numeric_parameter

Each Vee_profile has the profile_radius defined by zero or one Numeric_parameter. Each Numeric_-parameter defines the profile_radius for zero, one, or many Vee_profile objects.

©ISO 2000 - All rights reserved

Each Vee_profile has the profile_angle defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the profile_angle for zero, one, or many Vee_profile objects.

Each Vee_profile has the tilt_angle defined by exactly one Numeric_parameter. Each Numeric_-parameter defines the tilt_angle for zero, one, or many Vee_profile objects.

4.3.286 Woodruff_slot_end_type to Numeric_parameter

Each Woodruff_slot_end_type has the radius defined by exactly one Numeric_parameter. Each Numeric_parameter defines the radius for zero, one, or many Woodruff_slot_end_type objects.

5 Application interpreted model

5.1 Mapping table

This clause contains the mapping table that shows how each UoF and application object of this part of ISO 10303 (see clause 4) maps to one or more AIM constructs (see annex A). The mapping table is organized in five columns.

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

Column 2) AIM element: Name of an AIM element as it appears in the AIM (see annex A), the term 'IDENTICAL MAPPING', or the term 'PATH'. AIM entities are written in lower case. Attribute names of AIM entities are referred to as <entity name>.<attribute name>. The mapping of an application element may result in several related AIM elements. Each of these AIM elements requires a line of its own in the table. The term ‘IDENTICAL MAPPING' indicates that both application objects of the application assertion map to the same AIM element. The term 'PATH' indicates that the application assertion maps to the entire reference path.

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

Column 4) Rules: One or more numbers may be given that refer to rules that apply to the current AIM element or reference path. For rules that are derived from relationships between application objects, the same rule is referred to by the mapping entries of all the involved AIM elements. The expanded names of the rules are listed after the table.

Column 5) Reference path: To describe fully the mapping of an application object, it may be necessary to specify a reference path through several related AIM elements. The reference path column documents the role of an AIM element relative to the AIM element in the row succeeding it. Two or more such related AIM elements define the interpretation of the integrated resources that satisfies the requirement specified by the application object. For each AIM element that has been created for use within this part of ISO 10303, a reference path up to its supertype from an integrated resource is specified.

 ©ISO 2000 - All rights reserved

For the expression of reference paths and the relationships between AIM elements the following notational conventions apply:

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

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

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

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

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

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

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

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

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

j) = : the string, select, or enumeration type is constrained to a choice or value;

k) \ : the line continuation for strings that wrap.

©ISO 2000 - All rights reserved

Table 1 - Mapping table for design_exception UoF

Application element	AIM element	Source	Rules	Reference path
DESIGN_EXCEPTION_-NOTICE	versioned_action_request	41	3	
issuing_date	date	41	6	versioned_action_request feature_based_pp_dated_item = versioned_action_request feature_based_pp_dated_item <- feature_based_pp_date_assignment.items[i] feature_based_pp_date_assignment <= {date_assignment date_assignment.role -> date_role date_role.name = 'issuing date'} date_assignment date_assignment.assigned_date -> date
notice_description	versioned_action_request.purpose	41		
notice_number	versioned_action_request.id	41		
technical_-recommendation	action_method	41		versioned_action_request <- action_request_solution.request action_request_solution action_request_solution.method -> action_method

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 1 - Mapping table for design_exception UoF (continued)

Application element	AIM element	Source	Rules	Reference path
design_exception_notice to engineering_change_-proposal (as issues)	PATH			versioned_action_request <- action_request_solution.request action_request_solution action_request_solution.method -> action_method feature_based_pp_document_item = action_method feature_based_pp_document_item <- feature_based_pp_document_reference.items[i] feature_based_pp_document_reference <= document_reference document_reference.assigned_document -> document {document document.kind -> document_type document_type.product_data_type = 'engineering change proposal'}
design_exception_notice to part (as discrepant_part)	PATH			versioned_action_request <- action_request_assignment.assigned_action_request action_request_assignment => feature_based_pp_action_request_assignment feature_based_pp_action_request_assignment.items[i] -> feature_based_pp_action_request_item feature_based_pp_action_request_item = product_definition_formation product_definition_formation
ENGINEERING_-CHANGE_ORDER	action_directive	41		
change_order_number	action_directive.name	41		

Table 1 - Mapping table for design_exception UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
engineering_change_- order to part (as new_version)	PATH			action_directive <- directed_action.directive directed_action <= executed_action <= action <- action_assignment.assigned_action action_assignment => feature_based_pp_action_assignment feature_based_pp_action_assignment.items[i] -> feature_based_pp_action_item feature_based_pp_action_item = product_definition_formation product_definition_formation
ENGINEERING_- CHANGE_PROPOSAL	document	41		{document document.kind -> document_type document_type.product_data_type = 'engineering change proposal'
change_proposal_number	document.id	41		
engineering_change_- proposal to engineering_- change_order (as incorporated_- proposal)	PATH			document<- document_reference.assigned_document document_reference=> feature_based_pp_document_reference feature_based_pp_document_reference.items[i]-> feature_based_pp_document_item feature_based_pp_document_item = action_method action_method<- action.chosen_method action=> executed_action=> directed_action directed_action.directive-> action_directive

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF

Application element	AIM element	Source	Rules	Reference path
ANGLE_TAPER	taper	224	23	taper <= shape_aspect {shape_aspect shape_aspect.description = 'angle taper'} {shape_aspect shape_aspect.of_shape->product_definition_shape<= property_definition property_definition.definition->characterized_definition=characterized_object characterized_object=> feature_component_definition}

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
angle_taper to numeric_-parameter (as angle)	PATH		9, 19, 24	taper <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'taper angle'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
BLIND_BOTTOM_- CONDITION	hole_bottom	224	23	hole_bottom <= shape_aspect {shape_aspect (shape_aspect.description = 'conical') (shape_aspect.description = 'flat') (shape_aspect.description = 'flat with radius') (shape_aspect.description = 'flat with taper') (shape_aspect.description = 'spherical')} {shape_aspect.of_shape -> product_definition_shape <= property_definition property_definition.definition -> characterized_definition characterized_definition = characterized_object characterized_object => feature_component_definition}
start_or_end	shape_aspect_relationship.name	41		round_hole <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape shape_aspect <- shape_aspect_relationship.related_shape_aspect {shape_aspect_relationship => feature_component_relationship} shape_aspect_relationship shape_aspect_relationship.name = 'hole depth start' shape_aspect_relationship.name = 'hole depth end'

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
BOSS_TOP_CONDITION	boss_top	224	23	boss_top <= shape_aspect {shape_aspect.of_shape -> product_definition_shape <= property_definition property_definition.definition -> characterized_definition characterized_definition = characterized_object characterized_object => feature_component_definition}
start_or_end	shape_aspect_relationship.name			boss_top <= shape_aspect<- shape_aspect_relationship.relating_shape_aspect shape_aspect_relationship shape_aspect_relationship.name {(shape_aspect_relationship.name = 'boss height start') (shape_aspect_relationship.name = 'boss height end')}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
boss_top to numeric_- parameter (as top_radius)	PATH		9, 19, 24	boss_top <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- {property_definition=> product_definition_shape} property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'top radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
CHAMFER_ANGLE	chamfer_offset	224	23	chamfer_offset <= shape_aspect {shape_aspect shape_aspect.description = 'second offset'}

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
chamfer_angle to numeric_parameter (as angle_amount)	PATH		9, 19, 24	chamfer_offset <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'offset angle'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}
CIRCULAR_PATH	path_feature_component	224	23	path_feature_component <= shape_aspect {shape_aspect (shape_aspect.description = 'partial circular') (shape_aspect.description = 'complete circular')}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circular_path to numeric_parameter (as radius)	PATH		9, 19, 24	path_feature_component <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation ->{representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] ->{representation_item representation_item.name = 'radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
COMPLETE_-CIRCULAR_PATH	path_feature_component	224	23	path_feature_component <= shape_aspect {shape_aspect shape_aspect.description = 'complete circular'}
CONICAL_HOLE_-BOTTOM	hole_bottom	224	23	hole_bottom <= shape_aspect {shape_aspect shape_aspect.description = 'conical'}

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
conical_hole_bottom to numeric_parameter (as tip_angle)	PATH		9, 19, 24	hole_bottom <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'tip angle'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
conical_hole_bottom to numeric_parameter (as tip_radius)	PATH		9, 19, 24	hole_bottom <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation ->{representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] ->{representation_item representation_item.name = 'tip radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
DIAMETER_TAPER	taper	224	23	taper <= shape_aspect {shape_aspect shape_aspect.description = 'diameter taper'} {shape_aspect shape_aspect.of_shape->product_definition_shape<= property_definition property_definition.definition->characterized_definition=characterized_object characterized_object=> feature_component_definition}

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
diameter_taper to numeric_parameter (as final_diameter)	PATH		9, 19, 24	taper <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'final diameter'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
DIRECTED_TAPER	taper	224	23	taper <= shape_aspect {shape_aspect shape_aspect.description = 'directed taper'}{shape_aspect shape_aspect.of_shape-> product_definition_shape<= property_definition property_definition.definition-> characterized_definition=characterized_object characterized_object=> feature_component_definition}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
directed_taper to direction_element (as direction)	PATH		9, 19, 24	taper <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'direction'} representation => shape_representation=> direction_shape_representation

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
directed_taper to numeric_parameter (as angle)	PATH		9, 19, 24	taper <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'angle'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}
FIRST_OFFSET	chamfer_offset	224	23	chamfer_offset <= shape_aspect {shape_aspect shape_aspect.description = 'first offset'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
first_offset to face_- shape_element (as face_shape)	PATH		19	chamfer_offset <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'first face shape'} representation => shape_representation=> face_shape_representation

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
first_offset to numeric_-parameter (as offset_amount)	PATH		9, 19, 24	chamfer_offset <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'offset amount'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
FLAT_HOLE_BOTTOM	hole_bottom	224	23	hole_bottom <= shape_aspect {shape_aspect shape_aspect.description = 'flat'}
FLAT_SLOT_END_TYPE	slot_end	224	23	slot_end <= shape_aspect {shape_aspect shape_aspect.description = 'flat'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
flat_slot_end_type to numeric_parameter (as first_radius)	PATH		9, 19, 24	slot_end <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'first radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
flat_slot_end_type to numeric_parameter (as second_radius)	PATH		9, 19, 24	slot_end <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'second radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
FLAT_WITH_RADIUS_-HOLE_BOTTOM	hole_bottom	224	23	hole_bottom <= shape_aspect {shape_aspect shape_aspect.description = 'flat with radius'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
flat_with_radius_hole_-bottom to numeric_-parameter (as corner_radius)	PATH		9, 19, 24	hole_bottom <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'corner radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
FLAT_WITH_TAPER_-HOLE_BOTTOM	hole_bottom	224	23	hole_bottom <= shape_aspect {shape_aspect shape_aspect.description = 'flat with taper'}

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
flat_with_taper_hole_- bottom to numeric_- parameter (as final_diameter)	PATH		9, 19, 24	hole_bottom <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'final diameter'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
flat_with_taper_hole_- bottom to numeric_- parameter (as taper angle)	PATH		9, 19, 24	hole_bottom <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'taper angle'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}
GENERAL_PATH	path_feature_component	224	23	path_feature_component <= shape_aspect {shape_aspect shape_aspect.description = 'complex'}

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
general_path to path_-element (as sweep_path)	PATH		19	path_feature_component <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation ->{representation representation.name = 'sweep path'} representation => shape_representation=> path_shape_representation
GENERAL_PROFILE_-FLOOR	profile_floor	224	23	profile_floor <= shape_aspect {shape_aspect shape_aspect.description = 'complex'}
general_profile_floor to face_shape_element (as floor)	PATH		19	profile_floor <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation ->{representation representation.name = 'floor'} representation => shape_representation=> face_shape_representation
GENERAL_POCKET_-BOTTOM_CONDITION	pocket_bottom	224	23	pocket_bottom <= shape_aspect {shape_aspect shape_aspect.description = 'complex'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
general_pocket_bottom_-condition to face_shape_-element (as floor)	PATH		19	pocket_bottom <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'floor face'} representation => shape_representation=> face_shape_representation

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
general_pocket_bottom_-condition to numeric_-parameter (as floor_radius)	PATH		9, 19, 24	pocket_bottom <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
GENERAL_RIB_TOP_-FLOOR	rib_top_floor	224	23	rib_top_floor <= shape_aspect {shape_aspect shape_aspect.description = 'complex'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
general_rib_top_floor to ordered_face_element (as rib_top_face)	PATH		19	rib_top_floor<= shape_aspect shape_definition=shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'rib top face'} representation => shape_representation => face_shape_representation
GENERAL_TOP_-CONDITION	boss_top	224	23	boss_top <= shape_aspect {shape_aspect shape_aspect.description = 'complex'}
general_top_condition to face_shape_element (as top_face)	PATH		19	boss_top <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation => shape_representation=> face_shape_representation

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
LINEAR_PATH	path_feature_component	224	23	path_feature_component <= shape_aspect {shape_aspect shape_aspect.description = 'linear'}
linear_path to direction_element (as direction)	PATH			path_feature_component <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation => shape_representation=> direction_shape_representation

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
linear_path to numeric_-parameter (as distance)	PATH		9, 19, 24	path_feature_component <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation ->{representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] ->{representation_item representation_item.name = 'distance'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
OPEN_SLOT_END_TYPE	slot_end	224	23	slot_end <= shape_aspect {shape_aspect shape_aspect.description = 'open'}
PARTIAL_AREA_-DEFINITION	applied_area	224	23	applied_area <= shape_aspect {shape_aspect shape_aspect.of_shape->product_definition_shape}

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
partial_area_definition to numeric_parameter (as effective_length)	PATH		9, 19, 24	applied_area <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'effective length'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
partial_area_definition to numeric_parameter (as maximum_length)	PATH		9, 19, 24	applied_area <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'maximum length'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
partial_area_definition to orientation (as placement)	PATH		9, 19, 24	applied_area <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'orientation'} representation_item => geometric_representation_item => placement
PARTIAL_CIRCULAR_-PATH	path_feature_component	224	23	path_feature_component <= shape_aspect {shape_aspect shape_aspect.description = 'partial circular'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
partial_circular_path to numeric_parameter (as sweep_angle)	PATH		9, 19, 24	path_feature_component <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation ->{representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] ->{representation_item representation_item.name = 'sweep angle'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}
PATH	path_feature_component	224	23	path_feature_component <= shape_aspect {shape_aspect shape_aspect.of_shape->product_definition_shape<= property_definition property_definition.definition->characterized_definition=characterized_object characterized_object=> feature_component_definition}

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
path to orientation (as placement)	PATH		9, 19, 24	path_feature_component <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'orientation'} representation_item => geometric_representation_item => placement
PLANAR_POCKET_- BOTTOM_CONDITION	pocket_bottom	224	23	pocket_bottom <= shape_aspect {shape_aspect shape_aspect.description = 'planar'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
planar_pocket_bottom_-condition to direction_-element (as floor_normal)	PATH		19	pocket_bottom <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'floor normal'} representation => shape_representation=> direction_shape_representation
planar_pocket_bottom_-condition to location_-element (as floor_location)	PATH		19	pocket_bottom <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'floor location'} representation => shape_representation=> location_shape_representation

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
planar_pocket_bottom_-condition to numeric_-parameter (as floor_radius)	PATH		9, 19, 24	pocket_bottom <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
PLANAR_PROFILE_-FLOOR	profile_floor	224	23	profile_floor <= shape_aspect {shape_aspect shape_aspect.description = 'planar'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
planar_profile_floor to planar_element (as floor)	PATH		19	profile_floor <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'floor'} representation => planar_shape_representation
PLANAR_RIP_TOP_-FLOOR	rib_top_floor		23	rib_top_floor <= shape_aspect {shape_aspect shape_aspect.description = 'planar'}
planar_rib_top_floor to closed_profile (as boundary)	PATH		9, 19, 24	rib_top_floor <= {shape_aspect shape_aspect.description = 'boundary occurrence'} shape_aspect<- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> {shape_aspect shape_aspect.description= 'rib top floor boundary'} shape_aspect => (circular_closed_profile) (closed_path_profile) (ngon_closed_profile) (rectangular_closed_profile)

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
planar_rib_top_floor to planar_element (as planar_face)	PATH		19	rib_top_floor<= shape_aspect shape_definition=shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'rib top face'} representation => shape_representation => planar_shape_representation
PLANAR_TOP_-CONDITION	boss_top	224	23	boss_top <= shape_aspect {shape_aspect shape_aspect.description = 'planar'}
planar_top_condition to direction_element (as top_normal)	PATH		19	boss_top <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation => shape_representation=> direction_shape_representation

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
planar_top_condition to location_element (as top_location)	PATH		19	boss_top <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation => shape_representation=> location_shape_representation
POCKET_BOTTOM_-CONDITION	pocket_bottom	224	23	pocket_bottom <= shape_aspect {shape_aspect.of_shape -> product_definition_shape <= property_definition property_definition.definition -> characterized_definition characterized_definition = characterized_object characterized_object => feature_component_definition}

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
start_or_end	shape_aspect_relationship.name	41		pocket_bottom <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape shape_aspect <- shape_aspect_relationship.related_shape_aspect {shape_aspect_relationship => feature_component_relationship} shape_aspect_relationship shape_aspect_relationship.name = 'pocket depth start' shape_aspect_relationship.name = 'pocket depth end'
PROFILE_FLOOR	profile_floor	224	23	profile_floor<= shape_aspect {shape_aspect.of_shape -> product_definition_shape <= property_definition property_definition.definition -> characterized_definition characterized_definition = characterized_object characterized_object => feature_component_definition}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
start_or_end	descriptive_representation_- item.description	45	9, 19, 24	profile_floor <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation ->{representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] ->{representation_item representation_item.name = 'shape profile floor orientation'} representation_item => descriptive_representation_item descriptive_representation_item.description {(descriptive_representation_item.description = 'shape profile start') (descriptive_representation_item.description = 'shape profile end')}

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
profile_floor to numeric_parameter (as floor radius)	PATH		9, 19, 24	profile_floor <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
RADIUSED_SLOT_-END_TYPE	slot_end	224	23	slot_end <= shape_aspect {shape_aspect shape_aspect.description = 'radiused'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
RIP_TOP_FLOOR	rib_top_floor	224	23	rib_top_floor <= shape_aspect {shape_aspect.of_shape -> product_definition_shape <= property_definition property_definition.definition -> characterized_definition characterized_definition = characterized_object characterized_object => feature_component_definition}
SECOND_CHAMFER_-OFFSET	chamfer_offset	224	23	chamfer_offset <= shape_aspect {shape_aspect shape_aspect.description = 'second offset'}
second_chamfer_offset to face_shape_element (as second_face)	PATH		19	chamfer_offset <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'second face shape'} representation => shape_representation=> face_shape_representation
SECOND_OFFSET	chamfer_offset	224	23	chamfer_offset <= shape_aspect {shape_aspect shape_aspect.description = 'second offset'}

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
second_offset to numeric_parameter (as offset_amount)	PATH		9, 19, 24	chamfer_offset <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'offset amount'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
SLOT_END_TYPE	slot_end	224	23	slot_end <= shape_aspect {shape_aspect.of_shape -> product_definition_shape <= property_definition property_definition.definition -> characterized_definition characterized_definition = characterized_object characterized_object => feature_component_definition}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
first_or_second	shape_aspect_relationship.name	41		slot_end <= shape_aspect <- shape_aspect_relationship.relating_shape_aspect shape_aspect_relationship shape_aspect_relationship.name {(shape_aspect_relationship.name = 'course of travel start') (shape_aspect_relationship.name = 'course of travel end')}
SPHERICAL_HOLE_- BOTTOM	hole_bottom	224	23	hole_bottom <= shape_aspect {shape_aspect shape_aspect.description = 'spherical'}
spherical_hole_bottom to numeric_parameter (as radius)	PATH		9, 19, 24	hole_bottom <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 2 - Mapping table for feature_definition_item UoF (continued)

Application element	AIM element	Source	Rules	Reference path
THROUGH_BOTTOM_-CONDITION	hole_bottom	224	23	hole_bottom <= shape_aspect {shape_aspect shape_aspect.description = 'through'} {shape_aspect.of_shape -> product_definition_shape <= property_definition property_definition.definition -> characterized_definition characterized_definition = characterized_object characterized_object => feature_component_definition}
THROUGH_POCKET_-BOTTOM_CONDITION	pocket_bottom	224	23	pocket_bottom <= shape_aspect {shape_aspect shape_aspect.description = 'through'} {shape_aspect.of_shape -> product_definition_shape <= property_definition property_definition.definition -> characterized_definition characterized_definition = characterized_object characterized_object => feature_component_definition}
THROUGH_PROFILE_-FLOOR	profile_floor	224	23	profile_floor <= shape_aspect {shape_aspect shape_aspect.description = 'through'} {shape_aspect.of_shape -> product_definition_shape <= property_definition property_definition.definition -> characterized_definition characterized_definition = characterized_object characterized_object => feature_component_definition}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 2 - Mapping table for feature_definition_item UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
WOODRUFF_SLOT_- END_TYPE	slot_end	224	23	slot_end <= shape_aspect {shape_aspect shape_aspect.description = 'woodruff'}
woodruff_slot_end_- type to numeric_- parameter (as radius)	PATH		9, 19, 24	slot_end <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF

Application element	AIM element	Source	Rules	Reference path
CIRCULAR_CLOSED_-PROFILE	circular_closed_profile	224	23	circular_closed_profile <= shape_aspect
circular_closed_profile to numeric_parameter (as diameter)	PATH		9, 19, 24	circular_closed_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'diameter'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
CLOSED_PROFILE	(circular_closed_profile) (closed_path_profile) (ngon_closed_profile) (rectangular_closed_profile)	224 224 224 224	23	(circular_closed_profile <=) (closed_path_profile <=) (ngon_closed_profile <=) (rectangular_closed_profile <=) shape_aspect

©ISO 2000 - All rights reserve

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
GENERAL_CLOSED_-PROFILE	closed_path_profile	224	23	closed_path_profile <= shape_aspect
general_closed_profile to path_element (as closed_profile_shape)	PATH		19	closed_path_profile <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation => shape_representation=> path_shape_representation
GENERAL_OPEN_-PROFILE	open_path_profile	224	23	open_path_profile <= shape_aspect
general_open_profile to path_element (as enclosed_boundary)	PATH		19	open_path_profile <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation => shape_representation=> path_shape_representation
LINEAR_PROFILE	linear_profile	224	23	linear_profile <= shape_aspect

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
linear_profile to numeric_parameter (as profile_length)	PATH		9, 19, 24	linear_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'profile length'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
NGON_PROFILE	ngon_closed_profile	224	23	ngon_closed_profile <= shape_aspect

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circumscribed_or_- across_flats	(representation_item.name = 'circumscribed diameter') (representation_item.name = 'diameter across flats')			ngon_closed_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> representation_item (representation_item.name = 'circumscribed diameter') (representation_item.name = 'diameter across flats')

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
ngon_profile to numeric_parameter (as diameter)	PATH		9, 19, 24	ngon_closed_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item (representation_item.name = 'circumscribed diameter') (representation_item.name = 'diameter across flats')} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
ngon_profile to numeric_parameter (as corner_radius)	PATH		9, 19, 24	ngon_closed_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'corner radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
ngon_profile to numeric_parameter (as number_of_sides)	PATH		9, 19, 24	ngon_closed_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'number of sides'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit measure_with_unit.value_component -> measure_value measure_value = count_measure count_measure}
OPEN_PROFILE	(linear_profile) (open_path_profile) (partial_circular_profile) (rounded_u_profile) (square_u_profile) (tee_profile) (vee_profile)	224 224 224 224 224 224 224	23	(linear_profile <=) (open_path_profile <=) (partial_circular_profile <=) (rounded_u_profile <=) (square_u_profile <=) (tee_profile <=) (vee_profile <=) shape_aspect

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
open_profile to planar_element (as profile_limit)	PATH			(rounded_u_profile <=) (square_u_profile <=) (open_path_profile <=) (tee_profile <=) (vee_profile <=) shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> representation=> {representation.name='profile limit'} shape_representation=> planar_shape_representation
PARTIAL_CIRCULAR_-PROFILE	partial_circular_profile	224	23	partial_circular_profile <= shape_aspect

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
partial_circular_profile to numeric_parameter (as radius)	PATH		9, 19, 24	partial_circular_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
partial_circular_profile to numeric_parameter (as sweep_angle)	PATH		9, 19, 24	partial_circular_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'sweep angle'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
PROFILE	(circular_closed_profile)	224	23	(circular_closed_profile <=)
	(closed_path_profile)	224		(closed_path_profile <=)
	(ngon_closed_profile)	224		(ngon_closed_profile <=)
	(rectangular_closed_profile)	224		(rectangular_closed_profile <=)
	(linear_profile)	224		(linear_profile <=)
	(open_path_profile)	224		(open_path_profile <=)
	(partial_circular_profile)	224		(partial_circular_profile <=)
	(rounded_u_profile)	224		(rounded_u_profile <=)
	(square_u_profile)	224		(square_u_profile <=)
	(tee_profile)	224		(tee_profile <=)
	(vee_profile)	224		(vee_profile <=)
				shape_aspect
				{shape_aspect
				shape_aspect.of_shape->
				product_definition_shape<=
				property_definition
				property_definition.definition->
				characterized_definition=characterized_object
				characterized_object=>
				feature_component_definition}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
profile to orientation (as placement)	PATH		9, 19, 24	(circular_closed_profile <=) (closed_path_profile <=) (ngon_closed_profile <=) (rectangular_closed_profile <=) (linear_profile <=) (open_path_profile <=) (partial_circular_profile <=) (rounded_u_profile <=) (square_u_profile <=) (tee_profile <=) (vee_profile <=) shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'orientation'} representation_item => geometric_representation_item => placement
RECTANGULAR_-CLOSED_PROFILE	rectangular_closed_profile	224	23	rectangular_closed_profile <= shape_aspect

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_closed_- profile to numeric_- parameter (as corner_radius)	PATH		9, 19, 24	rectangular_closed_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'corner radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_closed_- profile to numeric_- parameter (as profile_length)	PATH		9, 19, 24	rectangular_closed_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'length'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_closed_- profile to numeric_- parameter (as profile_width)	PATH		9, 19, 24	rectangular_closed_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'width'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
ROUNDED_U_PROFILE	rounded_u_profile	224	23	rounded_u_profile <= shape_aspect

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rounded_u_profile to numeric_parameter (as width)	PATH		9, 19, 24	rounded_u_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'width'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
SQUARE_U_PROFILE	square_u_profile	224	23	square_u_profile <= shape_aspect

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
square_u_profile to numeric_parameter (as first_angle)	PATH		9, 19, 24	square_u_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'first angle'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
square_u_profile to numeric_parameter (as first_radius)	PATH		9, 19, 24	square_u_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'first radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
square_u_profile to numeric_parameter (as second_angle)	PATH		9, 19, 24	square_u_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'second angle'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
square_u_profile to numeric_parameter (as second_radius)	PATH		9, 19, 24	square_u_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'second radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
square_u_profile to numeric_parameter (as width)	PATH		9, 19, 24	square_u_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'width'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
TEE_PROFILE	tee_profile	224	23	tee_profile <= shape_aspect

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
tee_profile to numeric_- parameter (as cross_bar_depth)	PATH		9, 19, 24	tee_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'cross bar depth'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
tee_profile to numeric_-parameter (as cross_bar_width)	PATH		9, 19, 24	tee_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'cross bar width'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
tee_profile to numeric_- parameter (as depth)	PATH		9, 19, 24	tee_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'depth'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
tee_profile to numeric_- parameter (as first_angle)	PATH		9, 19, 24	tee_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'first angle'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
tee_profile to numeric_- parameter (as first_offset)	PATH		9, 19, 24	tee_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'first offset'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
tee_profile to numeric_- parameter (as radius)	PATH		9, 19, 24	tee_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
tee_profile to numeric_- parameter (as second_angle)	PATH		9, 19, 24	tee_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'second angle'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
tee_profile to numeric_- parameter (as second_offset)	PATH		9, 19, 24	tee_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'second offset'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
tee_profile to numeric_- parameter (as width)	PATH		9, 19, 24	tee_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'width'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
VEE_PROFILE	vee_profile	224	23	vee_profile <= shape_aspect

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
vee_profile to numeric_-parameter (as profile_angle)	PATH		9, 19, 24	vee_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'profile angle'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (continued)

Application element	AIM element	Source	Rules	Reference path
vee_profile to numeric_-parameter (as profile_radius)	PATH		9, 19, 24	vee_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'profile radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 3 - Mapping table for feature_profile UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
vee_profile to numeric_-parameter (as tilt_angle)	PATH		9, 19, 24	vee_profile <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'tilt angle'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF

Application element	AIM element	Source	Rules	Reference path
BOSS	boss	224	11, 23, 25	boss <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object (characterized_object.description = 'circular') (characterized_object.description = 'complex') (characterized_object.description = 'rectangular')}
boss to boss_top_-condition (as top_condition)	PATH		22, 23	boss <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'top condition occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'boss top usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => boss_top

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
boss to linear_path (as boss_height)	PATH		22, 23	boss <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'boss height occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship [shape_aspect_relationship.name = 'boss height'] [shape_aspect_relationship.description = 'path feature component usage']]} shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'linear'} shape_aspect => path_feature_component

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
boss to numeric_- parameter (as fillet_radius)	PATH		9, 19, 24	boss <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- {property_definition=> product_definition_shape} property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'fillet radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
CATALOGUE_KNURL	externally_defined_feature_definition	224	11, 23, 25	externally_defined_feature_definition <= [externally_defined_item {externally_definied_item.item_id-> source_item source_item=' external knurl'} {externally_definied_item.source-> external_source external_source.source_id-> source_item source_item=' external feature specification'}] [{feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'knurl'}]

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
catalogue_knurl to specification (as documentation) #1: (if specification has zero constrants) #2: (if specification has one or more constrants)	PATH			#1:(externally_defined_feature_definition document_reference_item = externally_defined_feature_definition document_reference_item <- applied_document_reference.items[i] applied_document_reference <= document_reference document_reference.assigned_document -> document) #2:(externally_defined_feature_definition document_reference_item = externally_defined_feature_definition document_reference_item <- applied_document_usage_constraint_assignment.items[i] applied_document_usage_constraint_assignment <= document_usage_constraint_assignment document_usage_constraint_assignment.assigned_document_usage -> document_usage_constraint document_usage_constraint.source-> document) {document<- document_representation_type.represented_document document_representation_type} {document=> document_file<= characterized_object}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
CATALOGUE_MARKING	externally_defined_feature_definition	224	11, 23, 25	externally_defined_feature_definition <= [externally_defined_item {externally_definied_item.item_id-> source_item source_item=' external marking'} {externally_definied_item.source-> external_source external_source.source_id-> source_item source_item=' external feature specification'}] [{feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'marking'}]

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
catalogue_marking to specification (as documentation) #1: (if specification has zero constrants) #2: (if specification has one or more constrants)	PATH			#1:(externally_defined_feature_definition document_reference_item = externally_defined_feature_definition document_reference_item <- applied_document_reference.items[i] applied_document_reference <= document_reference document_reference.assigned_document -> document) #2:(externally_defined_feature_definition document_reference_item = externally_defined_feature_definition document_reference_item <- applied_document_usage_constraint_assignment.items[i] applied_document_usage_constraint_assignment <= document_usage_constraint_assignment document_usage_constraint_assignment.assigned_document_usage -> document_usage_constraint document_usage_constraint.source-> document) {document<- document_representation_type.represented_document document_representation_type} {document=> document_file<= characterized_object}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
CATALOGUE_THREAD	externally_defined_feature_definition	224	11, 23, 25	externally_defined_feature_definition <= [externally_defined_item {externally_definied_item.item_id-> source_item source_item=' external thread'} {externally_definied_item.source-> external_source external_source.source_id-> source_item source_item=' external feature specification'}] [{feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'thread'}]

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
catalogue_thread to specification (as documentation) #1: (if specification has zero constrants) #2: (if specification has one or more constrants)	PATH			#1:(externally_defined_feature_definition document_reference_item = externally_defined_feature_definition document_reference_item <- applied_document_reference.items[i] applied_document_reference <= document_reference document_reference.assigned_document -> document) #2:(externally_defined_feature_definition document_reference_item = externally_defined_feature_definition document_reference_item <- applied_document_usage_constraint_assignment.items[i] applied_document_usage_constraint_assignment <= document_usage_constraint_assignment document_usage_constraint_assignment.assigned_document_usage -> document_usage_constraint document_usage_constraint.source-> document) {document<- document_representation_type.represented_document document_representation_type} {document=> document_file<= characterized_object}
CHAMFER	chamfer	224	22, 26, 27	chamfer <= transition_feature <= shape_aspect

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
chamfer to face_shape_-element (as chamfer_face)	PATH		19	chamfer <= transition_feature <= shape_aspect shape_definition=shape_aspect shape_definition characterized_definition=shape_definition characterized_definition<- property_definition.definition 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.name= 'chamfer face'} representation=> shape_representation=> face_shape_representation
chamfer to first_offset (as first_face_offset)	PATH		22, 23	chamfer <= transition_feature <= shape_aspect <- shape_aspect_relationship.relating_shape_aspect {shape_aspect_relationship => feature_component_relationship} shape_aspect_relationship shape_aspect_relationship.related_shape_aspect ->{shape_aspect shape_aspect.description = 'first offset'} shape_aspect => chamfer_offset

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
chamfer to second_- chamfer_offset (as second_face_offset)	PATH		22, 23	chamfer <= transition_feature <= shape_aspect <- shape_aspect_relationship.relating_shape_aspect {shape_aspect_relationship => feature_component_relationship} shape_aspect_relationship shape_aspect_relationship.related_shape_aspect ->{shape_aspect shape_aspect.description = 'second offset'} shape_aspect => chamfer_offset
CIRCULAR_BOSS	boss	224	11, 23, 25	boss <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'circular'}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circular_boss to angle_- taper (as change_in_diameter)	PATH		22, 23	boss <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'change in diameter occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'taper usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'angle taper'} shape_aspect => taper

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circular_boss to diameter_taper (as change_in_diameter)	PATH		22, 23	boss <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'change in diameter occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'taper usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'diameter taper'} shape_aspect => taper

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circular_boss to directed_taper (as change_in_diameter)	PATH		22, 23	boss <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'change in diameter occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'taper usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'directed taper'} shape_aspect => taper

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circular_boss to circular_- closed_profile (as circular_profile)	PATH		22, 23	boss <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'circular profile occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => circular_closed_profile
CIRCULAR_CLOSED_- SHAPE_PROFILE	outside_profile	224	11, 23, 25	outside_profile <= {feature_definition => instanced_feature} feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'closed circular boundary occurrence'}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circular_closed_shape_-profile to complete_circular_profile (as closed_boundary)	PATH		19	outside_profile <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'closed circular boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {shape_aspect_relationship => shape_defining_relationship shape_aspect_relationship shape_aspect_relationship.description = 'profile usage'} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => circular_closed_profile
CIRCULAR_CUTOUT	pocket	224	11, 23, 25	pocket <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'circular cutout'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circular_cutout to circular_closed_profile (as circular_boundary)	PATH		22, 23	pocket <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'enclosed boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => circular_closed_profile
CIRCULAR_OFFSET_-PATTERN	shape_aspect	41	22	[shape_aspect <- shape_aspect_relationship.related_shape_aspect shape_aspect_relationship => feature_component_relationship => pattern_offset_membership] [shape_aspect <- shape_aspect_relationship.related_shape_aspect {shape_aspect_relationship => feature_component_relationship} shape_aspect_relationship.relating_shape_aspect-> shape_aspect=> modified_pattern]

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circular_offset_pattern to numeric_parameter (as angular_offset)	PATH		9, 19, 24	shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = offset'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circular_offset_pattern to numeric_parameter (as index_number)	PATH		9, 19, 24	shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'offset index'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit measure_with_unit.value_component -> measure_value measure_value = count_measure count_measure}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
CIRCULAR_OMIT_-PATTERN	shape_aspect	41	22	[shape_aspect <- shape_aspect_relationship.related_shape_aspect shape_aspect_relationship => feature_component_relationship => pattern_omit_membership] [shape_aspect <- shape_aspect_relationship.related_shape_aspect {shape_aspect_relationship => feature_component_relationship} shape_aspect_relationship.relating_shape_aspect-> shape_aspect=> modified_pattern]

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circular_omit_pattern to numeric_parameter (as omit_index)	PATH		9, 19, 24	shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'omit index'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit measure_with_unit.value_component -> measure_value measure_value = count_measure count_measure}
CIRCULAR_PATTERN	circular_pattern	224	23	circular_pattern <= replicate_feature <= {feature_definition=> instanced_feature} feature_definition<= characterized_object

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circular_pattern to circular_offset_pattern (as relocated_base_-feature)	PATH		22	circular_pattern <= replicate_feature <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape [shape_aspect <- shape_aspect_relationship.relating_shape_aspect {shape_aspect_relationship => feature_component_relationship => pattern_offset_membership} shape_aspect_relationship shape_aspect_relationship.related_shape_aspect -> shape_aspect] [shape_aspect <- shape_aspect_relationship.related_shape_aspect {shape_aspect_relationship => feature_component_relationship} {shape_aspect_relationship.description='base pattern'} shape_aspect_relationship.relating_shape_aspect-> shape_aspect<- {shape_aspect=> modified_pattern} shape_aspect_relationship.relating_shape_aspect {shape_aspect_relationship => feature_component_relationship} {shape_aspect_relationship.description='modified pattern'} shape_aspect_relationship.related_shape_aspect-> shape_aspect]

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circular_pattern to circular_omit_pattern (as missing_base_feature)	PATH		22, 23	circular_pattern <= replicate_feature <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape [shape_aspect <- shape_aspect_relationship.relating_shape_aspect {shape_aspect_relationship => feature_component_relationship => pattern_omit_membership} shape_aspect_relationship shape_aspect_relationship.related_shape_aspect -> shape_aspect] [shape_aspect <- shape_aspect_relationship.related_shape_aspect {shape_aspect_relationship => feature_component_relationship} {shape_aspect_relationship.description='base pattern'} shape_aspect_relationship.relating_shape_aspect-> shape_aspect<- {shape_aspect=> modified_pattern} shape_aspect_relationship.relating_shape_aspect {shape_aspect_relationship => feature_component_relationship} {shape_aspect_relationship.description='modified pattern'} shape_aspect_relationship.related_shape_aspect-> shape_aspect]

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circular_pattern to numeric_parameter (as angular_spacing)	PATH		9, 19, 24	circular_pattern <= replicate_feature <= feature_definition<= characterized_object shape_definition = characterized_object shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'angular spacing'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circular_pattern to numeric_parameter (as base_feature_-diameter)	PATH		9, 19, 24	circular_pattern <= replicate_feature <= feature_definition<= characterized_object shape_definition = characterized_object shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'diameter'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circular_pattern to numeric_parameter (as base_feature_rotation)	PATH		9, 19, 24	circular_pattern <= replicate_feature <= feature_definition<= characterized_object shape_definition = characterized_object shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'base feature rotation'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circular_pattern to numeric_parameter (as number_of_features)	PATH		9, 19, 24	circular_pattern <= replicate_feature <= feature_definition<= characterized_object shape_definition = characterized_object shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'number of features'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit measure_with_unit.value_component -> measure_value measure_value = count_measure count_measure}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
COMPOUND_FEATURE	compound_feature	224	11, 23, 25	compound_feature <= {feature_definition=> instanced_feature} feature_definition<= characterized_object characterized_definition=characterized_object<- property_definition.definition property_definition property_definition=> product_definition_shape<- shape_aspect.of_shape shape_aspect shape_aspect.name = 'compound feature in solid'
feature_description	characterized_object.description	41		compound_feature <= feature_definition <= characterized_object characterized_object.description
feature_name	characterized_object.name	41		compound_feature <= feature_definition <= characterized_object characterized_object.name

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
compound_feature to compound_feature_-element #1: as element except for thread #2: as element for thread only	PATH		11, 22, 23, 26, 27	compound_feature <= feature_definition <= characterized_object characterized_definition=characterized_object<- property_definition.definition property_definition property_definition=> product_definition_shape<- shape_aspect.of_shape shape_aspect {shape_aspect=> composite_shape_aspect} shape_aspect<- shape_aspect_relationship.relating_shape_aspect shape_aspect_relationship {shape_aspect_relationship=> feature_component_relationship} shape_aspect_relationship.related_shape_aspect->shape_aspect=> #1:((instanced_feature) (transition_feature)) #2:(applied_area)
COMPOUND_-FEATURE_ELEMENT #1: as element except for thread #2: as element for thread only	#1: (instanced_feature) (transition_feature) #2(applied_area)	224 224	11, 23, 25, 26, 27	#1: (instanced_feature <= [feature_definition <= characterized_object] [shape_aspect]) (transition_feature <= shape_aspect) #2: (applied_area <= shape_aspect)
compound_feature_-element to machining_-feature (as element)	IDENTICAL MAPPING			

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
compound_feature_-element to transition_-feature (as element)	IDENTICAL MAPPING			
COMPOUND_-FEATURE_-RELATIONSHIP	shape_aspect_relationship	41		{shape_aspect_relationship shape_aspect_relationship.name = 'compound feature ordering'}
compound_feature_-relationship to compound_feature_-element (as predecessor)	PATH		11, 23, 26, 27	shape_aspect_relationship shape_aspect_relationship.related_shape_aspect -> shape_aspect => (instanced_feature) (transition_feature)
compound_feature_-relationship to compound_feature_-element (as successor)	PATH		11, 23, 26, 27	shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => (instanced_feature) (transition_feature)
CONSTANT_RADIUS_-EDGE_ROUND	edge_round	224	23, 26, 27	edge_round <= transition_feature <= shape_aspect {shape_aspect shape_aspect.description = 'constant radius'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
constant_radius_edge_- round to numeric_- parameter (as first_face_offset)	PATH		9, 19, 24	edge_round <= transition_feature <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'first offset'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
constant_radius_edge_- round to numeric_- parameter (as radius)	PATH		9, 19, 24	edge_round <= transition_feature <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
constant_radius_edge_- round to numeric_- parameter (as second_face_offset)	PATH		9, 19, 24	edge_round <= transition_feature <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'second offset'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
CONSTANT_RADIUS_- FILLET	fillet	224	23, 26, 27	fillet <= transition_feature <= shape_aspect {shape_aspect shape_aspect.description = 'constant radius'}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
constant_radius_fillet to numeric_parameter (as first_face_offset)	PATH		9, 19, 24	fillet <= transition_feature <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'first offset'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
constant_radius_fillet to numeric_parameter (as radius)	PATH		9, 19, 24	fillet <= transition_feature <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
constant_radius_fillet to numeric_parameter (as second_face_offset)	PATH		9, 19, 24	fillet <= transition_feature <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'second offset'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
COUNTERBORE_HOLE	composite_hole	224	11, 23, 25	composite_hole <= compound_feature <= feature_definition=> [feature_definition <= characterized_object] instanced_feature <= [{[shape_aspect => composite_shape_aspect] [shape_aspect shape_aspect.description = 'counterbore']} shape_aspect]

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
counterbore_hole to round_hole (as larger_hole)	PATH		22, 23	composite_hole <= compound_feature <= feature_definition <= characterized_object characterized_definition=characterized_object<- property_definition.definition property_definition property_definition=> product_definition_shape<- shape_aspect.of_shape shape_aspect {shape_aspect=> composite_shape_aspect} shape_aspect<- shape_aspect_relationship.relating_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.name = 'large hole']} shape_aspect_relationship shape_aspect_relationship.related_shape_aspect -> shape_aspect => instanced_feature <= feature_definition => round_hole

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
counterbore_hole to round_hole (as smaller_hole)	PATH		22, 23	composite_hole <= compound_feature <= feature_definition <= characterized_object characterized_definition=characterized_object<- property_definition.definition property_definition property_definition=> product_definition_shape<- shape_aspect.of_shape shape_aspect {shape_aspect=> composite_shape_aspect} shape_aspect<- shape_aspect_relationship.relating_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.name = 'small hole']} shape_aspect_relationship shape_aspect_relationship.related_shape_aspect -> shape_aspect => instanced_feature <= feature_definition => round_hole
COUNTERSUNK_HOLE	composite_hole	224	11, 23, 25	composite_hole <= compound_feature <= feature_definition=> [feature_definition <= characterized_object] instanced_feature <= [{[shape_aspect => composite_shape_aspect] [shape_aspect shape_aspect.description = 'countersunk']} shape_aspect]

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
countersunk_hole to round_hole (as constant_diameter_-hole)	PATH		22, 23	composite_hole <= compound_feature <= feature_definition <= characterized_object characterized_definition=characterized_object<- property_definition.definition property_definition property_definition=> product_definition_shape<- shape_aspect.of_shape shape_aspect {shape_aspect=> composite_shape_aspect} shape_aspect<- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.name = 'constant diameter hole']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => instanced_feature <= feature_definition => round_hole

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
countersunk_hole to round_hole (as tapered_hole)	PATH		22, 23	composite_hole <= compound_feature <= feature_definition <= characterized_object characterized_definition=characterized_object<- property_definition.definition property_definition property_definition=> product_definition_shape<- shape_aspect.of_shape shape_aspect {shape_aspect=> composite_shape_aspect} shape_aspect<- shape_aspect_relationship.relating_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.name = 'tapered hole']} shape_aspect_relationship shape_aspect_relationship.related_shape_aspect -> shape_aspect shape_aspect.description= 'change in diameter occurrence' shape_aspect <- shape_aspect_relationship.relating_shape_aspect {shape_aspect_relationship => feature_component_relationship} shape_aspect_relationship.description= 'taper usage' shape_aspect_relationship shape_aspect_relationship.related_shape_aspect -> shape_aspect => taper shape_aspect => instanced_feature <= feature_definition => round_hole

©ISO 2000 - All rights reserve

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
CUTOUT	pocket	224	11, 23, 25	pocket <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object (characterized_object.description = 'circular cutout') (characterized_object.description = 'complex cutout')}
cutout to through_pocket_bottom_condition (as bottom_condition)	PATH		22, 23	pocket <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'bottom condition occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'pocket bottom usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> {shape_aspect (shape_aspect.description = 'through')} shape_aspect => pocket_bottom
DEFINED_MARKING	marking	224	11, 23, 25	marking <= {feature_definition => instanced_feature} feature_definition <= characterized_object

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
defined_marking to descriptive_parameter (as font_name)	PATH		9, 19, 24	marking <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'font name'} representation_item => descriptive_representation_item

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
defined_marking to descriptive_parameter (as special_instructions)	PATH		9, 19, 24	marking <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'special instructions'} representation_item => descriptive_representation_item

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
defined_marking to numeric_parameter (as character_height)	PATH		9, 19, 24	marking <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'character height'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
defined_marking to numeric_parameter (as character_spacing)	PATH		9, 19, 24	marking <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'character spacing'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
DEFINED_THREAD	thread	224	11, 23, 25	thread <= {feature_definition => instanced_feature} feature_definition <= characterized_object

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
defined_thread to numeric_parameter (as crest)	PATH		9, 19, 24	thread <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'crest'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
defined_thread to numeric_parameter (as minor_diameter)	PATH		9, 19, 24	thread <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'minor diameter'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
defined_thread to numeric_parameter (as pitch_diameter)	PATH		9, 19, 24	thread <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'pitch diameter'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
DIAGONAL_KNURL	turned_knurl	224	11, 23, 25	turned_knurl <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'diagonal'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
diagonal_knurl to descriptive_parameter (as helix_hand)	PATH		9, 19, 24	turned_knurl <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'helix hand'} representation_item => descriptive_representation_item

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
diagonal_knurl to numeric_parameter (as helix_angle)	PATH		9, 19, 24	turned_knurl <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'helix angle'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}
DIAMOND_KNURL	turned_knurl	224	11, 23, 25	turned_knurl <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'diamond'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
diamond_knurl to numeric_parameter (as helix_angle)	PATH		9, 19, 24	turned_knurl <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'helix angle'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}
EDGE_ROUND	edge_round	224	23, 26, 27	edge_round <= transition_feature <= shape_aspect

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
edge_round to face_- shape_element (as edge_round_face)	PATH		19	edge_round <= transition_feature <= shape_aspect shape_definition=shape_aspect shape_definition characterized_definition=shape_definition characterized_definition<- property_definition.definition 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.name= 'edge round face'} representation=> shape_representation=> face_shape_representation
edge_round to face_- shape_element (as first_face_shape)	PATH		19	edge_round <= transition_feature <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- {property_definition=> product_definition_shape} property_definition_representation.definition property_definition_representation property_definition_representation.used_representation ->{representation representation.name = 'first face shape'} representation => shape_representation=> face_shape_representation

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
edge_round to face_- shape_element (as second_face_shape)	PATH		19	edge_round <= transition_feature <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- {property_definition=> product_definition_shape} property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'second face shape'} representation => shape_representation=> face_shape_representation
FILLET	fillet	224	23, 26, 27	fillet <= transition_feature <= shape_aspect

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
fillet to face_shape_-element (as fillet_face)	PATH		19	fillet <= transition_feature <= shape_aspect shape_definition=shape_aspect shape_definition characterized_definition=shape_definition characterized_definition<- property_definition.definition 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.name= 'fillet face'} representation=> shape_representation=> face_shape_representation
fillet to face_shape_-element (as first_face_shape)	PATH		19	fillet <= transition_feature <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- {property_definition=> product_definition_shape} property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'first face shape'} representation => shape_representation=> face_shape_representation

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
fillet to face_shape_- element (as second_face_shape)	PATH		19	fillet <= transition_feature <= shape_aspect shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- {property_definition=> product_definition_shape} property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'second face shape'} representation => shape_representation=> face_shape_representation
GENERAL_BOSS	boss	224	11, 23, 25	boss <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'complex'}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
general_boss to angle_- taper (as change_in_boundary)	PATH		22, 23	boss <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'change in boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'taper usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'angle taper'} shape_aspect => taper

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
general_boss to closed_-profile (as enclosed_boundary)	PATH		22, 23	boss <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'enclosed boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => (closed_path_profile) (ngon_closed_profile)
GENERAL_CUTOUT	pocket	224	11, 23, 25	pocket <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'complex cutout'}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
general_cutout to profile (as boundary)	PATH		22,23	pocket <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => (circular_closed_profile) (closed_path_profile) (ngon_closed_profile) (rectangular_closed_profile) (linear_profile) (open_path_profile) (partial_circular_profile) (rounded_u_profile) (square_u_profile) (tee_profile) (vee_profile)

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
GENERAL_OUTSIDE_-PROFILE	outside_profile	224	11, 23, 25	outside_profile <= {feature_definition => instanced_feature} feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'outside boundary'}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
general_outside_profile to profile (as boundary)	PATH		22, 23	outside_profile <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'outside boundary'} shape_aspect => (circular_closed_profile) (closed_path_profile) (ngon_closed_profile) (rectangular_closed_profile) (linear_profile) (open_path_profile) (partial_circular_profile) (rounded_u_profile) (square_u_profile) (tee_profile) (vee_profile)

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
GENERAL_PATTERN	feature_pattern	224	23	feature_pattern <= replicate_feature <= {feature_definition=> instanced_feature} feature_definition<= characterized_object
general_pattern to orientation (as feature_placement)	PATH		9, 19, 24	feature_pattern <= replicate_feature <= feature_definition<= characterized_object characterized_definition=charcterized_object characterized_definition<- property_definition.definition property_definition <- {property_definition=> product_definition_shape} property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'base feature placement'} representation_item => geometric_representation_item => placement

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
GENERAL_POCKET	pocket	224	11, 23, 25	pocket <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'complex'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
general_pocket to profile (as boundary)	PATH		22, 23	pocket <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'boundary'} shape_aspect => (circular_closed_profile) (closed_path_profile) (ngon_closed_profile) (rectangular_closed_profile) (open_path_profile) (partial_circular_profile) (rounded_u_profile) (square_u_profile) (tee_profile) (vee_profile)
GENERAL_REMOVAL_-VOLUME	removal_volume	224	11, 23, 25	removal_volume <= {feature_definition => instanced_feature} feature_definition <= characterized_object

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
general_removal_volume to shape_element (as removal_volume)	PATH		22	removal_volume <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <= product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'shape volume occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'volume shape usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'volume shape'} shape_aspect
GENERAL_-REVOLUTION	revolved_profile	224	11, 23, 25	revolved_profile <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'open profile'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
general_revolution to general_open_profile (as outer_edge_shape)	PATH		22, 23	revolved_profile feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'outer edge shape occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'outer edge shape'} shape_aspect => open_path_profile
GENERAL_SHAPE_-PROFILE	outside_profile	224	11, 23, 25	outside_profile <= {feature_definition => instanced_feature} feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'complex boundary occurrence'}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
general_shape_profile to path (as profile_boundary)	PATH		22,23	outside_profile <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'complex boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {shape_aspect_relationship => shape_defining_relationship shape_aspect_relationship shape_aspect_relationship.description = 'profile usage'} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> {shape_aspect shape_aspect.description = 'boundary'} shape_aspect => (circular_closed_profile) (closed_path_profile) (ngon_closed_profile) (rectangular_closed_profile) (open_path_profile) (partial_circular_profile) (rounded_u_profile) (square_u_profile) (tee_profile) (vee_profile)

©ISO 2000 - All rights reserve

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
GROOVE	revolved_profile	224	11, 23, 25	revolved_profile <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'groove'}
groove to open_profile (as sweep)	PATH		22, 23	revolved_profile <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'sweep occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> {shape_aspect shape_aspect.description = 'sweep'} shape_aspect => (open_path_profile) (partial_circular_profile) (rounded_u_profile) (square_u_profile) (tee_profile) (vee_profile)

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
HOLE	(round_hole) (composite_hole)	224 224	11, 23, 25	(round_hole <= {feature_definition => instanced_feature} feature_definition <= characterized_object) (composite_hole <= compound_feature <= feature_definition=> [feature_definition <= characterized_object] instanced_feature <= [{[shape_aspect => composite_shape_aspect])
KNURL	(turned_knurl) (externally_defined_feature_definition)	224 224	11, 23	(turned_knurl <= {feature_definition => instanced_feature} feature_definition <= characterized_object) (externally_defined_feature_definition <= [externally_defined_item] [{feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'knurl'}])

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
knurl to partial_area_-definition (as partial_profile)	PATH		22, 23	(turned_knurl <=) (externally_defined_feature_definition <=) feature_definition <= characterized_object<- property_definition.definition property_definition=> product_definition_shape shape_aspect.of_shape->* shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'applied area usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->* shape_aspect => applied_area
knurl to shape_element (as applied_shape)	PATH		22	(turned_knurl <=) (externally_defined_feature_definition <=) feature_definition => characterized_object<- property_definition.definition property_definition=> product_definition_shape shape_aspect.of_shape->* shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'applied shape']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->* shape_aspect

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
MACHINING_FEATURE	instanced_feature	224	11, 23, 25	instanced_feature <= [shape_aspect] [feature_definition <= characterized_object]
usage_name	shape_aspect.description	41		instanced_feature <= shape_aspect shape_aspect.description
machining_feature to orientation (as placement)	PATH		9, 19, 24	instanced_feature <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition {property_definition=> product_definition_shape} property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'orientation'} representation_item => geometric_representation_item => placement
MANUFACTURING_-FEATURE	characterized_object	41		

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
MANUFACTURING_- FEATURE_GROUP	group	41		
group_description	group.description			
group_name	group.name			
manufacturing_feature_- group to manufacturing_feature (as feature groups)	PATH			group<- group_assignment.assigned_group group_assignment=> applied_group_assignment applied_group_assignment.items-> group_item (group_item=instanced_feature) (group_item=replicate_feature) (group_item=transition_feature)
manufacturing_feature_- group to manufacturing_feature_- group (as feature groups)	PATH			group<- group_relationship.related_group group_relationship group_relationship.relating_group-> group
MARKING	(marking) (externally_defined_feature_definition)	224 224	11, 23	(marking <= {feature_definition => instanced_feature} feature_definition <= characterized_object) (externally_defined_feature_definition <= [externally_defined_item] [{feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'marking'}])

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
marking to shape_ (as applied_to_shape)	PATH		22	(marking <=) (externally_defined_feature_definition <=) feature_definition => characterized_object<- property_definition.definition property_definition=> product_definition_shape
marking to descriptive_- parameter (as text)	PATH			(marking <=) (externally_defined_feature_definition <=) feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation representation.items[i] -> {representation_item representation_item.name = 'marking text'} representation_item => descriptive_representation_item
MULTI_AXIS_FEATURE	instanced_feature	224		(boss<=) (outside_profile<=) (removal_volume<=) (round_hole<=) (flat_face<=) (pocket<=) (protrusion<=) (rib_top<=) (rounded_end<=) (slot<=) (step<=) instanced_feature

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
maximum_feature_limit	PATH			instanced_feature <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition {property_definition=> product_definition_shape} property_definition <- property_definition_representation.definition property_definition_representation {property_definition_representation => shape_definition_representation} property_definition_representation .used_representation-> representation=> {representation.name='maximum feature limit'} shape_representation=> planar_shape_representation
OUTER_DIAMETER	outer_round	224	11, 23, 25	outer_round <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'outer diameter'}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
outer_diameter to angle_- taper (as reduced_size)	PATH		22, 23	outer_round <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'reduced size occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship [shape_aspect_relationship.name = 'reduced size'] [shape_aspect_relationship.description = 'taper usage']]} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'angle taper'} shape_aspect => taper

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
outer_diameter to diameter_taper (as reduced_size)	PATH		22, 23	outer_round <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'reduced size occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship [shape_aspect_relationship.name = 'reduced size'] [shape_aspect_relationship.description = 'taper usage']]} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> {shape_aspect shape_aspect.description = 'diameter taper'} shape_aspect => taper

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
outer_diameter to directed_taper (as reduced_size)	PATH		22, 23	outer_round <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'reduced size occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship [shape_aspect_relationship.name = 'reduced size'] [shape_aspect_relationship.description = 'taper usage']]} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'directed taper'} shape_aspect => taper

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
outer_diameter to numeric_parameter (as diameter)	PATH		9, 19, 24	outer_round <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'diameter'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
outer_diameter to numeric_parameter (as feature_length)	PATH		9, 19, 24	outer_round <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'length'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
OUTER_DIAMETER_-TO_SHOULDER	outer_round	224	11, 23, 25	outer_round <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'outer diameter to shoulder'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
outer_diameter_to_- shoulder to numeric_- parameter (as diameter)	PATH		9, 19, 24	outer_round <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'diameter'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
outer_diameter_to_- shoulder to vee_profile (as v_shape_boundary)	PATH		22, 23	outer_round <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'v-shape boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'v-shape'} shape_aspect => vee_profile
OUTER_ROUND	outer_round	224	11, 23, 25	outer_round <= {feature_definition => instanced_feature} feature_definition <= characterized_object

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
PARTIAL_CIRCULAR_- SHAPE_PROFILE	outside_profile	224	11, 23, 25	outside_profile <= {feature_definition => instanced_feature} feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'partial circular boundary occurrence'}
partial_circular_shape_- profile to partial_circular_profile (as open_boundary)	PATH		22,23	outside_profile <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'partial circular boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {shape_aspect_relationship => shape_defining_relationship shape_aspect_relationship shape_aspect_relationship.description = 'profile usage'} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => partial_circular_profile

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
PLANAR_FACE	flat_face	224	11, 23, 25	flat_face <= {feature_definition => instanced_feature} feature_definition <= characterized_object
planar_face to direction_-element (as removal_direction)	PATH		19	flat_face <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- {property_definition=> product_definition_shape} property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'removal direction'} representation => shape_representation=> direction_shape_representation

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
planar_face to linear_path (as course_of_travel)	PATH		22, 23	flat_face <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'course of travel occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship [shape_aspect_relationship.name = 'course of travel'] [shape_aspect_relationship.description = 'path feature component usage']]} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'linear'} shape_aspect => path_feature_component

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
planar_face to linear_-profile (as removal_boundary)	PATH		22, 23	flat_face <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'removal boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship [shape_aspect_relationship.name = 'removal boundary'] [shape_aspect_relationship.description = 'profile usage']]} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => linear_profile

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
planar_face to linear_-profile (as face_boundary)	PATH		22, 23	flat_face <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'enclosed boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship [shape_aspect_relationship.name = 'boundary'] [shape_aspect_relationship.description = 'profile usage']]} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => (circular_closed_profile) (ngon_closed_profile) (rectangular_closed_profile) (closed_path_profile)
POCKET	pocket	224	11, 23, 25	pocket <= {feature_definition => instanced_feature} feature_definition <= characterized_object

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
pocket to numeric_parameter (as base_radius)	PATH		9, 19, 24	pocket <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- {property_definition=> product_definition_shape} property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'fillet radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
pocket to angle_taper (as change_in_boundary)	PATH		22, 23	pocket <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'change in boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'taper usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'angle taper'} shape_aspect => taper

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
pocket to directed_taper (as change_in_boundary)	PATH		22, 23	pocket <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'change in boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'taper usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'directed taper'} shape_aspect => taper

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
pocket to linear_path (as pocket_depth)	PATH		22, 23	pocket <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'pocket depth occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship [shape_aspect_relationship.name = 'pocket depth'] [shape_aspect_relationship.description = 'path feature component usage']]} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'linear'} shape_aspect => path_feature_component

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
pocket to pocket_- bottom_condition (as bottom_condition)	PATH		22, 23	pocket <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'bottom condition occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'pocket bottom usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect (shape_aspect.description = 'planar') (shape_aspect.description = 'complex')} shape_aspect => pocket_bottom

©ISO 2000 - All rights reserve

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
pocket to through_- pocket_bottom_condition (as bottom_condition)	PATH		22, 23	pocket <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'bottom condition occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'pocket bottom usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect (shape_aspect.description = 'through')} shape_aspect => pocket_bottom
PROFILE_FEATURE	outside_profile	224	11, 23, 25	(outside_profile <= {feature_definition => instanced_feature} feature_definition <= characterized_object)

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
Profile_feature to linear_profile (as profile_swept_shape)	PATH		22,23	outside_profile<= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect (shape_aspect.description = 'outside boundry'), (shape_aspect.description = 'complex boundary occurrence'), (shape_aspect.description = 'partial circular boundary occurrence'), (shape_aspect.description = 'closed circular boundary occurrence'), (shape_aspect.description = 'open rectangular boundary occurrence'), (shape_aspect.description = 'closed rectangular boundary occurrence')} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship [shape_aspect_relationship.name = 'profile swept shape'] [shape_aspect_relationship.description = 'path feature component usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect shape_aspect.name='linear' shape_aspect => linear_profile
PROTRUSION	protrusion	224	11, 23, 25	protrusion <= {feature_definition => instanced_feature} feature_definition <= characterized_object

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
protrusion to shape_-element (as shape_volume)	PATH		22	protrusion <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'shape volume occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'volume shape usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'volume shape'} shape_aspect
RECESS	pocket	224	11, 23, 25	pocket <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'recess'}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
recess to pocket_bottom_condition (as bottom_condition)	PATH		22, 23	pocket <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'bottom condition occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'pocket bottom usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> {shape_aspect (shape_aspect.description = 'planar') (shape_aspect.description = 'complex')} shape_aspect => pocket_bottom

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
recess to closed_profile (as fillet boundary)	PATH		22, 23	pocket <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => (circular_closed_profile) (closed_path_profile) (ngon_closed_profile) (rectangular_closed_profile) (linear_profile) (open_path_profile) (partial_circular_profile) (rounded_u_profile) (square_u_profile) (tee_profile) (vee_profile)

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
RECTANGULAR_BOSS	boss	224	11, 23, 25	boss <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'rectangular'}
rectangular_boss to angle_taper (as change_in_boundary)	PATH		22, 23	boss <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'change in boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'taper usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'angle taper'} shape_aspect => taper

©ISO 2000 - All rights reserve

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_boss to directed_taper (as change_in_boundary)	PATH		22, 23	boss <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'change in boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'taper usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'directed taper'} shape_aspect => taper

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_boss to directed_taper (as change_in_boundary)	PATH		22, 23	boss <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'change in boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'taper usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'directed taper'} shape_aspect => taper

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_boss to closed_profile (as enclosed_boundary)	PATH		22, 23	boss <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'rectangular profile occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => rectangular_closed_profile
RECTANGULAR_-CLOSED_POCKET	pocket	224	11, 23, 25	pocket <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'closed rectangular'}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_closed_- pocket to rectangular_- closed_profile (as closed_boundary)	PATH		22, 23	pocket <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'closed boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => rectangular_closed_profile
RECTANGULAR_- CLOSED_SHAPE_- PROFILE	outside_profile	224	11, 23, 25	outside_profile <= {feature_definition => instanced_feature} feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'closed rectangular boundary occurrence'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_closed_- shape_profile to rectangular_closed_- profile (as closed_boundary)	PATH		22,23	outside_profile <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'closed rectangular boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {shape_aspect_relationship => shape_defining_relationship shape_aspect_relationship shape_aspect_relationship.description = 'profile usage'} shape_aspect_relationship.relating_shape_aspect -> shape_aspect => rectangular_closed_profile
RECTANGULAR_- OFFSET_PATTERN	shape_aspect	41	22	[shape_aspect <- shape_aspect_relationship.related_shape_aspect shape_aspect_relationship => feature_component_relationship => pattern_offset_membership] [shape_aspect <- shape_aspect_relationship.related_shape_aspect {shape_aspect_relationship => feature_component_relationship} shape_aspect_relationship.relating_shape_aspect-> shape_aspect=> modified_pattern]

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_offset_pattern to direction_element (as offset_direction)	PATH		19	shape_aspect <- shape_definition = shape_aspect shape_definition <- property_definition.definition property_definition <- {property_definition=> product_definition_shape} property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'offset direction'} representation => shape_representation=> direction_shape_representation

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_offset_pattern to numeric_parameter (as column_index)	PATH		9, 19, 24	shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'column index'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit measure_with_unit.value_component -> measure_value measure_value = count_measure count_measure}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_offset_pattern to numeric_parameter (as offset_distance)	PATH		9, 19, 24	shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'offset distance'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_offset_pattern to numeric_parameter (as row_index)	PATH		9, 19, 24	shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'row index'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit measure_with_unit.value_component -> measure_value measure_value = count_measure}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
RECTANGULAR_OMIT_-PATTERN	shape_aspect	41	22	[shape_aspect <- shape_aspect_relationship.related_shape_aspect shape_aspect_relationship => feature_component_relationship => pattern_omit_membership] [shape_aspect <- shape_aspect_relationship.related_shape_aspect {shape_aspect_relationship => feature_component_relationship} shape_aspect_relationship.relating_shape_aspect-> shape_aspect=> modified_pattern]

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_omit_pattern to numeric_parameter (as column_index)	PATH		9, 19, 24	shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'column index'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit measure_with_unit.value_component -> measure_value measure_value = count_measure count_measure}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_omit_pattern to numeric_parameter (as row_index)	PATH		9, 19, 24	shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'row index'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit measure_with_unit.value_component -> measure_value measure_value = count_measure count_measure}
RECTANGULAR_OPEN_-POCKET	pocket	224	11, 23, 25	pocket <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'open rectangular'}

©ISO 2000 - All rights reserve

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_open_pocket to square_u_profile (as open_boundary)	PATH		22, 23	pocket <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'open boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => square_u_profile
RECTANGULAR_OPEN_ SHAPE_PROFILE	outside_profile	224	11, 23, 25	outside_profile <= {feature_definition => instanced_feature} feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'open rectangular boundary occurrence'}

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_open_- shape_profile to square_U_profile (as open_boundary)	PATH		22,23	outside_profile <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'open rectangular boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {shape_aspect_relationship => shape_defining_relationship shape_aspect_relationship shape_aspect_relationship.description = 'profile usage'} shape_aspect_relationship.relating_shape_aspect -> shape_aspect => square_u_profile
RECTANGULAR_- PATTERN	rectangular_pattern	224	23	rectangular_pattern <= replicate_feature <= {feature_definition=> instanced_feature} feature_definition<= characterized_object

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_pattern to direction_element (as column_layout_-direction)	PATH		19	rectangular_pattern <= replicate_feature <= feature_definition<= characterized_object shape_definition = characterized_object shape_definition <- property_definition.definition property_definition <- {property_definition=> product_definition_shape} property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'column layout direction'} representation => shape_representation=> direction_shape_representation
rectangular_pattern to direction_element (as row_layout_direction)	PATH		19	rectangular_pattern <= replicate_feature <= feature_definition<= characterized_object shape_definition = characterized_object shape_definition <- property_definition.definition property_definition <- {property_definition=> product_definition_shape} property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'row layout direction'} representation => shape_representation=> direction_shape_representation

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_pattern to numeric_parameter (as column_spacing)	PATH		9, 19, 24	rectangular_pattern <= replicate_feature <= feature_definition<= characterized_object shape_definition = characterized_object shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'column spacing'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_pattern to numeric_parameter (as columns)	PATH		9, 19, 24	rectangular_pattern <= replicate_feature <= feature_definition<= characterized_object shape_definition = characterized_object shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'number of columns'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit measure_with_unit.value_component -> measure_value measure_value = count_measure count_measure}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_pattern to numeric_parameter (as row_spacing)	PATH		9, 19, 24	rectangular_pattern <= replicate_feature <= feature_definition<= characterized_object shape_definition = characterized_object shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'row spacing'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_pattern to numeric_parameter (as rows)	PATH		9, 19, 24	rectangular_pattern <= replicate_feature <= feature_definition<= characterized_object shape_definition = characterized_object shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'number of rows'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit measure_with_unit.value_component -> measure_value measure_value = count_measure count_measure}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_pattern to rectangular_offset_pattern (as relocated_base_-feature)	PATH		22	rectangular_pattern <= replicate_feature <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape [shape_aspect <- shape_aspect_relationship.relating_shape_aspect {shape_aspect_relationship => feature_component_relationship => pattern_offset_membership} shape_aspect_relationship shape_aspect_relationship.related_shape_aspect -> shape_aspect] [shape_aspect <- shape_aspect_relationship.related_shape_aspect {shape_aspect_relationship => feature_component_relationship} {shape_aspect_relationship.description='base pattern'} shape_aspect_relationship.relating_shape_aspect-> shape_aspect<- {shape_aspect=> modified_pattern} shape_aspect_relationship.relating_shape_aspect {shape_aspect_relationship => feature_component_relationship} {shape_aspect_relationship.description='modified pattern'} shape_aspect_relationship.related_shape_aspect-> shape_aspect]

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rectangular_pattern to rectangular_omit_pattern (as missing_base_feature)	PATH		22	rectangular_pattern <= replicate_feature <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape [shape_aspect <- shape_aspect_relationship.relating_shape_aspect {shape_aspect_relationship => feature_component_relationship => pattern_omit_membership} shape_aspect_relationship shape_aspect_relationship.related_shape_aspect -> shape_aspect] [shape_aspect <- shape_aspect_relationship.related_shape_aspect {shape_aspect_relationship => feature_component_relationship} {shape_aspect_relationship.description='base pattern'} shape_aspect_relationship.relating_shape_aspect-> shape_aspect<- {shape_aspect=> modified_pattern} shape_aspect_relationship.relating_shape_aspect {shape_aspect_relationship => feature_component_relationship} {shape_aspect_relationship.description='modified pattern'} shape_aspect_relationship.related_shape_aspect-> shape_aspect]
REPLICATE_BASE	feature_component_relationship	224 224	11, 22, 23	feature_component_relationship <= shape_aspect_relationship

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
replicate_base to machining_feature (as base_feature)	PATH			feature_component_relationship <= shape_aspect_relationship shape_aspect_relationship.related_shape_aspect -> shape_aspect => instanced_feature
replicate_base to replicate_feature (as base_feature)	PATH			feature_component_relationship <= shape_aspect_relationship shape_aspect_relationship.related_shape_aspect -> shape_aspect => instanced_feature<= feature_definition<= replicate_feature
REPLICATE_FEATURE	replicate_feature	224	23	replicate_feature <= {feature_definition<= instanced_feature} feature_definition<= characterized_object

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
replicate_feature to orientation (as placement)	PATH		9, 19, 24	replicate_feature <= feature_definition=> instanced_feature<= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'orientation'} representation_item => geometric_representation_item => placement
replicate_feature to replicate_base (as replicate_base_-feature)	PATH		11, 22, 23	replicate_feature <= feature_definition=> instanced_feature<= shape_aspect <- shape_aspect_relationship.relating_shape_aspect shape_aspect_relationship => feature_component_relationship {shape_aspect_relationship.name='pattern basis'}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
REVOLVED_FEATURE	revolved_profile	224	11, 23, 25	revolved_profile <= {feature_definition => instanced_feature} feature_definition <= characterized_object
revolved_feature to direction_element (as material_side)	PATH		19	revolved_profile <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- {property_definition=> product_definition_shape} property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'removal direction'} representation => shape_representation=> direction_shape_representation

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
revolved_feature to numeric_parameter (as radius)	PATH		9, 19, 24	revolved_profile <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
REVOLVED_FLAT	revolved_profile	224	11, 23, 25	revolved_profile <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'flat'}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
revolved_flat to linear_-profile (as flat_edge_shape)	PATH		22, 23	revolved_profile <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'flat edge shape occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'flat edge shape'} shape_aspect => linear_profile
REVOLVED_ROUND	revolved_profile	224	11, 23, 25	revolved_profile <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'round'}

©ISO 2000 - All rights reserve

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
revolved_round to partial_circular_profile (as rounded_edge_shape)	PATH		22, 23	revolved_profile <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'rounded edge shape occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'rounded edge shape'} shape_aspect => partial_circular_profile
RIP_TOP	rib_top		11, 23, 25	rib_top <= {feature_definition => instanced_feature} feature_definition <= characterized_object

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rib_top to direction_element (as removal_direction)	PATH		19	ribtop <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- {property_definition=> product_definition_shape} property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation representation.name= 'removal direction'} representation => shape_representation => direction_shape_representation

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rib_top to rib_top_floor (as floor_condition)			22,23	ribtop <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'rib top condition occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'ribtop usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect (shape_aspect.description = 'planar') (shape_aspect.description = 'complex')} shape_aspect => ribtop_floor
ROUND_HOLE	round_hole	224	11, 23, 25	round_hole <= {feature_definition => instanced_feature} feature_definition <= characterized_object

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
round_hole to angle_taper (as change_in_diameter)	PATH		22, 23	round_hole <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'change in diameter occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'taper usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'angle taper'} shape_aspect => taper

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
round_hole to blind_- bottom_condition (as bottom_condition)	PATH		22, 23	round_hole <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'bottom condition occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'hole bottom usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> {shape_aspect (shape_aspect.description = 'conical') (shape_aspect.description = 'flat') (shape_aspect.description = 'flat with radius') (shape_aspect.description = 'flat with taper') (shape_aspect.description = 'spherical')} shape_aspect => hole_bottom

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
round_hole to circular_- closed_profile (as diameter)	PATH		22, 23	round_hole <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'diameter occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship [shape_aspect_relationship.name = 'diameter'] [shape_aspect_relationship.description = 'profile usage']]} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => circular_closed_profile

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
round_hole to diameter_- taper (as change_in_diameter)	PATH		22, 23	round_hole <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'change in diameter occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'taper usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'diameter taper'} shape_aspect => taper

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
round_hole to directed_- taper (as change_in_diameter)	PATH		22, 23	round_hole <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'change in diameter occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'taper usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'directed taper'} shape_aspect => taper

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
round_hole to linear_path (as hole_depth)	PATH		22, 23	round_hole <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'hole depth occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship [shape_aspect_relationship.name = 'hole depth'] [shape_aspect_relationship.description = 'path feature component usage']]} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'linear'} shape_aspect => path_feature_component

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
round_hole to through_- bottom_condition (as bottom_condition)	PATH		22, 23	round_hole <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'bottom condition occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'hole bottom usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'through'} shape_aspect => hole_bottom
ROUNDED_END	rounded_end	224	11, 23, 25	rounded_end <= {feature_definition => instanced_feature} feature_definition <= characterized_object

©ISO 2000 - All rights reserve

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rounded_end to linear_- path (as course_of_travel)	PATH		22, 23	rounded_end <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'course of travel occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship [shape_aspect_relationship.description = 'path feature component usage']]} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'linear'} shape_aspect => path_feature_component

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
rounded_end to partial_-circular_profile (as partial_circular_-boundary)	PATH		22, 23	rounded_end <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'partial circular boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship [shape_aspect_relationship.description = 'profile usage']]} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => partial_circular_profile
SHAPE_PROFILE	outside_profile	224	11, 23, 25	outside_profile <= {feature_definition => instanced_feature} feature_definition <= characterized_object

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
shape_profile to profile_floor (as floor_condition)	PATH		22,23	outside_profile <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect (shape_aspect.description = 'complex boundary occurrence') (shape_aspect.description = 'partial circular boundary occurrence') (shape_aspect.description = 'closed circular boundary occurrence') (shape_aspect.description = 'open rectangular boundary occurrence') (shape_aspect.description = 'closed rectangular boundary occurrence')} shape_aspect <- shape_aspect_relationship.related_shape_aspect {shape_aspect_relationship => feature_component_relationship shape_aspect_relationship shape_aspect_relationship.description = 'profile floor usage'} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> {shape_aspect (shape_aspect.description = 'planar') (shape_aspect.description = 'complex')} shape_aspect => profile_floor

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
shape_profile to through_profile_floor (as floor_condition)	PATH		22,23	outside_profile <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect (shape_aspect.description = 'complex boundary occurrence'), (shape_aspect.description = 'partial circular boundary occurrence'), (shape_aspect.description = 'closed circular boundary occurrence'), (shape_aspect.description = 'open rectangular boundary occurrence'), (shape_aspect.description = 'closed rectangular boundary occurrence')} shape_aspect <- shape_aspect_relationship.related_shape_aspect {shape_aspect_relationship => feature_component_relationship shape_aspect_relationship shape_aspect_relationship.description = 'profile floor usage'} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> {shape_aspect (shape_aspect.description = 'through')} shape_aspect => profile_floor

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
shape_profile to direction_element (as removal_direction)	PATH		19	outside_profile <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- {property_definition=> product_definition_shape} {product_definition_shape <- shape_aspect.of_shape shape_aspect (shape_aspect.description = 'complex boundary occurrence'), (shape_aspect.description = 'partial circular boundary occurrence'), (shape_aspect.description = 'closed circular boundary occurrence'), (shape_aspect.description = 'open rectangular boundary occurrence'), (shape_aspect.description = 'closed rectangular boundary occurrence')} property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> {representation representation.name = 'removal direction'} representation => shape_representation=> direction_shape_representation
SLOT	slot	224	11, 23, 25	slot <= {feature_definition => instanced_feature} feature_definition <= characterized_object

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
slot to open_profile (as sweep_shape)	PATH		22, 23	slot <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'swept shape occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => (open_path_profile) (partial_circular_profile) (rounded_u_profile) (square_u_profile) (tee_profile) (vee_profile)

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
slot to path (as course_of_travel)	PATH		22, 23	slot <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'course of travel occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship [shape_aspect_relationship.name = 'course of travel'] [shape_aspect_relationship.description = 'path feature component usage']]} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => path_feature_component

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
slot to slot_end_type (as end_conditions)	PATH		22, 23	slot <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'end condition occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => feature_component_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'slot end usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => slot_end
SPHERICAL_CAP	spherical_cap	224	11, 23, 25	spherical_cap <= {feature_definition => instanced_feature} feature_definition <= characterized_object

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
spherical_cap to numeric_parameter (as internal_angle)	PATH		9, 19, 24	spherical_cap <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'internal angle'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
spherical_cap to numeric_parameter (as radius)	PATH		9, 19, 24	spherical_cap <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
STEP	step	224	11, 23, 25	step <= {feature_definition => instanced_feature} feature_definition <= characterized_object

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
step to linear_path (as course_of_travel)	PATH		22, 23	step <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'course of travel occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship [shape_aspect_relationship.description = 'path feature component usage']]} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->{shape_aspect shape_aspect.description = 'linear'} shape_aspect => path_feature_component

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
step to vee_profile (as removal_boundary)	PATH		22, 23	step <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition => product_definition_shape <- shape_aspect.of_shape {shape_aspect shape_aspect.description = 'removal boundary occurrence'} shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'profile usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => vee_profile
STRAIGHT_KNURL	turned_knurl	224	11, 23, 25	turned_knurl <= {feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'straight'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
THREAD	(thread) (externally_defined_feature_definition)	224 224	11, 23	(thread <= {feature_definition => instanced_feature} feature_definition <= characterized_object) (externally_defined_feature_definition <= [externally_defined_item] [{feature_definition => instanced_feature} feature_definition <= characterized_object {characterized_object characterized_object.description = 'thread'}])
thread to descriptive_- parameter (as fit_class)	PATH		9,19, 24	(thread <=) (externally_defined_feature_definition <=) feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation ->{representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] ->{representation_item representation_item.name = 'fit class'} representation_item => descriptive_representation_item

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
thread to descriptive_- parameter (as form)	PATH		9,19, 24	(thread <=) (externally_defined_feature_definition <=) feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'form'} representation_item => descriptive_representation_item

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
thread to descriptive_- parameter (as qualifier)	PATH		9, 19, 24	(thread <=) (externally_defined_feature_definition <=) feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'qualifier'} representation_item => descriptive_representation_item

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
inner_or_outer_thread	descriptive_representation_- item.description	45	9, 19, 24	(thread <=) (externally_defined_feature_definition <=) feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'thread side'} representation_item => descriptive_representation_item descriptive_representation_item.description {(descriptive_representation_item.description = 'internal') (descriptive_representation_item.description = 'external')}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
thread to numeric_parameter (as major_diameter)	PATH		9,19, 24	(thread <=) (externally_defined_feature_definition <=) feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'major diameter'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
thread to numeric_parameter (as number_of_threads)	PATH		9,19, 24	(thread <=) (externally_defined_feature_definition <=) feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'number of threads'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => ratio_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
thread to partial_area_-definition (as partial_profile)	PATH		22, 23	(thread <=) (externally_defined_feature_definition <=) feature_definition <= characterized_object<- property_definition.definition property_definition=> product_definition_shape shape_aspect.of_shape->: shape_aspect <- shape_aspect_relationship.related_shape_aspect {[shape_aspect_relationship => shape_defining_relationship] [shape_aspect_relationship shape_aspect_relationship.description = 'applied area usage']} shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect ->: shape_aspect => applied_area
thread to shape_ (as applied_shape)	PATH		22	(thread <=) (externally_defined_feature_definition <=) feature_definition => characterized_object<- property_definition.definition property_definition=> product_definition_shape

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
thread to descriptive_- parameter (as thread_hand)	PATH		9,19, 24	(thread <=) (externally_defined_feature_definition <=) feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'hand'} {representation_item representation_item.description = 'left' representation_item.description = 'right'} representation_item => descriptive_representation_item
TRANSITION_FEATURE	transition_feature	224	23, 26, 27	transition_feature <= shape_aspect
TURNED_KNURL	turned_knurl	224	11, 23, 25	turned_knurl <= {feature_definition => instanced_feature} feature_definition <= characterized_object

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
turned_knurl to numeric_-parameter (as diametral_pitch)	PATH		9, 19, 24	turned_knurl <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'diametral pitch'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => plane_angle_measure_with_unit}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
turned_knurl to numeric_-parameter (as number_of_teeth)	PATH		9, 19, 24	turned_knurl <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'number of teeth'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit measure_with_unit.value_component -> measure_value measure_value = count_measure count_measure}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
turned_knurl to numeric_-parameter (as major_diameter)	PATH		9, 19, 24	turned_knurl <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'major diameter'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
turned_knurl to numeric_-parameter (as nominal_diameter)	PATH		9, 19, 24	turned_knurl <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'nominal diameter'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 4 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
turned_knurl to numeric_-parameter (as root_fillet)	PATH		9, 19, 24	turned_knurl <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'root fillet'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 4 - Mapping table for manufacturing_part_properties UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
turned_knurl to numeric_-parameter (as tooth_depth)	PATH		9, 19, 24	turned_knurl <= feature_definition <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'tooth depth'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 5 - Mapping table for manufacturing_part_properties UoF

Application element	AIM element	Source	Rules	Reference path
ALTERNATE_-MATERIAL	product_definition	41	12	
alternate_ranking	make_from_usage_option.ranking	44		product_definition <- product_definition_relationship.related_product_definition product_definition_relationship => product_definition_usage => make_from_usage_option make_from_usage_option.ranking
alternate_material to material (as material_substitute)	PATH			product_definition <- product_definition_relationship.related_product_definition {product_definition_relationship => product_definition_usage => make_from_usage_option} product_definition_relationship product_definition_relationship.relating_product_definition ->product_definition
DESCRIPTIVE_-PARAMETER	descriptive_representation_item	45		
descriptive_string	descriptive_representation_-item.description	45		
HARDNESS	material_property_representation	45		{material_property_representation material_property_representation.dependent_environment->data_environment}

Table 5 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
high_value	measure_representation_item	45		material_property_representation<= property_definition_representation property_definition_representation.used_representation->representation representation.items[1] ->{representation_item representation_item.name = 'high value'} {representation_item representation_item=> qualified_representation_item qualified_representation_item.qualifiers->type_qualifier type_qualifier.name='high value'} representation_item => measure_representation_item
low_value	measure_representation_item	45		material_property_representation<= property_definition_representation property_definition_representation.used_representation->representation representation.items[1] ->{representation_item representation_item.name = 'low value'} {representation_item representation_item=> qualified_representation_item qualified_representation_item.qualifiers->type_qualifier type_qualifier.name='low value'} representation_item => measure_representation_item

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 5 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
nominal	measure_representation_item	45		material_property_representation<= property_definition_representation property_definition_representation.used_representation->representation representation.items[1] -> {representation_item representation_item.name = 'nominal'} {representation_item representation_item=> qualified_representation_item qualified_representation_item.qualifiers->type_qualifier type_qualifier.name='nominal'} representation_item => measure_representation_item
scale	data_environment.name	41		material_property_representation material_property_representation.dependent_environment->data_environment data_environment.name = 'hardness'
MATERIAL	product_definition	41		
material_description	product_definition.description	41		
material_id	material_designation.name	45		product_definition characterized_product_definition = product_definition characterized_product_definition characterized_definition = characterized_product_definition characterized_definition <- material_designation.definitions[i] material_designation material_designation.name
stock_size	product_definition.id	41		

Table 5 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
material to material_property (as material_-characteristics)	PATH			product_definition characterized_product_definition = product_definition characterized_product_definition characterized_definition = characterized_product_definition characterized_definition <- property_definition.definition property_definition=> material_property
material to specification (as material_-specification)	PATH			product_definition => product_definition_with_associated_documents product_definition_with_associated_documents.documentation_ids[i] -> document
MATERIAL_PROPERTY	material_property	45		
material_property to hardness (as material_hardness)	PATH			material_property <= property_definition <- property_definition_representation.definition property_definition_representation=> material_property_representation
material_property to property_parameter (as property_-characteristic)	PATH			material_property <= property_definition <- property_definition_representation.definition {property_definition_representation=> material_property_representation} property_definition_representation property_definition_representation.used_representation -> representation representation.items[i] -> representation_item
NUMERIC_PARAMETER	measure_representation_item	45		

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 5 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
parameter_units	named_unit	41	7	measure_representation_item <= measure_with_unit measure_with_unit.unit_component -> unit unit = named_unit named_unit
parameter_value	measure_value	41		measure_representation_item <= measure_with_unit measure_with_unit.value_component -> measure_value
NUMERIC_- PARAMETER_WITH_- TOLERANCE	[measure_representation_item] [qualified_representation_item]	45 45		
numeric_parameter_- with_tolerance to plus_- minus_value (as implicit_tolerance)	IDENTICAL MAPPING			
numeric_parameter_- with_tolerance to tolerance_limit (as implicit_tolerance)	PATH			qualified_representation_item qualified_representation_item.qualifiers[i] -> value_qualifier value_qualifier = type_qualifier type_qualifier
PART_PROPERTY	property_definition	41		{property_definition property_definition.name = 'part property'}
part_property to property_parameter (as property_- characteristic)	PATH			property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation representation.items[i] -> representation_item
PROCESS_PROPERTY	property_definition	41		{property_definition property_definition.name = 'process property'}

Table 5 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
process_name	representation.name	43		property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation representation.name
process_property to property_parameter (as property_-characteristic)	PATH			property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation representation.items[i] -> representation_item
PROPERTY	property_definition	41		
property_name	property_definition.name	41		
property to material_-property (as material_-characteristic)	PATH			property_definition <- property_definition_relationship.relating_property_definition property_definition_relationship property_definition_relationship.related_property_definition -> property_definition => material_property
property to part_property (as part_property_-characteristic)	PATH			property_definition <- property_definition_relationship.relating_property_definition property_definition_relationship property_definition_relationship.related_property_definition -> property_definition {property_definition property_definition.name = 'part property'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 5 - Mapping table for manufacturing_part_properties UoF (continued)

Application element	AIM element	Source	Rules	Reference path
property to process_- property (as process_- characteristic)	PATH			property_definition <- property_definition_relationship.relating_property_definition property_definition_relationship property_definition_relationship.related_property_definition -> property_definition {property_definition property_definition.name = 'process property'}
property to shape_aspect (as property_- characteristic)	PATH			property_definition property_definition.definition -> characterized_definition characterized_definition = shape_definition shape_definition shape_definition = shape_aspect shape_aspect
property to specification (as property_description)	PATH			property_definition document_reference_item = property_definition document_reference_item <- applied_document_reference.items[i] applied_document_reference <= document_reference document_reference.assigned_document -> document
property to surface_- property (as surface_characteristic)	PATH			property_definition <- property_definition_relationship.relating_property_definition property_definition_relationship property_definition_relationship.related_property_definition -> property_definition {property_definition property_definition.name = 'surface property'}
PROPERTY_- PARAMETER	representation_item	43		
parameter_name	representation_item.name	43		
SURFACE_PROPERTY	property_definition	41		{property_definition property_definition.name = 'surface property'}

Table 5 - Mapping table for manufacturing_part_properties UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
surface_finish	property_definition.description	41		{property_definition property_definition.description = 'surface finish'}
surface_property to property_parameter (as property_-characteristic)	PATH			property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation ->representation representation.items[i] ->representation_item

©ISO 2000 - All rights reserve

Table 6 - Mapping table for manufacturing_process_control_documentation UoF

Application element	AIM element	Source	Rules	Reference path
CUSTOMER_ORDER	directed_action	41	18	{directed_action <= executed_action <= action action.name = 'customer order'}
delivery_date	date	41	6	directed_action feature_based_pp_dated_item = directed_action feature_based_pp_dated_item <- feature_based_pp_date_assignment.items[i] feature_based_pp_date_assignment <= {date_assignment date_assignment.role -> date_role date_role.name = 'delivery date'} date_assignment date_assignment.assigned_date -> date
material_disposition	action.description	41		directed_action <= executed_action <= action action.description
order_number	action_directive.name	41		directed_action directed_action.directive -> action_directive action_directive.name
order_status	action_status	41	4	directed_action <= executed_action <= action <- action_status.assigned_action action_status
special_instructions	action_directive.comment	41		directed_action directed_action.directive -> action_directive action_directive.comment

©ISO 2000 - All rights reserved

Table 6 - Mapping table for manufacturing_process_control_documentation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
customer_order to ordered_part (as quantity_ordered)	PATH			directed_action <= executed_action <= action <- action_assignment.assigned_action action_assignment => ordered_part
customer_order to person_in_organization (as customer)	PATH			directed_action directed_action.directive -> action_directive feature_based_pp_person_and_organization_item = action_directive feature_based_pp_person_and_organization_item <- feature_based_pp_person_and_organization_assignment.items[i] feature_based_pp_person_and_organization_assignment <= person_and_organization_assignment person_and_organization_assignment.assigned_person_and_organization -> person_and_organization
customer_order to project_order (as initiated_order)	PATH		18	directed_action <= executed_action <= action <- action_relationship.relating_action action_relationship action_relationship.related_action -> {action action.name = 'project order'} action => executed_action => directed_action
DIGITAL_TECHNICAL_-DATA_PACKAGE_-WORK_ORDER	directed_action	41	18	{directed_action <= executed_action <= action action.name = 'digital technical data package work order'}
order_id	action_directive.name	41		directed_action directed_action.directive -> action_directive action_directive.name

©ISO 2000 - All rights reserve

©ISO

Table 6 - Mapping table for manufacturing_process_control_documentation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
ORDERED_PART	ordered_part	224	13, 25	ordered_part <= [action_assignment] [characterized_object]
quantity_required	measure_with_unit.value_component	41		ordered_part <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation representation.items[i] -> representation_item => measure_representation_item <= measure_with_unit measure_with_unit.value_component {measure_with_unit.value_component -> measure_value measure_value = count_measure count_measure}

©ISO 2000 - All rights reserved

Table 6 - Mapping table for manufacturing_process_control_documentation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
quantity_unit_of_-measure	measure_with_unit.unit_component	41		ordered_part <= characterized_object characterized_definition = characterized_object characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation representation.items[i] -> representation_item => measure_representation_item <= measure_with_unit measure_with_unit.unit_component
PEDIGREE_CREATION_-ORDER	directed_action	41	18	{directed_action <= executed_action <= action action.name = 'pedigree creation order'}
order_id	action_directive.name	41		directed_action directed_action.directive -> action_directive action_directive.name
PROJECT_ORDER	directed_action	41	13, 18	{directed_action <= executed_action <= action action.name = 'project order'}
project_order_id	action_directive.name	41		directed_action directed_action.directive -> action_directive action_directive.name

©ISO 2000 - All rights reserve

©ISO

Table 6 - Mapping table for manufacturing_process_control_documentation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
project_order to approval (as release_authorization)	PATH		17	directed_action feature_based_pp_approved_item = directed_action feature_based_pp_approved_item <- feature_based_pp_approval_assignment.items[i] feature_based_pp_approval_assignment <= approval_assignment approval_assignment.assigned_approval -> approval
project_order to digital_-technical_data_package-work_order (as technical_data_-package_status)	PATH		18	directed_action <= executed_action <= action <- action_relationship.relating_action action_relationship action_relationship.related_action -> {action action.name = 'digital technical data package work order'} action => executed_action => directed_action
project_order to pedigree_creation_order (as pedigree_creation_-status)	PATH		18	directed_action <= executed_action <= action <- action_relationship.relating_action action_relationship action_relationship.related_action -> {action action.name = 'pedigree creation order'} action => executed_action => directed_action

©ISO 2000 - All rights reserved

Table 6 - Mapping table for manufacturing_process_control_documentation UoF (continued)

Application element	AIM element	Source	Rules	Reference path
project_order to part (as part_status)	PATH		13	directed_action <= executed_action <= action <- action_assignment.assigned_action action_assignment => feature_based_pp_action_assignment feature_based_pp_action_assignment.items[i] -> feature_based_pp_action_item feature_based_pp_action_item = product_definition_formation product_definition_formation
project_order to requisition (as ordered_resource)	PATH			directed_action feature_based_pp_document_item = directed_action feature_based_pp_document_item <- feature_based_pp_document_reference.items[i] feature_based_pp_document_reference <= document_reference document_reference.assigned_document -> document
project_order to resource_acquisition_- order (as resource_- acquisition_status)	PATH		18	directed_action <= executed_action <= action <- action_relationship.relating_action action_relationship action_relationship.related_action -> {action action.name = 'resource acquisition order'} action => executed_action => directed_action

©ISO 2000 - All rights reserve

©ISO

©ISO 2000 - All rights reserved

Table 6 - Mapping table for manufacturing_process_control_documentation UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
project_order to shop_- work_order (as shop_work_status)	PATH		18	directed_action <= executed_action <= action <- action_relationship.relating_action action_relationship action_relationship.related_action ->{action action.name = 'shop work order'} action => executed_action => directed_action
RESOURCE_- ACQUISITION_ORDER	directed_action	41	18	{directed_action <= executed_action <= action action.name = 'resource acquisition order'}
order_id	action_directive.name	41		directed_action directed_action.directive ->action_directive action_directive.name
SHOP_WORK_ORDER	directed_action	41	18	{directed_action <= executed_action <= action action.name = 'shop work order'}
order_id	action_directive.name	41		directed_action directed_action.directive ->action_directive action_directive.name

Table 7 - Mapping table for manufacturing_process_requirement_documents UoF

Application element	AIM element	Source	Rules	Reference path
SPECIFICATION	document	41		
specification_class	document_with_class.class	41		document=> document_with_class
specification_description	document.description	41		document document.description
specification_id	document.id	41		document document.id
specification to specification_usage_-constraint (as constraint)	PATH			document <- document_usage_constraint.source document_usage_constraint
SPECIFICATION_-USAGE_CONSTRAINT	document_usage_constraint	41		
class_id	document_usage_constraint.-subject_element_value	41		
element	document_usage_constraint.-subject_element	41		

©ISO 2000 - All rights reserve

©ISO

©ISO 2000 - All rights reserved

Table 8 - Mapping table for measurement_limitations UoF

Application element	AIM element	Source	Rules	Reference path
ANGULAR_- DIMENSION_- TOLERANCE	angular_location	47	22	
major_angle	angular_location.angle_selection	47		
mirror_origin #1: The origin element is not mirrored. #2: The origin element is mirrored.	#1: (shape_aspect) #2: (shape_aspect_relationship)	41 41		angular_location <= dimensional_location <= shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> #1: (shape_aspect) #2: (shape_aspect <- shape_aspect_relationship.relating_shape_aspect shape_aspect_relationship {shape_aspect_relationship shape_aspect_relationship.name = 'mirroring relationship'})
mirror_termination #1: The termination element is not mirrored. #2: The termination element is mirrored.	#1: (shape_aspect) #2: (shape_aspect_relationship)	41 41		angular_location <= dimensional_location <= shape_aspect_relationship shape_aspect_relationship.related_shape_aspect -> #1: (shape_aspect) #2: (shape_aspect <- shape_aspect_relationship.relating_shape_aspect shape_aspect_relationship {shape_aspect_relationship shape_aspect_relationship.name = 'mirroring relationship'})
ANGULAR_SIZE_- DIMENSION_- TOLERANCE	angular_size	47		

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
full_or_half	representation_item.name	43	9, 19, 24	angular_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 representation_item.name {(representation_item.name = 'full angle') (representation_item.name = 'half angle')}
major_angle	angular_size.angle_selection	47		
ANGULARITY_-TOLERANCE	angularity_tolerance	519	10	
segment_size	[measure_with_unit.value_component] [measure_with_unit.unit_component]	41 41		angularity_tolerance <= geometric_tolerance_with_datum_reference <= geometric_tolerance => geometric_tolerance_with_defined_unit geometric_tolerance_with_defined_unit.unit_size -> measure_with_unit [measure_with_unit.value_component] [measure_with_unit.unit_component]
angularity_tolerance to datum (as geometric_reference)	PATH			angularity_tolerance <= geometric_tolerance_with_datum_reference geometric_tolerance_with_datum_reference.datum_system[i] -> datum_reference
CIRCULAR_RUNOUT_-TOLERANCE	circular_runout_tolerance	519	10	

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
runout_angle	[measure_with_unit.value_component] [measure_with_unit.unit_component]	41 41	23	circular_runout_tolerance <= geometric_tolerance_with_datum_reference <= geometric_tolerance <- tolerance_zone.defining_tolerance[i] tolerance_zone <- tolerance_zone_definition.zone tolerance_zone_definition => runout_zone_definition runout_zone_definition.orientation -> runout_zone_orientation runout_zone_orientation.angle -> measure_with_unit [measure_with_unit.value_component] [measure_with_unit.unit_component]
circular_runout_tolerance to datum (as geometric_reference)	PATH			circular_runout_tolerance <= geometric_tolerance_with_datum_reference geometric_tolerance_with_datum_reference.datum_system[i] -> datum_reference
CIRCULARITY_-TOLERANCE	roundness_tolerance	519	10	
COMPOUND_DATUM	datum (see NOTE)	47	23	
compound_datum to datum_feature (as element)	PATH			datum <= shape_aspect <- shape_aspect_relationship.related _shape_aspect shape_aspect_relationship shape_aspect_relationship.relatingshape_aspect -> shape_aspect => datum_feature
CONCENTRICITY_-TOLERANCE	concentricity_tolerance	519	10	

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
value_qualifier	tolerance_zone_form.name	47		concentricity_tolerance <= geometric_tolerance_with_datum_reference <= geometric_tolerance <- tolerance_zone.defining_tolerance[i] tolerance_zone tolerance_zone.form -> tolerance_zone_form tolerance_zone_form.name
concentricity_tolerance to datum (as geometric_reference)	PATH			concentricity_tolerance <= geometric_tolerance_with_datum_reference geometric_tolerance_with_datum_reference.datum_system[i] -> datum_reference
CURVED_DIMENSION_-TOLERANCE	dimensional_size	47		{dimensional_size dimensional_size.name = 'curve length'}
CYLINDRICITY_-TOLERANCE	cylindricity_tolerance	519		
DATUM	datum_reference	47		
name	shape_aspect.name	41	23	datum_reference datum_reference.referenced_datum -> datum <= shape_aspect shape_aspect.name
precedence	datum_reference.precedence	47		
DATUM_FEATURE	(datum) ([datum] [datum_feature])	47 47 47	23	{(datum) (datum <= shape_aspect <- shape_aspect_relationship.related_shape_aspect shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => datum_feature)}
datum_feature to datum_-target_set (as datum_representation)	PATH			datum datum.established_by_relationships

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
datum_feature to material_condition_-modifier (as modifier)	PATH			datum <- datum_reference.referenced_datum datum_reference => referenced_modified_datum referenced_modified_datum.modifier -> limit_condition
datum_feature to shape_-element (as datum_representation)	IDENTICAL MAPPING			
DATUM_TARGET	datum_target	47	23	
identifier	datum_target.target_id	47		
DATUM_TARGET_SET	datum.established_by_relationships	47	23	{datum.established_by_relationships[i] -> shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => datum_target}
rule_description	shape_aspect_relationship.description	41		datum.established_by_relationships[i] -> shape_aspect_relationship shape_aspect_relationship.description
datum_target_set to datum_target (as target_shape)	PATH			datum.established_by_relationships datum <= shape_aspect<- shape_aspect_relationship.related_shape_aspect shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect => datum_target
DIAMETER_-DIMENSION_-TOLERANCE	dimensional_size	47		{dimensional_size dimensional_size.name = 'diameter'}
DIMENSIONAL_-TOLERANCE	shape_dimension_representation	47		

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
dimension_description	(dimensional_size.name) (shape_aspect_relationship.description)	47 41	22	shape_dimension_representation <- dimensional_characteristic_representation.representation dimensional_characteristic_representation dimensional_characteristic_representation.dimension -> dimensional_characteristic (dimensional_characteristic = dimensional_size dimensional_size dimensional_size.name) (dimensional_characteristic = dimensional_location dimensional_location <= shape_aspect_relationship shape_aspect_relationship.description)
dimension_value	measure_representation_item	45	9, 19, 24	shape_dimension_representation <= shape_representation <= representation representation.items[i] -> representation_item => measure_representation_item
significant_digits	precision_qualifier.precision_value	45	9, 19, 24	shape_dimension_representation <= shape_representation <= representation representation.items[i] -> {representation_item => measure_representation_item} representation_item => qualified_representation_item qualified_representation_item.qualifiers[1] -> value_qualifier value_qualifier = precision_qualifier precision_qualifier precision_qualifier.precision_value

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
unit_of_measure	unit		9, 19, 24	shape_dimension_representation <= shape_representation <= representation representation.items[i] -> representation_item => measure_representation_item <= measure_with_unit measure_with_unit.unit_component -> unit
dimensional_tolerance to tolerance_value (as limit)	PATH			shape_dimension_representation <- dimensional_characteristic_representation.representation dimensional_characteristic_representation
DISTANCE_ALONG_-CURVE_TOLERANCE	dimensional_location_with_path	47	22	
with_curve_direction	dimensional_location_with_path.path	47		
distance_along_curve_-tolerance to shape_aspect (as path)	PATH			dimensional_location_with_path dimensional_location_with_path.path -> shape_aspect
FLATNESS_-TOLERANCE #1: if segment size is specified #2: if segment_size is not specified	#1: (geometric_tolerance_with_defined_-unit) #2: (geometric_tolerance)	519 519		flatness_tolerance<= {#1: (geometric_tolerance_with_defined_unit <= geometric_tolerance) #2: (geometric_tolerance)}
segment_size	[measure_with_unit.value_component] [measure_with_unit.unit_component]	41 41		#1: geometric_tolerance_with_defined_unit geometric_tolerance_with_defined_unit.unit_size -> measure_with_unit [measure_with_unit.value_component] [measure_with_unit.unit_component]
GEOMETRIC_-TOLERANCE	geometric_tolerance	47	22	
geometric_tolerance_-value	measure_with_unit.value_component	41		geometric_tolerance geometric_tolerance.magnitude -> measure_with_unit measure_with_unit.value_component

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
significant_digits	precision_qualifier.precision_value	45		geometric_tolerance geometric_tolerance.magnitude -> measure_with_unit <- measure_qualification.qualifiers[1] -> value_qualifier value_qualifier = precision_qualifier precision_qualifier precision_qualifier.precision_value
unit_of_measure	unit	41		geometric_tolerance geometric_tolerance.magnitude -> measure_with_unit measure_with_unit.unit_component -> unit
geometric_tolerance to material_condition_-modifier (as modifier_control)	PATH			geometric_tolerance => modified_geometric_tolerance modified_geometric_tolerance.modifier -> limit_condition
geometric_tolerance to shape_aspect (as applied_shape)	PATH			geometric_tolerance geometric_tolerance.toleranced_shape_aspect -> shape_aspect
geometric_tolerance to tolerance_zone (as zone_definition)	PATH			geometric_tolerance <- tolerance_zone.defining_tolerance[i] tolerance_zone
GEOMETRIC_-TOLERANCE_-PRECEDENCE_-RELATIONSHIP	geometric_tolerance_relationship	47		{geometric_tolerance_relationship geometric_tolerance_relationship.name = 'precedence'}
geometric_tolerance_-precedence_relationship to geometric_tolerance (as base_shape_tolerance)	PATH			geometric_tolerance_precedence_relationship <= shape_aspect_relationship shape_aspect_relationship.related_shape_aspect -> shape_aspect <- geometric_tolerance.toleranced_shape_aspect geometric_tolerance

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
geometric_tolerance_-precedence_relationship to geometric_tolerance (as pattern_shape_-tolerance)	PATH			geometric_tolerance_precedence_relationship <= shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect <- geometric_tolerance.toleranced_shape_aspect geometric_tolerance
LIMITS_AND_FITS	limits_and_fits	47		
deviation	limits_and_fits.form_variance	47		
fitting_type	limits_and_fits.zone_variance	47		
grade	limits_and_fits.grade	47		
LINEAR_PROFILE_-TOLERANCE	line_profile_tolerance	519	10	
linear_profile_tolerance to datum (as geometric_reference)	PATH			line_profile_tolerance <= geometric_tolerance => geometric_tolerance_with_datum_reference geometric_tolerance_with_datum_reference.datum_system[i] -> datum_reference

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
linear_profile_tolerance to orientation (as affected_plane)	PATH			line_profile_tolerance <= geometric_tolerance <- tolerance_zone.defining_tolerance[i] tolerance_zone <= shape_aspect <- shape_aspect_relationship.relating_shape_aspect shape_aspect_relationship shape_aspect_relationship.related_shape_aspect-> shape_aspect {shape_aspect shape_aspect.description = 'affected plane'} shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation representation.items[i] -> {representation_item representation_item.name = 'affected plane'} representation_item => geometric_representation_item => placement
LOCATION_- DIMENSION_- TOLERANCE #1: if directed = true #2: if directed = false	#1: (directed_dimensional_location) #2: (dimensional_location)	224 47	22	#1: directed_dimensional_location <= dimensional_location
directed	IDENTICAL MAPPING			

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
location_dimension_- tolerance to orientation (as plane_and_direction)	PATH		22	#1: (directed_dimensional_location <= dimensional_location <=) #2: (dimensional_location <=) shape_aspect_relationship shape_definition = shape_aspect_relationship shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation representation.items[i] -> {representation_item representation_item.name = 'orientation'} representation_item => geometric_representation_item => placement
LOCATION_- TOLERANCE	dimensional_location	47	22	
location_tolerance to shape_element (as origin_shape)	PATH			dimensional_location <= shape_aspect_relationship shape_aspect_relationship.relating_shape_aspect -> shape_aspect
location_tolerance to shape_element (as termination_shape)	PATH			dimensional_location <= shape_aspect_relationship shape_aspect_relationship.related_shape_aspect -> shape_aspect
MATERIAL_- CONDITION_MODIFIER	limit_condition	47		
material_type	IDENTICAL_MAPPING			
PARALLELISM_- TOLERANCE	parallelism_tolerance	519	10	

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
segment_size	[measure_with_unit.value_component] [measure_with_unit.unit_component]	41 41		parallelism_tolerance <= geometric_tolerance_with_datum_reference <= geometric_tolerance => geometric_tolerance_with_defined_unit geometric_tolerance_with_defined_unit.unit_size -> measure_with_unit [measure_with_unit.value_component] [measure_with_unit.unit_component]
parallelism_tolerance to datum (as geometric_reference)	PATH			parallelism_tolerance <= geometric_tolerance_with_datum_reference geometric_tolerance_with_datum_reference.datum_system[i] -> datum_reference

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
parallelism_tolerance to orientation (as affected_plane)	PATH			parallelism_tolerance <= geometric_tolerance_with_datum_reference <= geometric_tolerance <- tolerance_zone.defining_tolerance[i] tolerance_zone <= shape_aspect <- shape_aspect_relationship.relating_shape_aspect shape_aspect_relationship shape_aspect_relationship.related_shape_aspect-> shape_aspect {shape_aspect shape_aspect.description = 'affected plane'} shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation representation.items[i] -> {representation_item representation_item.name = 'affected plane'} representation_item => geometric_representation_item => placement
PERPENDICULARITY_-TOLERANCE	perpendicularity_tolerance	519	10	

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
segment_size	[measure_with_unit.value_component] [measure_with_unit.unit_component]	41 41		perpendicularity_tolerance <= geometric_tolerance_with_datum_reference <= geometric_tolerance => geometric_tolerance_with_defined_unit geometric_tolerance_with_defined_unit.unit_size ->measure_with_unit [measure_with_unit.value_component] [measure_with_unit.unit_component]
perpendicularity_- tolerance to datum (as geometric_reference)	PATH			perpendicularity_tolerance <= geometric_tolerance_with_datum_reference geometric_tolerance_with_datum_reference.datum_system[i] ->datum_reference

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
perpendicularity_- tolerance to orientation (as affected_plane)	PATH			perpendicularity_tolerance <= geometric_tolerance_with_datum_reference <= geometric_tolerance <- tolerance_zone.defining_tolerance[i] tolerance_zone <= shape_aspect <- shape_aspect_relationship.relating_shape_aspect shape_aspect_relationship shape_aspect_relationship.related_shape_aspect-> shape_aspect {shape_aspect shape_aspect.description = 'affected plane'} shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation representation.items[i] -> {representation_item representation_item.name = 'affected plane'} representation_item => geometric_representation_item => placement
PLACED_TARGET	placed_datum_target_feature	224	23	placed_datum_target_feature <= datum_target {datum_target <= shape_aspect (shape_aspect.description = 'point') (shape_aspect.description = 'line') (shape_aspect.description = 'rectangle') (shape_aspect.description = 'circle')}

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
placed_target to orientation (as placement)	PATH		9, 19, 24	placed_datum_target_feature <= datum_target <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'orientation'} representation_item => geometric_representation_item => placement
PLUS_MINUS_VALUE #1: if the plus_minus_-value is selected for tolerance_value #2: if the plus_minus_value is selected for numeric_-parameter	#1: (tolerance_value) #2: (qualified_representation_item)	47 45		

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
lower_limit	#1: (tolerance_value.lower_bound) #2: (standard_uncertainty.- uncertainty_value)	47 45		#2: {qualified_representation_item <= representation_item} qualified_representation_item qualified_representation_item.qualifiers[i] -> value_qualifier value_qualifier = uncertainty_qualifier uncertainty_qualifier.measure_name='lower limit' uncertainty_qualifier => standard_uncertainty standard_uncertainty.uncertainty_value
significant_digits	precision_qualifier.precision_value	45		#1: (tolerance_value [tolerance_value.upper_bound ->] [tolerance_value.lower_bound ->] measure_with_unit <- measure_qualification.qualifiers[1] ->) #2: (qualified_representation_item qualified_representation_item.qualifiers[i] ->) value_qualifier value_qualifier = precision_qualifier precision_qualifier precision_qualifier.precision_value
upper_limit	#1: (tolerance_value.upper_bound) #2: (standard_uncertainty.- uncertainty_value)	47 45		#2: {qualified_representation_item <= representation_item} qualified_representation_item qualified_representation_item.qualifiers[i] -> value_qualifier value_qualifier = uncertainty_qualifier uncertainty_qualifier.measure_name='upper limit' uncertainty_qualifier => standard_uncertainty standard_uncertainty.uncertainty_value
POSITION_TOLERANCE	position_tolerance	519	10	

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
value_qualifier	tolerance_zone_form.name	47		position_tolerance <= geometric_tolerance_with_datum_reference <= geometric_tolerance <- tolerance_zone.defining_tolerance[i] tolerance_zone tolerance_zone.form -> tolerance_zone_form tolerance_zone_form.name
position_tolerance to datum (as geometric_reference)	PATH			position_tolerance <= geometric_tolerance_with_datum_reference geometric_tolerance_with_datum_reference.datum_system[i] -> datum_reference

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
position_tolerance to orientation (as affected_plane)	PATH			position_tolerance <= geometric_tolerance_with_datum_reference <= geometric_tolerance <- tolerance_zone.defining_tolerance[i] tolerance_zone <= shape_aspect <- shape_aspect_relationship.relating_shape_aspect shape_aspect_relationship shape_aspect_relationship.related_shape_aspect-> shape_aspect {shape_aspect shape_aspect.description = 'affected plane'} shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation representation.items[i] -> {representation_item representation_item.name = 'affected plane'} representation_item => geometric_representation_item => placement
PROJECTION	projected_zone_definition	47		
projection_length	[measure_with_unit.value_component] [measure_with_unit.unit_component]	41 41		projected_zone_definition projected_zone_definition.projected_length -> measure_with_unit [measure_with_unit.value_component] [measure_with_unit.unit_component]

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
projection to shape_-element (as projection_end)	PATH			projected_zone_definition projected_zone_definition.projection_end -> shape_aspect
RADIAL_DIMENSION_-TOLERANCE	dimensional_size	47		{dimensional_size dimensional_size.name = 'radius'}
SIZE_TOLERANCE	dimensional_size	47		
size_tolerance to shape_-element (as applied_shape)	PATH			dimensional_size dimensional_size.applies_to -> shape_aspect
STRAIGHTNESS_-TOLERANCE #1: if segment size is specified #2: if segment size is not specified	#1: (geometric_tolerance_with_defined_-unit) #2: (geometric_tolerance)	519 519		straightness_tolerance<= {#1: (geometric_tolerance_with_defined_unit <= geometric_tolerance) #2: (geometric_tolerance)}
segment_size	[measure_with_unit.value_component] [measure_with_unit.unit_component]	41 41		#1: geometric_tolerance_with_defined_unit geometric_tolerance_with_defined_unit.unit_size -> measure_with_unit [measure_with_unit.value_component] [measure_with_unit.unit_component]

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
straightness_tolerance to orientation (as affected_plane)	PATH			#1: (geometric_tolerance_with_defined_unit <= geometric_tolerance <-) #2: (geometric_tolerance <-) tolerance_zone.defining_tolerance[i] tolerance_zone <= shape_aspect <- shape_aspect_relationship.relating_shape_aspect shape_aspect_relationship shape_aspect_relationship.related_shape_aspect-> shape_aspect {shape_aspect shape_aspect.description = 'affected plane'} shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation representation.items[i] -> {representation_item representation_item.name = 'affected plane'} representation_item => geometric_representation_item => placement
SURFACE_PROFILE_-TOLERANCE	surface_profile_tolerance	519	10	
surface_profile_tolerance to datum (as geometric_reference)	PATH			surface_profile_tolerance <= geometric_tolerance => geometric_tolerance_with_datum_reference geometric_tolerance_with_datum_reference.datum_system[i] -> datum_reference

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
SYMMETRY_- TOLERANCE	symmetry_tolerance	519	10	
symmetry_tolerance to datum (as geometric_reference)	PATH			symmetry_tolerance <= geometric_tolerance_with_datum_reference geometric_tolerance_with_datum_reference.datum_system[i] -> datum_reference
symmetry_tolerance to orientation (as affected_plane)	PATH			symmetry_tolerance <= geometric_tolerance_with_datum_reference <= geometric_tolerance <- tolerance_zone.defining_tolerance[i] tolerance_zone <= shape_aspect <- shape_aspect_relationship.relating_shape_aspect shape_aspect_relationship shape_aspect_relationship.related_shape_aspect-> shape_aspect {shape_aspect shape_aspect.description = 'affected plane'} shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation representation.items[i] -> {representation_item representation_item.name = 'affected plane'} representation_item => geometric_representation_item => placement
TARGET_AREA	datum_target	47	23	

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
target_area to shape_-element (as area_shape)	IDENTICAL MAPPING			
TARGET_CIRCLE	placed_datum_target_feature	224	23	placed_datum_target_feature <= datum_target {datum_target <= shape_aspect shape_aspect.description = 'circle'}
target_diameter	measure_representation_item	45	9, 19, 24	placed_datum_target_feature <= datum_target <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'target diameter'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
TARGET_LINE	placed_datum_target_feature	224	23	placed_datum_target_feature <= datum_target {datum_target <= shape_aspect shape_aspect.description = 'line'}
target_length	measure_representation_item	45	9, 19, 24	placed_datum_target_feature <= datum_target <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation ->{representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] ->{representation_item representation_item.name = 'target length'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
TARGET_POINT	placed_datum_target_feature	224	23	placed_datum_target_feature <= datum_target {datum_target <= shape_aspect shape_aspect.description = 'point'}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
TARGET_RECTANGLE	placed_datum_target_feature	224	23	placed_datum_target_feature <= datum_target {datum_target <= shape_aspect shape_aspect.description = 'rectangle'}
target_length	measure_representation_item	45	9, 19, 24	placed_datum_target_feature <= datum_target <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation ->{representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] ->{representation_item representation_item.name = 'target length'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
target_width	measure_representation_item	45	9, 19, 24	placed_datum_target_feature <= datum_target <= shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> {representation => shape_representation => shape_representation_with_parameters} representation representation.items[i] -> {representation_item representation_item.name = 'target width'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
TOLERANCE_LIMIT	type_qualifier	45		
limit_qualifier	type_qualifier.name	45		
TOLERANCE_RANGE	shape_dimension_representation	47		

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
lower_range	measure_representation_item	45		shape_dimension_representation <= shape_representation<= representation representation.items[i] ->{representation_item representation_item.name = 'lower range'} representation_item => measure_representation_item
significant_digits	precision_qualifier.precision_value	45		shape_dimension_representation <= representation representation.items[i] ->representation_item => measure_representation_item <= measure_with_unit <- measure_qualification.qualifiers[1] ->value_qualifier = precision_qualifier precision_qualifier precision_qualifier.precision_value
upper_range	measure_representation_item	45		shape_dimension_representation <= representation representation.items[i] ->{representation_item representation_item.name = 'upper range'} representation_item => measure_representation_item
TOLERANCE_VALUE	dimensional_characteristic_-representation	47		
envelope	representation.name	43	9, 19, 24	dimensional_characteristic_representation dimensional_characteristic_representation.representation ->shape_dimension_representation <= shape_representation <= representation representation.name {representation.name = 'envelope tolerance'}

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
tolerance_value to limits_and_fits (as defined_value)	PATH			dimensional_characteristic_representation dimensional_characteristic_representation.dimension ->dimensional_characteristic <- plus_minus_tolerance.toleranced_dimension plus_minus_tolerance plus_minus_tolerance.range ->tolerance_method_definition tolerance_method_definition = limits_and_fits limits_and_fits
tolerance_value to plus_-minus_value (as defined_value)	PATH			dimensional_characteristic_representation dimensional_characteristic_representation.dimension ->dimensional_characteristic <- plus_minus_tolerance.toleranced_dimension plus_minus_tolerance plus_minus_tolerance.range ->tolerance_method_definition tolerance_method_definition = tolerance_value tolerance_value
tolerance_value to tolerance_limit (as defined_value)	PATH		9, 19, 24	dimensional_characteristic_representation dimensional_characteristic_representation.representation ->shape_dimension_representation <= shape_representation <= representation representation.items[i] ->{representation_item => measure_representation_item} representation_item => qualified_representation_item qualified_representation_item.qualifiers[i] ->value_qualifier value_qualifier = type_qualifier type_qualifier
tolerance_value to tolerance_range (as defined_value)	PATH			dimensional_characteristic_representation dimensional_characteristic_representation.representation ->shape_dimension_representation
TOLERANCE_ZONE	tolerance_zone	47	23	

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 8 - Mapping table for measurement_limitations UoF (continued)

Application element	AIM element	Source	Rules	Reference path
common_zone	tolerance_zone.defining_tolerance	47		
form_type	tolerance_zone_form.name	47		tolerance_zone tolerance_zone.form -> tolerance_zone_form tolerance_zone_form.name
tolerance_zone to projection (as extended_shape)	PATH			tolerance_zone <- tolerance_zone_definition.zone tolerance_zone_definition => projected_zone_definition
tolerance_zone to tolerance_zone_-definition (as zone_definition)	PATH			tolerance_zone <- tolerance_zone_definition.zone tolerance_zone_definition
TOLERANCE_ZONE_-DEFINITION	tolerance_zone_definition	47		
tolerance_zone_-definition to shape_-element (as first_element)	PATH			tolerance_zone_definition tolerance_zone_definition.boundaries[i] -> shape_aspect
tolerance_zone_-definition to shape_-element (as second_element)	PATH			tolerance_zone_definition tolerance_zone_definition.boundaries[i] -> shape_aspect
TOTAL_RUNOUT_-TOLERANCE	total_runout_tolerance	519	10	

Table 8 - Mapping table for measurement_limitations UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
runout	[measure_with_unit.value_component] [measure_with_unit.unit_component]	41 41	23	total_runout_tolerance <= geometric_tolerance_with_datum_reference <= geometric_tolerance <- tolerance_zone.defining_tolerance[i] tolerance_zone <- tolerance_zone_definition.zone tolerance_zone_definition => runout_zone_definition runout_zone_definition.orientation -> runout_zone_orientation runout_zone_orientation.angle -> measure_with_unit [measure_with_unit.value_component] [measure_with_unit.unit_component]
total_runout_tolerance to datum (as geometric_reference)	PATH			total_runout_tolerance <= geometric_tolerance_with_datum_reference geometric_tolerance_with_datum_reference.datum_system[i] -> datum_reference

©ISO 2000 - All rights reserve

Table 9 - Mapping table for part_administration_data UoF

Application element	AIM element	Source	Rules	Reference path
APPROVAL	approval	41	1, 2, 5, 14, 17, 20	
approval_date	date	41	1, 6	approval <- approval_date_time.dated_approval approval_date_time approval_date_time.date_time -> date_time_select date_time_select = date date
status	approval_status.name	41	5, 20	approval approval.status -> approval_status approval_status.name {(approval_status.name = 'approved') (approval_status.name = 'not yet approved') (approval_status.name = 'disapproved') (approval_status.name = 'withdrawn')}
approval to person_in_-organization (as approval_authority)	PATH		2	approval <- approval_person_organization.authorized_approval approval_person_organization approval_person_organization.person_organization -> person_organization_select person_organization_select = person_and_organization person_and_organization
ORGANIZATION	organization	41		
organization_address	address	41		organization <- organizational_address.organizations[i] organizational_address <= address
organization_id	organization.id	41		
organization_name	organization.name	41		

©ISO 2000 - All rights reserved

Table 9 - Mapping table for part_administration_data UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
PERSON	person	41		
person_address	address	41		person <- personal_address.people[i] personal_address <= address
person_id	person.id	41		
person_name	[person.first_name] [person.last_name]	41		
person_phone_number	address.telephone_number	41		person <- personal_address.people[i] personal_address <= address address.telephone_number
PERSON_IN_- ORGANIZATION	person_and_organization	41		
role	person_and_organization_role.name	41		person_and_organization <- person_and_organization_assignment.assigned_person_and_organization person_and_organization_assignment person_and_organization_assignment.role -> person_and_organization_role person_and_organization_role.name
person_in_organization to organization (as company)	PATH			person_and_organization person_and_organization.the_organization -> organization
person_in_organization to person (as employee)	PATH			person_and_organization person_and_organization.the_person -> person

©ISO 2000 - All rights reserved

©ISO

©ISO 2000 - All rights reserved

Table 10 - Mapping table for part model UoF

Application element	AIM element	Source	Rules	Reference path
MANUFACTURED_-ASSEMBLY	product_definition_formation	41		product_definition_formation<- product_definition.formation product_definition product_definition.frame_of_reference-> product_definition_context<= application_context_element application_context_element.name='assembly definition'
MANUFACTURED_-ASSEMBLY_-RELATIONSHIP	next_assembly_usage_occurrence	44		
manufactured_-assembly_relationship to manufactured_assembly (as assembly)	PATH			next_assembly_usage_occurrence<= assembly_component_usage<= product_definition_usage<= product_definition_relationship product_definition_relationship.relating_product_definition-> product_definition product_definition.formation-> product_definition_formation
manufactured_-assembly_relationship to part (as component)	PATH			next_assembly_usage_occurrence<= assembly_component_usage<= product_definition_usage<= product_definition_relationship product_definition_relationship.related_product_definition-> product_definition product_definition.formation-> product_definition_formation

Table 10 - Mapping table for part_model UoF (continued)

Application element	AIM element	Source	Rules	Reference path
manufactured_- assembly_relationship to part_placement (as orientation)	PATH			next_assembly_usage_occurrence<= assembly_component_usage<= product_definition_usage<= product_definiton_relationship characterized_product_definition = product_definition_relationship 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 => shape_representation} representation_map.mapped_representation representation_map <- mapped_item.mapping_source] mapped_item
MATING_DEFINITION	product_definition	41		product_definition product_definition.frame_of_reference->product_definition_context<= application_context_element application_context_element.name='mating definition'
mating_type	product_definition.name	41		

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 10 - Mapping table for part model UoF (continued)

Application element	AIM element	Source	Rules	Reference path
mating_definition to manufactured_assembly (as applied_assembly)	PATH			product_definition product_definition.formation->product_definition_formation {product_definition_formation<-product_definition.formation product_definition product_definition.frame_of_reference->product_definition_context<=application_context_element application_context_element.name='assembly definition'}
mating_definition to shape_element (as mating_ shape)	PATH			product_definition characterized_product_definition = product_definition characterized_product_definition characterized_definition = characterized_product_definition characterized_definition <-property_definition.definition property_definition =>product_definition_shape <-shape_aspect.of_shape shape_aspect
mating_definition to single_piece_part (as mating_solution)	PATH			product_definition<-product_definition_relationship.relating_product_definition product_definition_relationship product_definition_relationship.name='mating solution' product_definition_relationship.related_product_definition->product_definition
MATING_DEFINITON_-RELATIONSHIP	product_definiton_relationship	41		product_definiton_relationship product_definiton_relationship.name='mating material'

Table 10 - Mapping table for part_model UoF (continued)

Application element	AIM element	Source	Rules	Reference path
mating_definition_- relationship to mating_- definition (as mating_part_- definition)	PATH			product_definiton_relationship product_definiton_relationship.relating_product_definitoin->product_definition (product_definition product_definition.frame_of_reference->product_definition_context<= application_context_element application_context_element.name='mating definition')
mating_definition_- relationship to part_- placement (as orientation)	PATH			product_definiton_relationship characterized_product_definition = product_definition_relationship 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 => shape_representation} representation_map.mapped_representation representation_map <- mapped_item.mapping_source] mapped_item
mating_definition_- relationship to single_- piece_part (as mated_part)	PATH			product_definiton_relationship product_definiton_relationship.related_product_definitoin->product_definition
MATING_- RELATIONSHIP	product_definiton_relationship	41		product_definiton_relationship product_definiton_relationship.name='mating membership'

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 10 - Mapping table for part model UoF (continued)

Application element	AIM element	Source	Rules	Reference path
mating_relationship to single_piece_part (as predecessor)	PATH			product_definiton_relationship product_definiton_relationship.related_product_definitoin-> product_definition
mating_relationship to single_piece_part (as successor)	PATH			product_definiton_relationship product_definiton_relationship.relating_product_definitoin-> product_definition
PART	product_definition_formation	41	13, 14, 16	product_definition_formation<- product_definition.formation product_definition product_definition.frame_of_reference-> product_definition_context<= application_context_element application_context_element.name='part definition'
part_description	product.description	41	15	product_definition_formation product_definition_formation.of_product -> product product.description
part_id	product.id	41	15	product_definition_formation product_definition_formation.of_product -> product product.id
part_name	product.name	41	15	product_definition_formation product_definition_formation.of_product -> product product.name
part_revision_id	product_definition_formation.id	41		
security_classification	security_classification	41	8, 16, 21	product_definition_formation feature_based_pp_classified_item = product_definition_formation feature_based_pp_classified_item <- feature_based_pp_security_classification_assignment.items[i] feature_based_pp_security_classification_assignment <= security_classification_assignment security_classification_assignment.assigned_security_classification -> security_classification

Table 10 - Mapping table for part_model UoF (continued)

Application element	AIM element	Source	Rules	Reference path
part to approval (as manufacture_- authorization)	PATH		14	product_definition_formation feature_based_pp_approved_item = product_definition_formation feature_based_pp_approved_item <- feature_based_pp_approval_assignment.items[i] feature_based_pp_approval_assignment <= approval_assignment approval_assignment.assigned_approval -> approval
part to ordered_part (as quantity_ordered)	PATH		13	product_definition_formation feature_based_pp_ordered_item = product_definition_formation feature_based_pp_ordered_item <- ordered_part.items[i] ordered_part
part to organization (as manufactured_by_- organization)	PATH			product_definition_formation feature_based_pp_organization_item = product_definition_formation feature_based_pp_organization_item <- feature_based_pp_organization_assignment.items[i] feature_based_pp_organization_assignment <= {organization_assignment organization_assignment.role -> organization_role organization_role.name = 'manufacturer'} organization_assignment organization_assignment.assigned_organization -> organization

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 10 - Mapping table for part model UoF (continued)

Application element	AIM element	Source	Rules	Reference path
part to organization (as owned_by_- organization)	PATH			product_definition_formation feature_based_pp_organization_item = product_definition_formation feature_based_pp_organization_item <- feature_based_pp_organization_assignment.items[i] feature_based_pp_organization_assignment <= {organization_assignment organization_assignment.role -> organization_role organization_role.name = 'owner'} organization_assignment organization_assignment.assigned_organization -> organization
part to person_in_- organization (as manufactured_by_- person)	PATH			product_definition_formation feature_based_pp_person_and_organization_item = product_definition_formation feature_based_pp_person_and_organization_item <- feature_based_pp_person_and_organization_assignment.items[i] feature_based_pp_person_and_organization_assignment <= {person_and_organization_assignment person_and_organization_assignment.role -> person_and_organization_role person_and_organization_role.name = 'manufacturer'} person_and_organization_assignment person_and_organization_assignment.assigned_person_and_organization -> person_and_organization
part to person_in_- organization (as owned_by_person)	PATH			product_definition_formation feature_based_pp_person_and_organization_item = product_definition_formation feature_based_pp_person_and_organization_item <- feature_based_pp_person_and_organization_assignment.items[i] feature_based_pp_person_and_organization_assignment <= {person_and_organization_assignment person_and_organization_assignment.role -> person_and_organization_role person_and_organization_role.name = 'owner'} person_and_organization_assignment person_and_organization_assignment.assigned_person_and_organization -> person_and_organization

Table 10 - Mapping table for part_model UoF (continued)

Application element	AIM element	Source	Rules	Reference path
part to property (as property_- characteristics)	PATH			product_definition_formation <- product_definition.formation product_definition characterized_product_definition = product_definition characterized_product_definition characterized_definition = characterized_product_definition characterized_definition <- property_definition.definition property_definition
part to shape (as physical_form)	PATH			product_definition_formation <- product_definition.formation product_definition characterized_product_definition = product_definition characterized_product_definition characterized_definition = characterized_product_definition characterized_definition <- property_definition.definition property_definition => product_definition_shape
SINGLE_PIECE_PART	product_definition_formation	41		

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 10 - Mapping table for part model UoF (continued)

Application element	AIM element	Source	Rules	Reference path
single_piece_part to material (as material_-definition)	PATH		12	product_definition_formation <- product_definition.formation product_definition <- product_definition_relationship.relating_product_definition {product_definition_relationship => product_definition_usage => make_from_usage_option} product_definition_relationship product_definition_relationship.related_product_definition -> product_definition {product_definition characterized_product_definition = product_definition characterized_product_definition characterized_definition = characterized_product_definition characterized_definition <- material_designation.definitions material_designation}

Table 10 - Mapping table for part_model UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
single_piece_part to alternate_material (as alternate_material_-definition)	PATH		12	product_definition_formation <- product_definition.formation product_definition <- product_definition_relationship.relating_product_definition {product_definition_relationship => product_definition_usage => make_from_usage_option} product_definition_relationship product_definition_relationship.related_product_definition -> product_definition {[product_definition characterized_product_definition = product_definition characterized_product_definition characterized_definition = characterized_product_definition characterized_definition <- material_designation.definitions[i] material_designation] [product_definition <- product_definition_relationship.related_product_definition {product_definition_relationship => product_definition_usage => make_from_usage_option} product_definition_relationship product_definition_relationship.relating_product_definition -> product_definition characterized_product_definition = product_definition characterized_product_definition characterized_definition = characterized_product_definition characterized_definition <- material_designation.definitions[i] material_designation]}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 11 - Mapping table for requisitions UoF

Application element	AIM element	Source	Rules	Reference path
CUTTING_TOOL_-REQUISITION	document	41		{document document.kind -> document_type document_type.product_data_type = 'cutting tool requisition'}
DEDICATED_FIXTURE_-REQUISITION	document	41		{document document.kind -> document_type document_type.product_data_type = 'dedicated fixture requisition'}
INDIRECT_STOCK_-REQUISITION	document	41		{document document.kind -> document_type document_type.product_data_type = 'indirect stock requisition'}
MACHINE_-REQUISITION	document	41		{document document.kind -> document_type document_type.product_data_type = 'machine requisition'}
MATERIAL_-REQUISITION	document	41		{document document.kind -> document_type document_type.product_data_type = 'material requisition'}
MODULAR_FIXTURE_-REQUISITION	document	41		{document document.kind -> document_type document_type.product_data_type = 'modular fixture requisition'}
REQUISITION	document	41		
quantity_ordered	document.name	41		

Table 11 - Mapping table for requisitions UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
required_delivery_date	date	41	6	document=> document_with_class feature_based_pp_dated_item = document_with_class feature_based_pp_dated_item <- feature_based_pp_date_assignment.items[i] feature_based_pp_date_assignment <= {date_assignment date_assignment.role -> date_role date_role.name = 'required delivery date'} date_assignment date_assignment.assigned_date -> date
requisition_date	date	41	6	document=> document_with_class feature_based_pp_dated_item = document_with_class feature_based_pp_dated_item <- feature_based_pp_date_assignment.items[i] feature_based_pp_date_assignment <= {date_assignment date_assignment.role -> date_role date_role.name = 'requisition date'} date_assignment date_assignment.assigned_date -> date
requisition_description	document.description	41		
requisition_number	document.id	41		

©ISO 2000 - All rights reserved

Table 12 - Mapping table for shape_representation_for_machining UoF

Application element	AIM element	Source	Rules	Reference path
BASE_SHAPE	product_definition_shape	41		{product_definition_shape <= property_definition property_definition.definition -> characterized_definition <- material_designation.definitions[i]}
BLOCK_BASE_SHAPE	block_shape_representation	224	9, 19, 24	block_shape_representation <= shape_representation_with_parameters <= shape_representation
block_base_shape to numeric_parameter (as height)	PATH			block_shape_representation <= shape_representation_with_parameters <= shape_representation <= representation representation.items[i] -> {representation_item representation_item.name = 'height'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
block_base_shape to numeric_parameter (as width)	PATH			block_shape_representation <= shape_representation_with_parameters <= shape_representation <= representation representation.items[i] -> {representation_item representation_item.name = 'width'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
B-REP_MODEL	manifold_solid_brep	42		

Table 12 - Mapping table for shape_representation_for_machining UoF (continued)

Application element	AIM element	Source	Rules	Reference path
B-REP_MODEL_- ELEMENT	(geometric_representation_item) (topological_representation_item)	42 42		
B-rep_model_element to B-rep_model (as element)	PATH			(geometric_representation_item => surface <- face_surface.face_geometry face_surface <= face <-) (topological_representation_item => face <-) connected_face_set.cfs_faces[i] connected_face_set => closed_shell <- manifold_solid_brep.outer manifold_solid_brep
B-REP_SHAPE_- ASPECT_- REPRESENTATION	advanced_brep_shape_representation	514	9, 19, 24	advanced_brep_shape_representation <= shape_representation
B-rep_shape_aspect_- representation to B-rep_- model (as shape_definition)	PATH			advanced_brep_shape_representation <= shape_representation <= representation representation.items[i] -> representation_item => geometric_representation_item => solid_model => manifold_solid_brep
B-REP_SHAPE_- REPRESENTATION	advanced_brep_shape_representation	514	9, 19, 24	advanced_brep_shape_representation <= shape_representation
B-rep_shape_- representation to B-rep_- model (as shape_definition)	PATH			advanced_brep_shape_representation <= shape_representation <= representation representation.items[i] -> representation_item => geometric_representation_item => solid_model => manifold_solid_brep

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 12 - Mapping table for shape_representation_for_machining UoF (continued)

Application element	AIM element	Source	Rules	Reference path
CYLINDRICAL_BASE_-SHAPE	cylindrical_shape_representation	224	9, 19, 24	cylindrical_shape_representation <= shape_representation_with_parameters <= shape_representation
cylindrical_base_shape to numeric_parameter (as diameter)	PATH			cylindrical_shape_representation <= shape_representation_with_parameters <= shape_representation <= representation representation.items[i] -> {representation_item representation_item.name = 'diameter'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
DIRECTION_ELEMENT	direction_shape_representation	224	19	direction_shape_representation <= shape_representation<= representation {representation.items[i]-> representation_item=> geometric_representation_item=> direction}
EXPLICIT_BASE_-SHAPE_-REPRESENTATION	shape_representation	41	9	
explicit_base_shape_-representation to B-rep_-shape_representation (as B-rep_form)	IDENTICAL MAPPING			

Table 12 - Mapping table for shape_representation_for_machining UoF (continued)

Application element	AIM element	Source	Rules	Reference path
FACE_SHAPE_-ELEMENT	face_shape_representation	224	19	face_shape_representation <= shape_representation<= representation representation.items[i] -> representation_item => geometric_representation_item => (face_surface) (oriented_face)
FACE_SHAPE_-ELEMENT_-RELATIONSHIP	face_shape_representation_relationship	224		face_shape_representation_relationship<= representation_relationship
face_shape_element_-relationship to face_shape_element (as predecessor)	PATH			face_shape_representation_relationship<= representation_relationship representation_relationship.rep_1-> representation=> shape_representation=> face_shape_representation
face_shape_element_-relationship to face_shape_element (as successor)	PATH			face_shape_representation_relationship<= representation_relationship representation_relationship.rep_2-> representation=> shape_representation=> face_shape_representation
IMPLICIT_BASE_-SHAPE_-REPRESENTATION	shape_representation_with_parameters	224	9, 19, 24	shape_representation_with_parameters <= shape_representation

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 12 - Mapping table for shape_representation_for_machining UoF (continued)

Application element	AIM element	Source	Rules	Reference path
implicit_base_shape_- representation to numeric_parameter (as base_shape_length)	PATH			shape_representation_with_parameters <= shape_representation <= representation representation.items[i] ->{representation_item representation_item.name = 'length'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
implicit_base_shape_- representation to orientation (as placement)	PATH			shape_representation_with_parameters <= shape_representation <= representation representation.items[i] ->{representation_item representation_item.name = 'orientation'} representation_item => geometric_representation_item => placement
LOCATION_ELEMENT	location_shape_representation	224	19	location_shape_representation <= shape_representation<= representation representation.items[i] ->{representation_item representation_item.name = 'orientation'} {representation_item => geometric_representation_item => point=> cartesian_point}
NGON_BASE_SHAPE	ngon_shape_representation	224	9, 19, 24	ngon_shape_representation <= shape_representation_with_parameters <= shape_representation

Table 12 - Mapping table for shape_representation_for_machining UoF (continued)

Application element	AIM element	Source	Rules	Reference path
circumscribed_or_- across_flats	(representation_item.name = 'circumscribed diameter') (representation_item.name = 'diameter across flats')			ngon_shape_representation <= shape_representation_with_parameters <= shape_representation <= representation representation.items[i] -> representation_item (representation_item.name = 'circumscribed diameter') (representation_item.name = 'diameter across flats')
ngon_base_shape to numeric_parameter (as corner_radius)	PATH			ngon_shape_representation <= shape_representation_with_parameters <= shape_representation <= representation representation.items[i] -> {representation_item representation_item.name = 'corner radius'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}
ngon_base_shape to numeric_parameter (as diameter)	PATH			ngon_shape_representation <= shape_representation_with_parameters <= shape_representation <= representation representation.items[i] -> {representation_item (representation_item.name = 'circumscribed diameter') (representation_item.name = 'diameter across flats')} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit => length_measure_with_unit}

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 12 - Mapping table for shape_representation_for_machining UoF (continued)

Application element	AIM element	Source	Rules	Reference path
ngon_base_shape to numeric_parameter (as number_of_sides)	PATH			ngon_shape_representation <= shape_representation_with_parameters <= shape_representation <= representation representation.items[i] -> {representation_item representation_item.name = 'number of sides'} representation_item => measure_representation_item {measure_representation_item <= measure_with_unit measure_with_unit.value_component -> measure_value measure_value = count_measure count_measure}
ORIENTATION	placement	42		
axis	(axis1_placement.axis) ((axis2_placement_3d.axis) (axis2_placement_3d.ref_direction))	42 42 42		placement => (axis1_placement axis1_placement.axis) (axis2_placement_3d (axis2_placement_3d.axis) (axis2_placement_3d.ref_direction))
location	placement.location	42		
PART_PLACEMENT	mapped_item	43		
part_placement to orientation (as resulting_orientation)	PATH			mapped_item mapped_item.mapping_target-> representation_item=> geometric_representation_item=> placement=> axis2_placement_3d

Table 12 - Mapping table for shape_representation_for_machining UoF (continued)

Application element	AIM element	Source	Rules	Reference path
part_placement to orientation (as originating_-orientation)	PATH			mapped_item mapped_item.mapping_source->representation_map representation_map.mapping_origin->representation_item=> geometric_representation_item=>placement=> axis2_placement_3d
part_placement to shape (as oriented_physical_-form)	PATH			mapped_item mapped_item.mapping_source->representation_map representation_map.mapped_representation->representation<- {representation-=>shape_representation} property_definition_representation.used_representation property_definition_representation {property_definition_representation=>shape_definition_representation} property_definition_representation.definition->property_definition=> product_definition_shape
PATH_ELEMENT	path_shape_representation	224	19	path_shape_representation <= shape_representation<= representation representation.items[i] ->{representation_item representation_item.name = 'profile shape'} representation_item =>(geometric_representation_item =>curve=>bounded_curve) (topological_representation_item=>edge=>edge_curve)

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 12 - Mapping table for shape_representation_for_machining UoF (continued)

Application element	AIM element	Source	Rules	Reference path
PLANAR_ELEMENT	planar_shape_representation	224		planar_shape_representation <= shape_representation<= representation
location	PATH	42		planar_shape_representation<= shape_representation<= representation representation.items[i] -> representation_item => geometric_representation_item => surface=> elementary_surface {elementary_surface=> plane} elementary_surface.position-> axis2_placement_3d<= placement placement.location-> cartesian_point
normal	PATH	42		planar_shape_representation<= shape_representation<= representation representation.items[i] -> representation_item => geometric_representation_item => surface=> elementary_surface {elementary_surface=> plane} elementary_surface.position-> axis2_placement_3d.axis->direction
SHAPE	product_definition_shape	41		

Table 12 - Mapping table for shape_representation_for_machining UoF (continued)

Application element	AIM element	Source	Rules	Reference path
shape to base_shape (as base_shape_- definition)	PATH		12	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_relationship.relating_product_definition {product_definition_relationship => product_definition_usage => make_from_usage_option} product_definition_relationship product_definition_relationship.related_product_definition -> product_definition characterized_product_definition = product_definition characterized_product_definition characterized_definition = characterized_product_definition {characterized_definition <- material_designation.definitions[i] material_designation} characterized_definition <- property_definition.definition property_definition => product_definition_shape
shape to B-rep_shape_- representation (as B-rep_form)	PATH		9, 19, 24	product_definition_shape <= property_definition <- property_definition_representation.definition {property_definition_representation => shape_definition_representation} property_definition_representation property_definition_representation.used_representation -> representation => shape_representation => advanced_brep_shape_representation

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table 12 - Mapping table for shape_representation_for_machining UoF (continued)

Application element	AIM element	Source	Rules	Reference path
shape to shape_aspect (as element)	PATH			product_definition_shape <- shape_aspect.of_shape shape_aspect
SHAPE_ASPECT	shape_aspect	41	23	
shape_aspect to B-rep_- model_element (as B-rep_shape)	PATH			shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation representation.items[i] -> representation_item => (geometric_representation_item) (topological_representation_item)
shape_aspect to B-rep_- shape_aspect_- representation (as B-rep_form)	PATH		9, 19, 24	shape_aspect shape_definition = shape_aspect shape_definition characterized_definition = shape_definition characterized_definition <- property_definition.definition property_definition <- property_definition_representation.definition property_definition_representation property_definition_representation.used_representation -> representation => shape_representation => advanced_brep_shape_representation
shape_aspect to shape_- element (as element)	IDENTICAL MAPPING			

Table 12 - Mapping table for shape_representation_for_machining UoF (concluded)

Application element	AIM element	Source	Rules	Reference path
SHAPE_ELEMENT	shape_aspect	41	23	{shape_aspect shape_aspect.product_definitional = TRUE}

©ISO 2000 - All rights reserve

The following rules are referenced in the preceding tables:

1) approval_requires_approval_date_time
2) approval_requires_approval_person_organization
3) dependent_instantiable_action_request_status
4) dependent_instantiable_action_status
5) dependent_instantiable_approval_status
6) dependent_instantiable_date
7) dependent_instantiable_named_unit
8) dependent_instantiable_security_classification_level
9) dependent_instantiable_shape_representation
10) geometric_tolerance_subtype_exclusiveness
11) machining_feature_life_cycle
12) material_is_specified_for_part
13) part_requires_project_order
14) part_to_approval
15) product_requires_version
16) product_definition_formation_requires_security_classification
17) project_order_requires_approval
18) project_order_tracking_relationships
19) representation_subtype_exclusiveness
20) restrict_approval_status
21) restrict_security_classification_level
22) shape_aspect_relationship_subtype_exclusiveness
23) shape_aspect_subtype_exclusiveness
24) shape_representation_subtype_exclusiveness
25) subtype_mandatory_characterized_object
26) transition_feature_life_cycle
27) transition_feature_on_part_boundary

Note - The datum that represents a compound_datum uses at least four shape_aspect_relationships. At least two shape_aspect_relationship reference the datum associated to the compound_datum. Each datum referenced by the compound_datum has a shape_aspect_relationship to a datum_feature. The compound_datum has a shape_aspect_relationship that references the same datum_features.

5.2 AIM EXPRESS short listing

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

```
*)
SCHEMA feature_based_process_planning;

  USE FROM action_schema                                    -- ISO 10303-41
      (action_directive,
       action_method,
```

©ISO 2000 - All rights reserved

```
      action_relationship,
      action_request_solution,
      action_request_status,
      action_status,
      directed_action,
      versioned_action_request);

  USE FROM aic_advanced_brep                                -- ISO 10303-514
      (advanced_brep_shape_representation);

  USE FROM aic_topologically_bounded_surface                -- ISO 10303-511
      (advanced_face);

  USE FROM aic_geometric_tolerances                         -- ISO 10303-519
     (angularity_tolerance,
      concentricity_tolerance,
      circular_runout_tolerance,
      cylindricity_tolerance,
      flatness_tolerance,
      line_profile_tolerance,
      parallelism_tolerance,
      perpendicularity_tolerance,
      position_tolerance,
      roundness_tolerance,
      straightness_tolerance,
      surface_profile_tolerance,
      symmetry_tolerance,
      total_runout_tolerance);

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

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

  REFERENCE FROM basic_attribute_schema                      -- ISO 10303-41
     (description_attribute,
     object_role,
     role_association);

  USE FROM date_time_schema                                  -- ISO 10303-41
     (date,
      calendar_date,
      ordinal_date,
      week_of_year_and_day_date);

  USE FROM document_schema                                   -- ISO 10303-41
     (document,
      document_representation_type,
      document_with_class,
      document_usage_constraint,
      document_relationship);

  USE FROM external_reference_schema                         -- ISO 10303-41
     (externally_defined_item);
```

 ©ISO 2000 - All rights reserved

```
USE FROM geometric_model_schema                          -- ISO 10303-42
    (brep_with_voids,
     manifold_solid_brep);

REFERENCE FROM geometry_schema                           -- ISO 10303-42
    (dummy_gri);

USE FROM geometry_schema                                 -- ISO 10303-42
    (axis2_placement_2d,
     axis2_placement_3d,
     bezier_curve,
     bezier_surface,
     b_spline_curve_with_knots,
     b_spline_surface_with_knots,
     cartesian_point,
     cartesian_transformation_operator_3d,
     circle,
     conical_surface,
     cylindrical_surface,
     degenerate_toroidal_surface,
     direction,
     ellipse,
     geometric_representation_context,
     hyperbola,
     line,
     parabola,
     pcurve,
     plane,
     point,
     polyline,
     quasi_uniform_curve,
     quasi_uniform_surface,
     rational_b_spline_curve,
     rational_b_spline_surface,
     spherical_surface,
     surface_curve,
     surface_of_linear_extrusion,
     surface_of_revolution,
     swept_surface,
     toroidal_surface,
     uniform_curve,
     uniform_surface,
     vector);

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

USE FROM management_resources_schema                     -- ISO 10303-41
    (action_assignment,
     action_request_assignment,
     approval_assignment,
     date_assignment,
     document_reference,
     document_usage_constraint_assignment,
     group_assignment,
     organization_assignment,
     person_and_organization_assignment,
     security_classification_assignment);
```

©ISO 2000 - All rights reserved

```
USE FROM material_property_representation_schema          -- ISO 10303-45
    (material_property_representation);

USE FROM material_property_definition_schema              -- ISO 10303-45
    (material_designation,
     material_property,
     property_definition_relationship);

USE FROM measure_schema                                   -- ISO 10303-41
    (context_dependent_measure,
     context_dependent_unit,
     conversion_based_unit,
     count_measure,
     derived_unit,
     descriptive_measure,
     global_unit_assigned_context,
     length_measure_with_unit,
     length_unit,
     measure_with_unit,
     named_unit,
     parameter_value,
     plane_angle_measure_with_unit,
     plane_angle_unit,
     positive_plane_angle_measure,
     ratio_measure,
     ratio_measure_with_unit,
     ratio_unit,
     si_unit,
     solid_angle_measure_with_unit,
     solid_angle_unit);

USE FROM person_organization_schema                       -- ISO 10303-41
    (organizational_address,
     person_and_organization,
     personal_address);

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

USE FROM product_property_definition_schema               -- ISO 10303-41
    (characterized_object,
     product_definition_shape,
     property_definition,
     shape_aspect,
     shape_aspect_relationship);

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

USE FROM product_structure_schema                         -- ISO 10303-44
    (assembly_component_usage,
     next_assembly_usage_occurrence,
     make_from_usage_option);

USE FROM qualified_measure_schema                         -- ISO 10303-45
    (descriptive_representation_item,
```

 ©ISO 2000 - All rights reserved

```
     expanded_uncertainty,
     measure_representation_item,
     measure_qualification,
     precision_qualifier,
     qualified_representation_item,
     qualitative_uncertainty,
     standard_uncertainty,
     type_qualifier,
     uncertainty_qualifier);

USE FROM representation_schema                          -- ISO 10303-43
    (definitional_representation,
     global_uncertainty_assigned_context,
     mapped_item,
     parametric_representation_context,
     representation,
     representation_item,
     representation_relationship);

USE FROM security_classification_schema                 -- ISO 10303-41
    (security_classification,
     security_classification_level);

USE FROM shape_aspect_definition_schema                 -- ISO 10303-47
    (composite_shape_aspect,
     datum_feature,
     datum_reference,
     datum_target,
     referenced_modified_datum,
     datum);

USE FROM shape_dimension_schema                         -- ISO 10303-47
    (angular_location,
     angular_size,
     dimensional_characteristic_representation,
     dimensional_location,
     dimensional_location_with_path,
     dimensional_size,
     dimensional_size_with_path,
     shape_dimension_representation);

USE FROM shape_tolerance_schema                         -- ISO 10303-47
    (geometric_tolerance,
     geometric_tolerance_relationship,
     geometric_tolerance_with_datum_reference,
     geometric_tolerance_with_defined_unit,
     limits_and_fits,
     modified_geometric_tolerance,
     plus_minus_tolerance,
     projected_zone_definition,
     runout_zone_definition,
     runout_zone_orientation,
     tolerance_value,
     tolerance_zone,
     tolerance_zone_definition);

USE FROM support_resource_schema                        -- ISO 10303-41
    (identifier,
     label,
     text);
```

©ISO 2000 - All rights reserved

```
  REFERENCE FROM topology_schema                                -- ISO 10303-42
      (dummy_tri);

  USE FROM topology_schema                                      -- ISO 10303-42
      (closed_shell,
       connected_face_set,
       oriented_closed_shell,
       edge_curve,
       edge_loop,
       face_bound,
       face_outer_bound,
       face_surface,
       path,
       vertex_loop,
       vertex_point);

(*
```

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

action_schema	ISO 10303-41
application_context_schema	ISO 10303-41
approval_schema	ISO 10303-41
aic_advanced_brep	ISO 10303-514
aic_geometric_tolerances	ISO 10303-519
aic_topologically_bounded_surface	ISO 10303-511
basic_attribute_schema	ISO 10303-41
date_time_schema	ISO 10303-41
document_schema	ISO 10303-41
external_reference_schema	ISO 10303-41
geometric_model_schema	ISO 10303-42
geometry_schema	ISO 10303-42
management_resources_schema	ISO 10303-41
material_property_definition_schema	ISO 10303-45
measure_schema	ISO 10303-41
person_organization_schema	ISO 10303-41
product_definition_schema	ISO 10303-41
product_property_definition_schema	ISO 10303-41
product_property_representation_schema	ISO 10303-41
product_structure_schema	ISO 10303-44
qualified_measure_schema	ISO 10303-45
representation_schema	ISO 10303-43
security_classification_schema	ISO 10303-41
shape_aspect_definition_schema	ISO 10303-47
shape_dimension_schema	ISO 10303-47
shape_tolerance_schema	ISO 10303-47
support_resource_schema	ISO 10303-41
topology_schema	ISO 10303-42

 ©ISO 2000 - All rights reserved

5.2.1 Fundamental concepts and assumptions

ISO 10303-224 is designed to be used in the representation and exchange of data to a process planning function for the manufacture of mechanical parts. This schema embodies several fundamental concepts that are manifested in the information model. The process planning function in an organization can be assisted a great deal by identifying machining oriented part shape features so that the process planner can more readily identify machining equipment, tools, and processes to manufacture a part. The fundamental concept of this schema is to enable the representation and exchange of this information about a part.

5.2.1.1 Hybrid approach to feature representation

Two ways to represent the shape of part features are specified in ISO 10303-224, explicit shape representation and implicit shape representation. The explicit representation of shape is accomplished by specifying a boundary representation solid model that defines the shape of the part feature. The implicit representation of shape is accomplished by specifying the shape defining parameters that are specified in the model for each type of part feature. The implicit representation is always dependent upon a base shape representation to which the features are applied. The schema provides the capability to use a hybrid approach to represent the shape of a part feature. This capability allows for the specification of an explicit representation, an implicit representation, or both an explicit and implicit representation for any of the part features defined for a part. The base shape representation may be specified either implicitly or explicitly as well. The explicit representation may be used in conjunction with the implicit representation to verify the interpretation of the implicit representation or to share data with another application that does not rely on the implicit representation of part features.

5.2.1.2 Use of the integrated resources for feature definition

A part's machining features are elements or aspects of the shape of a part. As such, the **shape_aspect** entity from ISO 10303-41 is specialized in this schema to specify the concepts of the different types of features. Three basic types of features are specified as **instanced_feature**, **replicate_feature**, and **transition_feature**. These basic types are again specialized into the individual types of features. Two types of parameters are used to specify the implicit shape representation of a feature, a numeric and a descriptive parameter. The **measure_representation_item** is used to specify the numeric parameter providing for both a real value of length and angle or an integer value of count and the units by which each value is specified. The **descriptive_representation_item** is used to specify the descriptive parameter. The element of the implicit shape that is being defined for each feature is specified by the name of the parameter. Constraints are defined in each specialization to specify the required parameters and the value types for the implicit representation of each feature. The **shape_representation_with_-parameters** entity is used to specify the implicit representation of the shape of the feature.

Additionally, there are components of the part's features defined that are also elements of the shape of the part. As such, they are also specified using specializations of the **shape_aspect** entity. Unlike the feature representations, the feature components shall always have an implicit representation specified.

EXAMPLE - A profile to be swept or a taper definition are types of feature components.

©ISO 2000 - All rights reserved

The feature components define parts of the individual features and are required to be used in certain feature definitions. The feature components are related to the individual part's features through a specialization of the **shape_aspect_relationship** called the **feature_component_relationship**. Every **feature_component_relationship** specifies a component that has been defined for a particular feature.

There is also a specialization of the **shape_aspect_relationship** defined in the schema called **shape_-defining_relationship**. This entity is used to specify other elements of the shape of the part (**shape_-aspect**s) that participate in the implicit representation of the shape of the feature. These elements all have a geometric or topological representation that consists of a single **representation_item**.

EXAMPLE - A location or a path are types of shape elements that are shape defining.

5.2.2 Feature based process planning types

5.2.2.1 document_reference_item

A **document_reference_item** identifies an **action_method**, **externally_defined_feature_definition**, **directed_action**, **dimensional_characteristic_representation**, or **property_definition** to which a referenced document may be assigned.

EXPRESS specification:

```
*)
TYPE document_reference_item = SELECT
  (action_method,
   externally_defined_feature_definition,
   directed_action,
   dimensional_characteristic_representation,
   property_definition);
END_TYPE;
(*
```

5.2.2.2 feature_based_pp_action_item

A **feature_based_pp_action_item** identifies a **product_definition_formation** that is being tracked by an **action_directive** for a project.

EXPRESS specification:

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

5.2.2.3 feature_based_pp_action_request_item

A **feature_based_pp_action_request_item** identifies a **product_definition_formation** for which a design exception has been identified.

 ©ISO 2000 - All rights reserved

EXPRESS specification:

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

5.2.2.4 feature_based_pp_approved_item

A **feature_based_pp_approved_item** identifies a **directed_action** or **product_definition_formation** to which an **approval** may be assigned.

EXPRESS specification:

```
*)
TYPE feature_based_pp_approved_item = SELECT
  (directed_action,
   product_definition_formation);
END_TYPE;
(*
```

5.2.2.5 feature_based_pp_classified_item

A **feature_based_pp_classified_item** identifies a **product_definition_formation** to which a **security_-classification** may be assigned.

EXPRESS specification:

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

5.2.2.6 feature_based_pp_dated_item

A **feature_based_pp_dated_item** identifies a **directed_action**, **document**, or **versioned_action_-request** to which a **date** may be assigned.

EXPRESS specification:

```
*)
TYPE feature_based_pp_dated_item = SELECT
  (directed_action,
   document,
   versioned_action_request);
END_TYPE;
(*
```

©ISO 2000 - All rights reserved

5.2.2.7 feature_based_pp_ordered_item

A **feature_based_pp_ordered_item** identifies the **product_definition_formation** that is being ordered by a customer.

EXPRESS specification:

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

5.2.2.8 feature_based_pp_organization_item

A **feature_based_pp_organization_item** identifies an **action_directive** or **product_definition_-formation** to which an **organization** may be assigned.

EXPRESS specification:

```
*)
TYPE feature_based_pp_organization_item = SELECT
  (action_directive,
   product_definition_formation);
END_TYPE;
(*
```

5.2.2.9 feature_based_pp_person_and_organization_item

A **feature_based_pp_person_and_organization_item** identifies an **action_directive** or **product_-definition_formation** to which a **person_and_organization** may be assigned.

EXPRESS specification:

```
*)
TYPE feature_based_pp_person_and_organization_item = SELECT
  (action_directive, product_definition_formation);
END_TYPE;
(*
```

5.2.2.10 group_item

A **group_item** identifies an **instanced_feature**, **replicate_feature**, or **transition_feature** to which a referenced group may be assigned.

EXPRESS specification:

```
*)
  TYPE group_item = SELECT
    (instanced_feature, replicate_feature, transition_feature);
  END_TYPE;
(*
```

 ©ISO 2000 - All rights reserved

5.2.3 Feature based process planning entities

5.2.3.1 Feature based process planning entity definitions

5.2.3.1.1 applied_area

An **applied_area** is a type of **shape_aspect** that is the representation of the bounded enclosed area used in the definition of the **thread** and **turned_knurl feature_definition**s. See ARM definition for Partial_-area_definition in paragraph 4.2.142 for more information.

EXPRESS specification:

```
*)
  ENTITY applied_area
    SUBTYPE OF (shape_aspect);
    WHERE
      wr1: ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
           IN TYPEOF(SELF.of_shape));
      wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
           (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
           (('FEATURE_BASED_PROCESS_PLANNING.' +
           'SHAPE_REPRESENTATION_WITH_PARAMETERS')
           IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0);

      wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
           (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
           used_representation)) ) |
           (NOT({2 <= SIZEOF(impl_rep.used_representation.items) <= 3} )
            ) )) = 0)) )) = 0);

    wr4:  SIZEOF( QUERY( pd <* USEDIN( SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
          (SIZEOF( QUERY( pdr <* USEDIN( pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
          ('FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS'
          IN TYPEOF(pdr.used_representation)) AND
          (SIZEOF( QUERY( srwp_i <* pdr.used_representation.items |
          NOT (srwp_i.name IN ['orientation','effective length',
          'maximum length']) )) > 0 ) )) = 0 )   )) = 0;

      wr5: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
           (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
           (('FEATURE_BASED_PROCESS_PLANNING.' +
```

©ISO 2000 - All rights reserved

```
          'SHAPE_REPRESENTATION_WITH_PARAMETERS')
          IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
          ((SIZEOF([
          'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
        TYPEOF(it)) = 2) AND (it.name = 'effective length')) )) = 1)) ))
          = 0)) )) <= 1);
    wr6: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
          (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
          (('FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS')
          IN TYPEOF(pdr.used_representation)) ) |
         (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
          ((SIZEOF([
          'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
         TYPEOF(it)) = 2) AND (it.name = 'maximum length')) )) <= 1)) ))
          = 0)) )) = 0);

    wr7: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
          (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
          (('FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS')
          IN TYPEOF(pdr.used_representation)) ) |
         (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
          (('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT'
          IN TYPEOF(it))  AND (it.name = 'orientation')) )) = 1)) ))
          = 0)) )) = 0);

  END_ENTITY; -- applied_area
(*
```

Formal propositions:

WR1: The **applied_area** shall be an aspect of the shape of a **product_definition_shape**.

WR2: The **applied_area** shall have its implicit representation specified by exactly one **shape_-representation_with_parameters**.

WR3: The **applied_area** shall have an implicit representation that contains at least two and at most three **representation_item**s in its set of **items**.

WR4: The implicit representation of an **applied_area** shall contain only **representation_item**s in its set **name** of either 'orientation', 'effective length', or 'maximum length'.

WR5: Exactly one **representation_item** used for the implicit representation of the **applied_area** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'effective length'.

 ©ISO 2000 - All rights reserved

WR6: At most one **representation_item** used for the implicit representation of the **applied_area** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'maximum length'.

WR7: Exactly one **representation_item** used for the implicit representation of the **applied_area** shall be of type **placement** with a **name** of 'orientation'.

Informal Proposition

IP1: When applied to a **knurl**, the **length_measure_with_unit** with **name** of 'effective length' defines the overall length of the **knurl**.

5.2.3.1.2 applied_document_reference

A **applied_document_reference** specifies those **document_reference_item**s to which a referenced document is assigned.

EXPRESS specification:

```
*)
ENTITY applied_document_reference
  SUBTYPE OF (document_reference);
  items : SET [1:?] OF document_reference_item;
END_ENTITY;
(*
```

Attribute definition:

items: the set of **document_reference_item**s to which a referenced document is assigned.

5.2.3.1.3 applied_document_usage_constraint_assignment

A **applied_document_usage_constraint_assignment** specifies those **document_reference_item**s to which a referenced document is assigned.

EXPRESS specification:

```
*)
ENTITY applied_document_usage_constraint_assignment
  SUBTYPE OF (document_usage_constraint_assignment);
  items : SET [1:?] OF document_reference_item;
END_ENTITY;
(*
```

Attribute definition:

items: the set of **document_reference_item**s to which a referenced document is assigned.

5.2.3.1.4 applied_group_assignment

A **applied_group_assignment** specifies those **group_item**s to which an **group** is assigned.

©ISO 2000 - All rights reserved

EXPRESS specification:

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

Attribute definition:

items: the set of **group_item**s for which a particular **group** is applicable.

5.2.3.1.5 block_shape_representation

A **block_shape_representation** specifies the representation of a shape that is a rectangular volume defined as a rectangular area of a defined length. The enclosed area is defined by four straight sides with opposite sides equal in length. See ARM definition for Block_base_shape in paragraph 4.2.9 for more information.

EXPRESS specification:

```
*)
  ENTITY block_shape_representation
    SUBTYPE OF (shape_representation_with_parameters);
    WHERE
      wr1: (SIZEOF(SELF.items) = 4);

      wr2: (SIZEOF(QUERY ( it <* SELF.items |
           (('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT'
           IN TYPEOF(it)) AND (it.name = 'orientation')) )) = 1);

      wr3: (SIZEOF(QUERY ( it <* SELF.items |
           ((SIZEOF(
           ['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
           'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
            TYPEOF(it)) = 2) AND (it.name = 'length')) )) = 1);

      wr4: (SIZEOF(QUERY ( it <* SELF.items |
           ((SIZEOF([
           'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
           'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
           TYPEOF(it)) = 2) AND (it.name = 'width')) )) = 1);

      wr5: (SIZEOF(QUERY ( it <* SELF.items |
           ((SIZEOF([
           'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
           'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
           TYPEOF(it)) = 2) AND (it.name = 'height')) )) = 1);
  END_ENTITY; -- block_shape_representation
(*
```

 ©ISO 2000 - All rights reserved

Formal propositions:

WR1: The **block_shape_representation** shall contain exactly four **representation_item**s in its set of **items**.

WR2: One of the **representation_item**s used for the implicit representation of a **block_shape_-representation** shall be of type **placement** with a **name** of 'orientation'.

WR3: One of the **representation_item**s used for the implicit representation of a **block_shape_-representation** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'length'.

WR4: One of the **representation_item**s used for the implicit representation of a **block_shape_-representation** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'width'.

WR5: One of the **representation_item**s used for the implicit representation of a **block_shape_-representation** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'height'.

Informal propositions:

IP1: The **block_shape_representation** shall be defined at the center of the rectangular area in the X-Y plane with the width of the rectangle in the X direction, the height of the rectangle in the Y direction, and the length of the rectangle in the Z direction.

5.2.3.1.6 boss

A **boss** is a type of **feature_definition** that is an enclosed shape protrusion with a reference location and position. A **boss** shall be circular or complex. See ARM definition for Boss in paragraph 4.2.10 for more information.

EXPRESS specification:

```
*)
  ENTITY boss
    SUBTYPE OF (feature_definition);
    WHERE
      WR1: SELF\characterized_object.description IN
           ['circular', 'complex', 'rectangular'];

      WR2: SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
           'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
           TYPEOF (pd)) |
           NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
           'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
           (sa_occ.description = 'boss height occurrence') AND
           (SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
           'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
           'RELATED_SHAPE_ASPECT') |
           (sar.description = 'path feature component usage') AND
```

©ISO 2000 - All rights reserved

```
        ('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_DEFINING_RELATIONSHIP' IN TYPEOF (sar))) |
        ('FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT' IN
        TYPEOF (sdr.relating_shape_aspect)) AND
        (sdr.relating_shape_aspect.description = 'linear') AND
        (sdr.name = 'boss height'))) = 1))) = 1))) = 0;

 WR3:  SIZEOF( QUERY( pd <* USEDIN( SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       SIZEOF( QUERY( pdr <* USEDIN( pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.'+
      'SHAPE_REPRESENTATION_WITH_PARAMETERS'
      IN TYPEOF(pdr.used_representation)) AND
     ({1 <= SIZEOF(pdr.used_representation.items) <= 2} ) )) = 1 )) = 1;

WR4:  SIZEOF( QUERY( pd <* USEDIN( SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      SIZEOF( QUERY( pdr <* USEDIN( pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.'+
      'SHAPE_REPRESENTATION_WITH_PARAMETERS' IN
      TYPEOF(pdr.used_representation)) AND
      (SIZEOF( QUERY( srwp_i <* pdr.used_representation.items |
      (srwp_i.name = 'orientation') OR
      (srwp_i.name = 'fillet radius')))
      = SIZEOF(pdr.used_representation.items)) )) = 1 )) = 1;

WR5: SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation)) |
     NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
     (SIZEOF (
     ['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
     'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
     TYPEOF (it)) = 2) AND (it.name = 'fillet radius'))) <= 1)))
     = 0))) = 0;

WR6: (NOT (SELF\characterized_object.description = 'circular')) OR
     (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
     IN TYPEOF (pd)) |
     NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
     'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
     (sa_occ.description = 'circular profile occurrence') AND
     (SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
     'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
     'RELATED_SHAPE_ASPECT') |
     (sar.description = 'profile usage') AND
     ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
     IN TYPEOF (sar))) |
     'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE'
     IN TYPEOF (sdr.relating_shape_aspect))) = 1))) = 1))) = 0);
```

 ©ISO 2000 - All rights reserved

```
 WR7: SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
      TYPEOF (pd)) |
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (sa_occ.description = 'top condition occurrence') AND
      (SIZEOF (QUERY (fcr <* QUERY (sar <* USEDIN (sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      (sar.description = 'boss top usage') AND
      ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
      IN TYPEOF (sar))) |'FEATURE_BASED_PROCESS_PLANNING.BOSS_TOP'
      IN TYPEOF (fcr.relating_shape_aspect))) = 1))) = 1))) = 0;
 WR8: (NOT (SELF\characterized_object.description = 'circular')) OR
      (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
      TYPEOF (pd)) |
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (sa_occ.description = 'change in diameter occurrence') AND
      (SIZEOF (QUERY (fcr <* QUERY (sar <* USEDIN (sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      (sar.description = 'taper usage') AND
      ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
      IN TYPEOF (sar))) | 'FEATURE_BASED_PROCESS_PLANNING.TAPER'
      IN TYPEOF (fcr.related_shape_aspect)) )= 1))) <= 1))) = 0);
WR9: (NOT (SELF\characterized_object.description = 'complex')) OR
      (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
      TYPEOF (pd)) |
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (sa_occ.description = 'enclosed boundary occurrence') AND
      (SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      (sar.description = 'profile usage') AND
      ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
      IN TYPEOF (sar))) |
      SIZEOF (['FEATURE_BASED_PROCESS_PLANNING.NGON_CLOSED_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.CLOSED_PATH_PROFILE'] *
      TYPEOF (sdr.relating_shape_aspect)) = 1)) = 1)))= 1))) = 0);
WR10: (NOT (SELF\characterized_object.description
      IN ['complex','rectangular'])) OR
      (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF (pd)) |
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (sa_occ.description = 'change in boundary occurrence') AND
      (SIZEOF (QUERY (fcr <* QUERY (sar <* USEDIN (sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATING_SHAPE_ASPECT') |
      (sar.description = 'taper usage') AND
      ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
      IN TYPEOF (sar))) |
```

©ISO 2000 - All rights reserved

```
        ('FEATURE_BASED_PROCESS_PLANNING.TAPER'
        IN TYPEOF (fcr.related_shape_aspect)) AND
        (fcr.related_shape_aspect.description IN
        ['angle taper','directed taper']))
        = 1))) <= 1))) = 0);
  WR11: (NOT (SELF\characterized_object.description = 'rectangular')) OR
        (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
        IN TYPEOF (pd)) |
        NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
        (sa_occ.description = 'rectangular profile occurrence') AND
        (SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
        'RELATED_SHAPE_ASPECT') |
        (sar.description = 'profile usage') AND
        ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
        IN TYPEOF (sar))) |
        'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE'
        IN TYPEOF (sdr.relating_shape_aspect))) = 1))) = 1))) = 0);
  END_ENTITY;  -- Boss
(*
```

Formal propositions:

WR1: The **description** for the **boss** shall be either 'circular', 'complex', or 'rectangular'.

WR2: The **boss** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'boss height occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **name** of 'boss height' and a **description** of 'path feature component usage' in which the **relating_shape_aspect** is a **path_feature_component** with a **description** of 'linear'.

WR3: The implicit representation of the **boss** shall contain between one and two **representation_item**s.

WR4: The implicit representation of the **boss** shall contain only **representation_item**s that have a name of either 'orientation', or 'fillet radius'.

WR5: The implicit representation of the **boss** shall have at most one **representation_item** of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'fillet radius'.

WR6: If the **boss** has a **description** of 'circular' the **boss** shall be the basis shape for exactly one **shape_-aspect** with a **description** of 'circular profile occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' in which the **relating_shape_aspect** is a **circular_closed_profile**.

WR7: The **boss** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'top condition occurrence' that is the **related_shape_aspect** in exactly one **feature_component_relationship** with a **description** of 'boss top usage' in which the **relating_shape_aspect** is a **boss_top**.

WR8: The **boss** shall be the basis shape for at most one **shape_aspect** with a **description** of 'change in diameter occurrence' that is the **related_shape_aspect** in exactly one **feature_component_relationship** in with a **description** of 'taper usage' in which the **relating_shape_aspect** is a **taper**.

 ©ISO 2000 - All rights reserved

WR9: If the **boss** has a **description** of 'complex', the **boss** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'enclosed boundary occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' in which the **relating_-shape_aspect** is either a **ngon_closed_profile**, or**closed_path_profile**.

WR10: If the **boss** has a **description** of 'complex' or 'rectangular', the **boss** shall be the basis shape for at most one **shape_aspect** with a **description** of 'change in boundary occurrence' that is the **related_-shape_aspect** in exactly one **feature_component_relationship** with a **description** of 'taper usage' in which the **relating_shape_aspect** is a **taper** with a **description** of 'angle taper' or 'directed taper'.

WR11: If the **boss** has a **description** of 'rectangular', the **boss** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'rectangular profile occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' in which the **relating_-shape_aspect** is a **rectangular_closed_profile**.

Informal propositions:

IP1: The location shall be defined at the top of the **boss** at the approximate center of the feature.

IP2: The **boss** may be positioned on the face of a part with the Z-axis in the direction away from the part, or at the top of the Boss with the Z-axis in the direction of the part face.

IP3: The location, X direction, and Y direction of the **path_feature_component** with a **description** of 'boss height' shall be the same as the **boss**.

IP4: The placement of the **circular_closed_profile** that defines the diameter of a **boss** with a **description** of 'circular' shall be with the origin, X direction, and Y direction equal to that of the **boss**.

IP5: The placement of the **rectangular_closed_profile**, **ngon_closed_profile**, or **closed_path_profile** that defines the shape of a **boss** with a **description** of 'complex' shall be with the origin, X direction, and Y direction equal to that of the **boss**.

5.2.3.1.7 boss_top

A **boss_top** is a type of **shape_aspect** that is the end condition for a **boss feature_definition**. See ARM definition for Boss_top_condition in paragraph 4.2.11 for more information.

EXPRESS specification:

```
*)
  ENTITY boss_top
    SUBTYPE OF (shape_aspect);
    WHERE
      wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
            IN TYPEOF(SELF.of_shape.definition));

      wr2: (SELF.description IN ['planar','complex']);

      wr3: ((NOT (SELF.description = 'planar')) OR
           (SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
```

©ISO 2000 - All rights reserved

```
          (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION')
          IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0));

    wr4: ((NOT (SELF.description = 'planar')) OR
          (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
          (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
         (('FEATURE_BASED_PROCESS_PLANNING.LOCATION_SHAPE_REPRESENTATION')
          IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0));

    wr5: (NOT (SELF.description = 'complex')) OR
          (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
          (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.'+
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
          ('FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION')
          IN TYPEOF(pdr.used_representation) )) = 1)) )) = 0);

    wr6: (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'  +
          'RELATING_SHAPE_ASPECT') |
          ((sar.description = 'boss top usage') AND
          (sar.name  IN ['boss height start','boss height end']))AND
          ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
          IN TYPEOF(sar)))  |
          ((fcr.related_shape_aspect.description =
          'top condition occurrence') AND
          ('FEATURE_BASED_PROCESS_PLANNING.BOSS'
          IN TYPEOF(fcr.related_shape_aspect.of_shape.definition))) ))
          >= 1);

    wr7: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
          (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
          (('FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS')
          IN TYPEOF(pdr.used_representation)) )) <= 1)) )) = 0);

    wr8: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
          (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
          (('FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS')
          IN TYPEOF(pdr.used_representation)) ) |
          (NOT (SIZEOF(impl_rep.used_representation.items) = 1)) ))
          = 0)) )) = 0);

  WR9: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
```

 ©ISO 2000 - All rights reserved

```
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS'
        IN TYPEOF (pdr.used_representation)) |
        NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
        (SIZEOF (
        ['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
        TYPEOF (it)) = 2) AND (it.name = 'top radius'))) <= 1)))
        = 0))) = 0;
  END_ENTITY; -- boss_top
(*
```

Formal propositions:

WR1: The **boss_top** shall be an aspect of the shape of a **feature_component_definition**.

WR2: The **description** of the **boss_top** shall be either 'planar' or 'complex'.

WR3: If the **description** of the **boss_top** is 'planar', the **boss_top** shall have exactly one **direction_-shape_representation.**

WR4: If the **description** of the **boss_top** is 'planar', the **boss_top** shall have exactly one **location_-shape_representation**.

WR5: If the **description** of the **boss_top** is 'complex', the **boss_top** shall have exactly one **face_shape_-representation**.

WR6: The **boss_top** shall be the **relating_shape_aspect** in at least one **feature_component_-relationship** with a **description** of 'boss top usage' and a **name** of either 'boss height start' or 'boss height end' in which the **related_shape_aspect** is a **shape_aspect** of a **boss** with a **name** of 'top condition occurrence'.

WR7: The **boss_top** shall have at most one **shape_representation_with_parameters** to specify its implicit representation.

WR8: The **boss_top** shall be represented implicitly by exactly one **representation_item**.

WR9: The implicit representation of the **boss_top** shall have at most one **representation_item** of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'top radius'.

Informal propositions:

IP1: The location of the **boss_top** shall be at the center of the face that is the mating face to a **boss** feature.

IP2: The **boss_top** shall be defined on the mating face in the X-Y plane with the Z direction coincident to that of the mating **boss** feature.

©ISO 2000 - All rights reserved

5.2.3.1.8 chamfer

A **chamfer** is a type of **transition_feature** that is the representation of a linear transition between two **shape_aspects**. A **chamfer** defines the outside profile of a part. See ARM definition for Fillet in paragraph 4.2.19 for more information.

EXPRESS specification:

```
*)
  ENTITY chamfer
    SUBTYPE OF (transition_feature);
    WHERE
      wr1: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
           (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
           (('FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
           IN TYPEOF(pdr.used_representation)) AND
        (pdr.used_representation.name = 'chamfer face')) )) = 1)) )) = 0);

      wr2: (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
           'RELATING_SHAPE_ASPECT') |
           ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
           IN TYPEOF(sar)) ) |
           (('FEATURE_BASED_PROCESS_PLANNING.CHAMFER_OFFSET'
           IN TYPEOF(fcr.related_shape_aspect)) AND
          (fcr.related_shape_aspect.description = 'first offset')) )) = 1);

      wr3: (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
           'RELATING_SHAPE_ASPECT') |
           ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
           IN TYPEOF(sar)) ) |
           (('FEATURE_BASED_PROCESS_PLANNING.CHAMFER_OFFSET'
           IN TYPEOF(fcr.related_shape_aspect)) AND
           (fcr.related_shape_aspect.description = 'second offset')) )) =
1);
  END_ENTITY; -- chamfer
(*
```

Formal propositions:

WR1: The **chamfer** shall have exactly one **face_shape_representation**s in the role of the chamfer face.

WR2: The **chamfer** shall be related to exactly one **chamfer_offset** that specifies the first offset for the chamfer.

WR3: The **chamfer** shall be related to exactly one **chamfer_offset** that specifies the second offset for the chamfer.

 ©ISO 2000 - All rights reserved

5.2.3.1.9 chamfer_offset

A **chamfer_offset** is a type of **shape_aspect** that is the representation of the linear setbacks from the edge of two surfaces to create a **chamfer feature_definition**. See ARM definition for First_offset in paragraph 4.2.74, Second_chamfer_offset in paragraph 4.2.196, and Second_offset in paragraph 4.2.197 for more information.

EXPRESS specification:

```
*)
  ENTITY chamfer_offset
    SUBTYPE OF (shape_aspect);
    WHERE
      wr1: (SELF.description IN ['first offset','second offset']);

      wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
           (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
           (('FEATURE_BASED_PROCESS_PLANNING.' +
           'SHAPE_REPRESENTATION_WITH_PARAMETERS')
           IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0);

      wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
           (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
           (('FEATURE_BASED_PROCESS_PLANNING.' +
           'SHAPE_REPRESENTATION_WITH_PARAMETERS')
           IN TYPEOF(pdr.used_representation)) ) |
           (NOT (SIZEOF(impl_rep.used_representation.items) = 1)) ))
           = 0)) )) = 0);

      wr4: ((NOT (SELF.description = 'first offset')) OR
           (SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
           (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
           (('FEATURE_BASED_PROCESS_PLANNING.' +
           'SHAPE_REPRESENTATION_WITH_PARAMETERS')
           IN TYPEOF(pdr.used_representation)) ) |
           (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
           ((SIZEOF([
           'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
           'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
           TYPEOF(it)) = 2) AND (it.name = 'offset amount')) )) = 1)) ))
           = 0)) )) = 0));

      wr5: ((NOT (SELF.description = 'first offset')) OR
           (SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
           (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
           (('FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
```

©ISO 2000 - All rights reserved

```
          IN TYPEOF(pdr.used_representation)) AND
          (pdr.used_representation.name = 'first face shape')) )) = 1)) ))
          = 0));

     wr6: ((NOT (SELF.description = 'second offset')) OR
          (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
          (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
          (('FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS')
          IN TYPEOF(pdr.used_representation)) ) |
          (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
          (((SIZEOF([
          'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
          TYPEOF(it)) = 2) AND (it.name = 'offset amount')) OR
          ((SIZEOF([
          'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'offset angle'))) )) = 1)) ))
          = 0)) )) = 0));

     wr7: ((NOT (SELF.description = 'second offset')) OR
          (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
          (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
          (('FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
          IN TYPEOF(pdr.used_representation)) AND
         (pdr.used_representation.name = 'second face shape')) )) = 1)) ))
          = 0));

     wr8: (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
          'RELATED_SHAPE_ASPECT') |
          (('FEATURE_BASED_PROCESS_PLANNING.' +
          'FEATURE_COMPONENT_RELATIONSHIP')
          IN TYPEOF(sar)) ) |
          ('FEATURE_BASED_PROCESS_PLANNING.CHAMFER'
          IN TYPEOF(sdr.relating_shape_aspect)) )) = 1);
  END_ENTITY; -- chamfer_offset
(*
```

Formal propositions:

WR1: The **description** for the **chamfer_offset** shall be either 'first offset' or 'second offset'.

WR2: The **chamfer_offset** shall have its implicit representation specified by exactly one **shape_-representation_with_parameters**.

WR3: The **chamfer_offset** shall have an implicit representation that contains exactly one **representation_item** in its set of **items**.

 ©ISO 2000 - All rights reserved

WR4: If the **description** of the **chamfer_offset** is 'first offset', exactly one **representation_item** used for the implicit representation of the **chamfer_offset** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'offset amount'.

WR5: If the **description** of the **chamfer_offset** is 'first offset', the **chamfer_offset** shall have exactly one **face_shape_representation** with a **name** of 'first face shape'.

WR6: If the **description** of the **chamfer_offset** is 'second offset', exactly one **representation_item** used for the implicit representation of the **chamfer_offset** shall be either of type **measure_representation_-item** and **length_measure_with_unit** with a **name** of 'offset amount', or of type **measure_-representation_item** and **plane_angle_measure_with_unit** with a **name** of 'offset angle'.

WR7: If the **description** of the **chamfer_offset** is 'second offset', the **chamfer_offset** shall have exactly one **face_shape_representation** with a **name** of 'second face shape'.

WR8: The **chamfer_offset** shall be the **related_shape_aspect** in at exactly one **feature_component_-relationship** in which the **relating_shape_aspect** is a **chamfer**.

5.2.3.1.10 circular_closed_profile

A **circular_closed_profile** is a type of **shape_aspect** that is an enclosed 2D area defined by a radius, with orientation and location. The location is defined to be at the center of the circular arc, and the orientation of the **circular_closed_profile** is the X-Y plane. See ARM definition for Circular_closed_-profile in paragraph 4.2.22 for more information.

EXPRESS specification:

```
*)
  ENTITY circular_closed_profile
    SUBTYPE OF (shape_aspect);
    WHERE
      wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
           IN TYPEOF(SELF.of_shape.definition));

      wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
           (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
           (('FEATURE_BASED_PROCESS_PLANNING.' +
           'SHAPE_REPRESENTATION_WITH_PARAMETERS')
           IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0);

      wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
           (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
           'FEATURE_BASED_PROCESS_PLANNING.'+
           'SHAPE_REPRESENTATION_WITH_PARAMETERS')
           IN TYPEOF(pdr.used_representation)) ) |
           (NOT (SIZEOF(impl_rep.used_representation.items) = 2)) ))
           = 0)) )) = 0);
```

©ISO 2000 - All rights reserved

```
      wr4: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
           (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
           (('FEATURE_BASED_PROCESS_PLANNING.' +
           'SHAPE_REPRESENTATION_WITH_PARAMETERS')
           IN TYPEOF(pdr.used_representation)) ) |
           NOT(SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
           (('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT'
           IN TYPEOF(it))
           AND (it.name = 'orientation')) )) = 1) )) = 0)) )) = 0);

      wr5: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
           (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
           (('FEATURE_BASED_PROCESS_PLANNING.' +
           'SHAPE_REPRESENTATION_WITH_PARAMETERS')
           IN TYPEOF(pdr.used_representation)) ) |
           (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
           ((SIZEOF([
           'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
           'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
           TYPEOF(it)) = 2) AND (it.name = 'diameter')) )) = 1)) ))
           =  0)) )) = 0);
  END_ENTITY; -- circular_closed_profile
(*
```

Formal propositions:

WR1: The **circular_closed_profile** shall be an aspect of the shape of a **feature_component_definition**.

WR2: The **circular_closed_profile** shall have exactly one implicit representation defined by a relationship to a **shape_representation_with_parameters**.

WR3: The **shape_representation_with_parameters** that represents the **circular_closed_profile** shall contain two **representation_item**s in its set of **items**.

WR4: Exactly one **representation_item** used for the implicit representation of a **circular_closed_-profile** shall be of type **placement** with a **name** of 'orientation'.

WR5: Exactly one **representation_item** used the implicit representation of a **circular_closed_profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'diameter'.

Informal propositions:

IP1: The location of the **circular_closed_profile** shall be defined at the center of the circle.

IP2: The **circular_closed_profile** shall be definied in the X-Y plane.

 ©ISO 2000 - All rights reserved

5.2.3.1.11 circular_pattern

A **circular_pattern** is a type of **replicate_feature** that relates a base feature and one or more **shape_-aspect**s which are the placement of the base feature at a specified circular location on the base part. A diameter is defined with a placement on the part and a number of base features are equally spaced around this diameter. See ARM definition for Circular_pattern in paragraph 4.2.28 for more information.

EXPRESS specification:

```
*)
  ENTITY circular_pattern
    SUBTYPE OF (replicate_feature);
    WHERE
      wr1: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
           ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
           IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN
           (pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
           ((SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
           'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
           + 'RELATING_SHAPE_ASPECT') |
         ((('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP')
           IN TYPEOF(sar))) ) |
           ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT'
           IN TYPEOF(sdr.related_shape_aspect)) )) = 1)) )) <= 3)) )) = 0);

      wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
            (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
            (('FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS')
            IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0);
      wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
            (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
            ('FEATURE_BASED_PROCESS_PLANNING.'+
            'SHAPE_REPRESENTATION_WITH_PARAMETERS'
            IN TYPEOF(pdr.used_representation)) ) |
            (NOT ((SIZEOF (impl_rep.used_representation.items) >= 3)
            AND (SIZEOF(impl_rep.used_representation.items) <= 5))) ))
            = 0)) )) = 0);

     wr4:  (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
            (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
            ('FEATURE_BASED_PROCESS_PLANNING.'+
            'SHAPE_REPRESENTATION_WITH_PARAMETERS'
            IN TYPEOF(pdr.used_representation)) ) |
            (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
            ((SIZEOF(
            ['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
            'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
```

©ISO 2000 - All rights reserved

```
          TYPEOF(it)) = 2) AND (it.name = 'diameter')) )) <= 1)) ))
          = 0)) )) = 0);
     wr5: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
          (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
          ('FEATURE_BASED_PROCESS_PLANNING.'+
          'SHAPE_REPRESENTATION_WITH_PARAMETERS'
          IN TYPEOF(pdr.used_representation)) ) |
          (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
          ((SIZEOF(
          ['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
         'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT'] *
          TYPEOF(it)) = 2) AND (it.name = 'base feature rotation')) ))
          <= 1)) )) = 0)) )) = 0);
     wr6: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
          (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
          ('FEATURE_BASED_PROCESS_PLANNING.'+
          'SHAPE_REPRESENTATION_WITH_PARAMETERS'
          IN TYPEOF(pdr.used_representation)) ) |
          (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
          (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
          IN TYPEOF(it)) AND
          ('FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE'
          IN TYPEOF(it\measure_with_unit.value_component)) AND
          (it.name = 'number of features')) )) = 1)) )) = 0)) )) = 0);
     wr7: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
          (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
          ('FEATURE_BASED_PROCESS_PLANNING.'+
          'SHAPE_REPRESENTATION_WITH_PARAMETERS'
          IN TYPEOF(pdr.used_representation)) ) |
          (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
          ((SIZEOF(
          ['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
         'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT'] *
          TYPEOF(it)) = 2) AND (it.name = 'angular spacing'))))= 1)) ))
          = 0)) )) = 0);
     wr8: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
          (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
          ('FEATURE_BASED_PROCESS_PLANNING.'+
          'SHAPE_REPRESENTATION_WITH_PARAMETERS'
          IN TYPEOF(pdr.used_representation)) ) |
          (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
          (('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT'
          IN TYPEOF(it)) AND (it.name = 'orientation')) )) = 1)) ))
          = 0)) )) = 0);
  END_ENTITY; -- circular_pattern
(*
```

 ©ISO 2000 - All rights reserved

Formal propositions:

WR1: The **circular_pattern** shall be the **relating_shape_aspect** in no more than three **shape_aspect_-relationship**s.

WR2: The **circular_pattern** shall have exactly one implicit representation.

WR3: The implicit representation of the **circular_pattern** shall contain between two and five **representation_item**s in its set of **items**.

WR4: The implicit representation of the **circular_pattern** shall contain at most one **representation_-item** of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'diameter'.

WR5: The implicit representation of the **circular_pattern** shall contain at most one **representation_-item** of type **measure_representation_item** and **plane_angle_measure_with_unit** with a **name** of 'base feature rotation'.

WR6: The implicit representation of the **circular_pattern** shall contain exactly one **representation_-item** of type **measure_representation_item** with a **value_component** of type **count_measure** and a **name** of 'number of features'.

WR7: The implicit representation of the **circular_pattern** shall contain exactly one **representation_-item** of type **measure_representation_item** and **plane_angle_measure_with_unit** with a **name** of 'angular spacing'.

WR8: The implicit representation of the **circular_pattern** shall contain exactly one **representation_-item** of type **placement** with a **name** of 'orientation'.

Informal propositions:

IP1: The location of the **circular_pattern** shall be defined at the center of the circle.

IP2: The **circular_pattern** shall be defined in the X-Y plane, with the X direction intersecting the placement position of the first base feature.

5.2.3.1.12 closed_path_profile

A **closed_path_profile** is a type of **shape_aspect** that is a closed two dimensional area defined by a connected set of curves with a location and orientation. The start vertex and end vertex for the set of curves are the same point creating an enclosed area. The **closed_path_profile** is located at the approximate center of the enclosed area, and the orientation is in the X-Y plane. See ARM definition for General_closed_profile in paragraph 4.2.81 for more information.

EXPRESS specification:

```
*)
  ENTITY closed_path_profile
    SUBTYPE OF (shape_aspect);
    WHERE
      wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
```

©ISO 2000 - All rights reserved

```
            IN TYPEOF(SELF.of_shape.definition));

      wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
            'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
            'DEFINITION') |
            (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
            (('FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS')
            IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0);

      wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
            (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
            (('FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS')
            IN TYPEOF(pdr.used_representation)) ) |
            (NOT (SIZEOF(impl_rep.used_representation.items) = 1)) ))
            = 0)) )) = 0);

      wr4: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
            (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
            (('FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS')
            IN TYPEOF(pdr.used_representation)) ) |
            (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
            (('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT'
            IN TYPEOF(it)) AND (it.name = 'orientation')) )) = 1)) ))
            = 0)) )) = 0);

      wr5: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
            'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
            'DEFINITION') |
            (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
            (('FEATURE_BASED_PROCESS_PLANNING.PATH_SHAPE_REPRESENTATION')
            IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0);
  END_ENTITY; -- closed_path_profile
(*
```

Formal propositions:

WR1: The **closed_path_profile** shall be an aspect of the shape of a **feature_component_definition**.

WR2: The **closed_path_profile** shall have exactly one implicit representation defined by a relationship to a **shape_representation_with_parameters**.

WR3: The **shape_representation_with_parameters** that represents the **closed_path_profile** shall contain one **representation_item**s in its set of **items**.

 ©ISO 2000 - All rights reserved

WR4: Exactly one **representation_item** used for the implicit representation of a **closed_path_profile** shall be of type **placement** with a **name** of 'orientation'.

WR5: The **closed_path_profile** shall have exactly one **path_shape_representation**.

Informal propositions:

IP1: The location of the **closed_path_profile** shall be defined at the approximate center of the enclosed area.

IP2: The **closed_path_profile** shall be defined in the X-Y plane.

5.2.3.1.13 composite_hole

A **composite_hole** is a feature that is the representation of two **round_hole**s for the purpose of creating two commonly used **round_hole** combinations, the countersunk hole and the counterbore hole. The counterbore hole is the representation of two **round_hole**s with one having a larger diameter than the second. The countersunk hole is the representation of two **round_hole**s, the first having a larger diameter than the second with a linear decrease in diameter until it is the same as the second. This linear decrease shall be represented by a **taper**.

EXPRESS specification:

```
*)
  ENTITY composite_hole
    SUBTYPE OF (compound_feature);
    WHERE
      wr1: (SELF\characterized_object.description IN ['counterbore',
               'countersunk']);

      wr2:  (SIZEOF (QUERY (pds <* USEDIN (SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')|
            ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
            IN TYPEOF (pds)) AND
            (SIZEOF (QUERY (csa <* USEDIN (pds,
            'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')|
            ('FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT'
            IN TYPEOF (csa)) AND
            (SIZEOF (QUERY (sar <*  csa.component_relationships |
          ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
            IN TYPEOF(sar)) AND
            ('FEATURE_BASED_PROCESS_PLANNING.ROUND_HOLE'
            IN TYPEOF (sar.related_shape_aspect))  )) = 2)  )) = 1) )) = 1);

      WR3: (NOT (SELF\characterized_object.description = 'countersunk')) OR
            (SIZEOF (QUERY (pds <* USEDIN (SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')|
            ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
            IN TYPEOF (pds)) AND
            (SIZEOF (QUERY (csa <* USEDIN (pds,
            'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')|
            ('FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT'
            IN TYPEOF (csa)) AND
            (SIZEOF (QUERY (sar <*  csa.component_relationships |
            ('FEATURE_BASED_PROCESS_PLANNING.ROUND_HOLE'
```

©ISO 2000 - All rights reserved

```
            IN TYPEOF (sar.related_shape_aspect))
            AND
            NOT(SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN
            (sar.related_shape_aspect,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
            ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
            IN TYPEOF(pd)) ) |
            (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
            'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
            ((sa_occ.description = 'change in diameter occurrence')  AND
            (SIZEOF(QUERY ( fcr2 <* QUERY ( sar2 <* USEDIN(sa_occ,
            'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
            'RELATING_SHAPE_ASPECT') |
            ((sar2.description = 'taper usage') AND
          (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP')
            IN TYPEOF(sar2))) ) |
            ('FEATURE_BASED_PROCESS_PLANNING.TAPER'
            IN TYPEOF(fcr2.related_shape_aspect)) )) = 1)) )) = 0)) )) = 0)
            )) = 1)  )) = 1) )) = 1);
  END_ENTITY; -- composite_hole
(*
```

Formal propositions:

WR1: The **description** of the **composite_hole** shall be either 'counterbore' or 'countersunk'.

WR2: The **composite_hole** shall be the **relating_shape_aspect** in exactly two instances of **feature_componet_relationship** with a **related_shape_aspect** of type **round_hole**.

WR3: If the **description** of the **composite_hole** is 'countersunk', exactly one **round_hole** that comprises it shall be a tapered hole.

Informal propositions:

IP1: Each **round_hole**s that comprise the **composite_hole** shall not have diameters that are equal in length.

5.2.3.1.14 compound_feature

A **compound_feature** is a type of **feature_definition** that is the representation of several **instanced_-feature**s and their relationships to one another for the purpose of creating unique user defined features.

EXPRESS specification:

```
*)
  ENTITY compound_feature
    SUBTYPE OF (feature_definition);
      WHERE
WR1:  SIZEOF( QUERY( pds <* USEDIN( SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pds))  AND
     (SIZEOF( QUERY( csa <* USEDIN( pds,
     'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
```

 ©ISO 2000 - All rights reserved

```
     ((csa.name='compound feature in solid') AND
     ('FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT'
     IN TYPEOF(csa))) )) = 1)  )) = 1;

WR2:  SIZEOF( QUERY( pds <* USEDIN( SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pds))  AND
     (SIZEOF( QUERY( csa <* USEDIN( pds,
     'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
     'FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT'
     IN TYPEOF(csa) )) = 1)  )) = 1;

wr3: (SIZEOF(QUERY ( pds <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     (('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
     IN TYPEOF(pds)) AND (SIZEOF(QUERY ( csa <* USEDIN(pds,
     'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
     (('FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT'
     IN TYPEOF(csa)) AND
     (SIZEOF(QUERY ( fcr <* csa.component_relationships |
     (NOT ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
     IN TYPEOF(fcr))) )) = 0)) )) = 1)) )) = 1);

WR4: SIZEOF (QUERY (pds <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')|
     ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
     IN TYPEOF (pds)) AND
     (SIZEOF (QUERY (csa <* USEDIN (pds,
     'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')|
     ('FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT'
     IN TYPEOF (csa)) AND
     (SIZEOF (QUERY (sar <*  csa.component_relationships |
     (SIZEOF ([
     'FEATURE_BASED_PROCESS_PLANNING.INSTANCED_FEATURE',
     'FEATURE_BASED_PROCESS_PLANNING.TRANSITION_FEATURE',
     'FEATURE_BASED_PROCESS_PLANNING.APPLIED_AREA'] *
     TYPEOF (sar.related_shape_aspect)) = 1) )) = 2)  )) = 1) )) = 1;

wr5: (SIZEOF(QUERY ( pds <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pds)) AND
      (SIZEOF(QUERY ( csa <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (('FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT'
      IN TYPEOF(csa)) AND
      (SIZEOF(QUERY ( sar <* csa.component_relationships |
      ('FEATURE_BASED_PROCESS_PLANNING.THREAD'
      IN TYPEOF(sar.related_shape_aspect)) )) = 0)
      ) )) = 1)) )) = 1);

WR6: (SIZEOF (QUERY (pds <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')|
     ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
     IN TYPEOF (pds)) AND   (SIZEOF (QUERY (csa <* USEDIN (pds,
     'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')|
     ('FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT'
     IN TYPEOF (csa))  AND
     (SIZEOF (QUERY (sar <*  csa.component_relationships |
     (('FEATURE_BASED_PROCESS_PLANNING.COMPOUND_FEATURE'
```

©ISO 2000 - All rights reserved

```
     IN TYPEOF (sar.related_shape_aspect))  AND
     (sar.related_shape_aspect\characterized_object.name <>
     SELF\characterized_object.name)) )) = 0)  )) = 1) )) = 1);

END_ENTITY;  -- compound_feature
(*
```

Formal propositions:

WR1: The **name** of the **compound_feature** shall be 'compound feature in solid'.

WR2: Each instance of **compound_feature** shall have a shape that has exactly one **shape_aspect** identified on it that is of type **composite_shape_aspect**.

WR3: The **shape_aspect_relationship**s that are used to relate the single features with the **composite_shape_aspect** shall be **feature_component_relationship**s.

WR4: The **composite_shape_aspect** defining a **compound_feature** shall be the **relating_shape_aspect** only in **shape_aspect_relationships** with a **related_shape_aspect** that is of type **instanced_feature**, **applied_area**, or **transition_feature**.

WR5: The **composite_shape_aspect** defining a **compound_feature** shall be the **relating_shape_aspect** only in **shape_aspect_relationships** with a **related_shape_aspect** that is not of type **thread**.

NOTE - A **thread** feature can not be directly related to a **compound_feature**. The **applied_area** of the **thread** is related to a **compound_feature**.

WR6: The **compound_feature** shall only have constituent **compound_feature**s of the same type.

5.2.3.1.15 cylindrical_shape_representation

A **cylindrical_shape_representation** specifies representation of a shape that is a cylindrical volume defined as a circular area of a defined length. The enclosed area is defined by a circle with a specified radius. See ARM definition for Cylindrical_base_shape in paragraph 4.2.47 for more information.

EXPRESS specification:

```
*)
  ENTITY cylindrical_shape_representation
    SUBTYPE OF (shape_representation_with_parameters);
    WHERE
      wr1: (SIZEOF(SELF.items) = 3);

      wr2: (SIZEOF(QUERY ( it <* SELF.items | ((
           'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT'
           IN TYPEOF(it)) AND (it.name = 'orientation')) )) = 1);

      wr3: (SIZEOF(QUERY ( it <* SELF.items | ((SIZEOF([
           'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
           'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
            TYPEOF(it)) = 2) AND (it.name = 'length')) )) = 1);
```

 ©ISO 2000 - All rights reserved

```
      wr4: (SIZEOF(QUERY ( it <* SELF.items | ((SIZEOF([
           'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
           'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
            TYPEOF(it)) = 2) AND (it.name = 'diameter')) )) = 1);
  END_ENTITY; -- cylindrical_shape_representation
(*
```

Formal propositions:

WR1: The **cylindrical_shape_representation** shall contain exactly three **representation_item**s in its set of **items**.

WR2: One of the **representation_item**s used for the implicit representation of a **cylindrical_shape_-representation** shall be of type **placement** with a **name** of 'orientation'.

WR3: One of the **representation_item**s used for the implicit representation of a **cylindrical_shape_-representation** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'length'.

WR4: One of the **representation_item**s used for the implicit representation of a **cylindrical_shape_-representation** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'diameter'.

Informal propositions:

IP1: The location of the **cylindrical_shape_representation** shall be defined to be at the center of the circle that defines the cylinder.

IP2: The **cylindrical_shape_representation** shall be definied by forming a circular profile in the X-Y plane, and the length along the z direction.

5.2.3.1.16 directed_dimensional_location

A **directed_dimension_location** specifies is a type of **dimension_location** that identifies the direction to measure the location dimension.

EXPRESS specification:

```
*)
  ENTITY directed_dimensional_location
    SUBTYPE OF (dimensional_location);
  END_ENTITY; -- directed_dimensional_location
(*
```

Attribute definitions:

SELF\shape_aspect_relationship.relating_shape aspect: the origin of the directed dimension.

SELF\shape_aspect_relationship.related_shape_aspect: the target of the directed dimension.

©ISO 2000 - All rights reserved

5.2.3.1.17 direction_shape_representation

A **direction_shape_representation** is a type of **representation** that represents a direction vector in two or three dimensional space.

EXPRESS specification:

```
*)
  ENTITY direction_shape_representation
    SUBTYPE OF (shape_representation);
    WHERE
      wr1: (SIZEOF(SELF.items) = 1);

      wr2: (SIZEOF(QUERY ( it <* SELF.items |
           (NOT (
           'FEATURE_BASED_PROCESS_PLANNING.DIRECTION' IN TYPEOF(it))) ))
           = 0);
  END_ENTITY; -- direction_shape_representation

(*
```

Formal propositions:

WR1: The **direction_shape_representation** shall have exactly one **representation_item** in its set of **items**.

WR2: The geometric element that is used to represent the **direction_shape_representation** shall be a **direction**.

5.2.3.1.18 document_file

A document_file is a type of **document** and **characterized_object** that is the representation of the physical document that contains the information about marking, knurl, or thread specifications.

EXPRESS specification

```
*)
ENTITY document_file
   SUBTYPE OF (characterized_object, document);
   WHERE
 wr1: (SIZEOF(QUERY(adr<* QUERY(dr <* USEDIN(SELF,
   'FEATURE_BASED_PROCESS_PLANNING.DOCUMENT_REFERENCE.ASSIGNED_DOCUMENT') |
      'FEATURE_BASED_PROCESS_PLANNING.APPLIED_DOCUMENT_REFERENCE'
      IN TYPEOF(dr))|
     'FEATURE_BASED_PROCESS_PLANNING.EXTERNALLY_DEFINED_FEATURE_DEFINITION'
      IN TYPEOF(adr.items)
      ))=1) OR
      (SIZEOF(QUERY (duc <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.DOCUMENT_USAGE_CONSTRAINT.SOURCE') |
      NOT
      (SIZEOF(QUERY(aduc<* QUERY(duca <* USEDIN(duc,
    'FEATURE_BASED_PROCESS_PLANNING.DOCUMENT_USAGE_CONSTRAINT_ASSIGNMENT.'+
      'ASSIGNED_DOCUMENT_USAGE') |
      'FEATURE_BASED_PROCESS_PLANNING.'+
      'APPLIED_DOCUMENT_USAGE_CONSTRAINT_ASSIGNMENT'
```

 ©ISO 2000 - All rights reserved

```
      IN TYPEOF(duca))|
     'FEATURE_BASED_PROCESS_PLANNING.EXTERNALLY_DEFINED_FEATURE_DEFINITION'
      IN TYPEOF(aduc.items)
      ))=1))) = 0);

 wr2: (SIZEOF(QUERY(drt <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.'+
      'DOCUMENT_REPRESENTATION_TYPE.REPRESENTED_DOCUMENT') |
      (drt.name='physical')))=1);
  END_ENTITY; -- document_file
(*
```

Formal propositions:

WR1: The **document_file** shall be either the **associated_document** in exactly one **applied_document_-reference** that contains one or more **external_defined_feature_definition** in its set of **items**, or a **source** in exactly one **applied_document_usage_constraint_assignment** that contains one or more **external_-defined_feature_definition** in its set of **items**.

WR2: The **document_fil**e shall be the **represented_file** in exactly one **document_representation_type** with **name** = 'physical'.

5.2.3.1.19 edge_round

An **edge_round** is a type of **transition_feature** that is the representation of a circular convex transition between two **shape_aspects**. An **edge_round** defines the outside profile of a part. See ARM definition for Edge_round in paragraph 4.2.67 for more information.

EXPRESS specification:

```
*)
ENTITY edge_round
  SUBTYPE OF (transition_feature);
WHERE
  WR1: (NOT (SELF\shape_aspect.description = 'constant radius')) OR
       (SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <*  USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation))) = 1))) = 0);

  WR2: (NOT (SELF\shape_aspect.description = 'constant radius')) OR
       (SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       (NOT (SIZEOF (impl_rep.used_representation.items) >= 1)
       AND  (SIZEOF (impl_rep.used_representation.items) <= 3))))
       = 0))) = 0);
```

©ISO 2000 - All rights reserved

```
WR3: (NOT (SELF.description = 'constant radius')) OR
     (SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
    'FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation)) |
     NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
     (SIZEOF
     (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
     'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
     TYPEOF (it)) = 2) AND (it.name = 'radius'))) = 1))) = 0))) = 0);

WR4: (NOT (SELF.description = 'constant radius')) OR
     (SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
    'FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation)) |
     NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
     (SIZEOF
     (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
     'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
     TYPEOF (it)) = 2) AND (it.name = 'first offset'))) <= 1)))
     = 0))) = 0);

WR5: (NOT (SELF.description = 'constant radius')) OR
     (SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
    'FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation)) |
     NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
     (SIZEOF
     (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
     'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
     TYPEOF (it)) = 2) AND (it.name = 'second offset'))) <= 1)))
     = 0))) = 0);

WR6: SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     ('FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
     IN TYPEOF (pdr.used_representation)) AND
     (pdr.used_representation.name = 'edge round face'))) = 1))) = 0;

WR7: SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     ('FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
     IN TYPEOF (pdr.used_representation)) AND
     (pdr.used_representation.name = 'first face shape'))) = 1))) = 0;
```

 ©ISO 2000 - All rights reserved

```
  WR8: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
       IN TYPEOF (pdr.used_representation)) AND
       (pdr.used_representation.name = 'second face shape'))) = 1))) = 0;
END_ENTITY; -- edge_round

(*
```

Formal propositions:

WR1: An **edge_round** has an implicit representation if and only if it has a **description** of 'constant radius'.

WR2: If the description of the **edge_round** is 'constant radius', the implicit representation shall contain at least one and at most three **representation_item**s in its set of **items**.

WR3: If the description of the **edge_round** is 'constant radius', exactly one **representation_item** used for the implicit representation of the **edge_round** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'radius'.

WR4: If the description of the **edge_round** is 'constant radius', at most one **representation_item** used for the implicit representation of the **edge_round** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'first offset'.

WR5: If the description of the **edge_round** is 'constant radius', at most one **representation_item** used for the implicit representation of the **edge_round** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'second offset'.

WR6: The **edge_round** shall have exactly one **face_shape_representation** with a **name** of 'edge round face'.

WR7: The **edge_round** shall have exactly one **face_shape_representation** with a **name** of 'first face shape'.

WR8: The **edge_round** shall have exactly one **face_shape_representation** with a **name** of 'second face shape'.

5.2.3.1.20 externally_defined_feature_definition

An **externally_defined_feature_definition** relates the document containing a feature specification to a **feature_definition**. See ARM definition for Catalogue_thread in paragraph 4.2.18, Catalogue_marking in paragraph 4.2.17, and Catalogue_knurl in paragraph 4.2.16 for more information.

©ISO 2000 - All rights reserved

EXPRESS specification:

```
*)
  ENTITY externally_defined_feature_definition
    SUBTYPE OF (feature_definition, externally_defined_item);
    WHERE
   wr1: (((SELF\characterized_object.description = 'thread') AND
        (SELF\externally_defined_item.item_id='external thread') AND
        (SELF\externally_defined_item.source.source_id=
              'external feature specification') ) OR
        ((SELF\characterized_object.description = 'marking') AND
        (SELF\externally_defined_item.item_id='external marking') AND
        (SELF\externally_defined_item.source.source_id=
             'external feature specification')) OR
        ((SELF\characterized_object.description = 'knurl') AND
        (SELF\externally_defined_item.item_id='external knurl') AND
        (SELF\externally_defined_item.source.source_id=
             'external feature specification')));

      WR2: ((NOT (SELF\characterized_object.description = 'thread')) OR
           (SIZEOF( QUERY( pd <* USEDIN( SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
           SIZEOF( QUERY( pdr <* USEDIN( pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
           ('FEATURE_BASED_PROCESS_PLANNING.'+
           'SHAPE_REPRESENTATION_WITH_PARAMETERS'
           IN TYPEOF(pdr.used_representation)) AND
           ({8 <= SIZEOF(pdr.used_representation.items) <= 10} ) ))
           = 1  )) = 1));

      WR3: ((NOT (SELF\characterized_object.description = 'marking')) OR
           (SIZEOF( QUERY( pd <* USEDIN( SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
           SIZEOF( QUERY( pdr <* USEDIN( pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
           ('FEATURE_BASED_PROCESS_PLANNING.'+
           'SHAPE_REPRESENTATION_WITH_PARAMETERS'
           IN TYPEOF(pdr.used_representation)) AND
           (SIZEOF(pdr.used_representation.items) = 2)  )) = 1  )) = 1));

     WR4: ((NOT (SELF\characterized_object.description = 'knurl')) OR
          (SIZEOF( QUERY( pd <* USEDIN( SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
          SIZEOF( QUERY( pdr <* USEDIN( pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
          ('FEATURE_BASED_PROCESS_PLANNING.'+
          'SHAPE_REPRESENTATION_WITH_PARAMETERS'
          IN TYPEOF(pdr.used_representation)) AND
          (SIZEOF(pdr.used_representation.items) = 1)   )) = 1  )) = 1));

    wr5:  (NOT (SELF\characterized_object.description IN [ 'knurl',
          'thread'])) OR
          (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
          ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
          IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN
          (pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
```

 ©ISO 2000 - All rights reserved

```
        ((sa_occ.description = 'partial area occurrence') AND
        (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
        + 'RELATED_SHAPE_ASPECT') |
        ((sar.description = 'applied area usage') AND
        (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
        IN TYPEOF(sar))) ) |
        ('FEATURE_BASED_PROCESS_PLANNING.APPLIED_AREA'
        IN TYPEOF(sdr.relating_shape_aspect)) )) = 1)) )) <= 1)) )) = 0);

wr6: (NOT (SIZEOF(QUERY( doc <*(QUERY( adr <*USEDIN(SELF,
   'FEATURE_BASED_PROCESS_PLANNING.APPLIED_DOCUMENT_REFERENCE.ITEMS') |
       'FEATURE_BASED_PROCESS_PLANNING.DOCUMENT'
        IN TYPEOF (adr.assigned_document) )) |
      'FEATURE_BASED_PROCESS_PLANNING.DOCUMENT_FILE' IN TYPEOF(doc) ))
        =1) OR
        NOT (SIZEOF(QUERY( doc <*(QUERY( aduc <*USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.'+
       'APPLIED_DOCUMENT_USAGE_CONSTRAINT_ASSIGNMENT.ITEMS') |
       'FEATURE_BASED_PROCESS_PLANNING.DOCUMENT'
        IN TYPEOF (aduc\
        document_usage_constraint_assignment.
        assigned_document_usage.source) )) |
      'FEATURE_BASED_PROCESS_PLANNING.DOCUMENT_FILE' IN TYPEOF(doc) ))
        >=1));

  wr7: ((NOT (SELF\characterized_object.description = 'marking')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN
       (pd,'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.'+
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
       IN TYPEOF(it)) AND (it.name = 'marking text')) )) = 1)) ))
       = 0)) )) = 0));

wr8: ((NOT (SELF\characterized_object.description = 'thread')) OR
     (SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN
     (pd,'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     (('FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS')
     IN TYPEOF(pdr.used_representation)) ) |
     (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
     (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
     IN TYPEOF(it)) AND (it.name = 'removal direction') AND
     ((it.description = 'internal') OR
     (it.description = 'external'))) )) = 1)) )) = 0)) )) = 0));

wr9: ((NOT (SELF\characterized_object.description = 'thread')) OR
     (SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN
     (pd,'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
```

©ISO 2000 - All rights reserved

```
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
       IN TYPEOF(it)) AND (it.name = 'qualifier')) ))<= 1)) ))= 0)) ))
       = 0));

wr10: ((NOT (SELF\characterized_object.description = 'thread')) OR
       (SIZEOF(QUERY ( pd <*  USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN
       (pd,'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
       IN TYPEOF(it)) AND (it.name = 'hand')) )) = 1)) )) = 0)) ))
       = 0));

wr11: ((NOT (SELF\characterized_object.description = 'thread')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN
       (pd,'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
       IN TYPEOF(it)) AND (it.name = 'fit class')) )) = 1)) )) =
       0)) )) = 0));

wr12: ((NOT (SELF\characterized_object.description = 'thread')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN
       (pd,'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
       IN TYPEOF(it)) AND (it.name = 'form')) )) = 1)) )) = 0)) ))
       = 0));

wr13: ((NOT (SELF\characterized_object.description = 'thread')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN
       (pd,'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       ((SIZEOF([
```

 ©ISO 2000 - All rights reserved

```
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
        TYPEOF(it)) = 2) AND (it.name = 'major diameter')) )) = 1)) ))
        = 0)) )) = 0));

 wr14: ((NOT (SELF\characterized_object.description = 'thread')) OR
        (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN
        (pd,'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
        ((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.RATIO_MEASURE_WITH_UNIT'] *
        TYPEOF(it)) = 2) AND (it.name = 'number of threads')) )) = 1)) ))
        = 0)) )) = 0));

    wr15:  (NOT (SELF\characterized_object.description IN [ 'knurl',
           'thread'])) OR
           (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
           ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
           IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN
           (pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
           ((SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
           'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
           + 'RELATED_SHAPE_ASPECT') |
           ((sar.description = 'applied shape') AND
           (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
           IN TYPEOF(sar))) ) |
           ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT'
           IN TYPEOF(sdr.relating_shape_aspect)) )) = 1)) )) <= 1)) )) = 0);
END_ENTITY; -- externally_defined_feature_definition

(*
```

Formal propositions:

WR1: If the **description** of the **externally_defined_feature_definition** is 'thread', then the **item_id** is 'external thread', and the **source_id** of the **external_source** referenced by the **source** of the **externally_-defined_feature_definition** is 'external feature specification'. If the **description** of the **externally_-defined_feature_definition** is 'marking', then the **item_id** is 'external marking', and the **source_id** of the **external_source** referenced by the **source** of the **externally_defined_feature_definition** is 'external feature specification'. If the **description** of the **externally_defined_feature_definition** is 'knurl', then the **item_id** is 'external knurl', and the **source_id** of the **external_source** referenced by the **source** of the **externally_defined_feature_definition** is 'external feature specification'.

WR2: If the **description** of the **externally_defined_feature_definition** is 'thread', the implicit representation of the **externally_defined_feature_definition** shall contain between one and ten **representation_item**s in its set of **items**.

©ISO 2000 - All rights reserved

WR3: If the **description** of the **externally_defined_feature_definition** is 'marking', the implicit representation of the **externally_defined_feature_definition** shall contain exactly three **representation_item**s in its set of **items**.

WR4: If the **description** of the **externally_defined_feature_definition** is 'knurl', the implicit representation of the **externally_defined_feature_definition** shall contain exactly one **representation_-item**s in its set of **items**.

WR5: If the **description** of the **externally_defined_feature_definition** is 'marking', or 'thread' there shall be the basis shape for at most one **shape_aspect** with a **description** of 'partial area occurrence' that is the **related_shape_aspect** in at most one **shape_defining_relationship** with a description of 'applied area usage' in which the **relating_shape_aspect** is an **applied_area**.

WR6: The **externally_defined_feature_definition** shall be in the set of **items** of more than one **applied_document_reference** or exactly one **applied_document_usage_constraint_assignment** that defines the **document_file** containing the feature specification.

WR7: If the **description** of the **externally_defined_feature_definition** is 'marking', exactly one **representation_item** used for the implicit representation of the **externally_defined_feature_definition** shall be of type **descriptive_representation_item** with a **name** of 'marking text'.

WR8: If the **description** of the **externally_defined_feature_definition** is 'thread', then exactly one **representation_item** used for the implicit representation of the **externally_defined_feature_definition** shall be of type **descriptive_representation_item** with a **name** of 'removal direction' and a **description** of either 'internal' or 'external'.

WR9: If the **description** of the **externally_defined_feature_definition** is 'thread', at most one **representation_item** used for the implicit representation of the **externally_defined_feature_definition** shall be of type **descriptive_representation_item** with a **name** of 'qualifier'.

WR10: If the **description** of the **externally_defined_feature_definition** is 'thread', exactly one **representation_item** used for the implicit representation of the **externally_defined_feature_definition** shall be of type **descriptive_representation_item** with a **name** of 'hand'.

WR11: If the **description** of the **externally_defined_feature_definition** is 'thread', at most one **representation_item** used for the implicit representation of the **externally_defined_feature_definition** shall be of type **descriptive_representation_item** with a **name** of 'fit class'.

WR12: If the **description** of the **externally_defined_feature_definition** is 'thread', exactly one **representation_item** used for the implicit representation of the **externally_defined_feature_definition** shall be of type **descriptive_representation_item** with a **name** of 'form'.

WR13: If the **description** of the **externally_defined_feature_definition** is 'thread', at most one **representation_item** used for the implicit representation of the **externally_defined_feature_definition** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'major diameter'.

 ©ISO 2000 - All rights reserved

WR14: If the **description** of the **externally_defined_feature_definition** is 'thread', exactly one **representation_item** used for the implicit representation of the **externally_defined_feature_definition** shall be of type **measure_representation_item** and **ratio_measure_with_unit** with a **name** of 'number of threads'.

WR15: If the **description** of the **externally_defined_feature_definition** is 'marking', or 'thread' there shall be the basis shape for exactly one **shape_aspect** that is the **related_shape_aspect** in at most one **shape_defining_relationship** with a description of 'applied shape' in which the **relating_shape_aspect** is a **shape_aspect**.

Informal proposition:

IP1: If the **name** of the **descriptive_representation_item** is 'fit class' and the first character of the **description** is 'M', then the thread is metric.

5.2.3.1.21 face_shape_representation

A **face_shape_representation** is a type of **representation** that represents geometry defined by a portion of an associated surface.

EXPRESS specification:

```
*)
  ENTITY face_shape_representation
    SUBTYPE OF (shape_representation);
    WHERE
      wr1: (SIZEOF(SELF.items) >= 1);

      wr2: (SIZEOF(QUERY ( it <* SELF.items | (NOT (
        ('FEATURE_BASED_PROCESS_PLANNING.FACE_SURFACE' IN TYPEOF(it)) OR
        ('FEATURE_BASED_PROCESS_PLANNING.ORIENTED_FACE' IN TYPEOF(it)) ))
         )) = 0);
  END_ENTITY; -- face_shape_representation
(*
```

Formal propositions:

WR1: The **face_shape_representation** shall have one or more **representation_item** in its set of **items**.

WR2: The geometric element that is used to represent the **face_shape_representation** shall be a **face_-surface**, **oriented_face**.

5.2.3.1.22 face_shape_representation_relationship

A **face_shape_representation_relationship** is a type of **representation_relationship** that is the representation of several **face_shape_representation**s and their relationship to one another.

EXPRESS specification:

```
*)
ENTITY face_shape_representation_relationship
  SUBTYPE OF (representation_relationship);
      WHERE
```

©ISO 2000 - All rights reserved

```
      wr1: ('FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
            IN TYPEOF(SELF.rep_1));
      wr2: ('FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
            IN TYPEOF(SELF.rep_2));
END_ENTITY; -- face_shape_representation_relationship
(*
```

Formal propositions:

WR1: The **face_shape_representation_relationship** shall have only **face_shape_representation** in its **rep_1**.

WR2: The **face_shape_representation_relationship** shall have only **face_shape_representation** in its **rep_2**.

5.2.3.1.23 feature_based_pp_action_assignment

A **feature_based_pp_action_assignment** specifies those **feature_based_pp_action_item**s to which an **action** is assigned.

EXPRESS specification:

```
*)
ENTITY feature_based_pp_action_assignment
  SUBTYPE OF (action_assignment);
  items : SET [1:?] OF feature_based_pp_action_item;
END_ENTITY;
(*
```

Attribute definition:

items: the set of **feature_based_pp_action_item**s for which a particular **action** is applicable.

5.2.3.1.24 feature_based_pp_action_request_assignment

A **feature_based_pp_action_request_assignment** specifies those **feature_based_pp_action_request_-items** for which a design exception has been identified.

EXPRESS specification:

```
*)
ENTITY feature_based_pp_action_request_assignment
  SUBTYPE OF (action_request_assignment);
  items : SET [1:?] OF feature_based_pp_action_request_item;
END_ENTITY;
(*
```

Attribute definition:

items: the set of **feature_based_pp_action_request_item**s for which a particular **action_request** is applicable.

 ©ISO 2000 - All rights reserved

5.2.3.1.25 feature_based_pp_approval_assignment

A **feature_based_pp_approval_assignment** specifies those **feature_based_pp_approved_item**s to which an **approval** is assigned.

EXPRESS specification:

```
*)
ENTITY feature_based_pp_approval_assignment
  SUBTYPE OF (approval_assignment);
  items : SET [1:?] OF feature_based_pp_approved_item;
END_ENTITY;
(*
```

Attribute definition:

items: the set of **feature_based_pp_approved_item**s to which an **approval** is assigned.

5.2.3.1.26 feature_based_pp_date_assignment

A **feature_based_pp_date_assignment** specifies those **feature_based_pp_dated_item**s to which a date is assigned.

EXPRESS specification:

```
*)
ENTITY feature_based_pp_date_assignment
  SUBTYPE OF (date_assignment);
  items : SET [1:?] OF feature_based_pp_dated_item;
END_ENTITY;
(*
```

Attribute definition:

items: the set of **feature_based_pp_dated_item**s to which a date is assigned.

5.2.3.1.27 feature_based_pp_organization_assignment

A **feature_based_pp_organization_assignment** specifies those **feature_based_pp_organization_-item**s to which an **organization** is assigned.

EXPRESS specification:

```
*)
ENTITY feature_based_pp_organization_assignment
  SUBTYPE OF (organization_assignment);
  items : SET [1:?] OF feature_based_pp_organization_item;
END_ENTITY;
(*
```

©ISO 2000 - All rights reserved

Attribute definition:

items: the set of **feature_based_pp_organization_item**s to which an **organization** is assigned.

5.2.3.1.28 feature_based_pp_person_and_organization_assignment

A **feature_based_pp_person_and_organization_assignment** specifies those **feature_based_pp_-person_and_organization_item**s to which a **person_and_organization** is assigned.

EXPRESS specification:

```
*)
ENTITY feature_based_pp_person_and_organization_assignment
  SUBTYPE OF (person_and_organization_assignment);
  items : SET [1:?] OF feature_based_pp_person_and_organization_item;
END_ENTITY;
(*
```

Attribute definition:

items: the set of **feature_based_pp_person_and_organization_item**s to which a **person_and_-organization** is assigned.

5.2.3.1.29 feature_based_pp_security_classification_assignment

A **feature_based_pp_security_classification_assignment** specifies those **feature_based_pp_-classified_item**s to which a **security_classification** is assigned.

EXPRESS specification:

```
*)
ENTITY feature_based_pp_security_classification_assignment
  SUBTYPE OF (security_classification_assignment);
  items : SET [1:?] OF feature_based_pp_classified_item;
END_ENTITY;
(*
```

Attribute definition:

items: the set of **feature_based_pp_classified_item**s to which a **security_classification** is assigned.

5.2.3.1.30 feature_component_definition

A **feature_component_definition** is a type of **characterized_object** that is used as a constituent of a preconceived form pattern.

EXPRESS specification:

```
*)
  ENTITY feature_component_definition
    SUBTYPE OF (characterized_object);
```

 ©ISO 2000 - All rights reserved

```
    WHERE
      wr1: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
         (NOT (SIZEOF(USEDIN(pd,
         'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')) = 1)) ))
         = 0);
  END_ENTITY; -- feature_component_definition
(*
```

Formal propositions:

WR1: The **feature_component_definition** shall be the item for which exactly one **shape_aspect** is defined.

5.2.3.1.31 feature_component_relationship

A **feature_component_relationship** is a kind of **shape_aspect_relationship** in which the **related_-shape_aspect** is an implicitly represented component, such as a **taper**, of the **relating_shape_aspect**.

EXAMPLE - A Chamfer may have the implicit chamfer offset amount defined with a **feature_component_-relationship** and the explicit geometry for the chamfer defined with a **shape_defining_relationship**.

EXPRESS specification:

```
*)
  ENTITY feature_component_relationship
    SUPERTYPE OF (ONEOF
(pattern_omit_membership,pattern_offset_membership))
    SUBTYPE OF (shape_aspect_relationship);
    WHERE
      wr1: ((SIZEOF([
           'FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT',
           'FEATURE_BASED_PROCESS_PLANNING.REPLICATE_FEATURE',
           'FEATURE_BASED_PROCESS_PLANNING.TRANSITION_FEATURE',
           'FEATURE_BASED_PROCESS_PLANNING.MODIFIED_PATTERN'] * TYPEOF
           (SELF.relating_shape_aspect)) = 1) OR
           ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_DEFINITION'
           IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition)) OR
           ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
           IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition)));
  END_ENTITY; -- feature_component_relationship
(*
```

Formal proposition:

WR1: The **shape_aspect** referenced as the **relating_shape_aspect** shall be of type **replicate_feature**, **transition_feature, composite_shape_aspect**, or **modified_pattern** or shall be an aspect of the shape of a **feature_definition** or a **feature_component_definition**.

5.2.3.1.32 feature_definition

A **feature_definition** is a type of **characterized_object** that is a preconceived form pattern.

©ISO 2000 - All rights reserved

EXPRESS specification:

```
*)
  ENTITY feature_definition
    SUBTYPE OF (characterized_object);
    WHERE
  wr1: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0);


 WR2: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
       ('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT'
       IN TYPEOF (it)) AND (it.name = 'orientation'))) = 1))) = 0))) = 0;

  WR3: SIZEOF (['FEATURE_BASED_PROCESS_PLANNING.BOSS',
       'FEATURE_BASED_PROCESS_PLANNING.TURNED_KNURL',
       'FEATURE_BASED_PROCESS_PLANNING.THREAD',
       'FEATURE_BASED_PROCESS_PLANNING.MARKING',
       'FEATURE_BASED_PROCESS_PLANNING.RIB_TOP',
       'FEATURE_BASED_PROCESS_PLANNING.ROUND_HOLE',
       'FEATURE_BASED_PROCESS_PLANNING.OUTSIDE_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.POCKET',
       'FEATURE_BASED_PROCESS_PLANNING.REMOVAL_VOLUME',
       'FEATURE_BASED_PROCESS_PLANNING.REVOLVED_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.OUTER_ROUND',
       'FEATURE_BASED_PROCESS_PLANNING.FLAT_FACE',
       'FEATURE_BASED_PROCESS_PLANNING.PROTRUSION',
       'FEATURE_BASED_PROCESS_PLANNING.ROUNDED_END',
       'FEATURE_BASED_PROCESS_PLANNING.SLOT',
       'FEATURE_BASED_PROCESS_PLANNING.SPHERICAL_CAP',
       'FEATURE_BASED_PROCESS_PLANNING.STEP',
       'FEATURE_BASED_PROCESS_PLANNING.COMPOUND_FEATURE',
       'FEATURE_BASED_PROCESS_PLANNING.REPLICATE_FEATURE',
    'FEATURE_BASED_PROCESS_PLANNING.EXTERNALLY_DEFINED_FEATURE_DEFINITION']
       * TYPEOF (SELF)) = 1;

  WR4: (NOT (SIZEOF (['FEATURE_BASED_PROCESS_PLANNING.ROUND_HOLE',
       'FEATURE_BASED_PROCESS_PLANNING.BOSS',
       'FEATURE_BASED_PROCESS_PLANNING.OUTSIDE_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.REMOVAL_VOLUME',
       'FEATURE_BASED_PROCESS_PLANNING.FLAT_FACE',
       'FEATURE_BASED_PROCESS_PLANNING.POCKET',
       'FEATURE_BASED_PROCESS_PLANNING.PROTRUSION',
       'FEATURE_BASED_PROCESS_PLANNING.RIB_TOP',
       'FEATURE_BASED_PROCESS_PLANNING.ROUNDED_END',
       'FEATURE_BASED_PROCESS_PLANNING.SLOT',
```

 ©ISO 2000 - All rights reserved

```
        'FEATURE_BASED_PROCESS_PLANNING.STEP']
        * TYPEOF (SELF)) = 1)) OR
        (SIZEOF (QUERY (pd <* USEDIN (SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
        'DEFINITION') |
        NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        ('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
        IN TYPEOF (pdr.used_representation))AND
        (pdr.used_representation.name = 'maximum feature limit'))) = 1)))
        <= 1);
  END_ENTITY; -- feature_definition
(*
```

Formal propositions:

WR1: The **feature_definition** shall have exacttly one implicit **representation**.

WR2: Exactly one **representation_item** used for the implicit representation of a **feature_definition** shall be of type **placement** with a **name** of 'orientation'. The **placement** shall define a reference location and orientation for the origin of the **feature_definition**.

WR3: The **feature_definition** shall be either a **boss**, **turned_knurl**, **thread**, **marking**, **rib_top**, **round_hole**, **outside_profile**, **pocket**, **removal_volume**, **revolved_profile**, **outer_round**, **flat_face**, **protrusion**, **rounded_end**, **slot**, **spherical_cap**, **step**, **compound_feature**, **replicate_feature**, or **externally_defined_feature_definition**.

WR4: If the **feature_definition** is a **boss**, **outside_profile**, **flat_face**, **removal_volume**, **protrusion**, **pocket**, **rib**, **rounded_end**, **slot**, **step** the **instance_feature** shall have at most one **planar_shape_representation** with a **name** of 'maximum feature limit'.

5.2.3.1.33 feature_pattern

A **feature_pattern** is a type of **replicate_feature** that relates a base feature and one or more **shape_-aspect**s which are the placement of the base feature at a specified location on the base part.

EXPRESS specification:

```
*)
  ENTITY feature_pattern
    SUBTYPE OF (replicate_feature);
    WHERE
      WR1: SIZEOF( QUERY( pd <* USEDIN( SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
            SIZEOF( QUERY( pdr <* USEDIN( pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
            ('FEATURE_BASED_PROCESS_PLANNING.'+
            'SHAPE_REPRESENTATION_WITH_PARAMETERS'
            IN TYPEOF(pdr.used_representation)) AND
            (SIZEOF( QUERY( srwp_i <* pdr.used_representation.items |
            NOT ('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT'
            IN TYPEOF(srwp_i)) )) > 0 ) ))   > 0  )) = 0;
```

©ISO 2000 - All rights reserved

```
 WR2: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
       ('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT'
       IN TYPEOF (it)) AND (it.name = 'base feature placement'))) > 1)))
       = 0))) = 0;
  END_ENTITY; -- feature_pattern
(*
```

Formal propositions:

WR1: The implicit representation of a **feature_pattern** shall only contain **representation_item**s in its set of **items** that are of type **placement**.

WR2: The **feature_pattern** shall have an implicit representation defined by a **shape_representation_-with_parameters** which contains one or more **representation_item**s in its set of **items** of type **placement** with the name of 'base feature placement'.

5.2.3.1.34 fillet

A **fillet** is a type of **transition_feature** that is the representation of a circular concave transition between two **shape_aspects**. A **fillet** defines the outside profile of a part. See ARM definition for Fillet in paragraph 4.2.73 for more information.

EXPRESS specification:

```
*)
  ENTITY fillet
    SUBTYPE OF (transition_feature);
  WHERE
    WR1: (NOT (SELF\shape_aspect.description = 'constant radius')) OR
         (SIZEOF (QUERY (pd <* USEDIN (SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
         NOT (SIZEOF (QUERY (pdr <*  USEDIN (pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
         'FEATURE_BASED_PROCESS_PLANNING.'+
         'SHAPE_REPRESENTATION_WITH_PARAMETERS'
         IN TYPEOF (pdr.used_representation))) = 1))) = 0);

    WR2: (NOT (SELF\shape_aspect.description = 'constant radius')) OR
         (SIZEOF (QUERY (pd <* USEDIN (SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
         NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
         'FEATURE_BASED_PROCESS_PLANNING.'+
         'SHAPE_REPRESENTATION_WITH_PARAMETERS'
         IN TYPEOF (pdr.used_representation)) |
         (NOT (SIZEOF (impl_rep.used_representation.items) >= 1)
```

 ©ISO 2000 - All rights reserved

```
     AND(SIZEOF (impl_rep.used_representation.items) <= 3))))
     = 0))) = 0);

WR3: (NOT (SELF.description = 'constant radius')) OR
     (SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     'FEATURE_BASED_PROCESS_PLANNING.'+
     'SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation)) |
     NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
     (SIZEOF
     (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
     'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
     TYPEOF (it)) = 2) AND (it.name = 'radius'))) = 1))) = 0))) = 0);

WR4: (NOT (SELF.description = 'constant radius')) OR
     (SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation)) |
     NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
     (SIZEOF
     (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
     'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
     TYPEOF (it)) = 2) AND (it.name = 'first offset'))) <= 1)))
     = 0))) = 0);

WR5: (NOT (SELF.description = 'constant radius')) OR
     (SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation)) |
     NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
     (SIZEOF
     (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
     'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
     TYPEOF (it)) = 2) AND (it.name = 'second offset'))) <= 1)))
     = 0))) = 0);

WR6: SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
     'DEFINITION') |
     NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     ('FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
     IN TYPEOF (pdr.used_representation)) AND
     (pdr.used_representation.name = 'fillet face'))) = 1))) = 0;
```

©ISO 2000 - All rights reserved

```
    WR7: SIZEOF (QUERY (pd <* USEDIN (SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
         'DEFINITION') |
         NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
         ('FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
         IN TYPEOF (pdr.used_representation)) AND
         (pdr.used_representation.name = 'first face shape'))) = 1))) = 0;

    WR8: SIZEOF (QUERY (pd <* USEDIN (SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
         'DEFINITION') |
         NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
         ('FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
         IN TYPEOF (pdr.used_representation)) AND
         (pdr.used_representation.name = 'second face shape'))) = 1))) = 0;

END_ENTITY;
(*
```

Formal propositions:

WR1: A **fillet** has an implicit representation if and only if it has a **description** of 'constant radius'.

WR2: If the description of the **fillet** is 'constant radius', the implicit representation shall contain at least one and at most three **representation_item**s in its set of **items**.

WR3: If the description of the **fillet** is 'constant radius', exactly one **representation_item** used for the implicit representation of the **fillet** shall be of type **measure_representation_item** and **length_-measure_with_unit** with a **name** of 'radius'.

WR4: If the description of the **fillet** is 'constant radius', at most one **representation_item** used for the implicit representation of the **fillet** shall be of type **measure_representation_item** and **length_-measure_with_unit** with a **name** of 'first offset'.

WR5: If the description of the **fillet** is 'constant radius', at most one **representation_item** used for the implicit representation of the **fillet** shall be of type **measure_representation_item** and **length_-measure_with_unit** with a **name** of 'second offset'.

WR6: The **fillet** shall have exactly one **face_shape_representation** with a **name** of 'fillet face'.

WR7: The **fillet** shall have exactly one **face_shape_representation** with a **name** of 'first face shape'.

WR8: The **fillet** shall have exactly one **face_shape_representation** with a **name** of 'second face shape'.

 ©ISO 2000 - All rights reserved

5.2.3.1.35 flat_face

A **flat_face** is a type of **feature_definition** that is the representation of a volume that is removed from the base shape. This removal shall be planar and representation shall be a **linear_profile** moving along a linear **path_feature_component**. See ARM definition for Planar_face in paragraph 4.2.154 for more information.

EXPRESS specification:

```
*)
  ENTITY flat_face
    SUBTYPE OF (feature_definition);
   WHERE

   wr1: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
        IN TYPEOF(pdr.used_representation)) AND
        (pdr.used_representation.name = 'removal direction')) )) = 1)) ))
        = 0);

   wr2: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
        IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
        ((sa_occ.description = 'course of travel occurrence') AND
        (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
        'RELATED_SHAPE_ASPECT') |
        ((sar.description = 'path feature component usage') AND
        (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
        IN TYPEOF(sar))) ) |
        (('FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT'
        IN TYPEOF(sdr.relating_shape_aspect)) AND
        (sdr.relating_shape_aspect.description = 'linear') AND
        (sdr.name = 'course of travel')) )) = 1)) )) = 1)) )) = 0);

   wr3: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
        IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
        ((sa_occ.description = 'removal boundary occurrence') AND
        (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
        'RELATED_SHAPE_ASPECT') |
        ((sar.description = 'profile usage') AND
        (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
        IN TYPEOF(sar))) ) |
        (('FEATURE_BASED_PROCESS_PLANNING.LINEAR_PROFILE'
        IN TYPEOF(sdr.relating_shape_aspect)) AND
        (sdr.name = 'removal boundary')) )) = 1)) )) = 1)) )) = 0);
```

©ISO 2000 - All rights reserved

```
   WR4: SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
        IN TYPEOF (pd)) |
        NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
        (sa_occ.description = 'enclosed boundary occurrence') AND
        (SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
        'RELATED_SHAPE_ASPECT') |
        (sar.description = 'profile usage') AND
        ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
        IN TYPEOF (sar))) |
        (SIZEOF
        (['FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE',
        'FEATURE_BASED_PROCESS_PLANNING.NGON_CLOSED_PROFILE',
        'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE',
        'FEATURE_BASED_PROCESS_PLANNING.CLOSED_PATH_PROFILE'] *
        TYPEOF (sdr.relating_shape_aspect)) = 1) AND
        (sdr.relating_shape_aspect.description = 'boundary')))
        = 1))) <= 1))) = 0;
  END_ENTITY; -- flat_face

(*
```

Formal proposition:

WR1: The **flat_face** shall have exactly one **direction_shape_representation** with a **name** of 'removal direction'.

WR2: The **flat_face** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'course of travel occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'path feature component usage' and a **name** of 'course of travel' and a **relating_shape_-aspect** that references a **path_feature_component** with a **description** of 'linear'.

WR3: The **flat_face** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'removal boundary occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_-relationship** with a **description** of 'profile usage' and a **name** of 'removal boundary' and a **relating_-shape_aspect** that references a **linear_profile**.

WR4: The **flat_face** shall be the basis shape for at most one **shape_aspect** with a **description** of 'enclosed boundary occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_-relationship** with a **description** of 'profile usage' and a **name** of 'boundary' and a **relating_shape_aspect** that references a **circular_closed_profile**, **ngon_closed_profile**, **rectangular_closed_profile**, or **closed_path_profile**.

Informal propositions:

IP1: The location of the **flat_face** shall be defined at one of the corners of the removal volume.

IP2: The direction of volume removal of the **flat_face** shall be in the Z direction.

 ©ISO 2000 - All rights reserved

IP3: The origin, X direction, and Y direction of the **path_feature_component** shall be equal to that of the **flat_face**.

IP4: The origin, X direction, Y direction of the **flat_face** shall be equal to that of the **linear_profile**.

5.2.3.1.36 hole_bottom

A **hole_bottom** is a type of **shape_aspect** that is the representation of the end condition for a hole **feature_definition**. See ARM definition for Blind_bottom_condition in paragraph 4.2.8 and Through_-bottom_condition in paragraph 4.2.225 for more information.

EXPRESS specification:

```
*)
  ENTITY hole_bottom
    SUBTYPE OF (shape_aspect);
   WHERE
   wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
        IN TYPEOF(SELF.of_shape.definition));

   wr2: (SELF.description IN ['through', 'flat', 'flat with radius',
        'flat with taper','spherical', 'conical']);

   wr3: ((NOT (SELF.description = 'through')) OR
        (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION')) = 0)) ))
         = 0));
  wr4 : ((NOT (SELF.description IN ['flat with radius',
        'flat with taper', 'spherical','conical'])) OR
        (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0));

  wr5: ((NOT (SELF.description = 'flat')) OR
       (SIZEOF(QUERY ( pd <*  USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(impl_rep.used_representation.items) = 0)) ))
        = 0)) )) = 0));

  wr6: ((NOT (SELF.description IN ['flat with radius','spherical']))
       OR (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
```

©ISO 2000 - All rights reserved

```
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(impl_rep.used_representation.items) = 1)) ))
       = 0)) )) = 0));

  wr7: ((NOT (SELF.description = 'flat with taper')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(impl_rep.used_representation.items) = 2)) ))
       = 0)) )) = 0));

  wr8: ((NOT (SELF.description = 'conical')) OR (SIZEOF(
       QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF (impl_rep.used_representation.items) >= 1)
       AND  (SIZEOF (impl_rep.used_representation.items) <= 2)) ))
       = 0)) )) = 0));

  wr9: ((SELF.description = 'through') OR
       (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
       'RELATING_SHAPE_ASPECT') |
       ((sar.description = 'hole bottom usage') AND
       ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
       IN TYPEOF(sar))) ) |
       ((fcr.name IN ['hole depth start','hole depth end'])) )) >= 1));

 wr10: ((NOT (SELF.description = 'flat with radius')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       ((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF(it)) = 2) AND (it.name = 'corner radius')))) = 1)) ))
       = 0)) )) = 0));
 wr11: ((NOT (SELF.description = 'spherical')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
```

 ©ISO 2000 - All rights reserved

```
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF(it)) = 2) AND (it.name = 'radius')) )) = 1)) )) =
      0)) )) = 0));
wr12: ((NOT (SELF.description = 'conical')) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF(it)) = 2) AND (it.name = 'tip radius')) )) <= 1)) ))
      = 0)) )) = 0));
wr13: ((NOT (SELF.description = 'conical')) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT'] *
      TYPEOF(it)) = 2) AND (it.name = 'tip angle')) )) = 1)) ))
      = 0)) )) = 0));
wr14: (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATING_SHAPE_ASPECT') |
      ((sar.description = 'hole bottom usage') AND
      ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
      IN TYPEOF(sar))) ) |
    ((fcr.related_shape_aspect.description ='bottom condition occurrence')
      AND ('FEATURE_BASED_PROCESS_PLANNING.ROUND_HOLE'
      IN TYPEOF(fcr.related_shape_aspect.of_shape.definition))) )) >= 1);
WR15: ((NOT (SELF.description = 'flat with taper')) OR (SIZEOF(
      QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF([
```

```
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF(it)) = 2) AND (it.name = 'final diameter')))) = 1)) ))
       = 0)) )) = 0));
 wr16: ((NOT (SELF.description = 'flat with taper')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       ((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT'] *
       TYPEOF(it)) = 2) AND (it.name =  'taper diameter')) )) = 1)) ))
       = 0)) )) = 0));
   END_ENTITY; -- hole_bottom
(*
```

Formal propositions:

WR1: The **hole_bottom** shall be an aspect of the shape of a **feature_component_definition**.

WR2: The **description** of the **hole_bottom** shall be either 'through', 'flat', 'flat with radius', 'flat with taper', 'spherical', or 'conical'.

WR3: If the **description** of the **hole_bottom** is 'through', the **hole_bottom** shall not have any representation.

WR4: If the **description** of the **hole_bottom** is 'flat with radius', 'spherical', or 'conical', the **hole_-bottom** shall have exactly one **shape_representation_with_parameters** to specify its implicit representation.

WR5: If the **description** of the **hole_bottom** is 'flat', the **hole_bottom** shall be represented implicitly by exactly zero **representation_item**.

WR6: If the **description** of the **hole_bottom** is 'flat with radius' or 'spherical', the **hole_bottom** shall be represented implicitly by exactly one **representation_item**s.

WR7: If the **description** of the **hole_bottom** is 'flat with taper', the **hole_bottom** shall be represented implicitly by exactly two **representation_item**s.

WR8: If the **description** of the **hole_bottom** is 'conical', the **hole_bottom** shall be represented implicitly by at least one and at most two **representation_item**s.

WR9: If the **description** of the **hole_bottom** is not 'through ', the **feature_component_relationship** that refers to it, shall have a **name** of either 'hole depth start ' or 'hole depth end '.

 ©ISO 2000 - All rights reserved

WR10: If the **description** of the **hole_bottom** is 'flat with radius', the implicit representation of the **hole_bottom** shall have exactly one **representation_item** in its **items** set which is a **measure_-representation_item** and **length_measure_with_unit** with a **name** of 'corner radius'.

WR11: If the **description** of the **hole_bottom** is 'spherical', the implicit representation of the **hole_-bottom** shall have exactly one **representation_item** in its **items** set which is a **measure_-representation_item** and **length_measure_with_unit** with a **name** of 'radius'.

WR12: If the **description** of the **hole_bottom** is 'conical', the implicit representation of the **hole_-bottom** shall have at most one **representation_item** in its **items** set which is a **measure_-representation_item** and **length_measure_with_unit** with a **name** of 'tip radius'.

WR13: If the **description** of the **hole_bottom** is 'conical', the implicit representation of the **hole_-bottom** shall have exactly one **representation_item** in its **items** set which is a **measure_-representation_item** and **plane_angle_measure_with_unit** with a **name** of 'tip angle'.

WR14: The **hole_bottom** shall be the **relating_shape_aspect** in at least one **feature_component_-relationship** with a **description** of 'hole bottom usage' in which the **related_shape_aspect** is an aspect of the shape of a **round_hole** with a **description** of 'bottom condition occurrence'.

WR15: If the **description** of the **hole_bottom** is 'flat with taper', the implicit representation of the **hole_bottom** shall have exactly one **representation_item** in its **items** set which is a **measure_-representation_item** and **length_measure_with_unit** with a **name** of 'final diameter'.

WR16: If the **description** of the **hole_bottom** is 'flat with taper', the implicit representation of the **hole_bottom** shall have exactly one **representation_item** in its **items** set which is a **measure_-representation_item** and **plane_angle_measure_with_unit** with a **name** of 'taper diameter'.

Informal propositions:

IP1: The location of the **hole_bottom** shall be at the center of the circular face that is the mating face to a **hole** feature for **spherical**, **conical**, and **flat** hole bottom types. For **flat_with_radius** and **flat_-with_taper**, the location shall be at the absolute bottom.

IP2: Th **hole_bottom** shall be defined on the mating face in the X-Y plane with the Z direction coincident to that of the mating **hole** feature.

5.2.3.1.37 instanced_feature

An **instanced_feature** is a type of **shape_aspect** and of **feature_definition** that is the representation of a preconceived pattern for the machining application purpose.

EXPRESS specification:

```
*)
ENTITY instanced_feature
  SUBTYPE OF (feature_definition, shape_aspect);
WHERE
  WR1: 'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION' IN
       TYPEOF (SELF.of_shape.definition);
```

©ISO 2000 - All rights reserved

```
  WR2: SELF.product_definitional;

END_ENTITY;
(*
```

Formal propositions:

WR1: The **instanced_feature** shall be an aspect of the shape of a **product_definition**.

WR2: The **instanced_feature** shall lie on the boundary of the part.

Informal propositions:

IP1: If an **instanced_feature** has a **shape_representation** and if the **product_definition_shape** that is referenced as **of_shape** by the **instanced_feature** has a **shape_representation** then these **shape_representation**s shall have the same **geometric_representation_context**.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **instance_feature** entity:

— machining_feature_life_cycle (See 5.2.4.14).

5.2.3.1.38 linear_profile

A **linear_profile** is a type of **shape_aspect** that is the representation of a straight line with orientation and location. See ARM definition for Linear_profile in paragraph 4.2.105 for more information.

EXPRESS specification:

```
*)
  ENTITY linear_profile
    SUBTYPE OF (shape_aspect);
    WHERE
  wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
       IN TYPEOF(SELF.of_shape.definition));

  wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0);

  wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
```

 ©ISO 2000 - All rights reserved

```
      (NOT (SIZEOF(impl_rep.used_representation.items) = 2)) ))
      = 0)) )) = 0);

  wr4: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      (('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it)) AND
      (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);

  wr5: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF(it)) = 2) AND (it.name = 'profile length')) )) = 1)) ))
      = 0)) )) = 0);


  END_ENTITY; -- linear_profile
(*
```

Formal propositions:

WR1: The **linear_profile** shall be an aspect of the shape of a **feature_component_definition**.

WR2: The **linear_profile** shall have exactly one implicit representation defined by a relationship to a **shape_representation_with_parameters**.

WR3: The **shape_representation_with_parameters** that represents the **linear_profile** shall contain two **representation_item**s in its set of **items**.

WR4: Exactly one **representation_item** used for the implicit representation of a **linear_profile** shall be of type **placement** with a **name** of 'orientation'.

WR5: Exactly one **representation_item** used for the implicit representation of a **linear_profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'profile length'.

Informal propositions:

IP1: The location of the **linear_profile** shall be one end of the profile.

©ISO 2000 - All rights reserved

IP2: The X direction of the **linear_profile** shall be defined coincident to the X direction of the mating feature.

5.2.3.1.39 location_shape_representation

A **location_shape_representation** is a type of **representation** that represents a geometric position or a reference point.

EXPRESS specification:

```
*)
  ENTITY location_shape_representation
    SUBTYPE OF (shape_representation);
    WHERE
 wr1: (SIZEOF(SELF.items) = 1);

 wr2: (SIZEOF(QUERY ( it <* SELF.items |
      (NOT ('FEATURE_BASED_PROCESS_PLANNING.POINT'
      IN TYPEOF(it))) )) = 0);
  END_ENTITY; -- location_shape_representation
(*
```

Formal propositions:

WR1: The **location_shape_representation** shall have exactly one **representation_item** in its set of **items**.

WR2: The geometric element that is used to represent the **location_shape_representation** shall be a **point**.

5.2.3.1.40 marking

A **marking** is a type of **feature_definition** that is the representation of a text or symbol shapes that are applied to all or a portion of a **shape_aspect**. See ARM definition for Defined_marking in paragraph 4.2.54 for more information.

EXPRESS specification:

```
*)
  ENTITY marking
    SUBTYPE OF (feature_definition);
    WHERE
   WR1: SIZEOF( QUERY( pd <* USEDIN( SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        SIZEOF( QUERY( pdr <* USEDIN( pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        ('FEATURE_BASED_PROCESS_PLANNING.'+
        'SHAPE_REPRESENTATION_WITH_PARAMETERS' IN
        TYPEOF(pdr.used_representation)) AND
        ({2 <= SIZEOF(pdr.used_representation.items) <=6} ) ))
        = 1 )) = 1;
```

 ©ISO 2000 - All rights reserved

```
wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     (NOT (SIZEOF(QUERY ( impl_rep <*
     QUERY ( pdr <* USEDIN(pd,'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     (('FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS')
     IN TYPEOF(pdr.used_representation)) ) |
     (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
     (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
     IN TYPEOF(it)) AND (it.name = 'marking text')) )) = 1)) ))
     = 0)) )) = 0);

wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     (('FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS')
     IN TYPEOF(pdr.used_representation)) ) |
     (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
     (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
     IN TYPEOF(it)) AND (it.name = 'special instructions')) )) <= 1)) ))
     = 0)) )) = 0);

wr4: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     (('FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS')
     IN TYPEOF(pdr.used_representation)) ) |
     (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
     (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
     IN TYPEOF(it)) AND (it.name = 'font name')))) <= 1))))
     = 0)) )) = 0);

wr5: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     (('FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS')
     IN TYPEOF(pdr.used_representation)) ) |
     (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
     ((SIZEOF([
     'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
     'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
     TYPEOF(it)) = 2) AND (it.name = 'character height')) )) <= 1)) ))
     = 0)) )) = 0);

wr6: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     (('FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS')
```

©ISO 2000 - All rights reserved

```
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       ((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF(it)) = 2) AND (it.name = 'character spacing')) )) <= 1)) ))
       = 0)) )) = 0);

  WR7: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
       IN TYPEOF(pd)) ) |
       (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       ((SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
       + 'RELATED_SHAPE_ASPECT') |
       ((sar.description = 'applied shape') AND
       (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
       IN TYPEOF(sar))) ) |
       ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT'
       IN TYPEOF(sdr.relating_shape_aspect)) )) = 1)) )) = 1)) )) = 0);

  END_ENTITY; -- marking
(*
```

Formal propositions:

WR1: The implicit representation of the **marking** shall contain between two and six **representation_item**s.

WR2: Exactly one **representation_item** used for the implicit representation of the **marking** shall be of type **descriptive_representation_item** with a **name** of 'marking text'.

WR3: At most one **representation_item** used for the implicit representation of the **marking** shall be of type **descriptive_representation_item** with a **name** of 'special instructions'.

WR4: At most one **representation_item** used for the implicit representation of the **marking** shall be of type **descriptive_representation_item** with a **name** of 'font name'.

WR5: At most one **representation_item** used for the implicit representation of the **marking** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'character height'.

WR6: At most one **representation_item** used for the implicit representation of the **marking** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'character spacing'.

WR7: Exactly one **marking** shall be the basis shape for exactly one **shape_aspect** that is the **related_-shape_aspect** in at most one **shape_defining_relationship** with a description of 'applied shape' in which the **relating_shape_aspect** is a **shape_aspect**.

 ©ISO 2000 - All rights reserved

5.2.3.1.41 modified_pattern

A **modified_pattern** is a type of **replicate_feature** that represents the result of a collection of omissions and offsets of a base feature in either a **rectangular_pattern** or a **circular_pattern**.

EXPRESS specification:

```
*)
  ENTITY modified_pattern
    SUBTYPE OF (shape_aspect);
    WHERE
   WR1: SIZEOF( QUERY (fcr <* QUERY(sar <* USEDIN( SELF,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
        'RELATING_SHAPE_ASPECT') |
        'FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
        IN TYPEOF (sar)) |
        (SIZEOF(
        ['FEATURE_BASED_PROCESS_PLANNING.REPLICATE_FEATURE',
        'FEATURE_BASED_PROCESS_PLANNING.INSTANCED_FEATURE'] *
        TYPEOF (fcr.related_shape_aspect.of_shape.definition)) >= 1)  AND
        (fcr.description = 'base shape')) ) = 1;

  wr2: SIZEOF( QUERY (fcr <* QUERY(sar <* USEDIN( SELF,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
        'RELATING_SHAPE_ASPECT') |
        'FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
        IN TYPEOF (sar)) |
        (SIZEOF(
        ['FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN',
        'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN'] *
        TYPEOF(fcr.related_shape_aspect.of_shape.definition)) = 1) AND
        (fcr.description = 'base pattern')) ) = 1;

  wr3: SIZEOF(QUERY ( sar <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
        'RELATING_SHAPE_ASPECT') |
        (SIZEOF(QUERY ( msar <* USEDIN(sar.related_shape_aspect,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
        'RELATED_SHAPE_ASPECT') |
       (SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.PATTERN_OFFSET_MEMBERSHIP',
       'FEATURE_BASED_PROCESS_PLANNING.PATTERN_OMIT_MEMBERSHIP'] *
       TYPEOF(sar)) = 1) AND (sar.description='modified pattern')
       AND (sar :<>: msar) )) >= 1) )) = 0;

  END_ENTITY; -- modified_pattern
(*
```

Formal propositions:

WR1: The **modified_pattern** shall be the **relating_shape_aspect** in exactly one **shape_aspect_-relationship** that is of type **feature_component_relationship** with a name of 'base shape' and a **related_shape_aspect** that is either of type **replicate_feature** or **instanced_feature**.

©ISO 2000 - All rights reserved

WR2: The **modified_pattern** shall be the **relating_shape_aspect** in exactly one **shape_aspect_-relationship** and has the **description** of 'base pattern', with a **related_shape_aspect** that references either a **circular_pattern** or a **rectangular_pattern**.

WR3: The **modified_pattern** shall be the **related_shape_aspect** in at least one **shape_aspect_-relationship** and has the **description** of 'modified pattern', in which the **related_shape_aspect** is the **related_shape_aspect** in a different instance of **shape_aspect_relationship** that is either a **pattern_-offset_membership** or **pattern_omit_membership**.

5.2.3.1.42 ngon_closed_profile

A **ngon_closed_profile** is a type of **shape_aspect** that represents an enclosed 2D area with location and orientation. The enclosed area is defined by three or more straight sides. The location is defined to be at the center of the enclosed area, and the orientation of the **ngon_closed_profile** is the X-Y plane with the X direction parallel to one of the sides of the ngon. See ARM definition for Ngon_profile in paragraph 4.2.127 for more information.

EXPRESS specification:

```
*)
ENTITY ngon_closed_profile
  SUBTYPE OF (shape_aspect);
WHERE
  WR1: 'FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
       IN TYPEOF (SELF.of_shape.definition);

  WR2: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation))) = 1))) = 0;

  WR3: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT ((SIZEOF (impl_rep.used_representation.items) >= 3)
       AND (SIZEOF(impl_rep.used_representation.items) <= 4)))) = 0))) = 0;

  WR4:  SIZEOF( QUERY( pd <* USEDIN( SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       SIZEOF( QUERY( pdr <* USEDIN( pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.'+
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF(pdr.used_representation)) AND
       (SIZEOF( QUERY( srwp_i <* pdr.used_representation.items |
       (srwp_i.name = 'orientation') OR
```

 ©ISO 2000 - All rights reserved

```
     (srwp_i.name = 'number of sides') OR
     (srwp_i.name = 'circumscribed diameter') OR
     (srwp_i.name = 'corner radius') OR
     (srwp_i.name = 'diameter across flats')))
     = SIZEOF(pdr.used_representation.items))  )) = 1 )) = 1;

WR5: SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation)) |
     NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
     ('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF (it)) AND
     (it.name = 'orientation'))) = 1))) = 0))) = 0;

WR6: SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation)) |
     NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
     ('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
     IN TYPEOF (it)) AND
     ('FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE'
     IN TYPEOF (it\measure_with_unit.value_component)) AND
     (it.name = 'number of sides'))) = 1))) = 0))) = 0;

WR7: SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation)) |
      NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
     (SIZEOF
     (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
     'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
     TYPEOF (it)) = 2) AND
     (it.name IN ['circumscribed diameter','diameter across flats']) ))
     = 1) )) = 0))) = 0;

WR8: SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation)) |
     NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
     (SIZEOF
     (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
     'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
```

©ISO 2000 - All rights reserved

```
       TYPEOF (it)) = 2) AND
       (it.name = 'corner radius'))) <= 1))) = 0))) = 0;
END_ENTITY;
(*
```

Formal propositions:

WR1: The **ngon_closed_profile** shall be an aspect of the shape of a **feature_component_definition**.

WR2: The **ngon_closed_profile** shall have exactly one implicit representation defined by a **shape_-representation_with_parameters**.

WR3: The **shape_representation_with_parameters** that represents the **ngon_closed_profile** shall contain at least three and at most four **representation_item**s in its set of **items**.

WR4: The implicit representation of an **ngon_closed_profile** shall contain only **representation_item**s in its set of **items** with a **name** of 'orientation', 'number of sides', 'diameter across flats', 'circumscribed diameter', and 'corner radius'.

WR5: Exactly one **representation_item** used for the implicit representation of a **ngon_closed_profile** shall be of type **placement** with a **name** of 'orientation'.

WR6: Exactly one **representation_item** used for the implicit representation of a **ngon_closed_profile** shall be of type **measure_representation_item** with a **value_component** of type **count_measure** and a **name** of 'number of sides'.

WR7: Exactly one **representation_item** used for the implicit representation of a **ngon_closed_profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'circumscribed diameter' or 'diameter across flats'.

WR8: At most one **representation_item** used for the implicit representation of a **ngon_closed_profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'corner radius'.

Informal propositions:

IP1: The location of the **ngon_closed_profile** shall be defined at the center of the enclosed area.

IP2: The **ngon_closed_profile** shall be defined in the X-Y plane with one of the sides of the ngon parallel to the X direction intersecting the negative Y direction.

5.2.3.1.43 ngon_shape_representation

An **ngon_shape_representation** specifies representation of a shape that is a volume defined as a ngon area of a defined length. The enclosed area is defined by three or more straight sides. See ARM definition for Ngon_base_shape in paragraph 4.2.126 for more information.

 ©ISO 2000 - All rights reserved

EXPRESS specification:

```
*)
  ENTITY ngon_shape_representation
    SUBTYPE OF (shape_representation_with_parameters);
    WHERE
  wr1: (SIZEOF(SELF.items) = 5);

  wr2: (SIZEOF(QUERY ( it <* SELF.items |
       (('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT'
       IN TYPEOF(it)) AND (it.name = 'orientation')) )) = 1);

  wr3: (SIZEOF(QUERY ( it <* SELF.items |
       ((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF(it)) = 2) AND (it.name = 'length')) )) = 1);

  wr4: (SIZEOF(QUERY ( it <* SELF.items |
       ((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF(it)) = 2) AND (it.name = 'corner radius')) )) = 1);

  wr5: SIZEOF (QUERY (it <* SELF.items |
       (SIZEOF
       (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF (it)) = 2) AND
     (it.name IN ['circumscribed diameter','diameter across flats']) ))= 1;

  wr6: (SIZEOF(QUERY ( it <* SELF.items |
       (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
       IN TYPEOF(it)) AND
       ('FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE'
       IN TYPEOF(it\measure_with_unit.value_component)) AND
       (it.name = 'number of sides')) )) = 1);
  END_ENTITY; -- ngon_shape_representation
(*
```

Formal propositions:

WR1: The **ngon_shape_representation** shall contain exactly five **representation_item**s in its set of **items**.

WR2: One of the **representation_item**s used for the implicit representation of a **ngon_shape_-representation** shall be of type **placement** with a **name** of 'orientation'.

WR3: One of the **representation_item**s used for the implicit representation of a **ngon_shape_-representation** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'length'.

WR4: One of the **representation_item**s used for the implicit representation of a **ngon_shape_-representation** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'corner radius'.

©ISO 2000 - All rights reserved

WR5: One of the **representation_item**s used for the implicit representation of a **ngon_shape_-representation** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'circumscribed diameter' or 'diameter across flats'.

WR6: One of the **representation_item**s used for the implicit representation of a **ngon_shape_-representation** shall be of type **measure_representation_item** with a **value_component** of type **count_measure** and a **name** of 'number of sides'.

Informal propositions:

IP1: The location of the **ngon_shape_representation** shall be defined at the center of the enclosed area.

IP2: The **ngon_shape_representation** shall be defined with the enclosed area in the X-Y plane with one of the sides of the ngon parallel to the X direction intersecting the negative Y axis. The length is along the Z direction.

5.2.3.1.44 open_path_profile

An **open_path_profile** is a type of **shape_aspect** that represents a connected set of curves with a location and orientation. The start vertex and end vertex for the set of curves do not connect creating an area open on one side. The **open_path_profile** orientation is in the X-Y plane. See ARM definition for General_open_path_profile in paragraph 4.2.83 for more information.

EXPRESS specification:

```
*)
  ENTITY open_path_profile
    SUBTYPE OF (shape_aspect);
    WHERE
  wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
       IN TYPEOF(SELF.of_shape.definition));

  wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0);

  wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF (impl_rep.used_representation.items) = 1)) ))
       = 0)) )) = 0);

  wr4: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
```

 ©ISO 2000 - All rights reserved

```
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      (('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT'
      IN TYPEOF(it)) AND
      (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);

  wr5: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PATH_SHAPE_REPRESENTATION'
      IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0);

  WR6: SIZEOF (QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
      IN TYPEOF (pdr.used_representation))AND
      (pdr.used_representation.name = 'profile limit') )) <= 1))) = 0;

END_ENTITY;
(*
```

Formal propositions:

WR1: The **open_path_profile** shall be an aspect of the shape of a **feature_component_definition**.

WR2: The **open_path_profile** shall have exactly one implicit representation defined by a **shape_-representation_with_parameters**.

WR3: The **shape_representation_with_parameters** that represents the **open_path_profile** shall contain at least one and at most two **representation_item**s in its set of **items**.

WR4: Exactly one **representation_item** used for the implicit representation of an **open_path_profile** shall be of type **placement** with a **name** of 'orientation'.

WR5: The **open_path_profile** shall have exactly one **path_shape_representation**.

WR6: The **open_path_profile** shall have at most one **planar_shape_representation** with a **name** of 'profile limit'.

Informal propositions:

IP1: The **open_path_profile** shall be defined in the X-Y plane.

©ISO 2000 - All rights reserved

5.2.3.1.45 ordered_part

An **ordered_part** represents the measured amount of product that is to be manufactured.

EXPRESS specification:

```
*)
  ENTITY ordered_part
    SUBTYPE OF (action_assignment, characterized_object);
      items : SET [1:?] OF feature_based_pp_ordered_item;
    WHERE
  wr1: (SIZEOF(USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION'))= 1);

  wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION')) = 1)) ))
       = 0);

  wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (NOT (SIZEOF(items) = 1)) )) = 0)) )) = 0);

  wr4: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( it <* pdr.used_representation.items |
       (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
       IN TYPEOF(it)) AND (it.name = 'quantity')) )) = 1)) ))
       = 0)) )) = 0);
  END_ENTITY; -- ordered_part
(*
```

Attribute definition:

items: the set of **product_definition_formation**s for which an **action** is being assigned.

Formal propositions:

WR1: The **ordered_part** shall have exactly one property defined for it.

WR2: The property defined for the **ordered_part** shall have exactly one representation.

WR3: The representation of the property defined for the **ordered_part** shall contain exactly one **representation_item** in its set of **items**.

WR4: The **representation_item** in the set of **items** for the representation of the property for the **ordered_part** shall be of type **measure_representation_item** with a **name** of 'quantity'.

 ©ISO 2000 - All rights reserved

5.2.3.1.46 outer_round

An **outer_round** is a type of **feature_definition** that is the representation of a volume that is removed from the base shape. The **outer_round** may be represented in one of two ways. This first shall be a removal of a cylindrical volume represented by a diameter about the Z direction of the **feature_-definition** placement axis. It may have a change in diameter which is represented by a **taper**. The second shall be a removal of a cylindrical volume represented by sweeping a **vee_profile** about the Z direction of the **feature_definition** placement axis. See ARM definition for Outer_diameter in paragraph 4.2.135 and Outer_diameter_to_shoulder in paragraph 4.2.136 for more information.

EXPRESS specification:

```
*)
  ENTITY outer_round
    SUBTYPE OF (feature_definition);
    WHERE
  WR1: ((NOT (SELF\characterized_object.description = 'outer diameter')) OR
       (SIZEOF( QUERY( pd <* USEDIN( SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       SIZEOF( QUERY( pdr <* USEDIN( pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF(pdr.used_representation)) AND
       (SIZEOF (pdr.used_representation.items) = 3) )) = 1 )) = 1));

  WR2: ((NOT (SELF\characterized_object.description
       = 'outer diameter to shoulder')) OR
       (SIZEOF( QUERY( pd <* USEDIN( SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       SIZEOF( QUERY( pdr <* USEDIN( pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF(pdr.used_representation)) AND
       (SIZEOF (pdr.used_representation.items) = 2) )) = 1 )) = 1));

  wr3: (SELF\characterized_object.description IN
       ['outer diameter', 'outer diameter to shoulder']);

  wr4: ((NOT (SELF\characterized_object.description = 'outer diameter')) OR
       (SIZEOF( QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       ((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF(it)) = 2) AND (it.name = 'length')) )) = 1)) ))
       = 0)) )) = 0));

  wr5: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
```

©ISO 2000 - All rights reserved

```
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF
      (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF(it)) = 2) AND (it.name = 'diameter')) )) = 1)) ))
      = 0)) )) = 0);

  wr6: ((NOT (SELF\characterized_object.description
      = 'outer diameter to shoulder')) OR (
      SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
      TYPEOF(pd)) ) |
      (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      ((sa_occ.description = 'v-shape boundary occurrence') AND
      (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      ((sar.description = 'profile usage') AND
      (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
      IN TYPEOF(sar))) ) |
      (('FEATURE_BASED_PROCESS_PLANNING.VEE_PROFILE'
      IN TYPEOF(sdr.relating_shape_aspect)) AND
      (sdr.relating_shape_aspect.description = 'v-shape')) )) = 1)) ))
      = 1)) )) = 0));

  wr7: ((NOT (SELF\characterized_object.description = 'outer diameter')) OR
      (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      ((sa_occ.description = 'reduced size occurrence') AND
      (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      ((sar.description = 'taper usage') AND
      (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP')
      IN TYPEOF(sar))) ) |
      (('FEATURE_BASED_PROCESS_PLANNING.TAPER'
      IN TYPEOF(sdr.relating_shape_aspect)) AND
      (sdr.name = 'reduced size')) ))  = 1)) ))
      <= 1)) )) = 0));
  END_ENTITY; -- outer_round
(*
```

Formal proposition:

WR1: The implicit representation of the **outer_round** with **name** 'outer diameter' shall contain exactly three **representation_item**s.

 ©ISO 2000 - All rights reserved

WR2: The implicit representation of the **outer_round** with **name** 'outer diameter to shoulder' shall contain exactly two **representation_item**s.

WR3: The **outer_round** shall have a description of either 'outer diameter' or 'outer diameter to shoulder'.

WR4: If the description of the **outer_round** is 'outer diameter', exactly one **representation_item** used for the implicit representation of the **outer_round** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'length'.

WR5: Exactly one **representation_item** used for the implicit representation of the **outer_round** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'diameter'.

WR6: If the description of the **outer_round** is 'outer diameter to shoulder', the **outer_round** shall be the basis shape for at most one **shape_aspect** with a **description** of 'v-shape boundary occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' and a **relating_shape_aspect** that references a **vee_profile** with a **description** of 'v-shape'.

WR7: If the description of the **outer_round** is 'outer diameter', the **outer_round** shall be the basis shape for at most one **shape_aspect** with a description of 'reduced size occurrence' that is the **related_-shape_aspect** in exactly one **feature_component_relationship** with a **description** of 'taper usage' and a **relating_shape_aspect** that references a **taper** with a **description** of 'reduced size'.

Informal propositions:

IP1: The location of the **outer_round** shall be defined at a position in the Z direction.

IP2: The Z direction of the **outer_round** is the axis of revolution.

IP3: If the description of the **outer_round** is 'outer diameter to shoulder', the placement of the **vee_-profile** shall be along the X direction of the **outer_round** at a specified distance from the origin. The orientation of the **vee_profile** shall be in the same direction in the Y axis. The Y axis of the vee_profile shall be in the same direction as the X axis of the **outer_round**.

5.2.3.1.47 outside_profile

An **outside_profile** is a type of **feature_definition** that is the representation of the outside boundary of the base shape. This boundary may be enclosed and have either a rectangular base shape that is represented by a **rectangular_closed_profile** or it may have a non-rectangular base shape that is represented by either a **ngon_closed_profile**, **rectangular_closed_profile**, **closed_path_profile**, or a **circular_closed_profile**. If the boundary is not enclosed the base shape would be defined by a **linear_-profile**, **square_u_profile**, **partial_circular_profile**, **round_u_profile**, **vee_profile**, **tee_profile**, or an **open_path_profile**. See ARM definition for General_outside_profile in paragraph 4.2.84 for more information.

EXPRESS specification:

```
*)
  ENTITY outside_profile
    SUBTYPE OF (feature_definition);
```

©ISO 2000 - All rights reserved

```
    WHERE
  WR1: SIZEOF( QUERY( pd <* USEDIN( SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       SIZEOF( QUERY( pdr <* USEDIN( pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF(pdr.used_representation)) AND
       (SIZEOF (pdr.used_representation.items) = 1) )) = 1 )) = 1;

  WR2: SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
       IN TYPEOF (pd)) |
       NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       (sa_occ.description IN ['boundary occurrence',
       'non-planar boundary occurrence',
       'partial circular boundary occurrence',
       'closed circular boundary occurrence',
       'open rectangular boundary occurrence',
       'closed rectangular boundary occurrence']) )) =1)  ))=0;

  WR3: (NOT(SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
       IN TYPEOF (pd)) |
       NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       (sa_occ.description = 'boundary occurrence') )) =1)  ))=0))  OR
       (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
       IN TYPEOF (pd)) |
       NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       (SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
       'RELATED_SHAPE_ASPECT') |
       (sar.description = 'profile usage') AND
       ('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_DEFINING_RELATIONSHIP' IN TYPEOF (sar))) |
       (SIZEOF (['FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.NGON_CLOSED_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.CLOSED_PATH_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.SQUARE_U_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.ROUNDED_U_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.LINEAR_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.VEE_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.TEE_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.OPEN_PATH_PROFILE'] *
       TYPEOF (sdr.relating_shape_aspect)) = 1) AND
       (sdr.relating_shape_aspect.description = 'outside boundary')))
       = 1))) = 1))) = 0);

WR4:   (NOT(SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
       IN TYPEOF (pd)) |
```

 ©ISO 2000 - All rights reserved

```
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (sa_occ.description IN ['complex boundary occurrence',
      'partial circular boundary occurrence',
      'closed circular boundary occurrence',
      'open rectangular boundary occurrence',
      'closed rectangular boundary occurrence']) ))=1)   ))=0)) OR
      (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF (pd)) |
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      NOT(SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      (sar.description = 'profile floor usage') AND
      ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
      IN TYPEOF (sar))) |
      'FEATURE_BASED_PROCESS_PLANNING.PROFILE_FLOOR'
      IN TYPEOF (sdr.relating_shape_aspect))) = 1))) = 0))) = 0);

WR5:  (NOT(SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF (pd)) |
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (sa_occ.description IN ['outside boundary',
      'complex boundary occurrence',
      'partial circular boundary occurrence',
      'closed circular boundary occurrence',
      'open rectangular boundary occurrence',
      'closed rectangular boundary occurrence']) ))=1)   ))=0)) OR
      (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF (pd)) |
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      (sar.description = 'path feature component usage') AND
      ('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_DEFINING_RELATIONSHIP' IN TYPEOF (sar))) |
      (SIZEOF (['FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT'] *
      TYPEOF (sdr.relating_shape_aspect)) = 1) AND
      (sdr.name = 'profile swept shape') AND
      (sdr.relating_shape_aspect.description='linear') ))  = 1)))
      = 1))) = 0);

 WR6: (NOT(SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF (pd)) |
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (sa_occ.description = 'complex boundary occurrence') ))
      =1)  ))=0))  OR
      (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
```

```
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF (pd)) |
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      (sar.description = 'profile usage') AND
      ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
      IN TYPEOF (sar))) |
      (SIZEOF (['FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.NGON_CLOSED_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.CLOSED_PATH_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.SQUARE_U_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.ROUNDED_U_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.VEE_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.TEE_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.LINEAR_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.OPEN_PATH_PROFILE'] *
      TYPEOF (sdr.relating_shape_aspect)) =1))  )
      = 1)))= 1))) = 0);

  WR7: (NOT(SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF (pd)) |
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (sa_occ.description = 'partial circular boundary occurrence') ))
      =1)  ))=0))  OR
      (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF (pd)) |
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      (sar.description = 'profile usage') AND
      ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
      IN TYPEOF (sar))) |
      ('FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE' IN
      TYPEOF (sdr.relating_shape_aspect))))
      = 1)))= 1))) = 0);

  WR8: (NOT(SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF (pd)) |
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (sa_occ.description = 'closed circular boundary occurrence') ))
      =1)  ))=0))  OR
      (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF (pd)) |
```

 ©ISO 2000 - All rights reserved

```
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      (sar.description = 'profile usage') AND
      ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
      IN TYPEOF (sar))) |
      ('FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE' IN
      TYPEOF (sdr.relating_shape_aspect))))
      = 1)))= 1))) = 0);

WR9: (NOT(SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF (pd)) |
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (sa_occ.description = 'open rectangular boundary occurrence') ))
      =1)  ))=0))  OR
      (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF (pd)) |
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      (sar.description = 'profile usage') AND
      ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
      IN TYPEOF (sar))) |
      ('FEATURE_BASED_PROCESS_PLANNING.SQUARE_U_PROFILE' IN
      TYPEOF (sdr.relating_shape_aspect))))
      = 1)))= 1))) = 0);

WR10: (NOT(SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF (pd)) |
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (sa_occ.description = 'closed rectangular boundary occurrence') ))
      =1)  ))=0))  OR
      (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF (pd)) |
      NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      (sar.description = 'profile usage') AND
      ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
      IN TYPEOF (sar))) |
      ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE' IN
      TYPEOF (sdr.relating_shape_aspect)) )) = 1)))
      = 1))) = 0);
```

©ISO 2000 - All rights reserved

```
 wr11:  (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
       IN TYPEOF (pd)) |
       NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       (sa_occ.description IN ['boundary occurrence',
       'complex boundary occurrence',
       'partial circular boundary occurrence',
       'closed circular boundary occurrence',
       'open rectangular boundary occurrence',
       'closed rectangular boundary occurrence']) )) =1)  ))=0) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
       IN TYPEOF(pdr.used_representation)) AND
       (pdr.used_representation.name = 'removal direction')) )) = 1)) ))
       = 0);

  END_ENTITY; -- outside_profile
(*
```

Formal proposition:

WR1: The implicit representation of the **outside_profile** shall contain exactly one **representation_item**.

WR2: The **outside_profile** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'boundary occurrence' , 'complex boundary occurrence', 'partial circular boundary occurrence', 'closed circular boundary occurrence', 'open rectangular boundary occurrence', or 'closed rectangular boundary occurrence'.

WR3: If the **outside_profile** has a **shape_aspect** with a **description** of 'boundary occurrence', then the **shape_aspect** is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' and a **relating_shape_aspect** that references a **circular_closed_profile**, **ngon_closed_profile**, **rectangular_closed_profile**, **closed_path_profile**, **square_u_profile**, **partial_-circular_profile**, **rounded_u_profile**, **linear_profile**, **vee_profile**, **tee_profile**, or **open_path_profile** with a **description** of 'outside boundary'.

WR4: If the **outside_profile** has a **shape_aspect** with a **description** of 'complex boundary occurrence', 'partial circular boundary occurrence', 'closed circular boundary occurrence', 'open rectangular boundary occurrence', 'closed rectangular boundary occurrence', then the **outside_profile** shall be the basis shape for exactly one **shape_aspect** that is the **related_shape_aspect** in exactly one **feature_component_-relationship** with a **description** of 'profile floor usage' and a **relating_shape_aspect** that references a **profile_floor**.

WR5: If the **outside_profile** has a **shape_aspect** with a **description** of 'outside boundary', 'complex boundary occurrence', 'partial circular boundary occurrence', 'closed circular boundary occurrence', 'open rectangular boundary occurrence', 'closed rectangular boundary occurrence', then the **outside_profile** shall be the basis shape for exactly one **shape_aspect** that is the **related_shape_aspect** in exactly one

 ©ISO 2000 - All rights reserved

shape_defining_relationship with a **description** of 'path feature component usage' and a **relating_-shape_aspect** that references a **path_feature_component** with a **description** of 'linear'.

WR6: If the **outside_profile** has a **shape_aspect** with a **description** of 'complex boundary occurrence', then the **outside_profile** shall be the basis shape for exactly one **shape_aspect** that is the **related_-shape_aspect** in exactly one **shape_defining_relationship** with a **description** of of 'profile usage' in which the **relating_shape_aspect** is either a **circular_closed_profile**, **ngon_closed_profile**, **rectangular_closed_profile**, **closed_path_profile**, **square_u_profile**, **partial_circular_profile**, **rounded_u_profile**, **vee_profile**, **tee_profile**, **linear_profile**, or **open_path_profile**.

WR7: If the **outside_profile** has a **shape_aspect** with a **description** of 'partial circular boundary occurrence', then the **outside_profile** shall be the basis shape for exactly one **shape_aspect** that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' in which the **relating_shape_aspect** is a **partial_circular_profile**.

WR8: If the **outside_profile** has a **shape_aspect** with a **description** of 'closed circular boundary occurrence', then the **outside_profile** shall be the basis shape for exactly one **shape_aspect** that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' in which the **relating_shape_aspect** is a **circular_closed_profile**.

WR9: If the **outside_profile** has a **shape_aspect** with a **description** of 'open rectangular boundary occurrence', then the **outside_profile** shall be the basis shape for exactly one **shape_aspect** that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' in which the **relating_shape_aspect** is a **square_u_profile**.

WR10: If the **outside_profile** has a **shape_aspect** with a **description** of 'closed rectangular boundary occurrence', then the **outside_profile** shall be the basis shape for exactly one **shape_aspect** that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' in which the **relating_shape_aspect** is a **rectangular_closed_profile**.

WR11: If the **outside_profile** has a **shape_aspect** with a **description** of 'complex boundary occurrence', 'partial circular boundary occurrence', 'closed circular boundary occurrence', 'open rectangular boundary occurrence', 'closed rectangular boundary occurrence', then the **outside_profile** shall have exactly one **direction_shape_representation** with a **name** of 'removal direction'.

Informal propositions:

IP1: The **outside_profile** shall be linear along the Z-axis of the feature.

IP2: The origin, X direction, Y direction of the **profile** that defines the shape of the **outside_profile** shall be with the X and Y axes equal to those specified by the placement of the **outside_profile**.

5.2.3.1.48 partial_circular_profile

A **partial_circular_profile** is a type of **shape_aspect** that is the representation of arc of constant radius, with a location and a position. The **partial_circular_profile** is located at the arc origin , the X direction starts at the arc origin in the direction of one of the arc end points and the Y direction orthogonal in the direction of volume removal. See ARM definition for Partial_circular_profile in paragraph 4.2.144 for more information.

©ISO 2000 - All rights reserved

EXPRESS specification:

```
*)
ENTITY partial_circular_profile
    SUBTYPE OF (shape_aspect);
    WHERE
  wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
       IN TYPEOF(SELF.of_shape.definition));
  wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0);
  wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF (impl_rep.used_representation.items) >= 3)) ))
       = 0)) )) = 0);
  wr4: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       (('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT'
       IN TYPEOF(it)) AND (it.name = 'orientation')) )) = 1)) ))
       = 0)) )) = 0);
  wr5: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       ((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF(it)) = 2) AND (it.name = 'radius')))) = 1)) ))
       = 0)) )) = 0);
  wr6: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
```

 ©ISO 2000 - All rights reserved

```
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT'] *
      TYPEOF(it)) = 2) AND (it.name = 'sweep angle')) )) = 1)) ))
      = 0)) )) = 0);

  WR7: SIZEOF (QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
      IN TYPEOF (pdr.used_representation))AND
      (pdr.used_representation.name = 'profile limit'))) <= 1))) = 0;

  END_ENTITY; -- partial_circular_profile
(*
```

Formal propositions:

WR1: The **partial_circular_profile** shall be an aspect of the shape of a **feature_component_-definition**.

WR2: The **partial_circular_profile** shall have exactly one implicit representation defined by a **shape_-representation_with_parameters**.

WR3: The **shape_representation_with_parameters** that represents the **partial_circular_profile** shall contain at least three and at most four **representation_item**s in its set of **items**.

WR4: Exactly one **representation_item** used for the implicit representation of a **partial_circular_-profile** shall be of type **placement** with a **name** of 'orientation'.

WR5: Exactly one **representation_item** used for the implicit representation of a **partial_circular_-profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'radius'.

WR6: Exactly one **representation_item** used for the implicit representation of a **partial_circular_-profile** shall be of type **measure_representation_item** and **plane_angle_measure_with_unit** with a **name** of 'sweep angle'.

WR7: The **partial_ciruclar_profile** shall have at most one **planar_shape_representation** with a **name** of 'profile limit'.

Informal propositions:

IP1: The location of the **partial_circular_profile** shall be at the arc origin.

IP2: The **partial_circular_profile** shall be defined with the X direction starting at the arc origin in the direction of one of the arc end points and the Y direction orthogonal.

©ISO 2000 - All rights reserved

5.2.3.1.49 path_feature_component

A **path_feature_component** is a type of **shape_aspect** that is the representation 2D or 3D curve or set of curves. See ARM definition for Path in paragraph 4.2.146 for more information.

EXPRESS specification:

```
*)
  ENTITY path_feature_component
    SUBTYPE OF (shape_aspect);
    WHERE
  wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
       IN TYPEOF(SELF.of_shape.definition));

  wr2: (SELF.description IN ['partial circular','complete circular',
       'linear','complex']);

  wr3: ((NOT (SELF.description = 'complex')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0));

  wr4: ((SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       (('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT'
       IN TYPEOF(it)) AND (it.name = 'orientation')) )) = 1)) ))
       = 0)) )) = 0));

  wr5: ((NOT (SELF.description = 'partial circular')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(impl_rep.used_representation.items) = 3)) ))
       = 0)) )) = 0));

  wr6: ((NOT (SELF.description = 'partial circular')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
```

 ©ISO 2000 - All rights reserved

```
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF(it)) = 2) AND (it.name = 'radius')) )) = 1)) ))
      = 0)) )) = 0));

wr7: ((NOT (SELF.description = 'partial circular')) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT'] *
      TYPEOF(it)) = 2) AND (it.name = 'sweep angle')) )) = 1)) ))
      = 0)) )) = 0));

wr8: ((NOT (SELF.description = 'complete circular')) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(impl_rep.used_representation.items) = 2)) ))
      = 0)) )) = 0));
wr9: ((NOT (SELF.description = 'complete circular')) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF(it)) = 2) AND (it.name = 'radius')) )) = 1)) ))
      = 0)) )) = 0));
wr10: ((NOT (SELF.description = 'linear')) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN( pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(impl_rep.used_representation.items) = 2)) ))
```

©ISO 2000 - All rights reserved

```
        = 0)) )) = 0));
 wr11: ((NOT (SELF.description = 'linear')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       ((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF(it)) = 2) AND (it.name = 'distance')) )) = 1)) ))
       = 0)) )) = 0));
 wr12: ((NOT (SELF.description = 'linear')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
       IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0));

  wr13: ((NOT (SELF.description = 'complex')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.PATH_SHAPE_REPRESENTATION'
       IN TYPEOF(pdr.used_representation)) AND
       (pdr.used_representation.name = 'sweep path')AND
       (SIZEOF( QUERY( srwp_i <* pdr.used_representation.items |
       (srwp_i.name = 'profile shape') )) = 1)
       ) )) = 1)) )) = 0));

  END_ENTITY; -- path_feature_component
(*
```

Formal propositions:

WR1: The **path_feature_component** shall be an aspect of the shape of a **feature_component_-definition**.

WR2: The **description** for the **path_feature_component** shall be either 'partial circular', 'complete circular', 'linear', or 'complex'.

WR3: If the **description** of the **path_feature_component** is 'complex', the **path_feature_component** shall have its implicit representation specified by exactly one **shape_representation_with_parameters**.

WR4: The **path_feature_component** shall have one **representation_item** used for the implicit representation specified by a **shape_representation_with_parameters**. This **shape_representation_-with_parameters** shall contain exactly one **representation_item** of type **placement** with a **name** of 'orientation'.

 ©ISO 2000 - All rights reserved

WR5: If the **description** of the **path_feature_component** is 'partial circular', the **path_feature_-component** shall have an implicit representation that contains exactly three **representation_item**s in its set of **items**.

WR6: If the **description** of the **path_feature_component** is 'partial circular', exactly one **representation_item** used for the implicit representation of the **path_feature_component** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'radius'.

WR7: If the **description** of the **path_feature_component** is 'partial circular', exactly one **representation_item** used for the implicit representation of the **path_feature_component** shall be of type **measure_representation_item** and **plane_angle_measure_with_unit** with a **name** of 'sweep angle'.

WR8: If the **description** of the **path_feature_component** is 'complete circular', the **path_feature_-component** shall have an implicit representation that contains exactly two **representation_item**s in its set of **items**.

WR9: If the **description** of the **path_feature_component** is 'complete circular', exactly one **representation_item** used for the implicit representation of the **path_feature_component** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'radius'.

WR10: If the **description** of the **path_feature_component** is 'linear', the **path_feature_component** shall have an implicit representation that contains exactly two **representation_item**s in its set of **items**.

WR11: If the **description** of the **path_feature_component** is 'linear', exactly one **representation_item** used for the implicit representation of the **path_feature_component** shall be of type **measure_-representation_item** and **length_measure_with_unit** with a **name** of 'distance'.

WR12: If the **description** of the **path_feature_component** is 'linear', the **path_feature_component** shall have exactly one **direction_shape_representation**..

WR13: If the **description** of the **path_feature_component** is 'complex', the **path_feature_component** shall have exactly one **path_shape_representation** with a **name** of 'sweep path'. This **path_shape_representation** shall contain exactly one **representation_item** with a **name** of 'profile shape'.

Informal propositions:

IP1: The location of the **path_feature_component** may be the same location used by the feature that references **path_feature_component**.

IP2: The origin, X direction, and Y direction of the **path_feature_component** may be the same as that used by the feature that references **path_feature_component**.

5.2.3.1.50 path_shape_representation

A **path_shape_representation** is a type of **representation** which represents a curved path using an ordered collection of geometric curves such that the collection of curves are continuous and directed.

©ISO 2000 - All rights reserved

EXPRESS specification:

```
*)
  ENTITY path_shape_representation
    SUBTYPE OF (shape_representation);
    WHERE
      wr1: (SIZEOF(SELF.items) >= 1);
      wr2: (SIZEOF(QUERY ( i <* SELF.items | (SIZEOF([
               'FEATURE_BASED_PROCESS_PLANNING.BOUNDED_CURVE',
               'FEATURE_BASED_PROCESS_PLANNING.EDGE_CURVE',
               'FEATURE_BASED_PROCESS_PLANNING.PATH'] * TYPEOF(i)) =
               1) )) >= 1);
  END_ENTITY; -- path_shape_representation
(*
```

Formal propositions:

WR1: The **path_shape_representation** shall have one or more **representation_item** in its set of **items**.

WR2: The geometric element that is used to represent the **path_shape_representation** shall be a **bounded_curve**, **edge_curve**,or **path**.

5.2.3.1.51 pattern_offset_membership

A **pattern_offset_membership** specifies the relationships necessary to offset a base feature from either a **rectangular_pattern** or a **circular_pattern**.

EXPRESS specification:

```
*)
  ENTITY pattern_offset_membership
    SUBTYPE OF (feature_component_relationship);
    WHERE
  wr1 : (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(
        SELF.relating_shape_aspect,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
        + 'RELATING_SHAPE_ASPECT') |
        (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
        IN TYPEOF(sar)) AND (sar :<>: SELF)) ) |
        ((SIZEOF (QUERY( pdr <*(QUERY(pd <* USEDIN
        (fcr.related_shape_aspect.of_shape,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
        IN TYPEOF(pd))) |
        SIZEOF (['FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN',
        'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN']
        * TYPEOF(pdr.definition))  =1 )) = 0 )) )) = 0);

  wr2 : (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(
        SELF.related_shape_aspect,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
        'RELATED_SHAPE_ASPECT') |
        (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
        IN TYPEOF(sar)) AND (sar :<>: SELF)) ) |
       (fcr.description='modified pattern') AND
        ('FEATURE_BASED_PROCESS_PLANNING.MODIFIED_PATTERN'
```

 ©ISO 2000 - All rights reserved

```
      IN TYPEOF(fcr.relating_shape_aspect)) )) >= 1);

wr3 : (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(
      SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
      IN TYPEOF(sar)) AND (sar :<>: SELF)) ) |
      (('FEATURE_BASED_PROCESS_PLANNING.MODIFIED_PATTERN'
      IN TYPEOF(fcr.relating_shape_aspect)) AND
      (NOT (SIZEOF(QUERY ( modfcr <* QUERY ( modsar <* USEDIN(
      fcr.relating_shape_aspect,'FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_ASPECT_RELATIONSHIP.RELATING_SHAPE_ASPECT') |
      ((SIZEOF(['FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN',
      'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN'] *
      TYPEOF(modsar.related_shape_aspect.of_shape.definition)) = 1) AND
      (modsar :<>: fcr)) ) |
      (NOT (modfcr.related_shape_aspect.of_shape.definition :=:
      SELF.relating_shape_aspect.of_shape.definition)) ))
       = 0))) )) = 0);

wr4 : ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN'
      IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(USEDIN(pd,'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION')) = 2)) )) = 0));

wr5 : ((NOT ('FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN'
      IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(USEDIN(pd,'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION')) = 1)) )) = 0));

wr6 : ((NOT ('FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN'
      IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (NOT (SIZEOF(pdr.used_representation.items) = 2)) )) = 0)) ))
      = 0));

wr7 : ((NOT ('FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN'
      IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
      IN TYPEOF(it)) AND
      ('FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE'
      IN TYPEOF(it\measure_with_unit.value_component)) AND
      (it.name = 'index number')) )) = 1)) )) = 0)) )) = 0));
```

©ISO 2000 - All rights reserved

```
wr8 : ((NOT ('FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN'
      IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      (('FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT'
      IN TYPEOF(it)) AND (it.name = 'offset')) )) = 1)) )) = 0)) ))
      = 0));

wr9:  ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN'
      IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')|
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(impl_rep.used_representation.items ) = 3)) ))
      = 0)) )) = 0));

wr10: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN'
      IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')|
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
      IN TYPEOF(it)) AND
      ('FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE'
      IN TYPEOF(it\measure_with_unit.value_component)) AND
      (it.name = 'row index')) )) = 1)) )) = 0)) )) = 0));

wr11: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN'
      IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
      IN TYPEOF(it)) AND
      ('FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE'
      IN TYPEOF(it\measure_with_unit.value_component)) AND
      (it.name = 'column index')) )) = 1)) )) = 0)) )) = 0));
```

 ©ISO 2000 - All rights reserved

```
  wr12: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN'
        IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
        (SIZEOF(QUERY ( pd <* USEDIN(SELF.related_shape_aspect,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
        (('FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'
        IN TYPEOF(it)) AND (it.name = 'offset distance')) )) = 1)) ))
        = 0)) )) = 0));

  wr13: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN'
        IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
        (SIZEOF(QUERY ( pd <* USEDIN(SELF.related_shape_aspect,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
        IN TYPEOF(pdr.used_representation)) AND
        (pdr.used_representation.name = 'offset direction')) )) = 1)) ))
        = 0));

  END_ENTITY; -- pattern_offset_membership
(*
```

Formal propositions:

WR1: The **pattern_offset_membership** shall define an offset relationship for a member of either a **circular_pattern** or a **rectangular_pattern**

WR2: The **shape_aspect** that defines the offset is defined by a **related_shape_aspect** that defines the omission in the **pattern_offset_membership.** This shall be incorporated in at least one basis shape of a **modified_pattern** by a **feature_component_relationship**with a **description** of ’modified pattern’.

WR3: The **pattern_offset_membership** shall define a set of entity instances such that the **related_-shape_aspect** defines the offset which is used as the **related_shape_aspect** in an instance of a **feature_-component_relationship** that relates the offset to the **modified_pattern** that incorporates all of the offsets for exactly one **circular_pattern** or **rectangular_pattern**.

NOTE - Figure 111 depicts the instances defined by this constraint.

WR4: If the **shape_aspect** referenced by the **relating_shape_aspect** of the **pattern_offset_-membership** is a **rectangular_pattern**, **it** shall have exactly two **representation**s.

WR5: If the **shape_aspect** referenced by the **relating_shape_aspect** of the **pattern_offset_-membership** is a **circular_pattern**, it shall have exactly one **representation**.

©ISO 2000 - All rights reserved

WR6: If the **shape_aspect** referenced by the **relating_shape_aspect** of the **pattern_offset_-membership** is a **circular_pattern**, the **representation** of the **shape_aspect** referenced by the **related_-shape_aspect** of the **pattern_offset_membership** shall contain two **representation_item**s in its set of **items**.

WR7: If the **shape_aspect** referenced by the **relating_shape_aspect** of the **pattern_offset_-membership** is a **circular_pattern**, exactly one **representation_item** used for the representation of the **shape_aspect** referenced by the **related_shape_aspect** of the **pattern_offset_membership** shall be of type **measure_representation_item** with a **value_component** of type **count_measure** and a **name** of 'index number'.

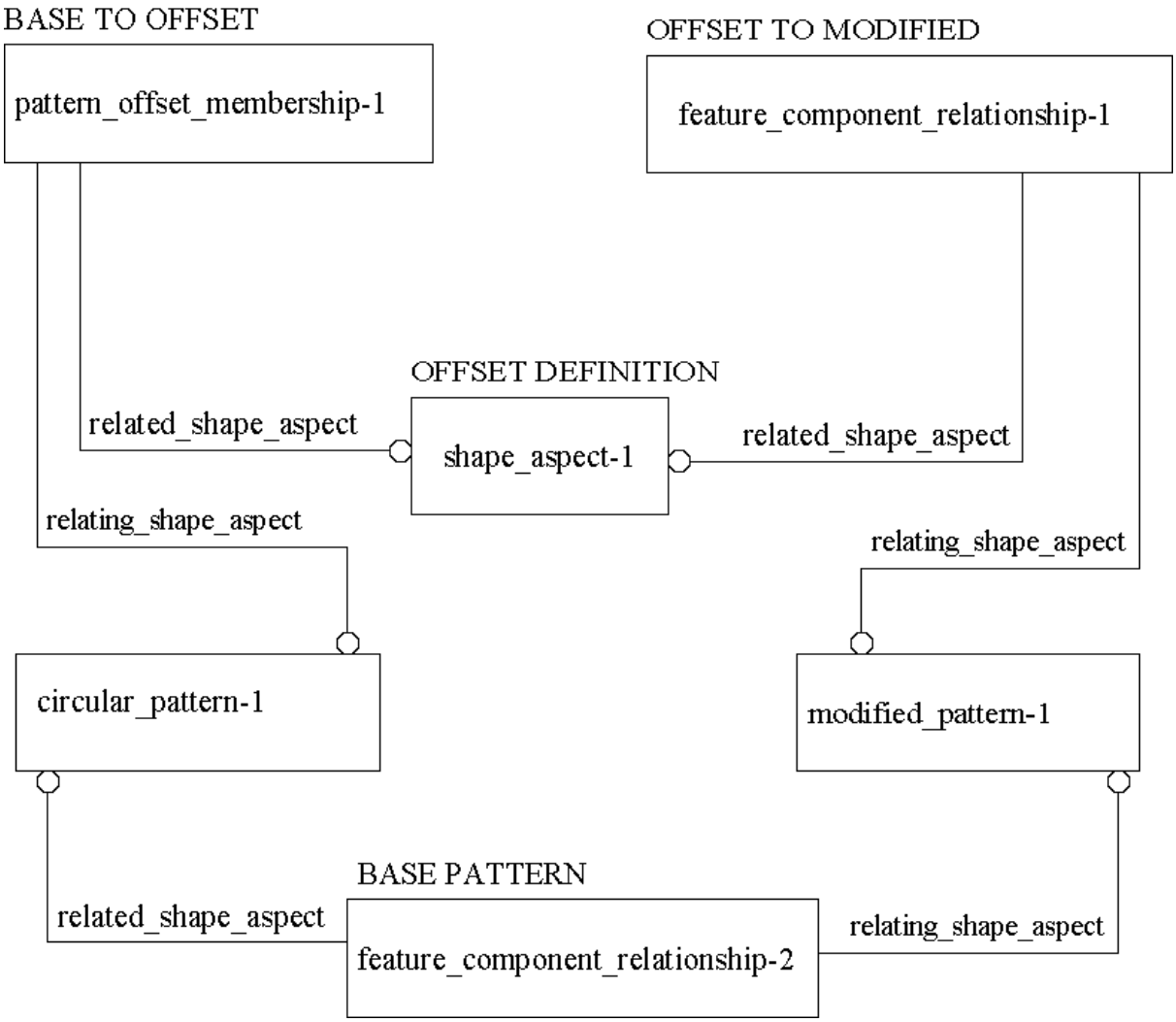


Figure 111 - Pattern offset required instances

 ©ISO 2000 - All rights reserved

WR8: If the **shape_aspect** referenced by the **relating_shape_aspect** of the **pattern_offset_-membership** is a **circular_pattern**, exactly one **representation_item** used for the representation of the **shape_aspect** referenced by the **related_shape_aspect** of the **pattern_offset_membership** shall be of type **measure_representation_item** and **plane_angle_measure_with_unit** with a **name** of 'offset'.

WR9: If the **shape_aspect** referenced by the **relating_shape_aspect** of the **pattern_offset_-membership** is a **rectangular_pattern**, the **representation** of the **shape_aspect** referenced by the **related_shape_aspect** of the **pattern_offset_membership** shall contain three **representation_item**s in its set of **items**.

WR10: If the **shape_aspect** referenced by the **relating_shape_aspect** of the **pattern_offset_-membership** is a **rectangular_pattern**, exactly one **representation_item** used for the representation of the **shape_aspect** referenced by the **related_shape_aspect** of the **pattern_offset_membership** shall be of type **measure_representation_item** with a **value_component** of type **count_measure** and a **name** of 'row index'.

WR11: If the **shape_aspect** referenced by the **relating_shape_aspect** of the **pattern_offset_-membership** is a **rectangular_pattern**, exactly one **representation_item** used for the representation of the **shape_aspect** referenced by the **related_shape_aspect** of the **pattern_offset_membership** shall be of type **measure_representation_item** with a **value_component** of type **count_measure** and a **name** of 'column index'.

WR12: If the **shape_aspect** referenced by the **relating_shape_aspect** of the **pattern_offset_-membership** is a **rectangular_pattern**, exactly one **representation_item** used for the representation of the **shape_aspect** referenced by the **related_shape_aspect** of the **pattern_offset_membership** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'offset distance'.

WR13: If the **shape_aspect** referenced by the **relating_shape_aspect** of the **pattern_offset_-membership** is a **rectangular_pattern**, the **shape_aspect** referenced by the **related_shape_aspect** of the **pattern_offset_membership** shall have exactly one **direction_shape_representation** with a **name** of 'offset direction'.

5.2.3.1.52 pattern_omit_membership

A **pattern_omit_membership** specifies the relationships necessary to omit a base feature from either a **rectangular_pattern** or a **circular_pattern**.

EXPRESS specification:

```
*)
  ENTITY pattern_omit_membership
    SUBTYPE OF (feature_component_relationship);
    WHERE
 wr1 : (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(
       SELF.relating_shape_aspect,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
       + 'RELATING_SHAPE_ASPECT') |
       (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
       IN TYPEOF(sar)) AND (sar :<>: SELF)) ) |
       ((SIZEOF (QUERY( pdr <*(QUERY(pd <* USEDIN
```

©ISO 2000 - All rights reserved

```
      (fcr.related_shape_aspect.of_shape,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pd))) |
      SIZEOF (['FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN',
      'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN']
      * TYPEOF(pdr.definition))  =1 )) = 0 )) )) = 0);

  wr2: (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(
      SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
      IN TYPEOF(sar)) AND (sar :<>: SELF)) ) |
      (fcr.description='modified pattern') AND
      ('FEATURE_BASED_PROCESS_PLANNING.MODIFIED_PATTERN'
      IN TYPEOF(fcr.relating_shape_aspect)) )) >= 1);

  wr3: (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(
      SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
      IN TYPEOF(sar)) AND (sar :<>: SELF)) ) |
      (('FEATURE_BASED_PROCESS_PLANNING.MODIFIED_PATTERN'
      IN TYPEOF(fcr.relating_shape_aspect)) AND
      (NOT (SIZEOF(QUERY ( modfcr <* QUERY ( modsar <* USEDIN(
      fcr.relating_shape_aspect,'FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_ASPECT_RELATIONSHIP.RELATING_SHAPE_ASPECT') |
      ((SIZEOF(['FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN',
      'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN'] *
      TYPEOF(modsar.related_shape_aspect.of_shape.definition)) = 1) AND
      (modsar :<>: fcr)) ) |
      (NOT (modfcr.related_shape_aspect.of_shape.definition :=:
      SELF.relating_shape_aspect.of_shape.definition)) ))
       = 0))) )) = 0);

  wr4: (SIZEOF(QUERY ( pd <* USEDIN(SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(USEDIN(pd,'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION')) = 1)) )) = 0);

  wr5: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN'
      IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (NOT (SIZEOF(pdr.used_representation.items) = 1)) )) = 0)) )) = 0));

  wr6: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN'
      IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
```

 ©ISO 2000 - All rights reserved

```
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
      IN TYPEOF(it)) AND
      ('FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE'
      IN TYPEOF(it\measure_with_unit.value_component)) AND
      (it.name = 'index number')) )) = 1)) )) = 0)) )) = 0));

  wr7: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN'
      IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (NOT (SIZEOF(pdr.used_representation.items) = 2)) )) = 0)) )) = 0));

  wr8: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN'
      IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
      IN TYPEOF(it)) AND ('FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE'
      IN TYPEOF(it\measure_with_unit.value_component)) AND
      (it.name = 'row index')) )) = 1)) )) = 0)) )) = 0));

  wr9: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN'
      IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
      IN TYPEOF(it)) AND ('FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE'
      IN TYPEOF(it\measure_with_unit.value_component)) AND
      (it.name = 'column index')) )) = 1)) )) = 0)) )) = 0));

  END_ENTITY; -- pattern_omit_membership
(*
```

Formal propositions:

WR1: The **pattern_omit_membership** shall define an omit relationship for a member of either a **circular_pattern** or a **rectangular_pattern**

©ISO 2000 - All rights reserved

WR2: The **related_shape_aspect** that defines the omission in the **pattern_offset_membership** shall be incorporated in at least one basis shape of a **modified_pattern** by a **feature_-component_relationship**with a **description** of 'modified pattern'.

WR3: The **pattern_omit_membership** shall define a set of entity instances such that the **related_-shape_aspect** defines the offset which is used as the **related_shape_aspect** in an instance of a **feature_-component_relationship** that relates the offset to the **modified_pattern** that incorporates all of the offsets for exactly one **circular_pattern** or **rectangular_pattern**.

NOTE - Figure 112 depicts the instances defined by this constraint.

WR4: The **shape_aspect** that is referenced by **related_shape_aspect** of the **pattern_omit_-membership** shall have exactly one **representation**.

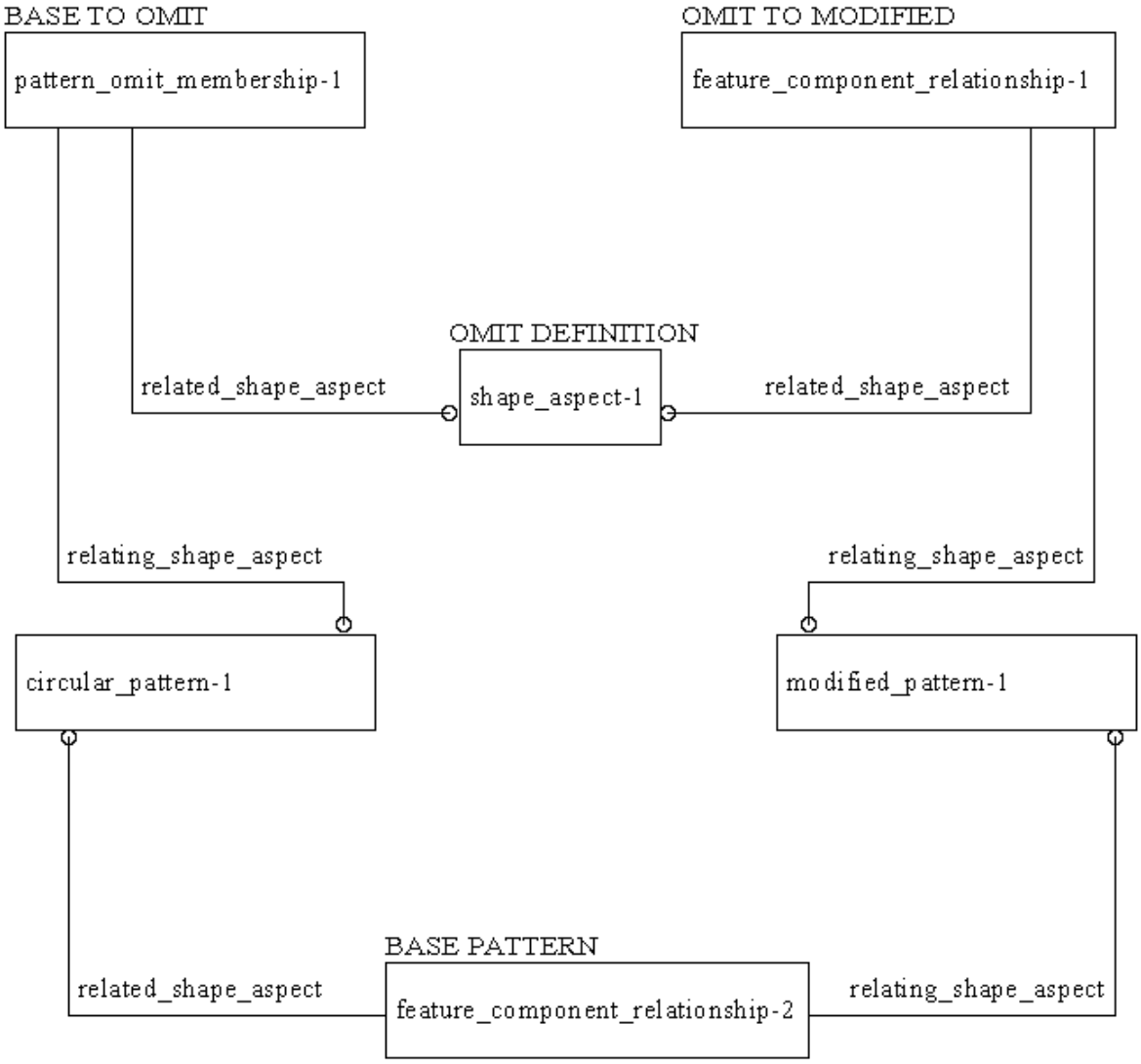


Figure 112 - Pattern omit required instances

 ©ISO 2000 - All rights reserved

WR5: If the **shape_aspect** referenced by the **relating_shape_aspect** of the **pattern_omit_membership** is a **circular_pattern**, the **representation** of the **shape_aspect** referenced by the **related_shape_aspect** of the **pattern_omit_membership** shall contain one **representation_item** in its set of **items**.

WR6: If the **shape_aspect** referenced by the **relating_shape_aspect** of the **pattern_omit_membership** is a **circular_pattern**, exactly one **representation_item** used for the representation of the **shape_aspect** referenced by the **related_shape_aspect** of the **pattern_omit_membership** shall be of type **measure_-representation_item** with a **value_component** of type **count_measure** and a **name** of 'index number'.

WR7: If the **shape_aspect** referenced by the **relating_shape_aspect** of the **pattern_omit_membership** is a **rectangular_pattern**, the **representation** of the **shape_aspect** referenced by the **related_shape_-aspect** of the **pattern_omit_membership** shall contain two **representation_item**s in its set of **items**.

WR8: If the **shape_aspect** referenced by the **relating_shape_aspect** of the **pattern_omit_membership** is a **rectangular_pattern**, exactly one **representation_item** used for the representation of the **shape_-aspect** referenced by the **related_shape_aspect** of the **pattern_omit_membership** shall be of type **measure_representation_item** with a **value_component** of type **count_measure** and a **name** of 'row index'.

WR9: If the **shape_aspect** referenced by the **relating_shape_aspect** of the **pattern_omit_membership** is a **rectangular_pattern**, exactly one **representation_item** used for the representation of the **shape_-aspect** referenced by the **related_shape_aspect** of the **pattern_omit_membership** shall be of type **measure_representation_item** with a **value_component** of type **count_measure** and a **name** of 'column index'.

5.2.3.1.53 placed_datum_target_feature

A **placed_datum_target_feature** is a type of **datum_target** that represents the implicit definition. See ARM definition for Placed_target in paragraph 4.2.152 for more information.

EXPRESS specification:

```
*)
  ENTITY placed_datum_target_feature
    SUBTYPE OF (datum_target);
    WHERE
  wr1 : (SELF.description IN ['point','line','rectangle','circle']);

  wr2 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0);

  wr3 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( dtm_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
```

©ISO 2000 - All rights reserved

```
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* dtm_rep.used_representation.items |
        (('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT'
        IN TYPEOF(it)) AND (it.name = 'orientation')) )) = 1)) ))
        = 0)) )) = 0);

  wr4 : ((NOT (SELF.description = 'point')) OR
        (SIZEOF(QUERY ( pd <*  USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( dtm_rep <*
        QUERY ( pdr <* USEDIN(pd,'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(dtm_rep.used_representation.items) = 1)) ))
        = 0)) )) = 0));

  wr5 : ((NOT (SELF.description IN ['line','circle'])) OR
        (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( dtm_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(dtm_rep.used_representation.items) = 2)) ))
         = 0)) )) = 0));

  wr6 : ((NOT (SELF.description = 'rectangle')) OR
        (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( dtm_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(dtm_rep.used_representation.items) = 3)) ))
        = 0)) )) = 0));

  wr7 : ((NOT (SELF.description = 'circle')) OR
        (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( dtm_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* dtm_rep.used_representation.items |
        ((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
        TYPEOF(it)) = 2) AND (it.name = 'target diameter')) ))
        = 1)) )) = 0)) )) = 0));
```

 ©ISO 2000 - All rights reserved

```
  wr8 : ((NOT (SELF.description = 'line')) OR
        (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( dtm_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* dtm_rep.used_representation.items |
        ((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
        TYPEOF(it)) = 2) AND (it.name = 'target length')))) = 1)) ))
        = 0)) )) = 0));

  wr9 : ((NOT (SELF.description = 'rectangle')) OR
        (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( dtm_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* dtm_rep.used_representation.items |
        ((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
        TYPEOF(it)) = 2) AND (it.name = 'target length')))) = 1)) ))
        = 0)) )) = 0));

  wr10: ((NOT (SELF.description = 'rectangle')) OR
        (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( dtm_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* dtm_rep.used_representation.items |
        ((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
        TYPEOF(it)) = 2) AND (it.name = 'target width'))))= 1))))
        = 0)) ))=0));
  END_ENTITY; -- placed_datum_target_feature
(*
```

Formal propositions:

WR1: The **description** for the **placed_datum_target_feature** shall be either 'point', 'line', 'rectangle' or 'circle'.

WR2: A **placed_datum_target_feature** shall have exactly one implicit representation.

©ISO 2000 - All rights reserved

WR3: Exactly one **representation_item** used for the representation of the **placed_datum_target_-feature** shall be of type **placement** with a **name** of 'orientation'.

WR4: If the **placed_datum_target_feature** is a point, the **representation** shall contain exactly one **representation_item**s in its set of **items**.

WR5: If the **placed_datum_target_feature** is a line or circle, the **representation** shall contain exactly two **representation_item**s in its set of **items**.

WR6: If the **placed_datum_target_feature** is a rectangle, the **representation** shall contain exactly three **representation_item**s in its set of **items**.

WR7: If the **description** of the **placed_datum_target_feature** is 'circle', exactly one **representation_-item** used for the implicit representation of the **placed_datum_target_feature** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'target diameter'.

WR8: If the **description** of the **placed_datum_target_feature** is 'line', exactly one **representation_-item** used for the implicit representation of the **placed_datum_target_feature** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'target length'.

WR9: If the **description** of the **placed_datum_target_feature** is 'rectangle', exactly one **representation_item** used for the implicit representation of the **placed_datum_target_feature** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'target length'.

WR10: If the **description** of the **placed_datum_target_feature** is 'rectangle', exactly one **representation_item** used for the implicit representation of the **placed_datum_target_feature** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'target width'.

5.2.3.1.54 planar_shape_representation

A **planar_shape_representation** is a type of **representation** that represents a direction vector in two or three dimensional space, and location.

EXPRESS specification:

```
*)
ENTITY planar_shape_representation
  SUBTYPE OF (shape_representation);
WHERE
  WR1: SIZEOF (SELF.items) = 1;
  WR2: (SIZEOF (QUERY (it <* SELF.items |
       NOT ('FEATURE_BASED_PROCESS_PLANNING.PLANE'
       IN TYPEOF (it)))) = 1) ;
END_ENTITY;
(*
```

Formal propositions:

WR1: The **planar_shape_representation** shall have exactly one **representation_item** in its set of **items**.

 ©ISO 2000 - All rights reserved

WR2: The geometric element that is used to represent the **planar_shape_representation** shall be a **plane**.

5.2.3.1.55 pocket

A **pocket** is a type of **feature_definition** that is the representation of a volume that is removed from the base shape. This volume may be enclosed and have either a rectangular base shape that is represented by a **rectangular_closed_profile** or it may have a non_rectangular base shape that is represented by either a **ngon_closed_profile**, **rectangular_closed_profile**, **closed_path_profile**, or a **circular_closed_-profile**. If the volume is not enclosed the base shape would be defined by a **square_u_profile**, **partial_-circular_profile**, **round_u_profile**, **vee_profile**, **tee_profile**, or an **open_path_profile**. The bottom of the pocket is represented by a **pocket_bottom**. See ARM definition for General_pocket in paragraph 4.2.87 and Rectangular_pocket in paragraph 4.2.181 for more information.

EXPRESS specification:

```
*)
  ENTITY pocket
    SUBTYPE OF (feature_definition);
    WHERE
  wr1: (SELF\characterized_object.description IN
       ['closed rectangular','open rectangular', 'complex',
       'circular cutout', 'complex cutout', 'recess']);

  wr2: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
       IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       ((sa_occ.description = 'pocket depth occurrence') AND
       (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
       'RELATED_SHAPE_ASPECT') |
       ((sar.description = 'path feature component usage')  AND
       (sar.name = 'pocket depth') AND
       (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
       IN TYPEOF(sar))) ) |
       (('FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT'
       IN TYPEOF(sdr.relating_shape_aspect)) AND
       (sdr.relating_shape_aspect.description ='linear')) ))
       = 1)) )) = 1)) )) = 0);

 WR3: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation))) = 1))) = 0;

    WR4:  SIZEOF( QUERY( pd <* USEDIN( SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
          SIZEOF( QUERY( pdr <* USEDIN( pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
         ('FEATURE_BASED_PROCESS_PLANNING.'+
```

©ISO 2000 - All rights reserved

```
        'SHAPE_REPRESENTATION_WITH_PARAMETERS'
        IN TYPEOF(pdr.used_representation)) AND
       ({1 <= SIZEOF(pdr.used_representation.items) <= 2} ) )) = 1 )) = 1;

  WR5:  SIZEOF( QUERY( pd <* USEDIN( SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        SIZEOF( QUERY( pdr <* USEDIN( pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        ('FEATURE_BASED_PROCESS_PLANNING.'+
        'SHAPE_REPRESENTATION_WITH_PARAMETERS' IN
        TYPEOF(pdr.used_representation)) AND
        (SIZEOF( QUERY( srwp_i <* pdr.used_representation.items |
        (srwp_i.name = 'orientation') OR
        (srwp_i.name = 'fillet radius')))
        = SIZEOF(pdr.used_representation.items)) )) = 1 )) = 1;

  WR6: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
       (SIZEOF (
       ['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF (it)) = 2) AND (it.name = 'fillet radius'))) <= 1)))
       = 0))) = 0;

 wr7: ((NOT (SELF\characterized_object.description IN ['complex',
      'non-circular cutout','recess'])) OR
      (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      ((sa_occ.description = 'boundary occurrence') AND
      (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      ((sar.description = 'profile usage') AND
      (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
      IN TYPEOF(sar))) ) | ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.NGON_CLOSED_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.CLOSED_PATH_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.SQUARE_U_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.ROUNDED_U_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.VEE_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.TEE_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.OPEN_PATH_PROFILE'] *
      TYPEOF(sdr.relating_shape_aspect)) = 1)) )) = 1)) ))
      = 1)) )) = 0));

 wr8: ((NOT (SELF\characterized_object.description =
      'closed rectangular'))
```

 ©ISO 2000 - All rights reserved

```
      OR (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      ((sa_occ.description = 'closed boundary occurrence') AND
      (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      ((sar.description = 'profile usage') AND
      (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
      IN TYPEOF(sar))) ) |
      (('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE'
      IN TYPEOF(sdr.relating_shape_aspect))) )) = 1)) ))
      = 1)) )) = 0));

 wr9: ((NOT (SELF\characterized_object.description = 'open rectangular'))
      OR (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      ((sa_occ.description = 'open boundary occurrence') AND
      (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      ((sar.description = 'profile usage') AND
      (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
      IN TYPEOF(sar))) ) |
      (('FEATURE_BASED_PROCESS_PLANNING.SQUARE_U_PROFILE'
      IN TYPEOF(sdr.relating_shape_aspect))) )) = 1)) ))
      = 1)) )) = 0));

 wr10: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      ((sa_occ.description = 'bottom condition occurrence') AND
      (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      ((sar.description = 'pocket bottom usage') AND
      (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP')
      IN TYPEOF(sar))) ) |
      ('FEATURE_BASED_PROCESS_PLANNING.POCKET_BOTTOM'
      IN TYPEOF(sdr.relating_shape_aspect)) )) = 1)) )) = 1)) )) = 0);

wr11: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      ((sa_occ.description = 'change in boundary occurrence') AND
      (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATING_SHAPE_ASPECT') |
      ((sar.description = 'taper usage') AND
      (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP')
      IN TYPEOF(sar))) ) |
      (('FEATURE_BASED_PROCESS_PLANNING.TAPER'
```

©ISO 2000 - All rights reserved

```
       IN TYPEOF(fcr.related_shape_aspect)) AND
       (fcr.related_shape_aspect.description
       IN ['angle taper','directed taper'])) )) = 1)) ))
       <= 1)) )) = 0);

 WR12: (NOT (SELF\characterized_object.description = 'circular cutout')) OR
       (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
       TYPEOF (pd)) | NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       (sa_occ.description = 'enclosed boundary occurrence') AND
       (SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
       'RELATED_SHAPE_ASPECT') |
       (sar.description = 'profile usage') AND
       ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
       IN TYPEOF (sar))) |
       ('FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE'
       IN TYPEOF (sdr.relating_shape_aspect))))
       = 1))) = 1))) = 0);

 WR13: (NOT (SELF\characterized_object.description IN
       ['circular cutout','complex cutout'])) OR
       (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
       TYPEOF (pd)) | NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       (sa_occ.description = 'bottom condition occurrence') AND
       (SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
       'RELATED_SHAPE_ASPECT') |
       (sar.description = 'pocket bottom usage') AND
       ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
       IN TYPEOF (sar))) |
       ('FEATURE_BASED_PROCESS_PLANNING.POCKET_BOTTOM'
       IN TYPEOF (sdr.relating_shape_aspect)) AND
       (sdr.relating_shape_aspect.description = 'through')))
       = 1))) = 1))) = 0);

 WR14: (NOT (SELF\characterized_object.description = 'recess')) OR
       (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
       TYPEOF (pd)) |
       NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       (sa_occ.description = 'bottom condition occurrence') AND
       (SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
       'RELATED_SHAPE_ASPECT') |
       (sar.description = 'pocket bottom usage') AND
       ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
       IN TYPEOF (sar))) |
       ('FEATURE_BASED_PROCESS_PLANNING.POCKET_BOTTOM'
       IN TYPEOF (sdr.relating_shape_aspect)) AND
       (sdr.relating_shape_aspect.description in ['planar','complex']))
       = 1))) = 1))) = 0);
  END_ENTITY; -- pocket
(*
```

 ©ISO 2000 - All rights reserved

Formal proposition:

WR1: The **pocket** shall have a description of 'open rectangular', 'closed rectangular', 'complex', 'circular cutout', 'complex cutout', or 'recess'.

WR2: The **pocket** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'pocket depth occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'path feature component usage' and a **name** of 'pocket depth' and a **relating_shape_-aspect** that references a **path_feature_component** with a **description** of 'linear'.

WR3: The **pocket** shall have exactly one implicit representation.

WR4: The implicit representation of the **pocket** shall contain between one and two **representation_item**s.

WR5: The implicit representation of the **pocket** shall contain only **representation_item**s that have a name of either 'orientation' or 'fillet radius'.

WR6: The implicit representation of the **pocket** shall have at most one **representation_item** of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'fillet radius'.

WR7: If the **pocket** has a description of 'complex', 'complex cutout', or 'recess', the **pocket** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'boundary occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' and a **relating_shape_aspect** that references a **circular_closed_profile**, **ngon_closed_profile**, **rectangular_closed_profile**, **closed_path_profile**, **square_u_profile**, **partial_circular_profile**, **rounded_u_profile**, **vee_profile**, **tee_profile**, or **open_path_profile**.

WR8: If the **pocket** has a description of 'closed rectangular', the **pocket** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'closed boundary occurrence' that is the **related_shape_-aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' and a **relating_-shape_aspect** that references a **rectangular_closed_profile**.

WR9: If the **pocket** has a description of 'open rectangular', the **pocket** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'open boundary occurrence' that is the **related_shape_-aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' and a **relating_-shape_aspect** that references a **square_u_profile**.

WR10: The **pocket** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'bottom condition occurrence' that is the **related_shape_aspect** in exactly one **feature_component_relationship** with a **description** of 'pocket bottom usage' and a **relating_shape_aspect** that references a **pocket_-bottom**.

WR11: The **pocket** shall be the basis shape for at most one **shape_aspect** with a **description** of 'change in boundary occurrence' that is the **related_shape_aspect** in exactly one **feature_component_-relationship** with a **description** of 'taper usage' in which the **relating_shape_aspect** is a **taper** with a **description** of 'angle taper' or 'directed taper'.

©ISO 2000 - All rights reserved

WR12: If the **pocket** has a description of 'circular cutout', the **pocket** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'enclosed boundary occurrence' that is the **related_-shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' and a **relating_shape_aspect** that references a **circular_closed_profile**.

WR13: If the **pocket** has a description of 'circular cutout',or 'complex cutout', the **pocket** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'bottom condition occurrence' that is the **related_shape_aspect** in exactly one **feature_component_relationship** with a **description** of 'pocket bottom usage' and a **relating_shape_aspect** that references a **pocket_bottom** with a **description** of 'through'.

WR14: If the **pocket** has a description of 'recess' the **pocket** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'bottom condition occurrence' that is the **related_shape_aspect** in exactly one **feature_component_relationship** with a **description** of 'pocket bottom usage' and a **relating_shape_aspect** that references a **pocket_bottom** with a **description** of 'planar' or 'complex'.

Informal propositions:

IP1: The volume removal of the **pocket** shall be in the Z direction.

IP2: The placement of the **circular_closed_profile**, **ngon_closed_profile**, **rectangular_closed_profile**, **closed_path_profile**, **square_u_profile**, **partial_circular_profile**, **rounded_u_profile**, **vee_profile**, **tee_profile**, or **open_path_profile** that defines the shape of the **pocket** shall be with the origin, X direction, and Y direction equal to that of the **pocket**.

IP3: If the **description** of the **pocket** is 'open rectangular', the origin, X direction, Y direction of the **square_u_profile** shall be equal to that of the **pocket**.

5.2.3.1.56 pocket_bottom

A **pocket_bottom** is a type of **shape_aspect** that is the representation of the end condition for a **pocket feature_definition**. See ARM definition for Pocket_bottom_condition in paragraph 4.2.161 for more information.

EXPRESS specification:

```
*)
ENTITY pocket_bottom
  SUBTYPE OF (shape_aspect);
WHERE
  WR1: 'FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
       IN TYPEOF (SELF.of_shape.definition);
  WR2: SELF.description IN ['planar', 'complex', 'through'];

  WR3 : ((NOT (SELF.description = 'planar')) OR
        (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION')
        IN TYPEOF(pdr.used_representation))  AND
```

 ©ISO 2000 - All rights reserved

```
      (pdr.used_representation.name = 'floor normal') )) = 1)) )) = 0));

WR4 : ((NOT (SELF.description = 'planar')) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.LOCATION_SHAPE_REPRESENTATION')
      IN TYPEOF(pdr.used_representation))  AND
      (pdr.used_representation.name = 'floor location'))) = 1)) )) = 0));

 WR5 : ((NOT (SELF.description = 'complex')) OR
     (SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     ('FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
     IN TYPEOF(pdr.used_representation)) AND
     (pdr.used_representation.name = 'floor face') )) = 1)) )) = 0));

WR6: (NOT (SELF.description IN  ['planar', 'complex']) OR
     (SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation))) = 1))) = 0));

WR7: (NOT (SELF.description IN  ['planar', 'complex']) OR
     (SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation)) |
     NOT (SIZEOF (impl_rep.used_representation.items)
     <= 1) )) = 0))) = 0));

WR8: (NOT (SELF.description = 'through') OR
     (SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation))) = 0))) = 0));

WR9: (NOT (SELF.description IN ['planar', 'complex'])) OR
     (SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     'FEATURE_BASED_PROCESS_PLANNING.' +
```

©ISO 2000 - All rights reserved

```
        'SHAPE_REPRESENTATION_WITH_PARAMETERS'
        IN TYPEOF (pdr.used_representation)) |
        NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
        (SIZEOF
        (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
        TYPEOF (it)) = 2) AND (it.name = 'radius'))) <= 1))) = 0))) = 0);

  WR10: SIZEOF (QUERY (fcr <* QUERY (sar <* USEDIN (SELF,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
        'RELATING_SHAPE_ASPECT') |
        (sar.description = 'pocket bottom usage') AND
        ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
        IN TYPEOF (sar))) |
      (fcr.related_shape_aspect.description = 'bottom condition occurrence')
        AND
        ('FEATURE_BASED_PROCESS_PLANNING.POCKET'
        IN TYPEOF(fcr.related_shape_aspect.of_shape.definition)))) >= 1;

 WR11: ((NOT (SELF.description IN ['planar','complex'])) OR
         (SIZEOF (QUERY (fcr <* QUERY (sar <* USEDIN (SELF,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
        'RELATING_SHAPE_ASPECT') |
        (sar.description = 'pocket bottom usage') AND
        ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
        IN TYPEOF (sar))) |
      (fcr.related_shape_aspect.description = 'bottom condition occurrence')
       AND
      (fcr.related_shape_aspect.name IN
       ['pocket depth start', 'pocket depth end'])
       )) = 0));
END_ENTITY;  -- Pocket_bottom
(*
```

Formal propositions:

WR1: The **pocket_bottom** shall be an aspect of the shape of a **feature_component_definition**.

WR2: The **description** of the **pocket_bottom** shall be either 'planar', 'complex', or 'through'.

WR3: If the **description** of the **pocket_bottom** is 'planar', the **pocket_bottom** shall have exactly one **direction_shape_representation** with **name** = 'floor normal'.

WR4: If the **description** of the **pocket_bottom** is 'planar', the **pocket_bottom** shall have exactly one **location_shape_representation** with **name** = 'floor location'.

WR5: If the **description** of the **pocket_bottom** is 'complex', the **pocket_bottom** shall have exactly one **face_shape_representation** with **name** = 'floor'.

WR6: If the **description** of the **pocket_bottom** is 'planar' or 'complex', the **pocket_bottom** shall have its implicit representation specified by exactly one **shape_representation_with_parameters**.

WR7: If the **description** of the **pocket_bottom** is 'planar' or 'complex', the **pocket_bottom** shall be represented implicitly by at most one **representation_item**s.

 ©ISO 2000 - All rights reserved

WR8: If the **description** of the **pocket_bottom** is 'through', the **pocket_bottom** shall have its implicit representation specified by exactly one **shape_representation_with_parameters**.

WR9: If the **description** of the **pocket_bottom** is 'planar' or 'complex', at most one **representation_-item** used for the implicit representation of the **pocket_bottom** shall be of type **measure_-representation_item** and **length_measure_with_unit** with a **name** of 'radius'.

WR10: The **pocket_bottom** shall be the **relating_shape_aspect** in at least one **feature_component_-relationship** with a **description** of 'pocket bottom usage' in which the **related_shape_aspect** is an aspect of the shape of a **pocket** with a **description** of 'bottom condition occurrence'.

WR11: If the **description** of the **pocket_bottom** is 'planar' or 'complex',the **pocket_bottom** shall be the **relating_shape_aspect** in at least one **feature_component_relationship** with a **description** of 'pocket bottom usage' and with a **name** of 'pocket depth start', or 'pocket depth end.

Informal propositions:

IP1: The location of the **pocket_bottom** shall be at the approximate center of the mating **pocket** feature.

IP2: The **pocket_bottom** shall be defined with the volume removal in Z direction of the **pocket** feature.

5.2.3.1.57 profile_floor

A **profile_floor** is a type of **shape_aspect** that is the representation of the bottom condition for an **outside_profile feature_definition**. See ARM definition for Profile_floor in paragraph ??? for more information.

EXPRESS specification:

```
*)
ENTITY profile_floor
  SUBTYPE OF (shape_aspect);
WHERE
  WR1: 'FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
       IN TYPEOF (SELF.of_shape.definition);

  WR2: SELF.description IN ['planar', 'complex', 'through'];

  WR3: (NOT (SELF.description IN ['planar', 'complex']) OR
       (SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation))) = 1))) = 0));

  WR4: (NOT (SELF.description IN  ['planar', 'complex'])) OR
       (SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
```

©ISO 2000 - All rights reserved

```
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS'
      IN TYPEOF (pdr.used_representation)) |
      NOT (SIZEOF (impl_rep.used_representation.items) >= 1)AND
     (SIZEOF (impl_rep.used_representation.items) <= 2))) = 0))) = 0);

  WR5: (NOT (SELF.description = 'through') OR
      (SIZEOF (QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS'
      IN TYPEOF (pdr.used_representation))) = 0))) = 0));

  WR6: (NOT (SELF.description IN ['planar', 'complex'])) OR
      (SIZEOF (QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS'
      IN TYPEOF (pdr.used_representation)) |
      NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
    (SIZEOF (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF (it)) = 2) AND (it.name = 'radius'))) = 1))) = 0))) = 0);

  WR7: SIZEOF (QUERY (fcr <* QUERY (sar <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATING_SHAPE_ASPECT') |
      (sar.description = 'profile floor usage') AND
      ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP' IN
      TYPEOF (sar))) |
      'FEATURE_BASED_PROCESS_PLANNING.OUTSIDE_PROFILE' IN TYPEOF
      (fcr.related_shape_aspect.of_shape.definition))) >= 1;

  WR8: ((NOT (SELF.description IN ['planar','complex'])) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
      | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
      used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
      impl_rep.used_representation.items |
      (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
      IN TYPEOF(it)) AND (it.name = 'shape profile floor orientation')
      AND (it.description IN ['shape profile start','shape profile end']))
      )) =  1)) )) = 0)) )) = 0));

 WR9: (NOT (SELF.description = 'complex')) OR
      (SIZEOF (QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
```

 ©ISO 2000 - All rights reserved

```
       ('FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
       IN TYPEOF (pdr.used_representation))  AND
       (pdr.used_representation.name='floor'))) = 1))) = 1);

 WR10: (NOT (SELF.description = 'planar')) OR
       (SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
       IN TYPEOF (pdr.used_representation))  AND
       (pdr.used_representation.name='floor'))) = 1))) = 1);

END_ENTITY;
(*
```

Formal propositions:

WR1: The **profile_floor** shall be an aspect of the shape of a **feature_component_definition**.

WR2: The **description** of the **profile_floor** shall be either 'planar', 'complex', or 'through'.

WR3: If the description of the profile_floor is 'planar' or 'complex', the **profile_floor** shall have its implicit representation specified by exactly one **shape_representation_with_parameters**.

WR4: If the **description** of the **profile_floor** is 'planar' or 'complex', the **profile_floor** shall be represented implicitly by at least one and at most two **representation_item**s.

WR5: If the **description** of the **profile_floor** is 'through', the **profile_floor** the profile_floor shall have its implicit representation specified by exactly zero shape_representation_with_parameters.

WR6: If the **description** of the **profile_floor** is 'planar' or 'complex', at most one **representation_item** used for the implicit representation of the **profile_floor** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'radius'.

WR7: The **profile_floor** shall be the **relating_shape_aspect** in at least one **feature_component_-relationship** with a **description** of 'profile floor usage' in which the **related_shape_aspect** is an aspect of the shape of a **outside_profile**.

WR8: If the **description** of the **profile_floor** is 'planar' or 'complex', the implicit representation of the **profile_floor** shall have exactly one **representation_item** in its **items** set which is a **descriptive_-representation_item** with a **name** of 'shape profile floor orientation' and a **description** of either 'shape profile start', or 'shape profile end'.

WR9: If the **description** of the **profile_floor** is 'complex', the **profile_floor** shall have exactly one **face_shape_representation** with name of 'floor'.

WR10: If the **description** of the **profile_floor** is 'planar', the **profile_floor** shall have exactly one **planar_shape_representation** with name of 'floor'.

©ISO 2000 - All rights reserved

5.2.3.1.58 protrusion

A **protrusion** is a type of **feature_definition** that is the representation of a volume that is extruding from the base shape. This volume shall be enclosed, except for the mating surface which may or may not be included in the definition of the volume, and have a shape defining representation. See ARM definition for Protrusion in paragraph 4.2.171 for more information.

EXPRESS specification:

```
*)
  ENTITY protrusion
    SUBTYPE OF (feature_definition);
    WHERE

  wr1: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(impl_rep.used_representation.items) = 1)) ))
       = 0)) )) = 0);

  wr2: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
       IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       ((sa_occ.description = 'shape volume occurrence') AND
       (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
       'RELATED_SHAPE_ASPECT') |
       ((sar.description = 'volume shape usage') AND
       (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
       IN TYPEOF(sar))) ) |
       (sdr.relating_shape_aspect.description = 'volume shape') )) = 1)) ))
       = 1)) )) = 0);
  END_ENTITY; -- protrusion
(*
```

Formal proposition:

WR1: The **protrusion** shall have an implicit representation that contains exactly one **representation_-item** in its set of **items**.

WR2: The **protrusion** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'shape volume occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'volume shape usage' and **relating_shape_aspect** with a **description** of 'volume shape'.

 ©ISO 2000 - All rights reserved

Informal propositions:

IP1: The representation of the **shape_aspect** that is the **relating_shape_aspect** in the **shape_defining_-relationship** shall define a volume of the **protrusion**.

IP2: The location of the **protrusion** shall be at a position on the base of the feature where it mates with the part.

IP3: The Z direction of the **protrusion** shall be with in a direction away from the part.

5.2.3.1.59 rectangular_closed_profile

A **rectangular_closed_profile** is a type of **shape_aspect** that represents an enclosed 2D area with location and orientation. The enclosed area is defined by four straight sides with opposite sides equal in length. The location is defined to be at the center of the rectangle, and the orientation of the **rectangular_closed_profile** is the X-Y plane with the X direction along the length of the rectangle and the Y direction orthogonal to the X direction in the direction of the width. See ARM definition for Rectangular_closed_profile in paragraph 4.2.177 for more information.

EXPRESS specification:

```
*)
ENTITY rectangular_closed_profile
  SUBTYPE OF (shape_aspect);
WHERE
  WR1: 'FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
       IN TYPEOF (SELF.of_shape.definition);

  WR2: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation))) = 1))) = 0;

  WR3: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT ((SIZEOF (impl_rep.used_representation.items) >= 3)
       AND (SIZEOF(impl_rep.used_representation.items) <= 4)))) = 0))) = 0;

  WR4: SIZEOF( QUERY( pd <* USEDIN( SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      SIZEOF( QUERY( pdr <* USEDIN( pd, 'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
      IN TYPEOF(pdr.used_representation)) AND
      (SIZEOF( QUERY( srwp_i <* pdr.used_representation.items |
```

©ISO 2000 - All rights reserved

```
     (srwp_i.name = 'orientation') OR
     (srwp_i.name = 'length') OR
     (srwp_i.name = 'width') OR
     (srwp_i.name = 'corner radius') )) =
     SIZEOF(pdr.used_representation.items))
      )) = 1 )) = 1;

  WR5: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
       ('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF (it)) AND
       (it.name = 'orientation'))) = 1))) = 0))) = 0;

  WR6: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
       (SIZEOF
       (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF (it)) = 2) AND
       (it.name = 'width'))) = 1))) = 0))) = 0;

  WR7: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
       (SIZEOF
       (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF (it)) = 2) AND
       (it.name = 'length'))) = 1))) = 0))) = 0;

  WR8: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
       (SIZEOF
       (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
```

 ©ISO 2000 - All rights reserved

```
        TYPEOF (it)) = 2) AND
        (it.name = 'corner radius'))) <= 1))) = 0))) = 0;
END_ENTITY;
(*
```

Formal propositions:

WR1: The **rectangular_closed_profile** shall be an aspect of the shape of a **feature_component_-definition**.

WR2: The **rectangular_closed_profile** shall have exactly one implicit representation defined by a relationship to a **shape_representation_with_parameters**.

WR3: The **shape_representation_with_parameters** that represents the **rectangular_closed_profile** shall contain at least three and at most four **representation_item**s in its set of **items**.

WR4: The implicit representation of a **rectangular_closed_profile** shall contain only **representation_-item**s in its set of **items** with a **name** of 'orientation', 'length', 'width', and 'corner radius'

WR5: Exactly one **representation_item** used for the implicit representation of a **rectangular_closed_-profile** shall be of type **placement** with a **name** of 'orientation'.

WR6: Exactly one **representation_item** used for the implicit representation of a **rectangular_closed_-profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'width'.

WR7: Exactly one **representation_item** used for the implicit representation of a **rectangular_closed_-profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'length'.

WR8: At most one **representation_item** used for the implicit representation of a **rectangular_closed_-profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'corner radius'.

Informal propositions:

IP1: The location of the **rectangular_closed_profile** shall be defined at the center of the rectangle.

IP2: The **rectangular_closed_profile** shall be defined in the X-Y plane with the length in the X direction and the width in the Y direction of the rectangle. The Xdirection is orthogonal to the Y direction.

5.2.3.1.60 rectangular_pattern

A **rectangular_pattern** is a type of **replicate_feature** that relates a base feature and one or more **shape_aspect**s which are the placement of the base feature at a specified row and column location on the base part. A position is defined on the part and a number of base features are equally place in rows and columns. See ARM definition for Rectangular_pattern in paragraph 4.2.183 for more information.

©ISO 2000 - All rights reserved

EXPRESS specification:

```
*)
  ENTITY rectangular_pattern
    SUBTYPE OF (replicate_feature);
    WHERE
   WR1: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
        IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN
        (pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
        ((SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
        + 'RELATING_SHAPE_ASPECT') |
        ((('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP')
         IN TYPEOF(sar))) ) |
        ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT'
        IN TYPEOF(sdr.related_shape_aspect)) )) = 1)) )) <= 5)) )) = 0);

  WR2 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
        IN TYPEOF(pdr.used_representation)) AND
       (pdr.used_representation.name = 'row layout direction') ) ))
       = 1)) ))  = 0);

  WR3 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
        IN TYPEOF(pdr.used_representation)) AND
        (pdr.used_representation.name = 'column layout direction')) ))
        = 1)) )) = 0);

  WR4 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
        'DEFINITION') | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        ('FEATURE_BASED_PROCESS_PLANNING.'+
        'SHAPE_REPRESENTATION_WITH_PARAMETERS'
        IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0);

  WR5 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        ('FEATURE_BASED_PROCESS_PLANNING.'+
        'SHAPE_REPRESENTATION_WITH_PARAMETERS'
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(impl_rep.used_representation.items) = 5)) ))
        = 0)) )) = 0);
```

 ©ISO 2000 - All rights reserved

```
WR6 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.'+
      'SHAPE_REPRESENTATION_WITH_PARAMETERS'
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
      IN TYPEOF(it)) AND
      ('FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE'
      IN TYPEOF(it\measure_with_unit.value_component)) AND
      (it.name = 'number of rows')) )) = 1)) )) = 0)) )) = 0);

WR7 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.'+
      'SHAPE_REPRESENTATION_WITH_PARAMETERS'
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
      IN TYPEOF(it)) AND
      ('FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE'
      IN TYPEOF(it\measure_with_unit.value_component)) AND
      (it.name = 'number of columns')) )) = 1)) )) = 0)) )) = 0);

WR8 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN( pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.'+
      'SHAPE_REPRESENTATION_WITH_PARAMETERS'
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF(it)) = 2) AND (it.name = 'row spacing')) )) = 1)) ))
      = 0)) )) = 0);

WR9: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.'+
      'SHAPE_REPRESENTATION_WITH_PARAMETERS'
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF(it)) = 2) AND (it.name = 'column spacing'))))= 1)) ))
      = 0)) )) = 0);
```

©ISO 2000 - All rights reserved

```
  WR10: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        ('FEATURE_BASED_PROCESS_PLANNING.'+
        'SHAPE_REPRESENTATION_WITH_PARAMETERS'
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
        (('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it)) AND
        (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
  END_ENTITY; -- rectangular_pattern
(*
```

Formal propositions:

WR1: The **rectangular_pattern** shall be the **relating_shape_aspect** in no more than five **shape_-aspect_relationship**s.

WR2: The **rectangular_pattern** shall have exactly one **direction_shape_representation** with a **name** of 'row layout direction'.

WR3: The **rectangular_pattern** shall have exactly one **direction_shape_representation** with a **name** of 'column layout direction'.

WR4: The **rectangular_pattern** shall have exactly one implicit representation.

WR5: The implicit representation of the **rectangular_pattern** shall contain five **representation_item**s in its set of **items**.

WR6: The implicit representation of the **rectangular_pattern** shall contain exactly one **representation_item** of type **measure_representation_item** with a **value_component** of type **count_-measure** and a **name** of 'number of rows'.

WR7: The implicit representation of the **rectangular_pattern** shall contain exactly one **representation_item** of type **measure_representation_item** with a **value_component** of type **count_-measure** and a **name** of 'number of columns'.

WR8: The implicit representation of the **rectangular_pattern** shall contain exactly one **representation_item** of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'row spacing'.

WR9: The implicit representation of the **rectangular_pattern** shall contain exactly one **representation_item** of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'column spacing'.

WR10: The implicit representation of the **rectangular_pattern** shall contain exactly one **representation_item** of type **placement** with a **name** of 'orientation'.

 ©ISO 2000 - All rights reserved

Informal propositions:

IP1: The location of the **rectangular_pattern** shall be defined at the position of the first base feature.

IP2: The **rectangular_pattern** shall be defined in the X-Y plane.

5.2.3.1.61 removal_volume

A **removal_volume** is a type of **feature_definition** that is the representation of a volume that is removed from the base shape. This volume may or may not be enclosed but in either case shall have a shape defining representation. See ARM definition for General_removal_volume in paragraph 4.2.90 for more information.

EXPRESS specification:

```
*)
  ENTITY removal_volume
    SUBTYPE OF (feature_definition);
    WHERE

  wr1: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(impl_rep.used_representation.items) = 1)) ))
       = 0)) )) = 0);

  wr2: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
       IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       ((sa_occ.description = 'shape volume occurrence') AND
       (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
       + 'RELATED_SHAPE_ASPECT') |
       ((sar.description = 'volume shape usage') AND
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_DEFINING_RELATIONSHIP')
       IN TYPEOF(sar))) ) |
       (sdr.relating_shape_aspect.description = 'volume shape') )) = 1)) ))
       = 1)) )) = 0);
  END_ENTITY; -- removal_volume
(*
```

Formal proposition:

WR1: The **removal_volume** shall have an implicit representation that contains exactly one **representation_item** in its set of **items**.

©ISO 2000 - All rights reserved

WR2: The **removal_volume** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'shape volume occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_-relationship** with a **description** of 'volume shape usage' and a **relating_shape_aspect** with a **description** of 'volume shape'.

Informal propositions:

IP1: The location of the **removal_volume** shall be defined at the bottom and approximate center of the removal_volume.

IP2: The Z direction of the **removal_volume** shall be in the direction of volume removal.

5.2.3.1.62 replicate_feature

A **replicate_feature** is a type of **shape_aspect** that is either a **feature_definition** or another **replicate_-feature** for the purpose of multiple representations.

EXPRESS specification:

```
*)
  ENTITY replicate_feature
    SUPERTYPE OF (ONEOF (circular_pattern, rectangular_pattern,
      feature_pattern))
    SUBTYPE OF (feature_definition);
  WHERE

      wr1: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
               used_representation)) ))  = 1)) )) = 0);

   wr2: (SIZEOF(QUERY ( fcr <* QUERY ( sar <*
        USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
        + 'RELATING_SHAPE_ASPECT') |
        ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
        IN TYPEOF(sar)) ) |
        ((SIZEOF(['FEATURE_BASED_PROCESS_PLANNING.REPLICATE_FEATURE',
        'FEATURE_BASED_PROCESS_PLANNING.INSTANCED_FEATURE'] *
        TYPEOF(fcr.related_shape_aspect)) >= 1) AND
        (fcr.name = 'pattern basis')) )) = 1);

   wr3: ((SIZEOF(QUERY ( sar <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
        + 'RELATING_SHAPE_ASPECT') |
        (NOT
        ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
        IN TYPEOF(sar))) )) +
        SIZEOF(QUERY ( sar <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
        + 'RELATED_SHAPE_ASPECT') |
        (NOT
```

 ©ISO 2000 - All rights reserved

```
        ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
        IN TYPEOF(sar))) ))) = 0);
END_ENTITY;  -- replicate_feature
(*
```

Formal proposition:

WR1: The **replicate_feature** shall have an implicit representation that contains exactly one **representation_item** in its set of **items**.

WR2: The **replicate_feature** shall be the **relating_shape_aspect** in exactly one **shape_aspect_-relationship** that is of type **feature_component_relationship** with a name of 'pattern basis' and a **related_shape_aspect** that is either of type **replicate_feature** or **instanced_feature**.

WR3: A **replicate_feature** shall be referenced by **shape_aspect_relationship**s only if they are of type **feature_component_relationship** or **shape_defining_relationship**.

5.2.3.1.63 revolved_profile

A **revolved_profile** is a type of **feature_definition** that is the representation of a volume that is removed from the base shape. The volume of removal is defined by either sweeping a linear_profile about an axis (see ARM definition for Revolved_flat in 4.2.189) or a partial_circular_profile about an axis (see ARM definition of Revolved_round in 4.2.190) or an open_path_profile about an axis (see ARM definition of General_revolution in 4.2.91) or open_profile about an axis (see ARM definition of Groove in 4.2.97).

EXPRESS specification:

```
*)
  ENTITY revolved_profile
    SUBTYPE OF (feature_definition);
    WHERE
  wr1: (SELF\characterized_object.description IN
       ['groove','flat','round','open profile']);

  wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(impl_rep.used_representation.items) = 2)) ))
       = 0)) )) = 0);

  WR3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       ((SIZEOF([
```

©ISO 2000 - All rights reserved

```
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF(it)) = 2) AND (it.name = 'radius')))) = 1)) ))
      = 0)) )) = 0);

  wr4: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
      IN TYPEOF(pdr.used_representation)) AND
      (pdr.used_representation.name = 'removal direction')) )) = 1)) ))
      = 0);

  wr5: ((NOT (SELF\characterized_object.description = 'open profile')) OR
      (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      ((sa_occ.description = 'outer edge shape occurrence') AND
      (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      ((sar.description = 'profile usage') AND
      (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
      IN TYPEOF(sar))) ) |
      (('FEATURE_BASED_PROCESS_PLANNING.OPEN_PATH_PROFILE'
      IN TYPEOF(sdr.relating_shape_aspect)) AND
      (sdr.relating_shape_aspect.description = 'outer edge shape')) ))
      = 1)) )) = 1)) )) = 0));

  wr6: (NOT (SELF\characterized_object.description = 'flat')) OR
      (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')|
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      (( sa_occ.description = 'flat edge shape occurrence') AND
      (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      ((sar.description = 'profile usage') AND
      (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
      IN TYPEOF(sar))) ) |
      (('FEATURE_BASED_PROCESS_PLANNING.LINEAR_PROFILE'
      IN TYPEOF(sdr.relating_shape_aspect)) AND
      (sdr.relating_shape_aspect.description = 'flat edge shape')) ))
      = 1)) )) = 1)) )) = 0);

  wr7: ((NOT (SELF\characterized_object.description = 'round')) OR
      (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      ((sa_occ.description = 'rounded edge shape occurrence') AND
      (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
```

 ©ISO 2000 - All rights reserved

```
      ((sar.description = 'profile usage') AND
      (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
      IN TYPEOF(sar))) ) |
      (('FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE'
      IN TYPEOF(sdr.relating_shape_aspect)) AND
      (sdr.relating_shape_aspect.description = 'rounded edge shape')) ))
      = 1)) )) = 1)) )) = 0));

  wr8: ((NOT (SELF\characterized_object.description = 'groove')) OR
      (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      ((sa_occ.description = 'sweep occurrence') AND
      (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      ((sar.description = 'profile usage') AND
      (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
      IN TYPEOF(sar))) ) |
      ((SIZEOF(['FEATURE_BASED_PROCESS_PLANNING.SQUARE_U_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.ROUNDED_U_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.VEE_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.TEE_PROFILE',
      'FEATURE_BASED_PROCESS_PLANNING.OPEN_PATH_PROFILE'] *
      TYPEOF(sdr.relating_shape_aspect)) = 1) AND
      (sdr.relating_shape_aspect.description = 'sweep')) )) = 1)) ))
      = 1)) )) = 0));
  END_ENTITY; -- revolved_profile

(*
```

Formal propositions:

WR1: The **revolved_profile** shall have a description of 'groove', 'flat', 'round', or 'open profile'.

WR2: The **revolved_profile** shall have an implicit representation that contains exactly two **representation_item** in its set of **items**.

WR3: The **revolved_profile** shall contain exactly one **representation_item** in its set of **items** which is a **measure_representation_item** and **length_measure_with_unit** with a **name** of 'radius'.

WR4: The **revolved_profile** shall have exactly one **direction_shape_representation** with a **name** of 'removal direction'.

WR4: If the **revolved_profile** has a description of 'open profile', the **revolved_profile** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'outer edge shape occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' and a **relating_shape_aspect** that references an **open_path_profile** with a **description** of 'outer edge shape'.

©ISO 2000 - All rights reserved

WR5: If the **revolved_profile** has a description of 'flat', the **revolved_profile** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'flat edge occurrence' that is the **related_shape_-aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' and a **relating_-shape_aspect** that references a **linear_profile** with a **description** of 'flat edge shape'.

WR6: If the **revolved_profile** has a description of 'round', the **revolved_profile** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'rounded edge shape occurrence' that is the **related_-shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' and a **relating_shape_aspect** that references an **partial_circular_profile** with a **description** of 'rounded edge shape'.

WR7: If the **revolved_profile** has a description of 'groove', the **revolved_profile** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'sweep occurrence' that is the **related_shape_-aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' and a **relating_-shape_aspect** that references a **square_u_profile**, **partial_circular_profile**, **rounded_u_profile**, **vee_-profile**, **tee_profile**, or **open_path_profile** with a **description** of 'sweep'.

Informal proposition:

IP1: If the orientation of the **revolved_profile** causes the feature to be self intersecting, the intersecting edges shall be trimmed at the axis of revolution.

IP2: If the description of the **revolved_profile** is 'open profile', then the placement of the **open_path_-profile** that defines the shape of the **revolved_profile** shall be with the origin, x direction, and y direction equal to that of the **revolved_profile**.

IP3: If the description of the **revolved_profile** is 'groove', then the placement of the **linear_profile**, **square_u_profile**, **partial_circular_profile**, **rounded_u_profile**, **vee_profile**, **tee_profile**, or **open_-path_profile** that defines the shape of the **revolved_profile** shall be placed a specified distance away from the origin of the **revolved_profile** along the X direction.

NOTE - The orientation of the **linear_profile**, **square_u_profile**, **partial_circular_profile**, **rounded_u_profile**, **vee_profile**, **tee_profile**, or **open_path_profile** is independent of the origin of the **revolved_profile**. The **revolved_profile** may be defined on different faces of a part dependent upon the orientation of the profile.

IP4: If the description of the **revolved_profile** is 'flat', then the placement of the **linear_profile** that defines the shape of the **revolved_profile** shall be placed a specified distance away from the origin of the **revolved_profile** along the X direction. The Y direction of the **linear_profile** shall be equal to that of the **revolved_profile**. The X direction and Z direction of the **linear_profile** shall be independent from that of the **revolved_profile**.

IP5: If the description of the **revolved_profile** is 'round', then the placement of the **partial_circular_-profile** that defines the shape of the **revolved_profile** shall be placed a specified distance away from the origin of the **revolved_profile** along the X direction. The Z direction of the **partial_circular_profile** shall be equal to the Y direction of the **revolved_profile**. The X direction and Y direction of the **partial_circular_profile** shall be independent from that of the **revolved_profile**.

 ©ISO 2000 - All rights reserved

5.2.3.1.64 rib_top

A **rib_top** is a type of **feature_definition** that is the representation of a volume that is removed from the top of a shape. The floor of the **rib_top** is represented by a **rib_top_floor**. See ARM definition for Rib_top in paragraph 4.2.191.

EXPRESS specification:

```
*)
ENTITY rib_top
   SUBTYPE OF (feature_definition);
  WHERE

  WR1: SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
       IN TYPEOF (pd)) |
       NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       (sa_occ.description = 'rib top condition occurrence') AND
       (SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
       'RELATED_SHAPE_ASPECT') |
       (sar.description = 'rib top usage') AND
       ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
       IN TYPEOF (sar))) |
       'FEATURE_BASED_PROCESS_PLANNING.RIB_TOP_FLOOR'
       IN TYPEOF (sdr.relating_shape_aspect))) = 1))) = 1))) = 0;

  WR2: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
       IN TYPEOF (pdr.used_representation))AND
       (pdr.used_representation.name = 'removal direction'))) = 1))) = 1;

END_ENTITY;
(*
```

Formal propositions:

WR1: The **rib_top** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'rib top condition occurrence' that is the **related_shape_aspect** in exactly one **feature_component_relationship** with a **description** of 'rib top usage' and a **relating_shape_aspect** that references a **rib_top_floor**.

WR2: The **rib_top** shall have exactly one **direction_shape_representation** with a **name** of 'removal direction'.

5.2.3.1.65 rib_top_floor

A **rib_top_floor** is a type of **shape_aspect** that is the representation of the flor condition for a **rib_top feature_definition**. See ARM definition for Rib_top_floor in paragraph ??? for more information.

©ISO 2000 - All rights reserved

EXPRESS specification:

```
*)
ENTITY rib_top_floor
  SUBTYPE OF (shape_aspect);
WHERE
  WR1: 'FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION' IN
       TYPEOF (SELF.of_shape.definition);

  WR2: SELF.description IN ['planar', 'complex'];

  WR3: SIZEOF (QUERY (fcr <* QUERY (sar <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
       'RELATING_SHAPE_ASPECT') |
       (sar.description = 'rib top usage') AND
       ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
       IN TYPEOF (sar))) |
     (fcr.related_shape_aspect.description = 'rib top condition occurrence')
       AND
       ('FEATURE_BASED_PROCESS_PLANNING.RIB_TOP' IN TYPEOF
       (fcr.related_shape_aspect.of_shape.definition)))) >= 1;

  WR4: (NOT (SELF.description = 'complex')) OR
       (SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
       IN TYPEOF (pdr.used_representation))AND
       (pdr.used_representation.name = 'rib top face'))) = 1))) = 0);

  WR5: (NOT (SELF.description = 'planar')) OR
       (SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
       IN TYPEOF (pdr.used_representation))AND
       (pdr.used_representation.name = 'rib top face'))) = 1))) = 0);

  WR6: (NOT (SELF.description = 'planar')) OR
       (SIZEOF (QUERY (pds <* QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
       IN TYPEOF (pd)) |
       NOT (SIZEOF (QUERY (sa_occ <* USEDIN (pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       (sa_occ.description = 'boundary occurrence') AND
       (SIZEOF (QUERY (sdr <* QUERY (sar <* USEDIN (sa_occ,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
       'RELATED_SHAPE_ASPECT') |
       (sar.description = 'profile usage') AND
       ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
       IN TYPEOF (sar))) |
       (SIZEOF (['FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.NGON_CLOSED_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.CLOSED_PATH_PROFILE'] *
```

 ©ISO 2000 - All rights reserved

```
        TYPEOF (sdr.relating_shape_aspect)) = 1) AND
        (sdr.relating_shape_aspect.description = 'rib top floor boundary')))
        = 1))) = 1))) = 0);

END_ENTITY;  -- rib_top_floor
(*
```

Formal propositions:

WR1: The **rib_top_floor** shall be an aspect of the shape of a **feature_component_definition**.

WR2: The **description** of the **rib_top_floor** shall be either 'planar' or 'complex'.

WR3: The **rib_top_floor** shall be the **relating_shape_aspect** in at least one **feature_component_-relationship** with a **description** of 'rib top usage' in which the **related_shape_aspect** is an aspect of the shape of a **rib_top** with a **description** of 'rib top condition occurrence'.

WR4: If the **description** of the **rib_top_floor** is 'complex', the **rib_top_floor** shall have exactly one **face_shape_representation** with **name** of 'rib top face'.

WR5: If the **description** of the **rib_top_floor** is 'planar', the **rib_top_floor** shall have exactly one **planar_shape_representation** with **name** of 'rib top face'.

WR6: If the **rib_top** has a **description** of 'planar', the **boss** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'boundary occurrence' that is the **related_shape_aspect** in exactly one **feature_component_relationship** with a **description** of 'profile usage' in which the **relating_-shape_aspect** is either a **circular_closed_profile**, **rectangular_closed_profile**, **ngon_closed_profile**, or **closed_path_profile**.

Informal propositions:

IP1: The location of the **rib_top_floor** shall be at the approximate center of the mating **rib_top** feature.

IP2: The **rib_top_floor** shall be defined with the volume removal in Z direction of the **rib_top** feature.

5.2.3.1.66 round_hole

A **round_hole** is a type of **feature_definition** that is the representation of a circular enclosed volume this is removed from the base shape. The bottom of the **round_hole** is represented by a **hole_bottom**. A **round_hole** may have a change in diameter which is represented by a **taper**. See ARM definition for Round_hole in paragraph 4.2.193 for more information.

EXPRESS specification:

```
*)
  ENTITY round_hole
    SUBTYPE OF (feature_definition);
    WHERE
  wr1: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
```

©ISO 2000 - All rights reserved

```
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')|
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      ((sa_occ.description = 'diameter occurrence') AND
      (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      ((sar.description = 'profile usage') AND
      (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
      IN TYPEOF(sar))) ) |
      (('FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE'
      IN TYPEOF(sdr.relating_shape_aspect)) AND
      (sdr.name = 'diameter'))  )) = 1)) )) = 1)) )) = 0);

  wr2: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      ((sa_occ.description = 'hole depth occurrence') AND
      (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      ((sar.description = 'path feature component usage') AND
      (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
      IN TYPEOF(sar))) ) |
      (('FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT'
      IN TYPEOF(sdr.relating_shape_aspect)) AND
      (sdr.name = 'hole depth') AND
      (sdr.relating_shape_aspect.description = 'linear')) )) = 1)) ))
      = 1)) )) = 0);

  wr3: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      ((sa_occ.description = 'bottom condition occurrence') AND
      (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      ((sar.description = 'hole bottom usage') AND
      (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP')
      IN TYPEOF(sar))) ) |
      ('FEATURE_BASED_PROCESS_PLANNING.HOLE_BOTTOM'
      IN TYPEOF(fcr.relating_shape_aspect)) )) = 1)) )) = 1)) )) = 0);

  wr4: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      ((sa_occ.description = 'change in diameter occurrence') AND
      (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      ((sar.description = 'taper usage') AND
      (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP')
      IN TYPEOF(sar))) ) |
      ('FEATURE_BASED_PROCESS_PLANNING.TAPER'
```

 ©ISO 2000 - All rights reserved

```
      IN TYPEOF(fcr.relating_shape_aspect)) )) = 1)) )) <= 1)) )) = 0);

  WR5: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation))) = 1))) = 0;

  END_ENTITY; -- round_hole

(*
```

Formal propositions:

WR1: The **round_hole** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'diameter occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' and a **name** of 'diameter' and a **relating_shape_aspect** that is a **circular_closed_profile**.

WR2: The **round_hole** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'hole depth occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'path feature component usage' and a **name** of 'hole depth' in which the **relating_-shape_aspect** is a **path_feature_component** with a **description** of 'linear'.

NOTE - The specification of the hole depth places a limitation on the depth of the **round_hole** so that it will not interfere with other features of the part that are nearby.

WR3: The **round_hole** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'bottom condition occurrence' that is the **related_shape_aspect** in exactly one **feature_component_-relationship** with a **description** of 'hole bottom usage' and the **relating_shape_aspect** is a **hole_bottom**.

WR4: The **round_hole** shall be the basis shape for at most one **shape_aspect** with a **description** of 'change in diameter occurrence' that is the **related_shape_aspect** in exactly one **feature_component_-relationship** with a **description** of 'taper usage' and the **relating_shape_aspect** is a **taper**.

WR5: The **round_hole** shall contain exactly one implicit representation.

Informal propositions:

IP1: The location of the **round_hole** shall be defined at either the bottom or the top center of the hole.

IP2: The Z direction of the **round_hole** shall be in the direction of volume removal.

IP3: The origin, X direction, Y direction of the **circular_closed_profile** shall be equal to that of the **round_hole**.

IP4: The origin, X direction, Y direction of the **path_feature_component** shall be equal to that of the **round_hole**.

©ISO 2000 - All rights reserved

5.2.3.1.67 rounded_end

A **rounded_end** is a type of **feature_definition** that is the representation of a volume that is removed from the base shape. The removal shall be represented by **partial_circular_profile** swept along about a linear path represented by a **path_feature_component**. See ARM definition for Rounded_end in paragraph 4.2.194 for more information.

EXPRESS specification:

```
*)
  ENTITY rounded_end
    SUBTYPE OF (feature_definition);
    WHERE

  wr1: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(impl_rep.used_representation.items) = 1)) ))
       = 0)) )) = 0);

  wr2: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
       IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       ((sa_occ.description = 'partial circular boundary occurrence') AND
       (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
       'RELATED_SHAPE_ASPECT') |
       ((sar.description = 'profile usage') AND
       (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
       IN TYPEOF(sar))) ) |
       (('FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE'
       IN TYPEOF(sdr.relating_shape_aspect)) ) )) = 1)) )) = 1)) )) = 0);

  wr3: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
       IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       ((sa_occ.description = 'course of travel occurrence') AND
       (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
       'RELATED_SHAPE_ASPECT') |
       ((sar.description = 'path feature component usage') AND
       (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
       IN TYPEOF(sar))) ) |
       (('FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT'
       IN TYPEOF(sdr.relating_shape_aspect)) AND
       (sdr.relating_shape_aspect.description = 'linear')) )) = 1)) ))
       = 1)) )) = 0);
  END_ENTITY; -- rounded_end
(*
```

 ©ISO 2000 - All rights reserved

Formal propositions:

WR1: The **rounded_end** shall have an implicit representation that contains exactly one **representation_item** in its set of **items**.

WR2: The **rounded_end** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'partial circular boundary occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_-relationship** with a **description** of 'profile usage' in which the **relating_shape_aspect** references a **partial_circular_profile**.

WR3: The **rounded_end** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'course of travel occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_-relationship** with a **description** of 'path feature component usage' and a **relating_shape_aspect** that references a **path_feature_component** with a **description** of 'linear'.

Informal propositions:

IP1: The origin, X direction, Y direction of the **partial_circular_profile** shall be equal to that of the **rounded_end**.

IP2: The origin, X direction, Y direction of the **path_feature_component** shall be equal to that of the **rounded_end**.

5.2.3.1.68 rounded_u_profile

A **rounded_u_profile** is a type of **shape_aspect** that is the representation of an arc with connected straight lines at each end, with a location and a position. The **rounded_u_profile** is located at the midpoint of the arc and positioned in the X-Y plane with the X direction tangent to the arc at the midpoint, and the Y direction orthogonal in the direction of volume removal. See ARM definition for Rounded_U_profile in paragraph 4.2.195 for more information.

EXPRESS specification:

```
*)
  ENTITY rounded_u_profile
    SUBTYPE OF (shape_aspect);
    WHERE
  wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
       IN TYPEOF(SELF.of_shape.definition));

  wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0);
```

©ISO 2000 - All rights reserved

```
  wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF (impl_rep.used_representation.items) = 2)) ))
       = 0)) )) = 0);

  WR4: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       (('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT'
       IN TYPEOF(it)) AND (it.name = 'orientation')) )) = 1)) ))
       = 0)) )) = 0);

  wr5: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       ((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF(it)) = 2) AND (it.name = 'width'))))= 1))))= 0))))= 0);

  WR6: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
       IN TYPEOF (pdr.used_representation))AND
       (pdr.used_representation.name = 'profile limit'))) <= 1))) = 0;

  END_ENTITY; -- rounded_u_profile
(*
```

Formal propositions:

WR1: The **rounded_u_profile** shall be an aspect of the shape of a **feature_component_definition**.

WR2: The **rounded_u_profile** shall have exactly one implicit representation defined by a relationship to a **shape_representation_with_parameters**.

 ©ISO 2000 - All rights reserved

WR3: The **shape_representation_with_parameters** that represents the **rounded_u_profile** shall contain exactly two **representation_item**s in its set of **items**.

WR4: Exactly one **representation_item** used for the implicit representation of a **rounded_u_profile** shall be of type **placement** with a **name** of 'orientation'.

WR5: Exactly one **representation_item** used for the implicit representation of a **rounded_u_profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'width'.

WR6: The **rounded_u_profile** shall have at most one **planar_shape_representation** with a **name** of 'profile limit'.

Informal propositions:

IP1: The location of the **rounded_u_profile** shall be at the midpoint of the arc.

IP2: The **rounded_u_profile** shall be defined with the X direction tangent to the arc at the midpoint, and the Y direction orthogonal in the direction of the open end of the profile..

5.2.3.1.69 shape_defining_relationship

A **shape_defining_relationship** is a kind of **shape_aspect_relationship** in which the **related_shape_-aspect** takes a part in the definition of the shape of the **relating_shape_aspect**.

EXAMPLE - A Chamfer may have the implicit chamfer offset amount defined with a **feature_component_-relationship** and the explicit geometry for the chamfer defined with a **shape_defining_relationship**.

EXPRESS specification:

```
*)
ENTITY shape_defining_relationship
  SUBTYPE OF (shape_aspect_relationship);
END_ENTITY;
(*
```

5.2.3.1.70 shape_representation_with_parameters

A **shape_representation_with_parameters** is a kind of **shape_representation** in which the shape of a **product_definition** or **shape_aspect** is defined implicitly using measurements and descriptive parameters.

EXAMPLE - The shape of a box may be specified by a **shape_representation_with_parameters** by giving measurements for its height, length, and width.

EXPRESS specification:

```
*)
  ENTITY shape_representation_with_parameters
    SUBTYPE OF (shape_representation);
    WHERE
  wr1: (SIZEOF(QUERY ( it <* SELF.items |
```

©ISO 2000 - All rights reserved

```
       (NOT (SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT',
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'] *
       TYPEOF(it)) = 1)) )) = 0);
  END_ENTITY; -- shape_representation_with_parameters
(*
```

Formal proposition:

WR1: The elements that specify the parameters for a **shape_representation_with_parameters** shall be of type **placement**, **measure_representation_item**, or **descriptive_representation_item**.

5.2.3.1.71 slot

A **slot** is a type of **feature_definition** that is the representation of a volume that is removed from the base shape. This removal shall be represented by sweeping either a **square_u_profile**, **partial_circular_-profile**, **round_u_profile**, **vee_profile**, **tee_profile**, or an **open_path_profile** swept along a path represented by a **path_feature_component**. Each end of the **slot** shall be represented by a **slot_end**. See ARM definition for Slot in paragraph 4.2.205 for more information.

EXPRESS specification:

```
*)
  ENTITY slot
    SUBTYPE OF (feature_definition);
    WHERE
  wr1: (SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       (NOT (SIZEOF (impl_rep.used_representation.items) = 1) ) ))
       = 0))) = 0);


  wr2: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
       IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       ((sa_occ.description = 'swept shape occurrence') AND
       (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
       'RELATED_SHAPE_ASPECT') |
       ((sar.description = 'profile usage') AND
       (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
       IN TYPEOF(sar))) ) |
       ((SIZEOF(['FEATURE_BASED_PROCESS_PLANNING.SQUARE_U_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.ROUNDED_U_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.VEE_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.TEE_PROFILE',
       'FEATURE_BASED_PROCESS_PLANNING.OPEN_PATH_PROFILE'] *
```

 ©ISO 2000 - All rights reserved

```
      TYPEOF(sdr.relating_shape_aspect)) = 1)) )) = 1)) ))
      = 1)) )) = 0);

  wr3: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      ((sa_occ.description = 'course of travel occurrence') AND
      (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      ((sar.description = 'path feature component usage') AND
      ((sar.name = 'course of travel') AND
      (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
      IN TYPEOF(sar))) )) |
      ('FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT'
      IN TYPEOF(sdr.relating_shape_aspect)) )) = 1)) )) = 1)) )) = 0);

  wr4: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
      IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
      ((sa_occ.description = 'end condition occurrence') AND
      (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(sa_occ,
      'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
      'RELATED_SHAPE_ASPECT') |
      ((sar.description = 'slot end usage') AND
      (sar.name IN ['course of travel start','course of travel end']) AND
      (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP')
      IN TYPEOF(sar))) ) |
      (('FEATURE_BASED_PROCESS_PLANNING.SLOT_END'
      IN TYPEOF(fcr.relating_shape_aspect)) )  ))
      = 1)) )) = 2)) )) = 0);
  END_ENTITY; -- slot
(*
```

Formal propositions:

WR1: The implicit representation of the **slot** shall contain exactly one **representation_item**s in its set of **items**.

WR2: The **slot** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'swept shape occurance' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** 'profile usage' and a **relating_shape_aspect** that references a **square_u_profile**, **partial_-circular_profile**, **rounded_u_profile**, **vee_profile**, **tee_profile**, or **open_path_profile**.

WR3: The **slot** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'course of travel occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'path feature component usage', and a **name** of of 'course of travel' and a **relating_-shape_aspect** that references a **path_feature_component**.

©ISO 2000 - All rights reserved

WR4: The **slot** shall be the basis shape for exactly two **shape_aspect**s with a **description** of 'end condition occurrence' that are each the **related_shape_aspect** in exactly one **feature_component_-relationship** with a **description** of 'slot end usage' and a **relating_shape_aspect** that references a **slot_-end** with a **description** of 'end condition'.

Informal propositions:

IP1: The location of the **slot** shall be defined at one end of the **slot**.

IP2: If the **description** of the **path_feature_component** relating to the **slot** is 'linear' or 'complex', then the Z direction of the **slot** shall be in the direction of **slot** end and the volume removal in the Y direction.

IP3: If the **description** of the **path_feature_component** relating to the **slot** is 'circular' or 'partial circular', then the Z direction of the **slot** shall be the axis of revolution.

IP4: If the **description** of the **path_feature_component** relating to the **slot** is 'linear' or 'complex', then the origin, X direction, Y direction of the **square_u_profile**, **partial_circular_profile**, **rounded_u_-profile**, **vee_profile**, **tee_profile**, or **open_path_profile** shall be equal to that of the **slot**.

IP5: If the **description** of the **path_feature_component** relating to the **slot** is 'circular' or 'partial circular', then the origin, X direction, Y direction of the **square_u_profile**, **partial_circular_profile**, **rounded_u_profile**, **vee_profile**, **tee_profile**, or **open_path_profile** shall be in the X direction of the **slot** at a distance equal to the radius of the **path_feature_component** away from the origin of the slot.

IP6: The origin, X direction, Y direction of the **path_feature_component** shall be equal to that of the **slot**.

5.2.3.1.72 slot_end

A **slot_end** is a type of **shape_aspect** that is the representation of the end condition for a **slot feature_-definition**. See ARM definition for Slot_end_type in paragraph 4.2.206 for more information.

EXPRESS specification:

```
*)
  ENTITY slot_end
    SUBTYPE OF (shape_aspect);
    WHERE
  wr1 : ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
        IN TYPEOF(SELF.of_shape.definition));

  wr2 : (SELF.description IN ['open','radiused','flat','woodruff']);

  wr3 : ((NOT (SELF.description IN ['open','radiused'])) OR
        (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) )) = 0)) )) = 0));
```

 ©ISO 2000 - All rights reserved

```
wr4 : ((NOT (SELF.description IN ['flat','woodruff'])) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0));

wr5 : ((NOT (SELF.description IN ['flat'])) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(impl_rep.used_representation.items) = 2)) ))
       = 0)) )) = 0));

wr6 : ((NOT (SELF.description = 'flat')) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF(it)) = 2) AND (it.name = 'first radius')) )) = 1)) ))
      = 0)) )) = 0));

wr7 : ((NOT (SELF.description = 'flat')) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF(it)) = 2) AND (it.name = 'second radius')))) = 1)) ))
      = 0)) )) = 0));

wr8 : ((NOT (SELF.description = 'woodruff')) OR
      (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
```

©ISO 2000 - All rights reserved

```
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
        (('FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'
        IN TYPEOF(it)) AND (it.name = 'radius')) )) = 1)) ))
        = 0)) )) = 0));

  wr9 : ((NOT (SELF.description IN ['woodruff'])) OR
        (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(impl_rep.used_representation.items) = 1)) ))
        = 0)) )) = 0));

  wr10: (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
        'RELATING_SHAPE_ASPECT') |
        ((sar.description = 'slot end usage') AND
        (sar.name IN ['course of travel start','course of travel end']) AND
        ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
        IN TYPEOF(sar))) ) |
        ((fcr.related_shape_aspect.description='end condition occurrence')
        AND
        ('FEATURE_BASED_PROCESS_PLANNING.SLOT'
        IN TYPEOF(fcr.related_shape_aspect.of_shape.definition))) ))
        >= 1);
  END_ENTITY; -- slot_end
(*
```

Formal propositions:

WR1: The **slot_end** shall be an aspect of the shape of a **feature_component_definition**.

WR2: The **description** for the **slot_end** shall be either 'open', 'radiused', 'flat', or 'woodruff'.

WR3: If the **description** of the **slot_end** is 'open' or 'radiused', the **slot_end** shall not have any implicit representation.

WR4: If the **description** of the **slot_end** is 'flat' or 'woodruff, 'the **slot_end** shall have its implicit representation specified by exactly one **shape_representation_with_parameters**.

WR5: If the **description** of the **slot_end** is 'flat", the **slot_end** shall have an implicit representation that contains exactly two **representation_item** in its set of **items**.

WR6: If the **description** of the **slot_end** is 'flat', exactly one **representation_item** used for the implicit representation of the **slot_end** shall be of type **measure_representation_item** and **length_measure_-with_unit** with a **name** of 'first radius'.

 ©ISO 2000 - All rights reserved

WR7: If the **description** of the **slot_end** is 'flat', exactly one **representation_item** used for the implicit representation of the **slot_end** shall be of type **measure_representation_item** and **length_measure_-with_unit** with a **name** of 'second radius'.

WR8: If the **description** of the **slot_end** is 'woodruff', exactly one **representation_item** used for the implicit representation of the **slot_end** shall be of type **measure_representation_item** and **length_-measure_with_unit** with a **name** of 'radius'.

WR9: If the **description** of the **slot_end** is 'woodruff', the **slot_end** shall have an implicit representation that contains exactly one **representation_item** in its set of **items**.

WR10: The **slot_end** shall be the **relating_shape_aspect** in at least one **feature_component_-relationship** with a **description** of 'slot end usage' and with a **name** of 'course of travel start', or 'course of travel end' in which the **related_shape_aspect** is an aspect of the shape of a **slot** with a **description** of 'end condition occurrence'.

Informal propositions:

IP1: If the **description** of the **slot_end** is 'open', then it shall have no placement or orientation.

IP2: If the **description** of the **slot_end** is 'flat', then location shall be at the midpoint of the base line.

IP3: If the **description** of the **slot_end** is 'radiused', then location shall be at the midpoint of the radius.

IP4: If the **description** of the **slot_end** is 'woodruff', then location shall be at the point where the slot floor makes contact with the radius.

IP5: If the **description** of the **slot_end** is 'open', 'radiused', or 'woodruff', then the **slot_end** is defined with the base line in the X direction and the Y direction orthogonal in the direction of volume removal. The Z direction is coincident to that of the mating **slot** feature.

5.2.3.1.73 spherical_cap

A **spherical_cap** is a type of **feature_definition** that is the representation of a volume that is extruding from the base shape. The volume shall be spherical of a defined radius. See ARM definition for Spherical_cap paragraph 4.2.209 for more information.

EXPRESS specification:

```
*)
  ENTITY spherical_cap
    SUBTYPE OF (feature_definition);
    WHERE

  wr1: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
```

©ISO 2000 - All rights reserved

```
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(impl_rep.used_representation.items) = 3)) ))
      = 0)) )) = 0);

  wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF(it)) = 2) AND (it.name = 'radius')) )) = 1))))
      = 0)) )) = 0);

  wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT'] *
      TYPEOF(it)) = 2) AND (it.name = 'internal angle')))) = 1)) ))
      = 0)) )) = 0);
  END_ENTITY; -- spherical_cap
(*
```

Formal propositions:

WR1: The **spherical_cap** shall have an implicit representation that contains exactly three **representation_item** in its set of **items**.

WR2: Exactly one **representation_item** used for the implicit representation of the **spherical_cap** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'radius'.

WR3: Exactly one **representation_item** used for the implicit representation of the **spherical_cap** shall be of type **measure_representation_item** and **plane_angle_measure_with_unit** with a **name** of 'internal angle'.

Informal propositions:

IP1: The orientation of the **spherical_cap** shall be with the Z-axis away from material.

IP2: The X-axis shall define the start of the **spherical_cap** and the internal angle is measured from the X-axis.

 ©ISO 2000 - All rights reserved

IP3: The location of the **spherical_cap** shall be at a position on the base of the feature where it mates with the part.

IP4: The Z direction of the **spherical_cap** shall be in the direction of away from the part.

5.2.3.1.74 square_u_profile

A **square_u_profile** is a type of **shape_aspect** that is the representation of three connected straight lines, a location and a position. The **square_u_profile** is located at the midpoint of the base line and positioned in the X-Y plane with the base line on the X direction and the Y direction orthogonal in the direction of volume removal. See ARM definition for Square_U_profile in paragraph 4.2.211 for more information.

EXPRESS specification:

```
*)
ENTITY square_u_profile
  SUBTYPE OF (shape_aspect);
WHERE
  WR1: 'FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
       IN TYPEOF (SELF.of_shape.definition);
  WR2: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation))) = 1))) = 0;

  WR3: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       (NOT (SIZEOF (impl_rep.used_representation.items) >= 4)
       AND  (SIZEOF (impl_rep.used_representation.items) <= 6))))
       = 0))) = 0;

  WR4: SIZEOF( QUERY( pd <* USEDIN( SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      SIZEOF( QUERY( pdr <* USEDIN( pd, 'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.'+
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF(pdr.used_representation)) AND
       (SIZEOF( QUERY( srwp_i <* pdr.used_representation.items |
       (srwp_i.name = 'orientation') OR
       (srwp_i.name = 'width') OR
       (srwp_i.name = 'first angle') OR
       (srwp_i.name = 'second angle') OR
       (srwp_i.name = 'first radius') OR
       (srwp_i.name = 'second radius') OR
       (srwp_i.name = 'profile limit') ))
```

©ISO 2000 - All rights reserved

```
       = SIZEOF(pdr.used_representation.items)) )) = 1 )) = 1;

  WR5: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
       ('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF (it)) AND
       (it.name = 'orientation'))) = 1))) = 0))) = 0;

  WR6: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
       (SIZEOF
       (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF (it)) = 2) AND
       (it.name = 'width'))) = 1))) = 0))) = 0;

  WR7: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
       (SIZEOF
       (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF (it)) = 2) AND
       (it.name = 'first radius'))) <= 1))) = 0))) = 0;

  WR8: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
       (SIZEOF
       (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF (it)) = 2) AND
       (it.name = 'second radius'))) <= 1))) = 0))) = 0;

  WR9: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
```

 ©ISO 2000 - All rights reserved

```
      NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS'
      IN TYPEOF (pdr.used_representation)) |
      NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
      (SIZEOF
      (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT'] *
      TYPEOF (it)) = 2) AND
      (it.name = 'first angle'))) = 1))) = 0))) = 0;

  WR10: SIZEOF (QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS'
      IN TYPEOF (pdr.used_representation)) |
      NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
      (SIZEOF
      (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT'] *
      TYPEOF (it)) = 2) AND
      (it.name = 'second angle'))) = 1))) = 0))) = 0;

  WR11: SIZEOF (QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
      'DEFINITION') |
      NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
      IN TYPEOF (pdr.used_representation))AND
      (pdr.used_representation.name = 'profile limit'))) <= 1))) = 0;

END_ENTITY; -- Square_U_profile
(*
```

Formal propositions:

WR1: The **square_u_profile** shall be an aspect of the shape of a **feature_component_definition**.

WR2: The **square_u_profile** shall have exactly one implicit representation defined by a relationship to a **shape_representation_with_parameters**.

WR3: The **shape_representation_with_parameters** that represents the **square_u_profile** shall contain at least four and at most six **representation_item**s in its set of **items**.

WR4: The implicit representation of a **square_u_profile** shall contain only **representation_-item**s in its set of **items** with a **name** of 'orientation','width', 'first angle', 'second angle', 'first radius', and 'second radius'.

WR5: Exactly one **representation_item** used for the implicit representation of a **square_u_profile** shall be of type **placement** with a **name** of 'orientation'.

©ISO 2000 - All rights reserved

WR6: Exactly one **representation_item** used for the implicit representation of a **square_u_profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'width'.

WR7: At most one **representation_item** used for the implicit representation of a **square_u_profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'first radius'.

WR8: At most one **representation_item** used for the implicit representation of a **square_u_profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'second radius'.

WR9: Exactly one **representation_item** used for the implicit representation of a **square_u_profile** shall be of type **measure_representation_item** and **plane_angle_measure_with_unit** with a **name** of 'first angle'.

WR10: Exactly one **representation_item** used for the implicit representation of a **square_u_profile** shall be of type **measure_representation_item** and **plane_angle_measure_with_unit** with a **name** of 'second angle'.

WR11: The **square_u_profile** shall have at most one **planar_shape_representation** with a **name** of 'profile limit'.

Informal propositions:

IP1: The location of the **square_u_profile** shall be defined at the midpoint of the base line.

IP2: The **square_u_profile** shall be defined in the X-Y plane with the base line in the X direction and the Y direction orthogonal in the direction of the open end of the profile.

5.2.3.1.75 step

A **step** is a type of **feature_definition** that is the representation of a volume that is removed from the base shape. This removal shall be represented by sweeping a **vee_profile** along a path represented by a **path_feature_component**. See ARM definition for Step in paragraph 4.2.212 for more information.

EXPRESS specification:

```
*)
  ENTITY step
    SUBTYPE OF (feature_definition);
    WHERE

  wr1: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(impl_rep.used_representation.items) = 1)) ))
       = 0)) )) = 0);
```

 ©ISO 2000 - All rights reserved

```
   wr2: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
        IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
        ((sa_occ.description = 'course of travel occurrence') AND
        (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
        'RELATED_SHAPE_ASPECT') |
        ((sar.description = 'path feature component usage') AND
        (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
        IN TYPEOF(sar))) ) |
        (('FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT'
        IN TYPEOF(sdr.relating_shape_aspect)) AND
        (sdr.relating_shape_aspect.description = 'linear')) )) = 1)) ))
        = 1)) )) = 0);

  wr3: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
        IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
        ((sa_occ.description = 'removal boundary occurrence') AND
        (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
        'RELATED_SHAPE_ASPECT') |
        ((sar.description = 'profile usage') AND
        (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
        IN TYPEOF(sar))) ) |
        ('FEATURE_BASED_PROCESS_PLANNING.VEE_PROFILE'
        IN TYPEOF(sdr.relating_shape_aspect)) )) = 1)) )) = 1)) )) = 0);
  END_ENTITY; -- step
(*
```

Formal proposition:

WR3: The **step** shall have an implicit representation that contains exactly one **representation_item** in its set of **items**.

WR1: The **step** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'course of travel occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'path feature component usage' and a **relating_shape_aspect** that references a **path_-feature_component** with a **description** of 'linear'.

WR2: The **step** shall be the basis shape for exactly one **shape_aspect** with a **description** of 'removal boundary occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'profile usage' and a **relating_shape_aspect** that references a **vee_profile**.

Informal propositions:

IP1: The origin, X direction, Y direction of the **vee_profile** shall be equal to that of the **step**.

IP2: The origin, X direction, Y direction of the **path_feature_component** shall be equal to that of the **step**.

©ISO 2000 - All rights reserved

5.2.3.1.76 taper

A **taper** is a type of **shape_aspect** which represents a linear change applied to a machining feature. See ARM definition for Angle_taper in paragraph 4.2.2 and Diameter_taper in paragraph 4.2.60 for more information.

EXPRESS specification:

```
*)
  ENTITY taper
    SUBTYPE OF (shape_aspect);
    WHERE
  wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
       IN TYPEOF(SELF.of_shape.definition));

  wr2: (SELF.description IN ['angle taper','diameter taper',
       'directed taper']);

  wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0);

  wr4: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(impl_rep.used_representation.items) = 1)) ))
       = 0)) )) = 0);

  wr5: ((NOT (SELF.description = 'angle taper')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       ((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT'] *
       TYPEOF(it)) = 2) AND (it.name = 'taper angle')) )) = 1)) ))
       = 0)) )) = 0));

  wr6: ((NOT (SELF.description = 'diameter taper')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
```

 ©ISO 2000 - All rights reserved

```
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       ((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF(it)) = 2) AND (it.name = 'final diameter')) )) = 1)) ))
       = 0)) )) = 0));

  wr7: ((NOT (SELF.description = 'directed taper')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       ((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT'] *
       TYPEOF(it)) = 2) AND (it.name = 'angle')) )) = 1)) ))
       = 0)) )) = 0));

 wr8: ((NOT (SELF.description = 'directed taper')) OR
       (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
       IN TYPEOF(pdr.used_representation)) AND
       (pdr.used_representation.name = 'direction') ))
       = 1)) )) = 0));
  END_ENTITY; -- taper
(*
```

Formal propositions:

WR1: The **taper** shall be an aspect of the shape of a **feature_component_definition**.

WR2: The **description** of the **taper** shall be either 'angle taper', 'diameter taper', or 'directed taper'.

WR3: The **taper** shall have exactly one implicit representation defined by a relationship to a **shape_-representation_with_parameters**.

WR4: The **shape_representation_with_parameters** that represents the **taper** shall contain exactly one **representation_item** in its set of **items**.

WR5: The **representation_item** used for the implicit representation of an angular **taper** shall be of type **measure_representation_item** and **plane_angle_measure_with_unit** with a **name** of 'taper angle'.

©ISO 2000 - All rights reserved

WR6: The **representation_item** used for the implicit representation of a diameter **taper** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'taper diameter'.

WR7: If the **description** of the **taper** is 'directed taper', the implicit representation of the **taper** shall contain exactly one **representation_item** of type **measure_representation_item** and **plane_angle_-measure_with_unit** with a **name** of 'angle'.

WR8: If the **description** of the **taper** is 'directed taper', its shape shall be represented by exactly one **direction_shape_representation**. This **direction_shape_representation** shall have a **name** of ' direction'.

Informal proposition:

IP1: The taper angle shall be -90 degrees < taper < 90 degrees.

5.2.3.1.77 tee_profile

A **tee_profile** is a type of **shape_aspect** which is the representation of an upside down T shape with a location and a position. The **tee_profile** consist of a stem and a perpendicular cross bar. The stem has two parallel lines connected at their ends by the cross bar. The cross bar is a rectangle. The profile is symmetric about the centerline of the stem. The **tee_profile** is located at the midpoint of the side of the cross bar opposite the stem, and positioned with the side of the cross bar opposite the stem on the X direction and the profile symmetric with respect to the Y direction. See ARM definition for Tee_profile in paragraph 4.2.223 for more information.

EXPRESS specification:

```
*)
ENTITY tee_profile
  SUBTYPE OF (shape_aspect);
WHERE
  WR1: 'FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
       IN TYPEOF (SELF.of_shape.definition);

  WR2: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation))) = 1))) = 0;

  WR3: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       (NOT (SIZEOF (impl_rep.used_representation.items) >= 9)
       AND  (SIZEOF (impl_rep.used_representation.items) <= 10))))
       = 0))) = 0;
```

 ©ISO 2000 - All rights reserved

```
WR4:  SIZEOF( QUERY( pd <* USEDIN( SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      SIZEOF( QUERY( pdr <* USEDIN( pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      ('FEATURE_BASED_PROCESS_PLANNING.'+
      'SHAPE_REPRESENTATION_WITH_PARAMETERS'
      IN TYPEOF(pdr.used_representation)) AND
      (SIZEOF( QUERY( srwp_i <* pdr.used_representation.items |
      (srwp_i.name = 'orientation') OR
      (srwp_i.name = 'width') OR
      (srwp_i.name = 'depth') OR
      (srwp_i.name = 'cross bar width') OR
      (srwp_i.name = 'cross bar depth') OR
      (srwp_i.name = 'first offset') OR
      (srwp_i.name = 'second offset') OR
      (srwp_i.name = 'first angle') OR
      (srwp_i.name = 'second angle') OR
      (srwp_i.name = 'radius') ))
      = SIZEOF(pdr.used_representation.items)) )) = 1 )) = 1;

  WR5: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
       ('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF (it)) AND
       (it.name = 'orientation'))) = 1))) = 0))) = 0;

  WR6: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
       (SIZEOF
       (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF (it)) = 2) AND  (it.name = 'width'))) = 1))) = 0))) = 0;

  WR7: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
       (SIZEOF
       (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF (it)) = 2) AND (it.name = 'depth'))) = 1))) = 0))) = 0;
```

©ISO 2000 - All rights reserved

```
 WR8: SIZEOF (QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS'
      IN TYPEOF (pdr.used_representation)) |
      NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
      (SIZEOF
      (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF (it)) = 2) AND (it.name = 'cross bar width'))) = 1)))
      = 0))) = 0;

 WR9: SIZEOF (QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS'
      IN TYPEOF (pdr.used_representation)) |
      NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
      (SIZEOF
      (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF (it)) = 2) AND (it.name = 'cross bar depth'))) = 1)))
      = 0))) = 0;

WR10: SIZEOF (QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS'
      IN TYPEOF (pdr.used_representation)) |
      NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
      (SIZEOF
      (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF (it)) = 2) AND (it.name = 'first offset'))) = 1)))
      = 0))) = 0;

WR11: SIZEOF (QUERY (pd <* USEDIN (SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS'
      IN TYPEOF (pdr.used_representation)) |
      NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
      (SIZEOF
      (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF (it)) = 2) AND (it.name = 'second offset'))) = 1)))
      = 0))) = 0;
```

 ©ISO 2000 - All rights reserved

```
 WR12: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
       (SIZEOF
       (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT'] *
       TYPEOF (it)) = 2) AND (it.name = 'first angle'))) = 1)))
       = 0))) = 0;

 WR13: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
       (SIZEOF
       (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT'] *
       TYPEOF (it)) = 2) AND (it.name = 'second angle'))) = 1)))
       = 0))) = 0;

 WR14: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
       (SIZEOF
       (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
       TYPEOF (it)) = 2) AND (it.name = 'radius'))) <= 1))) = 0))) = 0;

WR15: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
       'DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
       IN TYPEOF (pdr.used_representation))AND
       (pdr.used_representation.name = 'profile limit'))) <= 1))) = 0;

END_ENTITY;
(*
```

Formal propositions:

WR1: The **tee_profile** shall be an aspect of the shape of a **feature_component_definition**.

©ISO 2000 - All rights reserved

WR2: The **tee_profile** shall have exactly one implicit representation defined by a relationship to a **shape_representation_with_parameters**.

WR3: The **shape_representation_with_parameters** that represents the **tee_profile** shall contain at least nine and at most ten **representation_item**s in its set of **items**.

WR4: The implicit representation of a **tee_profile** shall contain only **representation_-item**s in its set of **items** with a **name** of 'orientation', 'width', 'depth', 'cross bar width', 'cross bar depth', 'first offset', 'second offset', 'first angle', 'second angle', and 'radius'.

WR5: Exactly one **representation_item** used for the implicit representation of a **tee_profile** shall be of type **placement** with a **name** of 'orientation'.

WR6: Exactly one **representation_item** used for the implicit representation of a **tee_profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'width'.

WR7: Exactly one **representation_item** used for the implicit representation of a **tee_profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'depth'.

WR8: Exactly one **representation_item** used for the implicit representation of a **tee_profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'cross bar width'.

WR9: Exactly one **representation_item** used for the implicit representation of a **tee_profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'cross bar depth'.

WR10: Exactly one **representation_item** used for the implicit representation of a **tee_profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'first offset'.

WR11: Exactly one **representation_item** used for the implicit representation of a **tee_profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'second offset'.

WR12: Exactly one **representation_item** used for the implicit representation of a **tee_profile** shall be of type **measure_representation_item** and **plane_angle_measure_with_unit** with a **name** of 'first angle'.

WR13: Exactly one **representation_item** used for the implicit representation of a **tee_profile** shall be of type **measure_representation_item** and **plane_angle_measure_with_unit** with a **name** of 'second angle'.

WR14: At most one **representation_item** used for the implicit representation of a **tee_profile** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'radius'.

WR15: The **tee_profile** shall have at most one **planar_shape_representation** with a **name** of 'profile limit'.

 ©ISO 2000 - All rights reserved

Informal propositions:

IP1: The location of the **tee_profile** shall be at the midpoint of the side of the cross bar opposite the stem.

IP2: The **tee_profile** shall be defined with the side of the cross bar opposite the stem in the X direction and the profile symmetric with respect to the Y direction.

5.2.3.1.78 thread

A **thread** is a type of **feature_definition** that is the representation of a thread shape that is applied to all or a portion of a **shape_aspect**. See ARM definition for Thread in paragraph 4.2.224 for more information.

EXPRESS specification:

```
*)
ENTITY thread
  SUBTYPE OF (feature_definition);
WHERE
  wr1 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (SIZEOF(QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        ((('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) AND
        (8 <= SIZEOF(pdr.used_representation.items)) AND
        (SIZEOF(pdr.used_representation.items) <= 11)) )) = 1) )) = 1);
  wr2 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN( pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
        ((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
        * TYPEOF(it)) = 2) AND (it.name = 'major diameter')) )) = 1)) ))
        = 0)) )) = 0);
  wr3 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
        ((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
        * TYPEOF(it)) = 2) AND (it.name = 'minor diameter')) ))
```

©ISO 2000 - All rights reserved

```
      <= 1)) )) = 0)) )) = 0);
  wr4 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN( pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
        ((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
        * TYPEOF(it)) = 2) AND (it.name = 'pitch diameter')) )) = 1)) ))
        = 0)) )) = 0);
  wr5 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
        ((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.RATIO_MEASURE_WITH_UNIT'] *
        TYPEOF(it)) = 2) AND (it.name = 'number of threads')) ))
        = 1)) )) = 0)) )) = 0);
  wr6 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN( pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
        (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
        IN TYPEOF(it)) AND (it.name = 'fit class')) )) = 1)) )) = 0)) ))
        = 0);
  wr7 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
        (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
        IN TYPEOF(it)) AND (it.name = 'form')) )) = 1)) )) = 0)) ))
        = 0);
  wr8 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN( pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
```

 ©ISO 2000 - All rights reserved

```
        (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
        (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
        IN TYPEOF(it)) AND (it.name = 'hand')AND
        (it.description IN ['left', 'right']) ) )) = 1)) )) = 0)) ))
        = 0);
 wr9 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
        (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
        IN TYPEOF(it)) AND (it.name = 'qualifier')) )) <= 1)) ))
        = 0)) )) = 0);
wr10: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN( pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
       IN TYPEOF(it)) AND (it.name = 'thread side') AND
       ((it.description = 'internal') OR (it.description = 'external'))) ))
       = 1)) )) = 0)) )) = 0);
wr11: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       ((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
       * TYPEOF(it)) = 2) AND (it.name = 'crest')) )) <= 1)) ))
       = 0)) )) = 0);
wr12: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
       IN TYPEOF(pd)) ) |
       (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       ((sa_occ.description = 'partial area occurrence') AND
       (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
       + 'RELATED_SHAPE_ASPECT') |
       ((sar.description = 'applied area usage') AND
       (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
       IN TYPEOF(sar))) ) |
       ('FEATURE_BASED_PROCESS_PLANNING.APPLIED_AREA'
       IN TYPEOF(sdr.relating_shape_aspect)) )) = 1)) )) = 1)) )) = 0);
wr13: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
```

©ISO 2000 - All rights reserved

```
        ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
        IN TYPEOF(pd)) ) |
        (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
        ((SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
        + 'RELATED_SHAPE_ASPECT') |
        ((sar.description = 'applied shape') AND
        (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
        IN TYPEOF(sar))) ) |
        ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT'
        IN TYPEOF(sdr.relating_shape_aspect)) )) = 1)) )) = 1)) )) = 0);

 END_ENTITY;
(*
```

Formal propositions:

WR1: The implicit representation of the **thread** shall contain between six and eleven **representation_item**s.

WR2: Exactly one **representation_item** used for the implicit representation of the **thread** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'major diameter'.

WR3: At most one **representation_item** used for the implicit representation of the **thread** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'minor diameter'.

WR4: Exactly one **representation_item** used for the implicit representation of the **thread** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'pitch diameter'.

WR5: Exactly one **representation_item** used for the implicit representation of the **thread** shall be of type **measure_representation_item** and **ratio_measure_with_unit** with a **name** of 'number of threads'.

WR6: Exactly one **representation_item** used for the implicit representation of the **thread** shall be of type **descriptive_representation_item** with a **name** of 'fit class'.

WR7: Exactly one **representation_item** used for the implicit representation of the **thread** shall be of type **descriptive_representation_item** with a **name** of 'form'.

WR8: Exactly one **representation_item** used for the implicit representation of the **thread** shall be of type **descriptive_representation_item** with a **name** of 'hand' and **description** of 'left' or 'right'.

WR9: At most one **representation_item** used for the implicit representation of the **thread** shall be of type **descriptive_representation_item** with a **name** of 'qualifier'.

WR10: Exactly one **representation_item** used for the implicit representation of the **thread** shall be of type **descriptive_representation_item** with a **name** of 'thread side' and a **description** of either 'internal' or 'external'.

WR11: At most one **representation_item** used for the implicit representation of the **thread** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'crest'.

 ©ISO 2000 - All rights reserved

WR12: The **thread** shall be the basis shape for at most one **shape_aspect** with a **description** of 'partial area occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'applied area usage' and the **relating_shape_aspect** is an **applied_area**.

WR13: Exactly one **thread** shall be the basis shape for exactly one **shape_aspect** that is the **related_-shape_aspect** in at most one **shape_defining_relationship** with a description of 'applied shape' in which the **relating_shape_aspect** is a **shape_aspect**.

Informal propositions:

IP1: The location of the **thread** shall be at a distance from the end of a cylindrical face, and on the axis of the cylindrical face.

IP2: The Z direction of the **thread** shall be in the direction of the center axis of the cylindrical face.

IP3: If the **name** of the **descriptive_representation_item** is 'fit class' and the first character of the **description** is 'M', then the thread is metric.

5.2.3.1.79 transition_feature

A **transition_feature** is a type of **shape_aspect** that is the representation of a transition between two **shape_aspect** entities. The placement of a **transition_feature** is dependent on the placement of the two **shape_aspect** entities.

EXPRESS specification:

```
*)
  ENTITY transition_feature
    SUPERTYPE OF (ONEOF (chamfer,edge_round,fillet))
    SUBTYPE OF (shape_aspect);
  WHERE

  wr1:  SIZEOF(['FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION',
                'FEATURE_BASED_PROCESS_PLANNING.COMPOUND_FEATURE']
                * TYPEOF(SELF.of_shape.definition)) = 1;

  wr2:  SIZEOF(['FEATURE_BASED_PROCESS_PLANNING.CHAMFER',
                 'FEATURE_BASED_PROCESS_PLANNING.EDGE_ROUND',
                 'FEATURE_BASED_PROCESS_PLANNING.FILLET'] *
        TYPEOF(SELF)) = 1;
  END_ENTITY; -- transition_feature
(*
```

Formal propositions:

WR1: The **transition_feature** shall identify an aspect of the shape of a **product_definition** or a **compound_feature**.

WR2: The **transition_feature** shall be a **fillet**, **edge_round**, or a **chamfer**.

©ISO 2000 - All rights reserved

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **named_unit** entity:

— transition_feature_life_cycle (See 5.2.4.32);
— transition_feature_on_part_boundary (See 5.2.4.33).

5.2.3.1.80 turned_knurl

A **turned_knurl** is a type of **feature_definition** that is the representation of a diamond, diagonal, or straight shape that is applied to all or a portion of a **shape_aspect**. See ARM definition for Turned_knurl in paragraph 4.2.235 for more information.

EXPRESS specification:

```
*)
  ENTITY turned_knurl
    SUBTYPE OF (feature_definition);
    WHERE
  wr1 : (SELF\characterized_object.description IN
         ['diamond','diagonal','straight']);

  WR2:  SIZEOF( QUERY( pd <* USEDIN( SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        SIZEOF( QUERY( pdr <* USEDIN( pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        ('FEATURE_BASED_PROCESS_PLANNING.'+
        'SHAPE_REPRESENTATION_WITH_PARAMETERS' IN
        TYPEOF(pdr.used_representation)) AND
        ({6 <= SIZEOF(pdr.used_representation.items) <= 9}) )) = 1 )) = 1;

  wr3 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
        (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
        IN TYPEOF(it)) AND
        ('FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE'
        IN TYPEOF(it\measure_with_unit.value_component)) AND
        (it.name = 'number of teeth')) )) <= 1)) )) = 0)) )) = 0);

  wr4 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.'
        + 'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
        ((SIZEOF([
```

 ©ISO 2000 - All rights reserved

```
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
      * TYPEOF(it)) = 2) AND (it.name = 'major diameter'))))= 1)) ))
      = 0)) )) = 0);

 wr5 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.'
       + 'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS')
       IN TYPEOF(pdr.used_representation)) ) |
       (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
       ((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
       * TYPEOF(it)) = 2) AND (it.name = 'nominal diameter')) )) = 1)) ))
       = 0)) )) = 0);

wr6 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
      'DEFINITION') | (NOT (SIZEOF(QUERY ( impl_rep <*
      QUERY ( pdr <* USEDIN(pd,'FEATURE_BASED_PROCESS_PLANNING.'
      + 'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
      used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
      impl_rep.used_representation.items | ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
      * TYPEOF(it)) = 2) AND (it.name = 'tooth depth')) )) <= 1)) ))
      = 0)) )) = 0);

wr7 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
      TYPEOF(it)) = 2) AND (it.name = 'root fillet')) )) <= 1)) ))
      = 0)) )) = 0);

wr8 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
      (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
      'FEATURE_BASED_PROCESS_PLANNING.' +
      'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.' +
      'SHAPE_REPRESENTATION_WITH_PARAMETERS')
      IN TYPEOF(pdr.used_representation)) ) |
      (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
      ((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
```

©ISO 2000 - All rights reserved

```
        TYPEOF(it)) = 2) AND (it.name = 'diametral pitch')) )) = 1)) ))
        = 0)) )) = 0);

  wr9 : ((NOT (SELF\characterized_object.description IN
        ['diamond','diagonal'])) OR (
        SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
        | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
        pd,'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
        used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
        impl_rep.used_representation.items | ((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
        * TYPEOF(it)) = 2) AND (it.name = 'helix angle')) )) = 1)) ))
        = 0)) )) = 0));

  wr10: ((NOT (SELF\characterized_object.description = 'diagonal')) OR
        (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS')
        IN TYPEOF(pdr.used_representation)) ) |
        (NOT (SIZEOF(QUERY ( it <* impl_rep.used_representation.items |
        (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
        IN TYPEOF(it)) AND (it.name = 'helix hand')) )) = 1)) ))
        = 0)) )) = 0));

  wr11: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
        ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
        IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
        ((sa_occ.description = 'partial area occurrence') AND
        (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.' +
        'RELATED_SHAPE_ASPECT') |
        ((sar.description = 'applied area usage') AND
        (('FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_DEFINING_RELATIONSHIP') IN TYPEOF(sar))) ) |
        ('FEATURE_BASED_PROCESS_PLANNING.APPLIED_AREA'
        IN TYPEOF(sdr.relating_shape_aspect)) )) = 1)) )) = 1)) )) = 0);

 WR12: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
       IN TYPEOF(pd)) ) |
       (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') |
       ((SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
       + 'RELATED_SHAPE_ASPECT') |
       ((sar.description = 'applied shape') AND
       (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP')
       IN TYPEOF(sar))) ) |
       ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT'
```

 ©ISO 2000 - All rights reserved

```
      IN TYPEOF(sdr.relating_shape_aspect)) )) = 1)) )) = 1)) )) = 0);

  END_ENTITY; -- turned_knurl
(*
```

Formal propositions:

WR1: The **description** for the **turned_knurl** shall be either 'diamond', 'diagonal', or 'straight'.

WR2: The implicit representation of the **thread** shall contain between six and nine **representation_item**s.

WR3: At most one **representation_item** used for the implicit representation of the **turned_knurl** shall be of type **measure_representation_item** with a **value_component** of type **count_measure** and a **name** of 'number of teeth'.

WR4: Exactly one **representation_item** used for the implicit representation of the **turned_knurl** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'major diameter'.

WR5: Exactly one **representation_item** used for the implicit representation of the **turned_knurl** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'nominal diameter'.

WR6: At most one **representation_item** used for the implicit representation of the **turned_knurl** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'tooth depth'.

WR7: At most one **representation_item** used for the implicit representation of the **turned_knurl** shall be of type **measure_representation_item** and **plane_angle_measure_with_unit** with a **name** of 'root fillet'.

WR8: Exactly one **representation_item** used for the implicit representation of the **turned_knurl** shall be of type **measure_representation_item** and **plane_angle_measure_with_unit** with a **name** of 'diametral pitch'.

WR9: If the **turned_knurl** has a **description** of either 'diagonal' or 'diamond', exactly one **representation_item** used for the implicit representation of the **turned_knurl** shall be of type **measure_representation_item** and **length_measure_with_unit** with a **name** of 'helix angle'.

WR10: If the **turned_knurl** has a **description** of 'diagonal' exactly one **representation_item** used for the implicit representation of the **turned_knurl** shall be of type **descriptive_representation_item** with a **name** of 'helix hand'.

WR11: The **turned_knurl** shall be the basis shape for at most one **shape_aspect** with a **description** of 'partial area occurrence' that is the **related_shape_aspect** in exactly one **shape_defining_relationship** with a **description** of 'applied area usage' and a **relating_shape_aspect** that is an **applied_area**.

WR12: Exactly one **turned_knurl** shall be the basis shape for exactly one **shape_aspect** that is the **related_shape_aspect** in at most one **shape_defining_relationship** with a description of 'applied shape' in which the **relating_shape_aspect** is a **shape_aspect**.

©ISO 2000 - All rights reserved

Informal propositions:

IP1: The location of the **turned_knurl** shall be at a distance from the end of a cylindrical face, and on the axis of the cylindrical face.

IP2: The Z direction of the **turned_knurl** shall be in the direction of the center axis of the cylindrical face.

5.2.3.1.81 vee_profile

A **vee_profile** is a type of **shape_aspect** that is the representation of two connected straight lines, with a location and a position. The **vee_profile** is located at the position where the two lines connect, the X direction intersects the angle between the two connected lines in the direction of volume removal, and the Y direction is orthogonal to the X direction. See ARM definition for Vee_profile in paragraph 4.2.236 for more information.

EXPRESS specification:

```
*)
ENTITY vee_profile
  SUBTYPE OF (shape_aspect);
WHERE
  WR1: 'FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
       IN TYPEOF (SELF.of_shape.definition);

  WR2: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation))) = 1))) = 0;

  WR3: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF (pdr.used_representation)) |
       (NOT (SIZEOF (impl_rep.used_representation.items) >= 3)
       AND  (SIZEOF (impl_rep.used_representation.items) <= 4))))
       = 0))) = 0;

  WR4: SIZEOF( QUERY( pd <* USEDIN( SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       SIZEOF( QUERY( pdr <* USEDIN( pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.'+
       'SHAPE_REPRESENTATION_WITH_PARAMETERS'
       IN TYPEOF(pdr.used_representation)) AND
       (SIZEOF( QUERY( srwp_i <* pdr.used_representation.items |
       (srwp_i.name = 'orientation') OR
```

 ©ISO 2000 - All rights reserved

```
     (srwp_i.name = 'profile angle') OR
     (srwp_i.name = 'tilt angle') OR
     (srwp_i.name = 'profile radius') ))
     = SIZEOF(pdr.used_representation.items)) )) = 1 )) = 1;

WR5: SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation)) |
     NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
     ('FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF (it)) AND
     (it.name = 'orientation'))) = 1))) = 0))) = 0;

WR6: SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation)) |
     NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
     ('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
     IN TYPEOF (it)) AND
     ('FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'
     IN TYPEOF (it\measure_with_unit.value_component)) AND
     (it.name = 'profile radius'))) <= 1))) = 0))) = 0;

WR7: SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation)) |
     NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
     (SIZEOF
     (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
     'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT'] *
     TYPEOF (it)) = 2) AND
     (it.name = 'profile angle'))) = 1))) = 0))) = 0;

WR8: SIZEOF (QUERY (pd <* USEDIN (SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
     NOT (SIZEOF (QUERY (impl_rep <* QUERY (pdr <* USEDIN (pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS'
     IN TYPEOF (pdr.used_representation)) |
     NOT (SIZEOF (QUERY (it <* impl_rep.used_representation.items |
     (SIZEOF
     (['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
     'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT'] *
     TYPEOF (it)) = 2) AND
     (it.name = 'tilt angle'))) = 1))) = 0))) = 0;
```

©ISO 2000 - All rights reserved

```
  WR9: SIZEOF (QUERY (pd <* USEDIN (SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       ('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
       IN TYPEOF (pdr.used_representation))AND
       (pdr.used_representation.name = 'profile limit'))) <= 1))) = 0;

END_ENTITY;
(*
```

Formal propositions:

WR1: The **vee_profile** shall be an aspect of the shape of a **feature_component_definition**.

WR2: The **vee_profile** shall have exactly one implicit representation defined by a relationship to a **shape_representation_with_parameters**.

WR3: The **shape_representation_with_parameters** that represents the **vee_profile** shall contain at least three and at most four **representation_item**s in its set of **items**.

WR4: The implicit representation of an **vee_profile** shall contain only **representation_-item**s in its set of **items** with a **name** of 'orientation', 'profile angle', 'tilt angle', and 'profile radius'.

WR5: Exactly one **representation_item** used for the implicit representation of a **vee_profile** shall be of type **placement** with a **name** of 'orientation'.

WR6: At most one **representation_item** used for the implicit representation of a **vee_profile** shall be of type **measure_representation_item** with a **value_component** of type **length_measure_with_unit** and a **name** of 'profile radius'.

WR7: Exactly one **representation_item** used for the implicit representation of a **vee_profile** shall be of type **measure_representation_item** and **plane_angle_measure_with_unit** with a **name** of 'profile angle'.

WR8: Exactly one **representation_item** used for the implicit representation of a **vee_profile** shall be of type **measure_representation_item** and **plane_angle_measure_with_unit** with a **name** of 'tilt angle'.

WR9: The **vee_profile** shall have at most one **planar_shape_representation** with a **name** of 'profile limit'.

Informal propositions:

IP1: The location of the **vee_profile** shall be at the position where the two lines connect.

IP2: The **vee_profile** shall be defined with the tilt angle measured from the X-axis and the opening of the profile in the direction of the Y-axis.

 ©ISO 2000 - All rights reserved

5.2.3.2 Feature based process planning imported entity modifications

5.2.3.2.1 action_relationship

The base definition of the **action_relationship** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **action_relationship** entity:

— project_order_tracking_relationships (See 5.2.4.22).

5.2.3.2.2 action_request_status

The base definition of the **action_request_status** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **action_request_status** entity:

— dependent_instantiable_action_request_status (See 5.2.4.4).

5.2.3.2.3 action_status

The base definition of the **action_status** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **action_status** entity:

— dependent_instantiable_action_status (See 5.2.4.5).

5.2.3.2.4 application_context

The base definition of the **application_context** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **application_context** entity:

— application_context_requires_ap_definition (See 5.2.4.1).

©ISO 2000 - All rights reserved

5.2.3.2.5 application_protocol_definition

The base definition of the **application_protocol_definition** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **application_protocol_-definition** entity:

— application_context_requires_ap_definition (See 5.2.4.1).

5.2.3.2.6 approval

The base definition of the **approval** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rules defined in this part of ISO 10303 apply to the **approval** entity:

— approval_requires_approval_date_time (See 5.2.4.2);

— approval_requires_approval_person_organization (See 5.2.4.3).

5.2.3.2.7 approval_date_time

The base definition of the **approval_date_time** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **approval_date_time** entity:

— approval_requires_approval_date_time (See 5.2.4.2).

5.2.3.2.8 approval_person_organization

The base definition of the **approval_person_organization** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **approval_person_-organization** entity:

— approval_requires_approval_person_organization (See 5.2.4.3).

 ©ISO 2000 - All rights reserved

5.2.3.2.9 approval_status

The base definition of the **approval_status** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rules defined in this part of ISO 10303 apply to the **approval_status** entity:

— dependent_instantiable_approval_status (See 5.2.4.5);

— restrict_approval_status (See 5.2.4.26).

5.2.3.2.10 characterized_object

The base definition of the **characterized_object** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **characterized_object** entity:

— subtype_mandatory_characterized_object (See 5.2.4.31).

5.2.3.2.11 date

The base definition of the **date** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **date** entity:

— dependent_instantiable_date (See 5.2.4.6).

5.2.3.2.12 description_attribute

The base definition of the **description_attribute** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **description_attribute** entity:

— non_instantiable_description_attribute (See 5.2.4.16).

©ISO 2000 - All rights reserved

5.2.3.2.13 dimensional_location

The base definition of the **dimensional_location** entity is given in ISO 10303-47. The following modifications apply to this part of ISO 10303.

Attribute definitions:

relating_shape aspect: the origin of the direction of measurement of the dimension.

related_shape_aspect: the target of the direction of measurement of the dimension.

5.2.3.2.14 directed_action

The base definition of the **directed_action** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **directed_action** entity:

— project_order_tracking_relationships (See 5.2.4.22).

5.2.3.2.15 geometric_tolerance

The base definition of the **geometric_tolerance** entity is given in ISO 10303-47. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **geometric_tolerance** entity:

— geometric_tolerance_subtype_exclusiveness (See 5.2.4.13).

5.2.3.2.16 make_from_usage_option

The base definition of the **make_from_usage_option** entity is given in ISO 10303-44. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **make_from_usage_option** entity:

— material_is_specified_for_part (See 5.2.4.15).

 ©ISO 2000 - All rights reserved

5.2.3.2.17 named_unit

The base definition of the **named_unit** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **named_unit** entity:

— dependent_instantiable_named_unit (See 5.2.4.7).

5.2.3.2.18 object_role

The base definition of the **object_role** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **object_role** entity:

— non_instantiable_object_role (See 5.2.4.17).

5.2.3.2.19 precision_qualifier

The base definition of the **precision_qualifier** entity is given in ISO 10303-45. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **precision_qualifier** entity:

— dependent_instantiable_precision_qualifier (See 5.2.4.8).

5.2.3.2.20 product

The base definition of the **product** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **product** entity:

— product_requires_version (See 5.2.4.21).

5.2.3.2.21 product_definition

The base definition of the **product_definition** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

©ISO 2000 - All rights reserved

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **product_definition** entity:

— material_is_specified_for_part (See 5.2.4.15).

5.2.3.2.22 product_definition_formation

The base definition of the **product_definition_formation** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rules defined in this part of ISO 10303 apply to the **product_definition_formation** entity:

— part_requires_project_order (See 5.2.4.19);

— part_to_approval (See 5.2.4.20);

— product_requires_version (See 5.2.4.21);

— product_version_requires_security_classification (See 5.2.4.22).

5.2.3.2.23 representation

The base definition of the **representation** entity is given in ISO 10303-43. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rules defined in this part of ISO 10303 apply to the **representation** entity:

— representation_subtype_exclusiveness (See 5.2.4.25).

5.2.3.2.24 role_association

The base definition of the **role_association** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **role_association** entity:

— non_instantiable_role_association (See 5.2.4.18).

 ©ISO 2000 - All rights reserved

5.2.3.2.25 security_classification_level

The base definition of the **security_classification_level** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rules defined in this part of ISO 10303 applies to the **security_classification_level** entity:

— dependent_instantiable_security_classification_level (See 5.2.4.11),

— restrict_security_classification_level (see 5.2.4.24).

5.2.3.2.26 shape_aspect

The base definition of the **shape_aspect** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **shape_aspect** entity:

— shape_aspect_subtype_exclusiveness (See 5.2.4.29).

5.2.3.2.27 shape_aspect_relationship

The base definition of the **shape_aspect_relationship** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **shape_aspect_relationship** entity:

— shape_aspect_relationship_subtype_exclusiveness (See 5.2.4.28).

5.2.3.2.28 shape_representation

The base definition of the **shape_representation** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

©ISO 2000 - All rights reserved

Associated global rules:

The following global rules defined in this part of ISO 10303 apply to the **shape_representation** entity:

— dependent_instantiable_shape_representation (See 5.2.4.12);

— shape_representation_subtype_exclusiveness (See 5.2.4.30).

5.2.3.2.29 type_qualifier

The base definition of the **type_qualifier** entity is given in ISO 10303-45. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **type_qualifier** entity:

— dependent_instantiable_type_qualifier (See 5.2.4.9).

5.2.3.2.30 uncertainty_qualifier

The base definition of the **uncertainty_qualifier** entity is given in ISO 10303-45. The following modifications apply to this part of ISO 10303.

Associated global rules:

The following global rule defined in this part of ISO 10303 applies to the **uncertainty_qualifier** entity:

— dependent_instantiable_uncertainty_qualifier (See 5.2.4.10).

5.2.4 Feature based process planning rules

5.2.4.1 application_context_requires_ap_definition

The **application_context_requires_ap_definition** rule specifies that each instance of **application_-context** shall be referenced by exactly one **application_protocol_definition** that specifies this part of ISO 10303.

EXPRESS specification:

```
*)
RULE application_context_requires_ap_definition FOR
  (application_context, application_protocol_definition);
WHERE
  WR1: SIZEOF (QUERY (ac <* application_context |
       NOT (SIZEOF (QUERY (apd <* application_protocol_definition |
       (ac :=: apd.application)
       AND
       (apd.application_interpreted_model_schema_name =
```

 ©ISO 2000 - All rights reserved

```
       'feature_based_process_planning'))) = 1 ))) = 0;
END_RULE;
(*
```

Argument definitions:

application_context: the set of all instances of **application_context** entities.

application_protocol_definition: the set of all instances of **application_protocol_definition** entities.

Formal propositions:

WR1: For each instance of **application_context**, there shall be exactly one instance of **application_-protocol_definition** that references the instance of **application_context** as its **application** with a value of 'feature_based_process_planning' as its **application_interpreted_model_schema_name**.

5.2.4.2 approval_requires_approval_date_time

The **approval_requires_approval_date_time** rule specifies that each instance of **approval** shall be referenced by exactly one **approval_date_time**. This rule enforces the requirement for every approval to have a date on which the approval obtained its specified status.

EXPRESS specification:

```
*)
RULE approval_requires_approval_date_time FOR (approval,
  approval_date_time);
WHERE
  WR1: SIZEOF (QUERY (app <* approval |
       NOT (SIZEOF (QUERY (adt <* approval_date_time |
       app :=: adt.dated_approval )) = 1 ))) = 0;
END_RULE;
(*
```

Argument definitions:

approval: the set of all instances of **approval** entities.

approval_date_time: the set of all instances of **approval_date_time** entities.

Formal propositions:

WR1: For each instance of **approval**, there shall be exactly one instance of **approval_date_time** which contains the instance of **approval** as its **dated_approval** attribute.

5.2.4.3 approval_requires_approval_person_organization

The **approval_requires_approval_person_organization** specifies that each instance of **approval** shall have at least one **approval_person_organization** referencing it. This rule enforces the requirement for an approval to be authorized by one or more people within their organizations.

©ISO 2000 - All rights reserved

EXPRESS specification:

```
*)
RULE approval_requires_approval_person_organization FOR
  (approval, approval_person_organization);
WHERE
  WR1: SIZEOF (QUERY (app <* approval |
       NOT (SIZEOF (QUERY (apo <* approval_person_organization |
       app :=: apo.authorized_approval )) >= 1 ))) = 0;
END_RULE;
(*
```

Argument definitions:

approval: the set of all instances of **approval** entities.

approval_person_organization: the set of all instances of **approval_person_organization** entities.

Formal propositions:

WR1: For each instance of **approval**, there shall be one or more instances of **approval_person_-organization** which contains the instance of **approval** as its **authorized_approval** attribute.

5.2.4.4 dependent_instantiable_action_request_status

The **dependent_instantiable_action_request_status** rule specifies that all instances of **action_-request_status** are dependent on the usage to define another entity.

EXPRESS specification:

```
*)
RULE dependent_instantiable_action_request_status FOR
(action_request_status);
WHERE
  WR1: SIZEOF (QUERY (arst <* action_request_status |
       NOT (SIZEOF (USEDIN (arst, '')) >= 1))) = 0;
END_RULE;
(*
```

Argument definition:

action_request_status: the set of all instances of **action_request_status**.

Formal proposition:

WR1: For each instance of **action_request_status**, there shall be a reference to the **action_request_-status** instance from an attribute of another entity.

5.2.4.5 dependent_instantiable_approval_status

The **dependent_instantiable_approval_status** rule specifies that all instances of **approval_status** are dependent on the usage to define another entity.

 ©ISO 2000 - All rights reserved

EXPRESS specification:

```
*)
RULE dependent_instantiable_approval_status FOR (approval_status);
WHERE
  WR1: SIZEOF (QUERY (ast <* approval_status |
       NOT (SIZEOF (USEDIN (ast, '')) >= 1))) = 0;
END_RULE;
(*
```

Argument definition:

approval_status: the set of all instances of **approval_status**.

Formal proposition:

WR1: For each instance of **approval_status**, there shall be a reference to the **approval_status** instance from an attribute of another entity.

5.2.4.6 dependent_instantiable_date

The **dependent_instantiable_date** rule specifies that all instances of **date** are dependent on the usage to define another entity.

EXPRESS specification:

```
*)
RULE dependent_instantiable_date FOR (date);
WHERE
  WR1: SIZEOF (QUERY (dt <* date |NOT(SIZEOF (USEDIN (dt,'')) >= 1))) = 0;
END_RULE;
(*
```

Argument definition:

date: the set of all instances of **date**.

Formal proposition:

WR1: For each instance of **date**, there shall be a reference to the **date** instance from an attribute of another entity.

5.2.4.7 dependent_instantiable_named_unit

The **dependent_instantiable_named_unit** rule specifies that all instances of **named_unit** are dependent on the usage to define another entity.

EXPRESS specification:

```
*)
RULE dependent_instantiable_named_unit FOR (named_unit);
WHERE
```

©ISO 2000 - All rights reserved

```
  WR1: SIZEOF (QUERY (nu <* named_unit |
       NOT (SIZEOF (USEDIN (nu, '')) >= 1))) = 0;
END_RULE;
(*
```

Argument definition:

named_unit: the set of all instances of **named_unit**.

Formal proposition:

WR1: For each instance of **named_unit**, there shall be a reference to the **named_unit** instance from an attribute of another entity.

5.2.4.8 dependent_instantiable_precision_qualifier

The **dependent_instantiable_precision_qualifier** rule specifies that all instances of **precision_qualifier** are dependent on the usage to define another entity.

EXPRESS specification:

```
*)
RULE dependent_instantiable_precision_qualifier FOR (precision_qualifier);
WHERE
  WR1: SIZEOF (QUERY (pq <* precision_qualifier |
       NOT (SIZEOF (USEDIN (pq, '')) >= 1))) = 0;
END_RULE;
(*
```

Argument definition:

precision_qualifier: the set of all instances of **precision_qualifier**.

Formal proposition:

WR1: For each instance of **precision_qualifier**, there shall be a reference to the **precision_qualifier** instance from an attribute of another entity.

5.2.4.9 dependent_instantiable_type_qualifier

The **dependent_instantiable_type_qualifier** rule specifies that all instances of **type_qualifier** are dependent on the usage to define another entity.

EXPRESS specification:

```
*)
RULE dependent_instantiable_type_qualifier FOR (type_qualifier);
WHERE
  WR1: SIZEOF (QUERY (tq <* type_qualifier |
       NOT (SIZEOF (USEDIN (tq, '')) >= 1))) = 0;
END_RULE;
(*
```

 ©ISO 2000 - All rights reserved

Argument definition:

type_qualifier: the set of all instances of **type_qualifier**.

Formal proposition:

WR1: For each instance of **type_qualifier**, there shall be a reference to the **type_qualifier** instance from an attribute of another entity.

5.2.4.10 dependent_instantiable_uncertainty_qualifier

The **dependent_instantiable_uncertainty_qualifier** rule specifies that all instances of **uncertainty_qualifier** are dependent on the usage to define another entity.

EXPRESS specification:

```
*)
RULE dependent_instantiable_uncertainty_qualifier FOR
(uncertainty_qualifier);
WHERE
  WR1: SIZEOF (QUERY (uq <* uncertainty_qualifier |
       NOT (SIZEOF (USEDIN (uq, '')) >= 1))) = 0;
END_RULE;
(*
```

Argument definition:

uncertainty_qualifier: the set of all instances of **uncertainty_qualifier**.

Formal proposition:

WR1: For each instance of **uncertainty_qualifier**, there shall be a reference to the **uncertainty_qualifier** instance from an attribute of another entity.

5.2.4.11 dependent_instantiable_security_classification_level

The **dependent_instantiable_security_classification_level** rule specifies that all instances of **security_classification_level** are dependent on the usage to define another entity.

EXPRESS specification:

```
*)
RULE dependent_instantiable_security_classification_level FOR
  (security_classification_level);
WHERE
  WR1: SIZEOF (QUERY (scl <* security_classification_level |
       NOT (SIZEOF (USEDIN (scl, '')) >= 1))) = 0;
END_RULE;
(*
```

©ISO 2000 - All rights reserved

Argument definition:

security_classification_level: the set of all instances of **security_classification_level**.

Formal proposition:

WR1: For each instance of **security_classification_level**, there shall be a reference to the **security_-classification_level** instance from an attribute of another entity.

5.2.4.12 dependent_instantiable_shape_representation

The **dependent_instantiable_shape_representation** rule specifies that all instances of **shape_-representation** are dependent on the usage to define another entity.

EXPRESS specification:

```
*)
RULE dependent_instantiable_shape_representation FOR
(shape_representation);
WHERE
  WR1: SIZEOF (QUERY (sr <* shape_representation |
       NOT (SIZEOF (USEDIN (sr, '')) >= 1))) = 0;
END_RULE;
(*
```

Argument definition:

shape_representation: the set of all instances of **shape_representation**.

Formal proposition:

WR1: For each instance of **shape_representation**, there shall be a reference to the **shape_-representation** instance from an attribute of another entity.

5.2.4.13 geometric_tolerance_subtype_exclusiveness

The **geometric_tolerance_subtype_exclusiveness** rule specifies that an instance of the subtypes of a **geometric_tolerance** shall be only one of **angularity_tolerance**, **circular_runout_tolerance**, **concentricity_tolerance**, **cylindricity_tolerance**, **flatness_tolerance**, **line_profile_tolerance**, **parallelism_tolerance**, **perpendicularity_tolerance**, **position_tolerance**, **roundness_tolerance**, **straightness_tolerance**, **surface_profile_tolerance**, **symmetry_tolerance**, or **total_runout_tolerance**.

EXPRESS specification:

```
*)
RULE geometric_tolerance_subtype_exclusiveness FOR (geometric_tolerance);
WHERE
  WR1: SIZEOF (QUERY (gt <* geometric_tolerance |
       NOT (SIZEOF (TYPEOF (gt) *
       ['FEATURE_BASED_PROCESS_PLANNING.ANGULARITY_TOLERANCE',
       'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_RUNOUT_TOLERANCE',
       'FEATURE_BASED_PROCESS_PLANNING.CONCENTRICITY_TOLERANCE',
```

 ©ISO 2000 - All rights reserved

```
        'FEATURE_BASED_PROCESS_PLANNING.CYLINDRICITY_TOLERANCE',
        'FEATURE_BASED_PROCESS_PLANNING.FLATNESS_TOLERANCE',
        'FEATURE_BASED_PROCESS_PLANNING.LINE_PROFILE_TOLERANCE',
        'FEATURE_BASED_PROCESS_PLANNING.PARALLELISM_TOLERANCE',
        'FEATURE_BASED_PROCESS_PLANNING.PERPENDICULARITY_TOLERANCE',
        'FEATURE_BASED_PROCESS_PLANNING.POSITION_TOLERANCE',
        'FEATURE_BASED_PROCESS_PLANNING.ROUNDNESS_TOLERANCE',
        'FEATURE_BASED_PROCESS_PLANNING.STRAIGHTNESS_TOLERANCE',
        'FEATURE_BASED_PROCESS_PLANNING.SURFACE_PROFILE_TOLERANCE',
        'FEATURE_BASED_PROCESS_PLANNING.SYMMETRY_TOLERANCE',
        'FEATURE_BASED_PROCESS_PLANNING.TOTAL_RUNOUT_TOLERANCE'])
        <= 2))) = 0;
END_RULE;
(*
```

Argument definitions:

geometric_tolerance: the set of all instances of **geometric_tolerance** entities.

Formal propositions:

WR1: Each instance of the subtypes of **geometric_tolerance** shall be one of **angularity_tolerance**, **circular_runout_tolerance**, **concentricity_tolerance**, **cylindricity_tolerance**, **flatness_tolerance**, **line_profile_tolerance**, **parallelism_tolerance**, **perpendicularity_tolerance**, **position_tolerance**, **roundness_tolerance**, **straightness_tolerance**, **surface_profile_tolerance**, **symmetry_tolerance**, or **total_runout_tolerance**.

5.2.4.14 machining_feature_life_cycle

The **machining_feature_life_cycle** rule specifies that each instance of **instanced_feature** shall be defined for the manufacturing planning stage of the part on which it is specified.

EXPRESS specification:

```
*)
 RULE machining_feature_life_cycle FOR
 (instanced_feature);
WHERE
  WR1: SIZEOF (QUERY (mf <* instanced_feature |
       NOT (mf.of_shape.definition.frame_of_reference.life_cycle_stage =
       'manufacturing planning'))) = 0;
END_RULE;
(*
```

Argument definitions:

instanced_feature: the set of all instances of **instanced_feature** entities.

Formal propositions:

WR1: For each instance of **instanced_feature**, the **life_cycle_stage** of the **product_definition** for which it is defined has a value of 'manufacturing planning'.

©ISO 2000 - All rights reserved

5.2.4.15 material_is_specified_for_part

The **material_is_specified_for_part** rule specifies that every **product_definition** that is not designated a material shall be related to a material designated **product_definition** through the **make_from_usage_-option**.

EXPRESS specification:

```
*)
RULE material_is_specified_for_part FOR (product_definition,
  make_from_usage_option);
WHERE
  WR1: SIZEOF (QUERY (nmpd <* QUERY (pd <* product_definition |
       SIZEOF (USEDIN (pd, 'FEATURE_BASED_PROCESS_PLANNING.' +
       'MATERIAL_DESIGNATION.DEFINITIONS')) = 0) |
       NOT (SIZEOF (QUERY (mfuo <* make_from_usage_option |
       NOT (nmpd :=: mfuo.relating_product_definition))) >= 1))) = 0;
END_RULE;
(*
```

Argument definitions:

product_definition: the set of all instances of **product_definition** entities.

make_from_usage_option: the set of all instances of **make_from_usage_option** entities.

Formal propositions:

WR1: For each instance of **product_definition** that is not designated as a material through a reference by the **definitions** attribute of the **material_designation**, there shall be at least one instance of **make_-from_usage_option** in which the non-material **product_definition** is the **relating_product_definition**.

5.2.4.16 non_instantiable_description_attribute

The **non_instantiable_description_attribute** rule specifies that no instances of **description_attribute** are making a reference to another entity.

EXPRESS specification:

```
*)
  RULE non_instantiable_description_attribute FOR (
             description_attribute);
  WHERE
    wr1: (SIZEOF(QUERY ( da <* description_attribute | (NOT (SIZEOF(
             USEDIN(da,'')) = 1)) )) = 0);
  END_RULE; -- non_description_attribute
(*
```

Argument definition:

description_attribute: the set of all instances of **description_attribute**.

 ©ISO 2000 - All rights reserved

5.2.4.17 non_instantiable_object_role

The **non_instantiable_object_role** rule specifies that no instances of **object_role** are making a reference to another entity.

EXPRESS specification:

```
*)
  RULE non_instantiable_object_role FOR (
             object_role);

  WHERE
    wr1: (SIZEOF(QUERY ( da <* object_role | (NOT (SIZEOF(
             USEDIN(da,'')) = 1)) )) = 0);
  END_RULE; --  non_object_role
(*
```

Argument definition:

object_role: the set of all instances of **object_role**.

5.2.4.18 non_instantiable_role_association

The **non_instantiable_role_association** rule specifies that no instances of **role_association** are making a reference to another entity.

EXPRESS specification:

```
*)
  RULE non_instantiable_role_association FOR (
             role_association);
  WHERE
    wr1: (SIZEOF(QUERY ( da <* role_association | (NOT (SIZEOF(
             USEDIN(da,'')) = 1)) )) = 0);
  END_RULE; --   non_role_association
(*
```

Argument definition:

role_association: the set of all instances of **role_association**.

5.2.4.19 part_requires_project_order

The **part_requires_project_order** rule specifies that each instance of **product_definition_formation** shall be referenced by exactly one instance of **feature_based_pp_action_assignment** that assigns a project order to the part.

EXPRESS specification:

```
*)
RULE part_requires_project_order FOR (product_definition_formation,
  feature_based_pp_action_assignment);
```

©ISO 2000 - All rights reserved

```
WHERE
  WR1: SIZEOF (QUERY (pdf <* product_definition_formation |
       NOT (SIZEOF (QUERY (fbppaa <* feature_based_pp_action_assignment |
       (pdf IN fbppaa.items) AND
       (fbppaa.assigned_action.name = 'project order'))) = 1 ))) = 0;
END_RULE;
(*
```

Argument definitions:

product_definition_formation: the set of all instances of **product_definition_formation** entities.

feature_based_pp_action_assignment: the set of all instances of **feature_based_pp_action_-assignment** entities.

Formal propositions:

WR1: For each instance of **product_definition_formation**, there shall be exactly one instance of **feature_based_pp_action_assignment** that contains the instance of **product_definition_formation** in its set of **items** and references an **action** with a name of 'project order'.

5.2.4.20 part_to_approval

The **part_to_approval** rule specifies that each instance of **product_definition_formation** shall be referenced by at most one instance of **feature_based_pp_approval_assignment**.

EXPRESS specification:

```
*)
RULE part_to_approval FOR (product_definition_formation,
  feature_based_pp_approval_assignment);
WHERE
  WR1: SIZEOF (QUERY (pdf <* product_definition_formation |
       NOT (SIZEOF (QUERY (fbppa <* feature_based_pp_approval_assignment |
       pdf IN fbppa.items )) <= 1 ))) = 0;
END_RULE;
(*
```

Argument definitions:

product_definition_formation: the set of all instances of **product_definition_formation** entities.

feature_based_pp_approval_assignment: the set of all instances of **feature_based_pp_approval_-assignment** entities.

Formal propositions:

WR1: For each instance of **product_definition_formation**, there shall be exactly one instance of **feature_based_pp_approval_assignment** that contains the instance of **product_definition_formation** in its set of **items**.

 ©ISO 2000 - All rights reserved

5.2.4.21 product_requires_version

The **product_requires_version** rule specifies that each instance of **product** shall be referenced by at least one instance of **product_definition_formation**. This rule enforces the requirement for every product to have one or more versions.

EXPRESS specification:

```
*)
RULE product_requires_version FOR (product, product_definition_formation);
WHERE
  WR1: SIZEOF (QUERY (prod <* product |
       NOT (SIZEOF (QUERY (pdf <* product_definition_formation |
       prod :=: pdf.of_product )) >= 1 ))) = 0;
END_RULE;
(*
```

Argument definitions:

product: the set of all instances of **product** entities.

product_definition_formation: the set of all instances of **product_definition_formation** entities.

Formal propositions:

WR1: For each instance of **product**, there shall be one or more instances of **product_definition_-formation** that contains an **of_product** attribute value equal to that instance of **product**.

5.2.4.22 product_definition_formation_requires_security_classification

The **product_definition_formation_requires_security_classification** rule specifies that each instance of **product_definition_formation** shall be referenced by exactly one instance of **feature_based_pp_-security_classification_assignment**. This rule enforces the requirement for every version of a design to have a security classification.

EXPRESS specification:

```
*)
RULE product_definition_formation_requires_security_classification FOR
  (product_definition_formation,
  feature_based_pp_security_classification_assignment);
WHERE
  WR1: SIZEOF (QUERY (pdf <* product_definition_formation |
       NOT (SIZEOF (QUERY (fbppsca <*
       feature_based_pp_security_classification_assignment |
       pdf IN fbppsca.items )) = 1 ))) = 0;
END_RULE;
(*
```

©ISO 2000 - All rights reserved

Argument definitions:

product_definition_formation: the set of all instances of **product_definition_formation** entities.

feature_based_pp_security_classification_assignment: the set of all instances of **feature_based_pp_-security_classification_assignment** entities.

Formal propositions:

WR1: For each instance of **product_definition_formation**, there shall be exactly one instance of **feature_based_pp_security_classification_assignment** that contains the instance of **product_-definition_formation** in its set of **items**.

5.2.4.23 project_order_requires_approval

The **project_order_requires_approval** rule specifies that each instance of **directed_action** that references an **action** with a **name** of 'project order' shall be referenced by exactly one instance of **feature_based_pp_approval_assignment**. This rule enforces the requirement for every project order to have an approval.

EXPRESS specification:

```
*)
RULE project_order_requires_approval FOR
  (directed_action, feature_based_pp_approval_assignment);
WHERE
  WR1: SIZEOF (QUERY (po <* QUERY (da <* directed_action |
       da.name = 'project order') |
      NOT (SIZEOF (QUERY (fbppapp <* feature_based_pp_approval_assignment |
       po IN fbppapp.items )) = 1 ))) = 0;
END_RULE;
(*
```

Argument definitions:

directed_action: the set of all instances of **directed_action** entities.

feature_based_pp_approval_assignment: the set of all instances of **feature_based_pp_approval_-assignment** entities.

Formal propositions:

WR1: For each instance of **directed_action** there shall be exactly one instance of **feature_based_pp_-approval_assignment** that contains the instance of **directed_action** in its set of **items**.

 ©ISO 2000 - All rights reserved

5.2.4.24 project_order_tracking_relationships

The **project_order_tracking_relationships** rule specifies the relationship between a project order and the other types of orders within the process planning preparation. Each order shall be carried out under a project order, and an individual project order shall be defined for tracking at most one of each type of order within the project. The types of orders defined for this part of ISO 10303 are shop work order, resource acquisition order, digital technical data package work order, and pedigree creation order.

EXPRESS specification:

```
*)
RULE project_order_tracking_relationships FOR (directed_action,
  action_relationship);
WHERE
  WR1: SIZEOF (QUERY (da <* directed_action |
       (da.name IN ['shop work order', 'resource acquisition order',
       'digital technical data package work order',
       'pedigree creation order']) AND
       NOT (SIZEOF (QUERY (ar <* action_relationship |
       (da :=: ar.related_action) AND
       (('FEATURE_BASED_PROCESS_PLANNING.DIRECTED_ACTION' IN
       TYPEOF (ar.relating_action)) AND
       (ar.relating_action.name = 'project order')))) = 1))) = 0;
  WR2: SIZEOF (QUERY (da <* directed_action |
       (da.name = 'project order') AND
       NOT (SIZEOF (QUERY (ar <* action_relationship |
       (da :=: ar.relating_action) AND
       (('FEATURE_BASED_PROCESS_PLANNING.DIRECTED_ACTION' IN
       TYPEOF (ar.related_action)) AND
       (ar.related_action.name  = 'shop work order')))) <= 1))) = 0;
  WR3: SIZEOF (QUERY (da <* directed_action |
       (da.name = 'project order') AND
       NOT (SIZEOF (QUERY (ar <* action_relationship |
       (da :=: ar.relating_action) AND
       (('FEATURE_BASED_PROCESS_PLANNING.DIRECTED_ACTION' IN
       TYPEOF (ar.related_action)) AND
       (ar.related_action.name  = 'resource acquisition order'))))
       <= 1))) = 0;
  WR4: SIZEOF (QUERY (da <* directed_action |
       (da.name = 'project order') AND
       NOT (SIZEOF (QUERY (ar <* action_relationship |
       (da :=: ar.relating_action) AND
       (('FEATURE_BASED_PROCESS_PLANNING.DIRECTED_ACTION' IN
       TYPEOF (ar.related_action)) AND
       (ar.related_action.name  =
       'digital technical data package work order')))) <= 1))) = 0;
  WR5: SIZEOF (QUERY (da <* directed_action |
       (da.name = 'project order') AND
       NOT (SIZEOF (QUERY (ar <* action_relationship |
       (da :=: ar.relating_action) AND
       (('FEATURE_BASED_PROCESS_PLANNING.DIRECTED_ACTION' IN
       TYPEOF (ar.related_action)) AND
       (ar.related_action.name  = 'pedigree creation order'))))
       <= 1))) = 0;
  WR6: SIZEOF (QUERY (da <* directed_action |
       (da.name = 'customer order') AND
       NOT (SIZEOF (QUERY (ar <* action_relationship |
       (da :=: ar.related_action) AND
```

©ISO 2000 - All rights reserved

```
       (('FEATURE_BASED_PROCESS_PLANNING.DIRECTED_ACTION' IN
       TYPEOF (ar.relating_action)) AND
       (ar.relating_action.name = 'project order')))) >= 1))) = 0;
  WR7: SIZEOF (QUERY (da <* directed_action |
       (da.name = 'project order' ) AND
       NOT (SIZEOF (QUERY (ar <* action_relationship |
       (da :=: ar.relating_action) AND
       (('FEATURE_BASED_PROCESS_PLANNING.DIRECTED_ACTION' IN
       TYPEOF (ar.related_action)) AND
       (ar.related_action.name  = 'customer order'))))
       <= 1))) = 0;
END_RULE;
(*
```

Argument definitions:

directed_action: the set of all instances of **directed_action** entities.

action_relationship: the set of all instances of **action_relationship** entities.

Formal propositions:

WR1: For each instance of **directed_action** with a **name** of either 'shop work order', 'resource acquisition order', 'digital technical data package work order', or 'pedigree creation order', there shall be exactly one instance of **action_relationship** that references that **directed_action** as the **related_action**, and references an instance of **directed_action** with a **name** of 'project order' as the **relating_action**.

WR2: For each instance of **directed_action** with a **name** of 'project order', there shall be at most one instance of **action_relationship** that references that **directed_action** as the **relating_action**, and references an instance of **directed_action** with a **name** of 'shop work order'.

WR3: For each instance of **directed_action** with a **name** of 'project order', there shall be at most one instance of **action_relationship** that references that **directed_action** as the **relating_action**, and references an instance of **directed_action** with a **name** of 'resource acquisition order'.

WR4: For each instance of **directed_action** with a **name** of 'project order', there shall be at most one instance of **action_relationship** that references that **directed_action** as the **relating_action**, and references an instance of **directed_action** with a **name** of 'digital technical data package work order'.

WR5: For each instance of **directed_action** with a **name** of 'project order', there shall be at most one instance of **action_relationship** that references that **directed_action** as the **relating_action**, and references an instance of **directed_action** with a **name** of 'pedigree creation order'.

WR6: For each instance of **directed_action** with a **name** of 'customer order' there shall be at least one instance of **action_relationship** that references that **directed_action** as the **related_action**, and references an instance of **directed_action** with a **name** of 'project order' as the **relating_action**.

WR7: For each instance of **directed_action** with a **name** of 'project order', there shall be at most one instance of **action_relationship** that references that **directed_action** as the **relating_action**, and references an instance of **directed_action** with a **name** of 'customer order'.

 ©ISO 2000 - All rights reserved

5.2.4.25 representation_subtype_exclusiveness

The **representation_subtype_exclusiveness** rule specifies that an instance of the subtypes of a **representation** shall be only one of **shape_representation**, **definitional_representation**, **direction_-shape_representation**, **face_shape_representation**, **location_shape_representation**, **planar_shape_-representation** or **path_shape_representation**.

EXPRESS specification:

```
*)
RULE representation_subtype_exclusiveness FOR (representation);
WHERE
  WR1: SIZEOF (QUERY (rep <* representation |
       NOT (SIZEOF (TYPEOF (rep) *
       ['FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION',
       'FEATURE_BASED_PROCESS_PLANNING.DEFINITIONAL_REPRESENTATION',
       'FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION',
       'FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION',
       'FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION',
       'FEATURE_BASED_PROCESS_PLANNING.LOCATION_SHAPE_REPRESENTATION',
       'FEATURE_BASED_PROCESS_PLANNING.PATH_SHAPE_REPRESENTATION'])
       <= 2))) = 0;
END_RULE;
(*
```

Argument definitions:

representation: the set of all instances of **representation** entities.

Formal propositions:

WR1: Each instance of the subtypes of **representation** shall be one of the **shape_representation**, **-definitional_representation**, **direction_shape_representation**, **face_shape_representation**, **location_-shape_representation**, **planar_shape_representation** or **path_shape_representation** subtypes.

5.2.4.26 restrict_approval_status

The **restrict_approval_status** rule specifies that the only values of **approval_status** permitted shall be 'approved', 'not_yet_approved', 'disapproved', or 'withdrawn'.

EXPRESS specification:

```
*)
RULE restrict_approval_status FOR (approval_status);
WHERE
  WR1: SIZEOF (QUERY (ast <* approval_status |
       NOT (ast.name IN
       ['approved', 'not yet approved', 'disapproved', 'withdrawn']))) = 0;
END_RULE;
(*
```

©ISO 2000 - All rights reserved

Argument definitions:

approval_status: the set of all instances of **approval** entities.

Formal propositions:

WR1: For each instance of **approval**, the value of the **approval_status** attribute shall be either 'approved', 'not yet approved', 'disapproved', or 'withdrawn'.

Attribute value definitions:

approved: specifies that the required authorizations have been obtained for a particular role of approval for a piece of product data.

not_yet_approved: specifies that the required authorizations are being awaited for a particular role of approval for a piece of product data.

disapproved: specifies that the required authorizations have been denied for a particular role of approval for a piece of product data.

withdrawn: specifies that the required authorizations have been revoked for a particular role of approval for a piece of product data.

5.2.4.27 restrict_security_classification_level

The **restrict_security_classification_level** rule specifies the permitted levels of security. This rule enforces the requirement for the levels of security to be "unclassified", "classified", "proprietary", "confidential", "secret", or "top_secret".

EXPRESS specification

```
*)
RULE restrict_security_classification_level FOR
  (security_classification_level);
WHERE
  WR1: SIZEOF (QUERY (scl <* security_classification_level |
       NOT (scl.name IN
       ['unclassified', 'classified', 'proprietary','confidential',
'secret',
       'top_secret']))) = 0;
END_RULE;
(*
```

Argument definitions:

security_classification_level: identifies the set of all instances of **security_classification_level** entities.

Formal propositions:

WR1: For each instance of **security_classification_level**, the **name** attribute shall contain a value of "unclassified", "classified", "proprietary", "confidential", "secret", or "top_secret".

 ©ISO 2000 - All rights reserved

Attribute value definitions:

unclassified: identifies the classification level for which no security is necessary.

classified: identifies the classification level for which security is necessary, but the classification details are not given.

proprietary: identifies the classification level for which the disclosure of information about the part or the design of the part would risk an organization's market or competitive advantage.

confidential: identifies the classification level for which the disclosure of information about the part or the design of the part would cause damage to national or organizational security.

secret: identifies the classification level for which the disclosure of information about the part or the design of the part would cause serious damage to national or organizational security.

top_secret: identifies the classification level for which the disclosure of information about the part or the design of the part would cause exceptionally grave damage to national or organizational security.

5.2.4.28 shape_aspect_relationship_subtype_exclusiveness

The **shape_aspect_relationship_subtype_exclusiveness** rule specifies that an instance of the subtypes of a **shape_aspect_relationship** shall be only one of **dimensional_location**, **geometric_tolerance_-relationship**, **feature_component_relationship**, or **shape_defining_relationship**.

EXPRESS specification:

```
*)
RULE shape_aspect_relationship_subtype_exclusiveness FOR
  (shape_aspect_relationship);
WHERE
  WR1: SIZEOF (QUERY (sr <* shape_aspect_relationship |
       NOT (SIZEOF (TYPEOF (sr) *
       ['FEATURE_BASED_PROCESS_PLANNING.DIMENSIONAL_LOCATION',
       'FEATURE_BASED_PROCESS_PLANNING.GEOMETRIC_TOLERANCE_RELATIONSHIP',
       'FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP',
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'])
       <= 2))) = 0;
END_RULE;
(*
```

Argument definitions:

shape_aspect_relationship: the set of all instances of **shape_aspect_relationship** entities.

Formal propositions:

WR1: Each instance of the subtypes of **shape_aspect_relationship** shall be one of the **dimensional_-location**, **geometric_tolerance_relationship**, **feature_component_relationship**, or **shape_defining_-relationship**.

©ISO 2000 - All rights reserved

5.2.4.29 shape_aspect_subtype_exclusiveness

The **shape_aspect_subtype_exclusiveness** rule specifies that an instance of the subtypes of **shape_-aspect** shall be one of **path_feature_component**, **slot_end**, **pocket_bottom**, **boss_top**, **hole_bottom**, **applied_area**, **taper**, **chamfer_offset**, **circular_closed_profile**, **ngon_closed_profile**, **closed_path_-profile**, **square_u_profile**, **tee_profile**, **vee_profile**, **rib_top_floor**, **profile_floor**, **rectangular_-closed_profile**, **partial_circular_profile**, **rounded_u_profile**, **open_path_profile**, **instanced_feature**, **replicate_feature**, **transition_feature**, **datum**, **datum_feature**, or **datum_target**.

EXPRESS specification:

```
*)
RULE shape_aspect_subtype_exclusiveness FOR (shape_aspect);
  WHERE
    wr1: (SIZEOF(QUERY ( sr <* shape_aspect | (NOT (SIZEOF(TYPEOF(sr) * [
              'FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT',
              'FEATURE_BASED_PROCESS_PLANNING.SLOT_END',
              'FEATURE_BASED_PROCESS_PLANNING.POCKET_BOTTOM',
              'FEATURE_BASED_PROCESS_PLANNING.PROFILE_FLOOR',
              'FEATURE_BASED_PROCESS_PLANNING.RIB_TOP_FLOOR',
              'FEATURE_BASED_PROCESS_PLANNING.BOSS_TOP',
              'FEATURE_BASED_PROCESS_PLANNING.HOLE_BOTTOM',
              'FEATURE_BASED_PROCESS_PLANNING.APPLIED_AREA',
              'FEATURE_BASED_PROCESS_PLANNING.TAPER',
              'FEATURE_BASED_PROCESS_PLANNING.CHAMFER_OFFSET',
              'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE',
              'FEATURE_BASED_PROCESS_PLANNING.NGON_CLOSED_PROFILE',
              'FEATURE_BASED_PROCESS_PLANNING.CLOSED_PATH_PROFILE',
              'FEATURE_BASED_PROCESS_PLANNING.SQUARE_U_PROFILE',
              'FEATURE_BASED_PROCESS_PLANNING.LINEAR_PROFILE',
              'FEATURE_BASED_PROCESS_PLANNING.TEE_PROFILE',
              'FEATURE_BASED_PROCESS_PLANNING.VEE_PROFILE',
              'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE',
              'FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE',
              'FEATURE_BASED_PROCESS_PLANNING.ROUNDED_U_PROFILE',
              'FEATURE_BASED_PROCESS_PLANNING.OPEN_PATH_PROFILE',
              'FEATURE_BASED_PROCESS_PLANNING.INSTANCED_FEATURE',
              'FEATURE_BASED_PROCESS_PLANNING.REPLICATE_FEATURE',
              'FEATURE_BASED_PROCESS_PLANNING.TRANSITION_FEATURE',
              'FEATURE_BASED_PROCESS_PLANNING.DATUM',
              'FEATURE_BASED_PROCESS_PLANNING.DATUM_FEATURE',
              'FEATURE_BASED_PROCESS_PLANNING.DATUM_TARGET']) <= 2)) ))
              = 0);
  END_RULE; -- shape_aspect_subtype_exclusiveness
(*
```

Argument definitions:

shape_aspect: the set of all instances of **shape_aspect** entities.

 ©ISO 2000 - All rights reserved

Formal propositions:

WR1: Each instance of the subtypes of **shape_aspect** shall be one of **path_feature_component**, **slot_-end**, **pocket_bottom**, **boss_top**, **hole_bottom**, **applied_area**, **taper**, **chamfer_offset**, **circular_closed_-profile**, **ngon_closed_profile**, **closed_path_profile**, **square_u_profile**, **tee_profile**, **vee_profile**, **rib_top_floor**, **profile_floor**, **rectangular_closed_profile**, **partial_circular_profile**, **rounded_u_-profile**, **open_path_profile**, **instanced_feature**, **replicate_feature**, **transition_feature**, **datum**, **datum_feature**, or **datum_target**.

5.2.4.30 shape_representation_subtype_exclusiveness

The **shape_representation_subtype_exclusiveness** rule specifies that an instance of the subtypes of a **shape_representation** shall be only one of **advanced_brep_shape_representation**, **shape_-representation_with_parameters**, or **shape_dimension_representation**.

EXPRESS specification:

```
*)
RULE shape_representation_subtype_exclusiveness FOR (shape_representation);
WHERE
  WR1: SIZEOF (QUERY (sr <* shape_representation |
       NOT (SIZEOF (TYPEOF (sr) *
      ['FEATURE_BASED_PROCESS_PLANNING.ADVANCED_BREP_SHAPE_REPRESENTATION',
     'FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS',
     'FEATURE_BASED_PROCESS_PLANNING.SHAPE_DIMENSION_REPRESENTATION'])
       <= 2))) = 0;
END_RULE;
(*
```

Argument definitions:

shape_representation: the set of all instances of **shape_representation** entities.

Formal propositions:

WR1: Each instance of the subtypes of **shape_representation** shall be one of the **advanced_brep_-shape_representation**, **shape_representation_with_parameters**, or **shape_dimension_-representation**.

5.2.4.31 subtype_mandatory_characterized_object

The **subtype_mandatory_characterized_object** rule specifies the permitted usage of the **characterized_object**. The **characterized_object** entity shall be limited to its use in the definition of a **feature_definition**, **feature_component_definition**, or **ordered_part**.

EXPRESS specification:

```
*)

  RULE subtype_mandatory_characterized_object FOR (characterized_object);
  WHERE
```

©ISO 2000 - All rights reserved

```
    wr1: ((SIZEOF(QUERY ( csa <* characterized_object |
         (NOT (SIZEOF([
         'FEATURE_BASED_PROCESS_PLANNING.DOCUMENT_FILE',
         'FEATURE_BASED_PROCESS_PLANNING.FEATURE_DEFINITION',
         'FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION',
         'FEATURE_BASED_PROCESS_PLANNING.ORDERED_PART'] *
         TYPEOF(csa)) = 1))
         AND
         (NOT(SIZEOF(QUERY ( pd <* USEDIN(csa,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION') |
         ('FEATURE_BASED_PROCESS_PLANNING.MATERIAL_PROPERTY'
         IN TYPEOF(pd)) )) = 1))
         )) = 0) );
  END_RULE; -- subtype_mandatory_characterized_object
(*
```

Argument definitions:

characterized_object: the set of all instances of **characterized_object** entities.

Formal propositions:

WR1: Each instance of **characterized_object** shall be either a **feature_definition**, **feature_-component_definition**, or **ordered_part**, or referenced by the **definition** attribute of **material_property**.

5.2.4.32 transition_feature_life_cycle

The **transition_feature_life_cycle** rule specifies that each instance of **transition_feature** shall be defined for the manufacturing planning stage of the part on which it is specified.

EXPRESS specification:

```
*)
RULE transition_feature_life_cycle FOR
 (transition_feature);
WHERE
  WR1: SIZEOF (QUERY (tf <* transition_feature |
       NOT (tf.of_shape.definition.frame_of_reference.life_cycle_stage =
       'manufacturing planning'))) = 0;
END_RULE;
(*
```

Argument definitions:

transition_feature: the set of all instances of **transition_feature** entities.

Formal propositions:

WR1: For each instance of **transition_feature**, the **life_cycle_stage** of the **product_definition** for which it is defined has a value of 'manufacturing planning'.

 ©ISO 2000 - All rights reserved

5.2.4.33 transition_feature_on_part_boundary

The **transition_feature_on_part_boundary** rule specifies that each instance of **transition_feature** shall lie on the boundary of the part for which it is defined.

EXPRESS specification:

```
*)
RULE transition_feature_on_part_boundary FOR
 (transition_feature);
WHERE
  WR1: SIZEOF (QUERY (tf <* transition_feature |
       NOT (tf.product_definitional))) = 0;
END_RULE;
(*
```

Argument definitions:

transition_feature: the set of all instances of **transition_feature** entities.

Formal propositions:

WR1: For each instance of **transition_feature**, **product_definitional** shall have a value of true.

```
*)
END_SCHEMA;                                -- feature_based_process_planning
(*
```

6 Conformance requirements

Conformance to this part of ISO 10303 includes satisfying the requirements stated in this part, the requirements of the implementation methods supported, and the relevant requirements of the normative references.

An implementation shall support at least one of the following implementation methods: ISO 10303-21. Requirements with respect to implementation methods are specified in annex C.

The Protocol Information Conformance Statement (PICS) proforma lists the options or the combinations of options that may be included in the implementation. The PICS proforma is provided in annex D.

This part of ISO 10303 provides for only one option that may be supported by an implementation. This option shall all be supported by a single class of conformance which consist of all the units of functionality for this part of ISO 10303.

This conformance class is characterized as follows:

— Feature based process planning and shape represented by advanced b-rep.

NOTE - ISO/TS 10303-324 defines the abstract test suite to be used in the assessment of conformance. ISO 10303-32[1)] describes the conformance assessment process.

©ISO 2000 - All rights reserved

Annex A
(normative)

AIM EXPRESS expanded listing

The following EXPRESS is the expanded form of the short form schema given in 5.2. In the event of any discrepancy between the short form and this expanded listing, the expanded listing shall be used.

```
SCHEMA feature_based_process_planning;

  CONSTANT
  dummy_gri : geometric_representation_item := representation_item('') ||
                 geometric_representation_item();
  dummy_tri : topological_representation_item := representation_item('')
                 || topological_representation_item();
  END_CONSTANT;

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

  TYPE attribute_type = SELECT
    (label,
     text);
  END_TYPE; -- attribute_type

  TYPE axis2_placement = SELECT
    (axis2_placement_2d,
     axis2_placement_3d);
  END_TYPE; -- axis2_placement

  TYPE b_spline_curve_form = ENUMERATION OF
    (polyline_form,
     circular_arc,
     elliptic_arc,
     parabolic_arc,
     hyperbolic_arc,
     unspecified);
  END_TYPE; -- b_spline_curve_form

  TYPE b_spline_surface_form = ENUMERATION OF
    (plane_surf,
     cylindrical_surf,
     conical_surf,
     spherical_surf,
     toroidal_surf,
     surf_of_revolution,
     ruled_surf,
     generalised_cone,
     quadric_surf,
     surf_of_linear_extrusion,
     unspecified);
  END_TYPE; -- b_spline_surface_form

  TYPE boolean_operand = SELECT
```

 ©ISO 2000 - All rights reserved

```
  (solid_model);
END_TYPE; -- boolean_operand

TYPE characterized_definition = SELECT
  (characterized_object,
   characterized_product_definition,
   shape_definition);
END_TYPE; -- characterized_definition

TYPE characterized_material_property = SELECT
  (material_property_representation);
END_TYPE; -- characterized_material_property

TYPE characterized_product_definition = SELECT
  (product_definition,
   product_definition_relationship);
END_TYPE; -- characterized_product_definition

TYPE context_dependent_measure = REAL;
END_TYPE; -- context_dependent_measure

TYPE count_measure = NUMBER;
END_TYPE; -- count_measure

TYPE curve_on_surface = SELECT
  (pcurve,
   surface_curve);
END_TYPE; -- curve_on_surface

TYPE date_time_or_event_occurrence = SELECT
  (date_time_select);
END_TYPE; -- date_time_or_event_occurrence

TYPE date_time_select = SELECT
  (date);
END_TYPE; -- date_time_select

TYPE day_in_month_number = INTEGER;
WHERE
  wr1: ((1 <= SELF) AND (SELF <= 31));
END_TYPE; -- day_in_month_number

TYPE day_in_week_number = INTEGER;
WHERE
  wr1: ((1 <= SELF) AND (SELF <= 7));
END_TYPE; -- day_in_week_number

TYPE day_in_year_number = INTEGER;
WHERE
  wr1: ((1 <= SELF) AND (SELF <= 366));
END_TYPE; -- day_in_year_number

TYPE derived_property_select = SELECT
  (property_definition);
END_TYPE; -- derived_property_select

TYPE description_attribute_select = SELECT
  (action_request_solution,
   application_context,
   approval_role,
   date_role,
```

©ISO 2000 - All rights reserved

```
    external_source,
    organization_role,
    person_and_organization_role,
    person_and_organization,
    property_definition_representation,
    representation);
END_TYPE; -- description_attribute_select

TYPE descriptive_measure = STRING;
END_TYPE; -- descriptive_measure

TYPE dimension_count = INTEGER;
WHERE
  wr1: (SELF > 0);
END_TYPE; -- dimension_count

TYPE dimensional_characteristic = SELECT
  (dimensional_location,
   dimensional_size);
END_TYPE; -- dimensional_characteristic

TYPE document_reference_item = SELECT
  (action_method,
   externally_defined_feature_definition,
   directed_action,
   dimensional_characteristic_representation,
   property_definition);
END_TYPE; -- document_reference_item

TYPE feature_based_pp_action_item = SELECT
  (product_definition_formation);
END_TYPE; -- feature_based_pp_action_item

TYPE feature_based_pp_action_request_item = SELECT
  (product_definition_formation);
END_TYPE; -- feature_based_pp_action_request_item

TYPE feature_based_pp_approved_item = SELECT
  (directed_action,
   product_definition_formation);
END_TYPE; -- feature_based_pp_approved_item

TYPE feature_based_pp_classified_item = SELECT
  (product_definition_formation);
END_TYPE; -- feature_based_pp_classified_item

TYPE feature_based_pp_dated_item = SELECT
  (directed_action,
   document,
   versioned_action_request);
END_TYPE; -- feature_based_pp_dated_item

TYPE feature_based_pp_ordered_item = SELECT
  (product_definition_formation);
END_TYPE; -- feature_based_pp_ordered_item

TYPE feature_based_pp_organization_item = SELECT
  (action_directive,
   product_definition_formation);
END_TYPE; -- feature_based_pp_organization_item
```

 ©ISO 2000 - All rights reserved

```
TYPE feature_based_pp_person_and_organization_item = SELECT
  (action_directive,
   product_definition_formation);
END_TYPE; -- feature_based_pp_person_and_organization_item

TYPE founded_item_select = SELECT
  (founded_item,
   representation_item);
END_TYPE; -- founded_item_select

TYPE geometric_set_select = SELECT
  (point,
   curve,
   surface);
END_TYPE; -- geometric_set_select

TYPE group_item = SELECT
  (instanced_feature,
   replicate_feature,
   transition_feature);
END_TYPE; -- group_item

TYPE id_attribute_select = SELECT
  (action,
   application_context,
   representation);
END_TYPE; -- id_attribute_select

TYPE identifier = STRING;
END_TYPE; -- identifier

TYPE knot_type = ENUMERATION OF
  (uniform_knots,
   quasi_uniform_knots,
   piecewise_bezier_knots,
   unspecified);
END_TYPE; -- knot_type

TYPE label = STRING;
END_TYPE; -- label

TYPE length_measure = REAL;
END_TYPE; -- length_measure

TYPE limit_condition = ENUMERATION OF
  (maximum_material_condition,
   least_material_condition,
   regardless_of_feature_size);
END_TYPE; -- limit_condition

TYPE list_of_reversible_topology_item = LIST [0:?] OF
            reversible_topology_item;
END_TYPE; -- list_of_reversible_topology_item

TYPE measure_value = SELECT
  (length_measure,
   plane_angle_measure,
   ratio_measure,
   parameter_value,
   context_dependent_measure,
   descriptive_measure,
```

©ISO 2000 - All rights reserved

```
    positive_length_measure,
    positive_plane_angle_measure,
    count_measure);
END_TYPE; -- measure_value

TYPE month_in_year_number = INTEGER;
WHERE
  wr1: ((1 <= SELF) AND (SELF <= 12));
END_TYPE; -- month_in_year_number

TYPE name_attribute_select = SELECT
  (action_request_solution,
   derived_unit,
   person_and_organization,
   product_definition,
   property_definition_representation);
END_TYPE; -- name_attribute_select

TYPE parameter_value = REAL;
END_TYPE; -- parameter_value

TYPE pcurve_or_surface = SELECT
  (pcurve,
   surface);
END_TYPE; -- pcurve_or_surface

TYPE person_organization_select = SELECT
  (person,
   organization,
   person_and_organization);
END_TYPE; -- person_organization_select

TYPE plane_angle_measure = REAL;
END_TYPE; -- plane_angle_measure

TYPE positive_length_measure = length_measure;
WHERE
  wr1: (SELF > 0);
END_TYPE; -- positive_length_measure

TYPE positive_plane_angle_measure = plane_angle_measure;
WHERE
  wr1: (SELF > 0);
END_TYPE; -- positive_plane_angle_measure

TYPE preferred_surface_curve_representation = ENUMERATION OF
  (curve_3d,
   pcurve_s1,
   pcurve_s2);
END_TYPE; -- preferred_surface_curve_representation

TYPE product_or_formation_or_definition = SELECT
  (product,
   product_definition_formation,
   product_definition);
END_TYPE; -- product_or_formation_or_definition

TYPE ratio_measure = REAL;
END_TYPE; -- ratio_measure

TYPE represented_definition = SELECT
```

 ©ISO 2000 - All rights reserved

```
  (property_definition,
   property_definition_relationship,
   shape_aspect,
   shape_aspect_relationship);
END_TYPE; -- represented_definition

TYPE reversible_topology = SELECT
  (reversible_topology_item,
   list_of_reversible_topology_item,
   set_of_reversible_topology_item);
END_TYPE; -- reversible_topology

TYPE reversible_topology_item = SELECT
  (edge,
   path,
   face,
   face_bound,
   closed_shell,
   open_shell);
END_TYPE; -- reversible_topology_item

TYPE role_select = SELECT
  (action_assignment,
   action_request_assignment,
   approval_assignment,
   approval_date_time,
   document_reference,
   group_assignment,
   security_classification_assignment);
END_TYPE; -- role_select

TYPE set_of_reversible_topology_item = SET [0:?] OF
            reversible_topology_item;
END_TYPE; -- set_of_reversible_topology_item

TYPE shape_definition = SELECT
  (product_definition_shape,
   shape_aspect,
   shape_aspect_relationship);
END_TYPE; -- shape_definition

TYPE shell = SELECT
  (open_shell,
   closed_shell);
END_TYPE; -- shell

TYPE si_prefix = ENUMERATION OF
  (exa,
   peta,
   tera,
   giga,
   mega,
   kilo,
   hecto,
   deca,
   deci,
   centi,
   milli,
   micro,
   nano,
   pico,
```

©ISO 2000 - All rights reserved

```
    femto,
    atto);
END_TYPE; -- si_prefix

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; -- si_unit_name

TYPE source_item = SELECT
  (identifier);
END_TYPE; -- source_item

TYPE supported_item = SELECT
  (action_directive,
   action,
   action_method);
END_TYPE; -- supported_item

TYPE text = STRING;
END_TYPE; -- text

TYPE tolerance_method_definition = SELECT
  (tolerance_value,
   limits_and_fits);
END_TYPE; -- tolerance_method_definition

TYPE tolerance_select = SELECT
  (geometric_tolerance,
   plus_minus_tolerance);
END_TYPE; -- tolerance_select

TYPE transformation = SELECT
  (functionally_defined_transformation);
END_TYPE; -- transformation
```

 ©ISO 2000 - All rights reserved

```
TYPE transition_code = ENUMERATION OF
  (discontinuous,
   continuous,
   cont_same_gradient,
   cont_same_gradient_same_curvature);
END_TYPE; -- transition_code

TYPE trimming_select = SELECT
  (cartesian_point,
   parameter_value);
END_TYPE; -- trimming_select

TYPE unit = SELECT
  (named_unit,
   derived_unit);
END_TYPE; -- unit

TYPE value_qualifier = SELECT
  (precision_qualifier,
   type_qualifier,
   uncertainty_qualifier);
END_TYPE; -- value_qualifier

TYPE vector_or_direction = SELECT
  (vector,
   direction);
END_TYPE; -- vector_or_direction

TYPE week_in_year_number = INTEGER;
WHERE
  wr1: ((1 <= SELF) AND (SELF <= 53));
END_TYPE; -- week_in_year_number

TYPE year_number = INTEGER;
END_TYPE; -- year_number

ENTITY action;
    name          : label;
    description   : OPTIONAL text;
    chosen_method : action_method;
  DERIVE
    id : identifier := get_id_value(SELF);
  WHERE
    wr1: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
              'ID_ATTRIBUTE.IDENTIFIED_ITEM')) <= 1);
END_ENTITY; -- action

ENTITY action_assignment
  ABSTRACT SUPERTYPE;
    assigned_action : action;
  DERIVE
    role : object_role := get_role(SELF);
  WHERE
    wr1: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
              'ROLE_ASSOCIATION.ITEM_WITH_ROLE')) <= 1);
END_ENTITY; -- action_assignment

ENTITY action_directive;
    name        : label;
    description : OPTIONAL text;
    analysis    : text;
```

©ISO 2000 - All rights reserved

```
    comment     : text;
    requests    : SET [1:?] OF versioned_action_request;
END_ENTITY; -- action_directive

ENTITY action_method;
    name        : label;
    description : OPTIONAL text;
    consequence : text;
    purpose     : text;
END_ENTITY; -- action_method

ENTITY action_relationship;
    name            : label;
    description     : OPTIONAL text;
    relating_action : action;
    related_action  : action;
END_ENTITY; -- action_relationship

ENTITY action_request_assignment
  ABSTRACT SUPERTYPE;
    assigned_action_request : versioned_action_request;
  DERIVE
    role : object_role := get_role(SELF);
  WHERE
    wr1: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
              'ROLE_ASSOCIATION.ITEM_WITH_ROLE')) <= 1);
END_ENTITY; -- action_request_assignment

ENTITY action_request_solution;
    method  : action_method;
    request : versioned_action_request;
  DERIVE
    description : text := get_description_value(SELF);
    name        : label := get_name_value(SELF);
  WHERE
    wr1: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
              'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1);
    wr2: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
              'NAME_ATTRIBUTE.NAMED_ITEM')) <= 1);
END_ENTITY; -- action_request_solution

ENTITY action_request_status;
    status           : label;
    assigned_request : versioned_action_request;
END_ENTITY; -- action_request_status

ENTITY action_status;
    status          : label;
    assigned_action : executed_action;
END_ENTITY; -- action_status

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;
```

 ©ISO 2000 - All rights reserved

```
      telephone_number        : OPTIONAL label;
      electronic_mail_address : OPTIONAL label;
      telex_number            : OPTIONAL label;
    WHERE
      wr1: (EXISTS(internal_location) OR EXISTS(street_number) OR EXISTS(
               street) OR EXISTS(postal_box) OR EXISTS(town) OR EXISTS(
               region) OR EXISTS(postal_code) OR EXISTS(country) OR EXISTS(
               facsimile_number) OR EXISTS(telephone_number) OR EXISTS(
               electronic_mail_address) OR EXISTS(telex_number));
  END_ENTITY; -- address

  ENTITY advanced_brep_shape_representation
    SUBTYPE OF (shape_representation);
    WHERE
      wr1: (SIZEOF(QUERY ( it <* SELF.items | (NOT (SIZEOF([
               'FEATURE_BASED_PROCESS_PLANNING.MANIFOLD_SOLID_BREP',
               'FEATURE_BASED_PROCESS_PLANNING.FACETED_BREP',
               'FEATURE_BASED_PROCESS_PLANNING.MAPPED_ITEM',
               'FEATURE_BASED_PROCESS_PLANNING.AXIS2_PLACEMENT_3D'] *
               TYPEOF(it)) = 1)) )) = 0);
      wr2: (SIZEOF(QUERY ( it <* SELF.items | (SIZEOF([
               'FEATURE_BASED_PROCESS_PLANNING.MANIFOLD_SOLID_BREP',
               'FEATURE_BASED_PROCESS_PLANNING.MAPPED_ITEM'] * TYPEOF(it))
               = 1) )) > 0);
      wr3: (SIZEOF(QUERY ( msb <* QUERY ( it <* SELF.items | (
               'FEATURE_BASED_PROCESS_PLANNING.MANIFOLD_SOLID_BREP' IN
               TYPEOF(it)) ) | (NOT (SIZEOF(QUERY ( csh <* msb_shells(msb)
                | (NOT (SIZEOF(QUERY ( fcs <* csh\connected_face_set.
               cfs_faces | (NOT (
               'FEATURE_BASED_PROCESS_PLANNING.ADVANCED_FACE' IN
                TYPEOF(fcs))) ))
               = 0)) )) = 0)) )) = 0);
      wr4: (SIZEOF(QUERY ( msb <* QUERY ( it <* items | (
               'FEATURE_BASED_PROCESS_PLANNING.MANIFOLD_SOLID_BREP' IN
               TYPEOF(it)) ) | (
               'FEATURE_BASED_PROCESS_PLANNING.ORIENTED_CLOSED_SHELL' IN
               TYPEOF(msb\manifold_solid_brep.outer)) )) = 0);
      wr5: (SIZEOF(QUERY ( brv <* QUERY ( it <* items | (
               'FEATURE_BASED_PROCESS_PLANNING.BREP_WITH_VOIDS' IN TYPEOF(
               it)) ) | (NOT (SIZEOF(QUERY ( csh <* brv\brep_with_voids.
               voids | csh\oriented_closed_shell.orientation )) = 0)) )) =
               0);
      wr6: (SIZEOF(QUERY ( mi <* QUERY ( it <* items | (
            'FEATURE_BASED_PROCESS_PLANNING.MAPPED_ITEM' IN TYPEOF(it)) )
                | (NOT (
'FEATURE_BASED_PROCESS_PLANNING.ADVANCED_BREP_SHAPE_REPRESENTATION'
               IN TYPEOF(mi\mapped_item.mapping_source.
               mapped_representation))) )) = 0);
  END_ENTITY; -- advanced_brep_shape_representation

  ENTITY advanced_face
    SUBTYPE OF (face_surface);
    WHERE
      wr1 : (SIZEOF(['FEATURE_BASED_PROCESS_PLANNING.ELEMENTARY_SURFACE',
                'FEATURE_BASED_PROCESS_PLANNING.B_SPLINE_SURFACE',
                'FEATURE_BASED_PROCESS_PLANNING.SWEPT_SURFACE'] * TYPEOF(
                face_geometry)) = 1);
      wr2 : (SIZEOF(QUERY ( elp_fbnds <* QUERY ( bnds <* bounds | (
                'FEATURE_BASED_PROCESS_PLANNING.EDGE_LOOP' IN TYPEOF(bnds.
                bound)) ) | (NOT (SIZEOF(QUERY ( oe <* elp_fbnds.bound\path
                .edge_list | (NOT (
```

```
            'FEATURE_BASED_PROCESS_PLANNING.EDGE_CURVE' IN TYPEOF(oe\
            oriented_edge.edge_element))) )) = 0)) )) = 0);
wr3 : (SIZEOF(QUERY ( elp_fbnds <* QUERY ( bnds <* bounds | (
      'FEATURE_BASED_PROCESS_PLANNING.EDGE_LOOP' IN TYPEOF(bnds.
      bound)) ) | (NOT (SIZEOF(QUERY ( oe <* elp_fbnds.bound\path
      .edge_list | (NOT (SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.LINE',
      'FEATURE_BASED_PROCESS_PLANNING.CONIC',
      'FEATURE_BASED_PROCESS_PLANNING.POLYLINE',
      'FEATURE_BASED_PROCESS_PLANNING.SURFACE_CURVE',
      'FEATURE_BASED_PROCESS_PLANNING.B_SPLINE_CURVE'] * TYPEOF(
      oe.edge_element\edge_curve.edge_geometry)) = 1)) )) = 0)) ))
      = 0);
wr4 : (SIZEOF(QUERY ( elp_fbnds <* QUERY ( bnds <* bounds | (
      'FEATURE_BASED_PROCESS_PLANNING.EDGE_LOOP' IN TYPEOF(bnds.
      bound)) ) | (NOT (SIZEOF(QUERY ( oe <* elp_fbnds.bound\path
      .edge_list | (NOT ((
      'FEATURE_BASED_PROCESS_PLANNING.VERTEX_POINT' IN TYPEOF(oe\
      edge.edge_start)) AND (
      'FEATURE_BASED_PROCESS_PLANNING.CARTESIAN_POINT' IN TYPEOF(
      oe\edge.edge_start\vertex_point.vertex_geometry)) AND (
      'FEATURE_BASED_PROCESS_PLANNING.VERTEX_POINT' IN TYPEOF(oe\
      edge.edge_end)) AND (
      'FEATURE_BASED_PROCESS_PLANNING.CARTESIAN_POINT' IN TYPEOF(
      oe\edge.edge_end\vertex_point.vertex_geometry)))) )) = 0)) ))
      = 0);
wr5 : (SIZEOF(QUERY ( elp_fbnds <* QUERY ( bnds <* bounds | (
            'FEATURE_BASED_PROCESS_PLANNING.EDGE_LOOP' IN TYPEOF(bnds.
            bound)) ) | ('FEATURE_BASED_PROCESS_PLANNING.ORIENTED_PATH'
            IN TYPEOF(elp_fbnds.bound)) )) = 0);
wr6 : ((NOT ('FEATURE_BASED_PROCESS_PLANNING.SWEPT_SURFACE' IN
            TYPEOF(face_geometry))) OR (SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.LINE',
            'FEATURE_BASED_PROCESS_PLANNING.CONIC',
            'FEATURE_BASED_PROCESS_PLANNING.POLYLINE',
            'FEATURE_BASED_PROCESS_PLANNING.B_SPLINE_CURVE'] * TYPEOF(
            face_geometry\swept_surface.swept_curve)) = 1));
wr7 : (SIZEOF(QUERY ( vlp_fbnds <* QUERY ( bnds <* bounds | (
            'FEATURE_BASED_PROCESS_PLANNING.VERTEX_LOOP' IN TYPEOF(bnds
            .bound)) ) | (NOT ((
            'FEATURE_BASED_PROCESS_PLANNING.VERTEX_POINT' IN TYPEOF(
            vlp_fbnds\face_bound.bound\vertex_loop.loop_vertex)) AND (
            'FEATURE_BASED_PROCESS_PLANNING.CARTESIAN_POINT' IN TYPEOF(
            vlp_fbnds\face_bound.bound\vertex_loop.loop_vertex\
            vertex_point.vertex_geometry)))) )) = 0);
wr8 : (SIZEOF(QUERY ( bnd <* bounds | (NOT (SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.EDGE_LOOP',
            'FEATURE_BASED_PROCESS_PLANNING.VERTEX_LOOP'] * TYPEOF(bnd.
            bound)) = 1)) )) = 0);
wr9 : (SIZEOF(QUERY ( elp_fbnds <* QUERY ( bnds <* bounds | (
      'FEATURE_BASED_PROCESS_PLANNING.EDGE_LOOP' IN TYPEOF(bnds.
      bound)) ) | (NOT (SIZEOF(QUERY ( oe <* elp_fbnds.bound\path
      .edge_list | ((
      'FEATURE_BASED_PROCESS_PLANNING.SURFACE_CURVE' IN TYPEOF(oe
      \oriented_edge.edge_element\edge_curve.edge_geometry)) AND
      (NOT (SIZEOF(QUERY ( sc_ag <* oe.edge_element\edge_curve.
      edge_geometry\surface_curve.associated_geometry | (NOT (
      'FEATURE_BASED_PROCESS_PLANNING.PCURVE' IN TYPEOF(sc_ag))) ))
      = 0))) )) = 0)) )) = 0);
wr10: (((NOT ('FEATURE_BASED_PROCESS_PLANNING.SWEPT_SURFACE' IN
            TYPEOF(face_geometry))) OR (NOT (
```

 ©ISO 2000 - All rights reserved

```
              'FEATURE_BASED_PROCESS_PLANNING.POLYLINE' IN TYPEOF(
              face_geometry\swept_surface.swept_curve))) OR (SIZEOF(
              face_geometry\swept_surface.swept_curve\polyline.points) >=
               3)) AND (SIZEOF(QUERY ( elp_fbnds <* QUERY ( bnds <*
              bounds | ('FEATURE_BASED_PROCESS_PLANNING.EDGE_LOOP' IN
              TYPEOF(bnds.bound)) ) | (NOT (SIZEOF(QUERY ( oe <*
              elp_fbnds.bound\path.edge_list | ((
              'FEATURE_BASED_PROCESS_PLANNING.POLYLINE' IN TYPEOF(oe\
              oriented_edge.edge_element\edge_curve.edge_geometry)) AND (
              NOT (SIZEOF(oe\oriented_edge.edge_element\edge_curve.
              edge_geometry\polyline.points) >= 3))) )) = 0)) )) = 0));
END_ENTITY; -- advanced_face

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

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

ENTITY angularity_tolerance
  SUBTYPE OF (geometric_tolerance_with_datum_reference);
  WHERE
    wr1: (SIZEOF(SELF\geometric_tolerance_with_datum_reference.
             datum_system) < 3);
END_ENTITY; -- angularity_tolerance

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,'FEATURE_BASED_PROCESS_PLANNING.' +
             'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1);
    wr2: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
             'ID_ATTRIBUTE.IDENTIFIED_ITEM')) <= 1);
END_ENTITY; -- application_context

ENTITY application_context_element
  SUPERTYPE OF (ONEOF (product_context,product_definition_context));
    name               : label;
    frame_of_reference : application_context;
END_ENTITY; -- application_context_element

ENTITY application_protocol_definition;
    status                                  : label;
    application_interpreted_model_schema_name : label;
    application_protocol_year               : year_number;
    application                             : application_context;
END_ENTITY; -- application_protocol_definition

ENTITY applied_area
  SUBTYPE OF (shape_aspect);
  WHERE
```

©ISO 2000 - All rights reserved

```
wr1: ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
         TYPEOF(SELF.of_shape));
wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
     | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
     used_representation)) )) = 1)) )) = 0);
wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
     | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
     used_representation)) ) | (NOT ((2 <= SIZEOF(impl_rep.
     used_representation.items)) AND (SIZEOF(impl_rep.
     used_representation.items) <= 3))) )) = 0)) )) = 0);
wr4: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
     | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (((
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
     used_representation)) AND (SIZEOF(QUERY ( srwp_i <* pdr.
     used_representation.items | (NOT (srwp_i.name IN [
     'orientation','effective length','maximum length'])) )) > 0)) ))
     = 0) )) = 0);
wr5: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
     | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
     used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
     impl_rep.used_representation.items | ((SIZEOF([
     'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
     'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
     TYPEOF(it)) = 2) AND (it.name = 'effective length')) )) = 1)) ))
     = 0)) )) <= 1);
wr6: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
     | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
     'FEATURE_BASED_PROCESS_PLANNING.' +
     'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
     used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
     impl_rep.used_representation.items | ((SIZEOF([
     'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
     'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
     TYPEOF(it)) = 2) AND (it.name = 'maximum length')) )) <= 1)) ))
     = 0)) )) = 0);
wr7: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
     | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
     'FEATURE_BASED_PROCESS_PLANNING.' +
```

 ©ISO 2000 - All rights reserved

```
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items | ((
          'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
          AND (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
END_ENTITY; -- applied_area

ENTITY applied_document_reference
  SUBTYPE OF (document_reference);
    items : SET [1:?] OF document_reference_item;
END_ENTITY; -- applied_document_reference

ENTITY applied_document_usage_constraint_assignment
  SUBTYPE OF (document_usage_constraint_assignment);
    items : SET [1:?] OF document_reference_item;
END_ENTITY; -- applied_document_usage_constraint_assignment

ENTITY applied_group_assignment
  SUBTYPE OF (group_assignment);
    items : SET [1:?] OF group_item;
END_ENTITY; -- applied_group_assignment

ENTITY approval;
    status : approval_status;
    level  : label;
END_ENTITY; -- approval

ENTITY approval_assignment
  ABSTRACT SUPERTYPE;
    assigned_approval : approval;
  DERIVE
    role : object_role := get_role(SELF);
  WHERE
    wr1: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
              'ROLE_ASSOCIATION.ITEM_WITH_ROLE')) <= 1);
END_ENTITY; -- approval_assignment

ENTITY approval_date_time;
    date_time      : date_time_select;
    dated_approval : approval;
  DERIVE
    role : object_role := get_role(SELF);
  WHERE
    wr1: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
              'ROLE_ASSOCIATION.ITEM_WITH_ROLE')) <= 1);
END_ENTITY; -- approval_date_time

ENTITY approval_person_organization;
    person_organization : person_organization_select;
    authorized_approval : approval;
    role                : approval_role;
END_ENTITY; -- approval_person_organization

ENTITY approval_role;
    role : label;
  DERIVE
    description : text := get_description_value(SELF);
  WHERE
    wr1: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
```

©ISO 2000 - All rights reserved

```
               'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1);
  END_ENTITY; -- approval_role

  ENTITY approval_status;
      name : label;
  END_ENTITY; -- approval_status

  ENTITY assembly_component_usage
    SUPERTYPE OF (next_assembly_usage_occurrence)
    SUBTYPE OF (product_definition_usage);
      reference_designator : OPTIONAL identifier;
  END_ENTITY; -- assembly_component_usage

  ENTITY axis1_placement
    SUBTYPE OF (placement);
      axis : OPTIONAL direction;
    DERIVE
      z : direction := NVL(normalise(axis),dummy_gri ||
direction([0,0,1]));
    WHERE
      wr1: (SELF\geometric_representation_item.dim = 3);
  END_ENTITY; -- axis1_placement

  ENTITY axis2_placement_2d
    SUBTYPE OF (placement);
      ref_direction : OPTIONAL direction;
    DERIVE
      p : LIST [2:2] OF direction := build_2axes(ref_direction);
    WHERE
      wr1: (SELF\geometric_representation_item.dim = 2);
  END_ENTITY; -- axis2_placement_2d

  ENTITY 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));
  END_ENTITY; -- axis2_placement_3d

  ENTITY b_spline_curve
    SUPERTYPE OF (ONEOF (uniform_curve,b_spline_curve_with_knots,
        quasi_uniform_curve,bezier_curve) ANDOR rational_b_spline_curve)
    SUBTYPE OF (bounded_curve);
      degree              : INTEGER;
      control_points_list : LIST [2:?] OF cartesian_point;
      curve_form          : b_spline_curve_form;
      closed_curve        : LOGICAL;
      self_intersect      : LOGICAL;
    DERIVE
      upper_index_on_control_points : INTEGER := SIZEOF(
                                          control_points_list) - 1;
      control_points                : ARRAY [0:
                                          upper_index_on_control_points] OF
                                           cartesian_point := list_to_array(
```

 ©ISO 2000 - All rights reserved

```
                                        control_points_list,0,
                                        upper_index_on_control_points);
  WHERE
    wr1: (('FEATURE_BASED_PROCESS_PLANNING.UNIFORM_CURVE' IN
              TYPEOF(SELF))
              OR ('FEATURE_BASED_PROCESS_PLANNING.QUASI_UNIFORM_CURVE' IN
              TYPEOF(SELF)) OR (
              'FEATURE_BASED_PROCESS_PLANNING.BEZIER_CURVE' IN
              TYPEOF(SELF))
              OR (
              'FEATURE_BASED_PROCESS_PLANNING.B_SPLINE_CURVE_WITH_KNOTS'
              IN TYPEOF(SELF)));
END_ENTITY; -- b_spline_curve

ENTITY b_spline_curve_with_knots
  SUBTYPE OF (b_spline_curve);
    knot_multiplicities : LIST [2:?] OF INTEGER;
    knots               : LIST [2:?] OF parameter_value;
    knot_spec           : knot_type;
  DERIVE
    upper_index_on_knots : INTEGER := SIZEOF(knots);
  WHERE
    wr1: constraints_param_b_spline(degree,upper_index_on_knots,
              upper_index_on_control_points,knot_multiplicities,knots);
    wr2: (SIZEOF(knot_multiplicities) = upper_index_on_knots);
END_ENTITY; -- b_spline_curve_with_knots

ENTITY b_spline_surface
  SUPERTYPE OF (ONEOF (b_spline_surface_with_knots,uniform_surface,
      quasi_uniform_surface,bezier_surface) ANDOR
      rational_b_spline_surface)
  SUBTYPE OF (bounded_surface);
    u_degree            : INTEGER;
    v_degree            : INTEGER;
    control_points_list : LIST [2:?] OF LIST [2:?] OF cartesian_point;
    surface_form        : b_spline_surface_form;
    u_closed            : LOGICAL;
    v_closed            : LOGICAL;
    self_intersect      : LOGICAL;
  DERIVE
    u_upper        : INTEGER := SIZEOF(control_points_list) - 1;
    v_upper        : INTEGER := SIZEOF(control_points_list[1]) - 1;
    control_points : ARRAY [0:u_upper] OF ARRAY [0:v_upper] OF
                         cartesian_point := make_array_of_array(
                        control_points_list,0,u_upper,0,v_upper);
  WHERE
    wr1: (('FEATURE_BASED_PROCESS_PLANNING.UNIFORM_SURFACE' IN TYPEOF(
              SELF)) OR (
              'FEATURE_BASED_PROCESS_PLANNING.QUASI_UNIFORM_SURFACE' IN
              TYPEOF(SELF)) OR (
              'FEATURE_BASED_PROCESS_PLANNING.BEZIER_SURFACE' IN TYPEOF(
              SELF)) OR
             ('FEATURE_BASED_PROCESS_PLANNING.B_SPLINE_SURFACE_WITH_KNOTS'
              IN TYPEOF(SELF)));
END_ENTITY; -- b_spline_surface

ENTITY b_spline_surface_with_knots
  SUBTYPE OF (b_spline_surface);
    u_multiplicities : LIST [2:?] OF INTEGER;
    v_multiplicities : LIST [2:?] OF INTEGER;
    u_knots          : LIST [2:?] OF parameter_value;
```

©ISO 2000 - All rights reserved

```
    v_knots          : LIST [2:?] OF parameter_value;
    knot_spec        : knot_type;
  DERIVE
    knot_u_upper : INTEGER := SIZEOF(u_knots);
    knot_v_upper : INTEGER := SIZEOF(v_knots);
  WHERE
    wr1: constraints_param_b_spline(SELF\b_spline_surface.u_degree,
             knot_u_upper,SELF\b_spline_surface.u_upper,u_multiplicities,
             u_knots);
    wr2: constraints_param_b_spline(SELF\b_spline_surface.v_degree,
             knot_v_upper,SELF\b_spline_surface.v_upper,v_multiplicities,
             v_knots);
    wr3: (SIZEOF(u_multiplicities) = knot_u_upper);
    wr4: (SIZEOF(v_multiplicities) = knot_v_upper);
END_ENTITY; -- b_spline_surface_with_knots

ENTITY bezier_curve
  SUBTYPE OF (b_spline_curve);
END_ENTITY; -- bezier_curve

ENTITY bezier_surface
  SUBTYPE OF (b_spline_surface);
END_ENTITY; -- bezier_surface

ENTITY block_shape_representation
  SUBTYPE OF (shape_representation_with_parameters);
  WHERE
    wr1: (SIZEOF(SELF.items) = 4);
    wr2: (SIZEOF(QUERY ( it <* SELF.items | ((
             'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
             AND (it.name = 'orientation')) )) = 1);
    wr3: (SIZEOF(QUERY ( it <* SELF.items | ((SIZEOF([
         'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
         'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
         TYPEOF(it)) = 2) AND (it.name = 'length')) )) = 1);
    wr4: (SIZEOF(QUERY ( it <* SELF.items | ((SIZEOF([
         'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
         'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
         TYPEOF(it)) = 2) AND (it.name = 'width')) )) = 1);
    wr5: (SIZEOF(QUERY ( it <* SELF.items | ((SIZEOF([
         'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
         'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
         TYPEOF(it)) = 2) AND (it.name = 'height')) )) = 1);
END_ENTITY; -- block_shape_representation

ENTITY boss
  SUBTYPE OF (feature_definition);
  WHERE
    wr1 : (SELF\characterized_object.description IN ['circular',
             'complex','rectangular']);
    wr2 : (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (
          'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
          IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
          pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
          | ((sa_occ.description = 'boss height occurrence') AND (
          SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
          'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
          + 'RELATED_SHAPE_ASPECT') | ((sar.description =
          'path feature component usage') AND ((
```

 ©ISO 2000 - All rights reserved

```
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_DEFINING_RELATIONSHIP') IN TYPEOF(sar))) ) | ((
       'FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT' IN
       TYPEOF(sdr.relating_shape_aspect)) AND (sdr.
       relating_shape_aspect.description = 'linear') AND (sdr.name
       = 'boss height')) )) = 1)) )) = 1)) )) = 0);
wr3 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
       | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (((
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
       used_representation)) AND (1 <= SIZEOF(pdr.
       used_representation.items)) AND (SIZEOF(pdr.
       used_representation.items) <= 2)) )) = 1) )) = 1);
wr4 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
       | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (((
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
       used_representation)) AND (SIZEOF(QUERY ( srwp_i <* pdr.
       used_representation.items | ((srwp_i.name = 'orientation')
       OR (srwp_i.name = 'fillet radius')) )) = SIZEOF(pdr.
       used_representation.items))) )) = 1) )) = 1);
wr5 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
       | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
       pd,'FEATURE_BASED_PROCESS_PLANNING.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
       'FEATURE_BASED_PROCESS_PLANNING.' +
       'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
       used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
       impl_rep.used_representation.items | ((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
       * TYPEOF(it)) = 2) AND (it.name = 'fillet radius')) )) <= 1)) )
       = 0)) )) = 0);
wr6 : ((NOT (SELF\characterized_object.description = 'circular')) OR
       (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
       | (
       'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
       IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
       pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
       | ((sa_occ.description = 'circular profile occurrence')
       AND (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
       'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
       + 'RELATED_SHAPE_ASPECT') | ((sar.description =
       'profile usage') AND
       ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
       IN TYPEOF(sar))) ) | (
       'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE' IN
       TYPEOF(sdr.relating_shape_aspect)) )) = 1)) )) = 1)) ))
       = 0));
wr7 : (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
       | (
       'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
```

©ISO 2000 - All rights reserved

```
        IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
        pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
        | ((sa_occ.description = 'top condition occurrence') AND (
        SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(sa_occ,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
        + 'RELATED_SHAPE_ASPECT') | ((sar.description =
        'boss top usage') AND
    ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
        IN TYPEOF(sar))) ) | (
        'FEATURE_BASED_PROCESS_PLANNING.BOSS_TOP' IN TYPEOF(fcr.
        relating_shape_aspect)) )) = 1)) )) = 1)) )) = 0);
wr8 : ((NOT (SELF\characterized_object.description = 'circular')) OR
         (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
        | (
        'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
        IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
        pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
         | ((sa_occ.description = 'change in diameter occurrence')
        AND (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(sa_occ,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
        + 'RELATED_SHAPE_ASPECT') | ((sar.description =
        'taper usage') AND
       ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
        IN TYPEOF(sar))) ) | (
        'FEATURE_BASED_PROCESS_PLANNING.TAPER' IN TYPEOF(fcr.
        related_shape_aspect)) )) = 1)) )) <= 1)) )) = 0));
wr9 : ((NOT (SELF\characterized_object.description = 'complex')) OR
        (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
        | (
        'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
        IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
        pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
        | ((sa_occ.description = 'enclosed boundary occurrence')
        AND (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
        + 'RELATED_SHAPE_ASPECT') | ((sar.description =
        'profile usage') AND
        ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
        IN TYPEOF(sar))) ) | (SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.NGON_CLOSED_PROFILE',
        'FEATURE_BASED_PROCESS_PLANNING.CLOSED_PATH_PROFILE'] *
        TYPEOF(sdr.relating_shape_aspect)) = 1) )) = 1)) )) = 1)) ))
        = 0));
wr10: ((NOT (SELF\characterized_object.description IN ['complex',
        'rectangular'])) OR (SIZEOF(QUERY ( pds <* QUERY ( pd <*
        USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
        | (
        'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
        IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
        pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
        | ((sa_occ.description = 'change in boundary occurrence')
        AND (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(sa_occ,
        'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
        + 'RELATING_SHAPE_ASPECT') | ((sar.description =
        'taper usage') AND
    ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
        IN TYPEOF(sar))) ) | ((
        'FEATURE_BASED_PROCESS_PLANNING.TAPER' IN TYPEOF(fcr.
```

 ©ISO 2000 - All rights reserved

```
          related_shape_aspect)) AND (fcr.related_shape_aspect.
          description IN ['angle taper','directed taper'])) )) = 1)) ))
          <= 1)) )) = 0));
    wr11: ((NOT (SELF\characterized_object.description = 'rectangular'))
          OR (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (
          'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
          IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
          pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
          | ((sa_occ.description = 'rectangular profile occurrence')
          AND (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
          'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
          + 'RELATED_SHAPE_ASPECT') | ((sar.description =
          'profile usage') AND
          ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
          IN TYPEOF(sar))) ) |
          ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE'
          IN TYPEOF(sdr.relating_shape_aspect)) )) = 1)) )) = 1)) ))
          = 0));
END_ENTITY; -- boss

ENTITY boss_top
  SUBTYPE OF (shape_aspect);
  WHERE
    wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
         IN TYPEOF(SELF.of_shape.definition));
    wr2: (SELF.description IN ['planar','complex']);
    wr3: ((NOT (SELF.description = 'planar')) OR (SIZEOF(QUERY ( pd <*
         USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
         | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
         ('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
         IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0));
    wr4: ((NOT (SELF.description = 'planar')) OR (SIZEOF(QUERY ( pd <*
         USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
         | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
         ('FEATURE_BASED_PROCESS_PLANNING.LOCATION_SHAPE_REPRESENTATION'
         IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0));
    wr5: ((NOT (SELF.description = 'complex')) OR (SIZEOF(QUERY ( pd <*
         USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
         | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (
         'FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
         IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0));
    wr6: (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
         + 'RELATING_SHAPE_ASPECT') | ((sar.description =
         'boss top usage') AND (sar.name IN ['boss height start',
         'boss height end']) AND
       ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
         IN TYPEOF(sar))) ) | ((fcr.related_shape_aspect.description
         = 'top condition occurrence') AND (
         'FEATURE_BASED_PROCESS_PLANNING.BOSS' IN TYPEOF(fcr.
```

```
          related_shape_aspect.of_shape.definition))) )) >= 1);
     wr7: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) )) <= 1)) )) = 0);
     wr8: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(impl_rep.
          used_representation.items) = 1)) )) = 0)) )) = 0);
     wr9: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items | ((SIZEOF([
          'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
          TYPEOF(it)) = 2) AND (it.name = 'top radius')) )) <= 1)) ))
          = 0)) )) = 0);
END_ENTITY; -- boss_top

ENTITY bounded_curve
  SUPERTYPE OF (ONEOF (polyline,b_spline_curve,composite_curve))
  SUBTYPE OF (curve);
END_ENTITY; -- bounded_curve

ENTITY bounded_surface
  SUPERTYPE OF (b_spline_surface)
  SUBTYPE OF (surface);
END_ENTITY; -- bounded_surface

ENTITY brep_with_voids
  SUBTYPE OF (manifold_solid_brep);
    voids : SET [1:?] OF oriented_closed_shell;
END_ENTITY; -- brep_with_voids

ENTITY calendar_date
  SUBTYPE OF (date);
    day_component   : day_in_month_number;
    month_component : month_in_year_number;
  WHERE
    wr1: valid_calendar_date(SELF);
END_ENTITY; -- calendar_date

ENTITY cartesian_point
  SUBTYPE OF (point);
    coordinates : LIST [1:3] OF length_measure;
END_ENTITY; -- cartesian_point
```

 ©ISO 2000 - All rights reserved

```
ENTITY cartesian_transformation_operator
  SUPERTYPE OF (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);
  WHERE
    wr1: (scl > 0);
END_ENTITY; -- cartesian_transformation_operator

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; -- cartesian_transformation_operator_3d

ENTITY chamfer
  SUBTYPE OF (transition_feature);
  WHERE
    wr1: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
         | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
         'FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
         IN TYPEOF(pdr.used_representation)) AND (pdr.
         used_representation.name = 'chamfer face')) )) = 1)) )) = 0);
    wr2: (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
         + 'RELATING_SHAPE_ASPECT') |
       ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
         IN TYPEOF(sar)) ) | ((
         'FEATURE_BASED_PROCESS_PLANNING.CHAMFER_OFFSET' IN TYPEOF(
         fcr.related_shape_aspect)) AND (fcr.related_shape_aspect.
         description = 'first offset')) )) = 1);
    wr3: (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
         + 'RELATING_SHAPE_ASPECT') |
         ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
         IN TYPEOF(sar)) ) | ((
         'FEATURE_BASED_PROCESS_PLANNING.CHAMFER_OFFSET' IN TYPEOF(
         fcr.related_shape_aspect)) AND (fcr.related_shape_aspect.
         description = 'second offset')) )) = 1);
END_ENTITY; -- chamfer

ENTITY chamfer_offset
  SUBTYPE OF (shape_aspect);
  WHERE
    wr1: (SELF.description IN ['first offset','second offset']);
    wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
         | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
```

©ISO 2000 - All rights reserved

```
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
        used_representation)) )) = 1)) )) = 0);
wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
        | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
        used_representation)) ) | (NOT (SIZEOF(impl_rep.
        used_representation.items) = 1)) )) = 0)) )) = 0);
wr4: ((NOT (SELF.description = 'first offset')) OR (SIZEOF(
        QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
        | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
        used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
        impl_rep.used_representation.items | ((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
        TYPEOF(it)) = 2) AND (it.name = 'offset amount')) )) = 1)) ))
        = 0)) )) = 0));
wr5: ((NOT (SELF.description = 'first offset')) OR (SIZEOF(
        QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
        | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
        'FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
        IN TYPEOF(pdr.used_representation)) AND (pdr.
        used_representation.name = 'first face shape')) )) = 1)) ))
        = 0));
wr6: ((NOT (SELF.description = 'second offset')) OR (SIZEOF(
        QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
        | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
        'FEATURE_BASED_PROCESS_PLANNING.' +
        'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
        used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
        impl_rep.used_representation.items | (((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
        TYPEOF(it)) = 2) AND (it.name = 'offset amount')) OR ((
        SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
        * TYPEOF(it)) = 2) AND (it.name = 'offset angle'))) )) = 1)) ))
        = 0)) )) = 0));
wr7: ((NOT (SELF.description = 'second offset')) OR (SIZEOF(
        QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
        | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
        'FEATURE_BASED_PROCESS_PLANNING.' +
```

 ©ISO 2000 - All rights reserved

```
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
          IN TYPEOF(pdr.used_representation)) AND (pdr.
          used_representation.name = 'second face shape')) )) = 1)) ))
          = 0));
    wr8: (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
          + 'RELATED_SHAPE_ASPECT') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'FEATURE_COMPONENT_RELATIONSHIP') IN TYPEOF(sar)) ) | (
          'FEATURE_BASED_PROCESS_PLANNING.CHAMFER' IN TYPEOF(sdr.
          relating_shape_aspect)) )) = 1);
END_ENTITY; -- chamfer_offset

ENTITY characterized_object;
    name        : label;
    description : OPTIONAL text;
END_ENTITY; -- characterized_object

ENTITY circle
  SUBTYPE OF (conic);
    radius : positive_length_measure;
END_ENTITY; -- circle

ENTITY circular_closed_profile
  SUBTYPE OF (shape_aspect);
  WHERE
    wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
              IN TYPEOF(SELF.of_shape.definition));
    wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) )) = 1)) )) = 0);
    wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(impl_rep.
          used_representation.items) = 2)) )) = 0)) )) = 0);
    wr4: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items | ((
          'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
          AND (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
    wr5: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
```

©ISO 2000 - All rights reserved

```
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items | ((SIZEOF([
          'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
          TYPEOF(it)) = 2) AND (it.name = 'diameter')) )) = 1)) )) =
          0)) )) = 0);
END_ENTITY; -- circular_closed_profile

ENTITY circular_pattern
  SUBTYPE OF (replicate_feature);
  WHERE
    wr1: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (
          'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
          TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
          'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | (
          SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
          'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
          + 'RELATING_SHAPE_ASPECT') |
          ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
          IN TYPEOF(sar)) ) | (
          'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT' IN TYPEOF(sdr.
          related_shape_aspect)) )) = 1) )) <= 3)) )) = 0);
    wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) )) = 1)) )) = 0);
    wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT ((SIZEOF(impl_rep.
          used_representation.items) >= 3) AND (SIZEOF(impl_rep.
          used_representation.items) <= 5))) )) = 0)) )) = 0);
    wr4: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items | ((SIZEOF([
          'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
          TYPEOF(it)) = 2) AND (it.name = 'diameter')) )) <= 1)) )) =
          0)) )) = 0);
    wr5: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
```

 ©ISO 2000 - All rights reserved

```
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items | ((SIZEOF([
          'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'base feature rotation')) ))
          <= 1)) )) = 0)) )) = 0);
     wr6: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |
          (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
          IN TYPEOF(it)) AND (
          'FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE' IN TYPEOF(it\
          measure_with_unit.value_component)) AND (it.name =
          'number of features')) )) = 1)) )) = 0)) )) = 0);
     wr7: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items | ((SIZEOF([
          'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'angular spacing')) )) = 1))
           )) = 0)) )) = 0);
     wr8: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items | ((
          'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
          AND (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
END_ENTITY; -- circular_pattern

ENTITY circular_runout_tolerance
  SUBTYPE OF (geometric_tolerance_with_datum_reference);
  WHERE
    wr1: (SIZEOF(SELF\geometric_tolerance_with_datum_reference.
             datum_system) <= 2);
END_ENTITY; -- circular_runout_tolerance

ENTITY closed_path_profile
  SUBTYPE OF (shape_aspect);
  WHERE
    wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
```

©ISO 2000 - All rights reserved

```
              IN TYPEOF(SELF.of_shape.definition));
     wr2: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
              'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
              'DEFINITION') | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) )) = 1)) )) = 0);
     wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(impl_rep.
              used_representation.items) = 1)) )) = 0)) )) = 0);
     wr4: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items | ((
              'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
              AND (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
     wr5: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
              'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
              'DEFINITION') | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (
              'FEATURE_BASED_PROCESS_PLANNING.PATH_SHAPE_REPRESENTATION'
              IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0);
END_ENTITY; -- closed_path_profile

ENTITY closed_shell
  SUBTYPE OF (connected_face_set);
END_ENTITY; -- closed_shell

ENTITY composite_curve
  SUBTYPE OF (bounded_curve);
    segments       : LIST [1:?] OF composite_curve_segment;
    self_intersect : LOGICAL;
  DERIVE
    n_segments   : INTEGER := SIZEOF(segments);
    closed_curve : LOGICAL := segments[n_segments].transition <>
                       discontinuous;
  WHERE
    wr1: (((NOT closed_curve) AND (SIZEOF(QUERY ( temp <* segments | (
              temp.transition = discontinuous) )) = 1)) OR (closed_curve
              AND (SIZEOF(QUERY ( temp <* segments | (temp.transition =
              discontinuous) )) = 0)));
END_ENTITY; -- composite_curve

ENTITY composite_curve_segment
  SUBTYPE OF (founded_item);
    transition   : transition_code;
    same_sense   : BOOLEAN;
```

 ©ISO 2000 - All rights reserved

```
    parent_curve : curve;
  INVERSE
    using_curves : BAG [1:?] OF composite_curve FOR segments;
  WHERE
    wr1: ('FEATURE_BASED_PROCESS_PLANNING.BOUNDED_CURVE' IN TYPEOF(
              parent_curve));
END_ENTITY; -- composite_curve_segment

ENTITY composite_hole
  SUBTYPE OF (compound_feature);
  WHERE
    wr1: (SELF\characterized_object.description IN ['counterbore',
              'countersunk']);
    wr2: (SIZEOF(QUERY ( pds <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | ((
              'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
               TYPEOF(pds)) AND (SIZEOF(QUERY ( csa <* USEDIN(pds,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
              'FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT' IN
              TYPEOF(csa)) AND (SIZEOF(QUERY ( sar <* csa.
              component_relationships |
       (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
              IN TYPEOF(sar)) AND (
              'FEATURE_BASED_PROCESS_PLANNING.ROUND_HOLE' IN TYPEOF(sar.
              related_shape_aspect))) )) = 2)) )) = 1)) )) = 1);
    wr3: ((NOT (SELF\characterized_object.description = 'countersunk'))
              OR (SIZEOF(QUERY ( pds <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | ((
              'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
               TYPEOF(pds)) AND (SIZEOF(QUERY ( csa <* USEDIN(pds,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
              'FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT' IN
              TYPEOF(csa)) AND (SIZEOF(QUERY ( sar <* csa.
              component_relationships | ((
              'FEATURE_BASED_PROCESS_PLANNING.ROUND_HOLE' IN TYPEOF(sar.
              related_shape_aspect)) AND (NOT (SIZEOF(QUERY ( pds <*
              QUERY ( pd <* USEDIN(sar.related_shape_aspect,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
              'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
               TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
              sa_occ.description = 'change in diameter occurrence') AND (
              SIZEOF(QUERY ( fcr2 <* QUERY ( sar2 <* USEDIN(sa_occ,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
              + 'RELATING_SHAPE_ASPECT') | ((sar2.description =
              'taper usage') AND
        ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
              IN TYPEOF(sar2))) ) | (
              'FEATURE_BASED_PROCESS_PLANNING.TAPER' IN TYPEOF(fcr2.
            related_shape_aspect)) )) = 1)) )) = 0)) )) = 0))) )) = 1)) ))
              = 1)) )) = 1));
END_ENTITY; -- composite_hole

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

©ISO 2000 - All rights reserved

```
  END_ENTITY; -- composite_shape_aspect

  ENTITY compound_feature
    SUBTYPE OF (feature_definition);
    WHERE
      wr1: (SIZEOF(QUERY ( pds <* USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | ((
               'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
                TYPEOF(pds)) AND (SIZEOF(QUERY ( csa <* USEDIN(pds,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
               csa.name = 'compound feature in solid') AND (
               'FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT' IN
               TYPEOF(csa))) )) = 1)) )) = 1);
      wr2: (SIZEOF(QUERY ( pds <* USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | ((
               'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
                TYPEOF(pds)) AND (SIZEOF(QUERY ( csa <* USEDIN(pds,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | (
               'FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT' IN
               TYPEOF(csa)) )) = 1)) )) = 1);
      wr3: (SIZEOF(QUERY ( pds <* USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | ((
               'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
                TYPEOF(pds)) AND (SIZEOF(QUERY ( csa <* USEDIN(pds,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
               'FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT' IN
               TYPEOF(csa)) AND (SIZEOF(QUERY ( fcr <* csa.
               component_relationships | (NOT
          ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
               IN TYPEOF(fcr))) )) = 0)) )) = 1)) )) = 1);
      wr4: (SIZEOF(QUERY ( pds <* USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | ((
               'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
                TYPEOF(pds)) AND (SIZEOF(QUERY ( csa <* USEDIN(pds,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
               'FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT' IN
               TYPEOF(csa)) AND (SIZEOF(QUERY ( sar <* csa.
               component_relationships | (SIZEOF([
               'FEATURE_BASED_PROCESS_PLANNING.INSTANCED_FEATURE',
               'FEATURE_BASED_PROCESS_PLANNING.TRANSITION_FEATURE',
               'FEATURE_BASED_PROCESS_PLANNING.APPLIED_AREA'] * TYPEOF(sar.
               related_shape_aspect)) = 1) )) = 2)) )) = 1)) )) = 1);
      wr5: (SIZEOF(QUERY ( pds <* USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | ((
               'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
                TYPEOF(pds)) AND (SIZEOF(QUERY ( csa <* USEDIN(pds,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
               'FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT' IN
               TYPEOF(csa)) AND (SIZEOF(QUERY ( sar <* csa.
               component_relationships | (
               'FEATURE_BASED_PROCESS_PLANNING.THREAD' IN TYPEOF(sar.
               related_shape_aspect)) )) = 0)) )) = 1)) )) = 1);
      wr6: (SIZEOF(QUERY ( pds <* USEDIN(SELF,

'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | ((
```

 ©ISO 2000 - All rights reserved

```
              'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
               TYPEOF(pds)) AND (SIZEOF(QUERY ( csa <* USEDIN(pds,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
              'FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT' IN
              TYPEOF(csa)) AND (SIZEOF(QUERY ( sar <* csa.
              component_relationships | ((
              'FEATURE_BASED_PROCESS_PLANNING.COMPOUND_FEATURE' IN TYPEOF(
              sar.related_shape_aspect)) AND (sar.related_shape_aspect\
          characterized_object.name <> SELF\characterized_object.name)) ))
              = 0)) )) = 1)) )) = 1);
END_ENTITY; -- compound_feature

ENTITY concentricity_tolerance
  SUBTYPE OF (geometric_tolerance_with_datum_reference);
  WHERE
    wr1: (SIZEOF(SELF\geometric_tolerance_with_datum_reference.
              datum_system) = 1);
END_ENTITY; -- concentricity_tolerance

ENTITY conic
  SUPERTYPE OF (ONEOF (circle,ellipse,hyperbola,parabola))
  SUBTYPE OF (curve);
    position : axis2_placement;
END_ENTITY; -- conic

ENTITY conical_surface
  SUBTYPE OF (elementary_surface);
    radius     : length_measure;
    semi_angle : plane_angle_measure;
  WHERE
    wr1: (radius >= 0);
END_ENTITY; -- conical_surface

ENTITY connected_face_set
  SUPERTYPE OF (ONEOF (closed_shell,open_shell))
  SUBTYPE OF (topological_representation_item);
    cfs_faces : SET [1:?] OF face;
END_ENTITY; -- connected_face_set

ENTITY context_dependent_unit
  SUBTYPE OF (named_unit);
    name : label;
END_ENTITY; -- context_dependent_unit

ENTITY conversion_based_unit
  SUBTYPE OF (named_unit);
    name              : label;
    conversion_factor : measure_with_unit;
END_ENTITY; -- conversion_based_unit

ENTITY curve
  SUPERTYPE OF (ONEOF (line,conic,pcurve,surface_curve))
  SUBTYPE OF (geometric_representation_item);
END_ENTITY; -- curve

ENTITY cylindrical_shape_representation
  SUBTYPE OF (shape_representation_with_parameters);
  WHERE
    wr1: (SIZEOF(SELF.items) = 3);
    wr2: (SIZEOF(QUERY ( it <* SELF.items | ((
              'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
```

©ISO 2000 - All rights reserved

```
                AND (it.name = 'orientation')) )) = 1);
      wr3: (SIZEOF(QUERY ( it <* SELF.items | ((SIZEOF([
           'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
               'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
                TYPEOF(it)) = 2) AND (it.name = 'length')) )) = 1);
      wr4: (SIZEOF(QUERY ( it <* SELF.items | ((SIZEOF([
           'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
               'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
                TYPEOF(it)) = 2) AND (it.name = 'diameter')) )) = 1);
  END_ENTITY; -- cylindrical_shape_representation

  ENTITY cylindrical_surface
    SUBTYPE OF (elementary_surface);
      radius : positive_length_measure;
  END_ENTITY; -- cylindrical_surface

  ENTITY cylindricity_tolerance
    SUBTYPE OF (geometric_tolerance);
    WHERE
      wr1: (NOT (('FEATURE_BASED_PROCESS_PLANNING.' +
               'GEOMETRIC_TOLERANCE_WITH_DATUM_REFERENCE') IN
TYPEOF(SELF)));
  END_ENTITY; -- cylindricity_tolerance

  ENTITY data_environment;
      name        : label;
      description : text;
      elements    : SET [1:?] OF property_definition_representation;
  END_ENTITY; -- data_environment

  ENTITY date
    SUPERTYPE OF (ONEOF (calendar_date,ordinal_date,
        week_of_year_and_day_date));
      year_component : year_number;
  END_ENTITY; -- date

  ENTITY date_assignment
    ABSTRACT SUPERTYPE;
      assigned_date : date;
      role          : date_role;
  END_ENTITY; -- date_assignment

  ENTITY date_role;
      name : label;
    DERIVE
      description : text := get_description_value(SELF);
    WHERE
      wr1: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
               'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1);
  END_ENTITY; -- date_role

  ENTITY datum
    SUBTYPE OF (shape_aspect);
      identification : identifier;
    INVERSE
      established_by_relationships : SET [1:?] OF
                                        shape_aspect_relationship FOR
                                       related_shape_aspect;
    WHERE
      wr1: (SIZEOF(QUERY ( x <* SELF.established_by_relationships | (
                SIZEOF(TYPEOF(x.relating_shape_aspect) * [
```

 ©ISO 2000 - All rights reserved

```
            'FEATURE_BASED_PROCESS_PLANNING.DATUM_FEATURE',
            'FEATURE_BASED_PROCESS_PLANNING.DATUM_TARGET']) <> 1) )) = 0);
  END_ENTITY; -- datum

  ENTITY datum_feature
    SUBTYPE OF (shape_aspect);
    INVERSE
      feature_basis_relationship : shape_aspect_relationship FOR
                                     relating_shape_aspect;
    WHERE
      wr1: (SIZEOF(QUERY ( sar <* bag_to_set(USEDIN(SELF,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
               + 'RELATING_SHAPE_ASPECT')) | (NOT (
               'FEATURE_BASED_PROCESS_PLANNING.DATUM' IN TYPEOF(sar.
               related_shape_aspect))) )) = 0);
      wr2: (SELF.product_definitional = TRUE);
  END_ENTITY; -- datum_feature

  ENTITY datum_reference;
      precedence       : INTEGER;
      referenced_datum : datum;
    WHERE
      wr1: (precedence > 0);
  END_ENTITY; -- datum_reference

  ENTITY datum_target
    SUBTYPE OF (shape_aspect);
      target_id : identifier;
    INVERSE
      target_basis_relationship : shape_aspect_relationship FOR
                                    relating_shape_aspect;
    WHERE
      wr1: (SIZEOF(QUERY ( sar <* bag_to_set(USEDIN(SELF,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
               + 'RELATING_SHAPE_ASPECT')) | (NOT (
               'FEATURE_BASED_PROCESS_PLANNING.DATUM' IN TYPEOF(sar.
               related_shape_aspect))) )) = 0);
      wr2: (SELF.product_definitional = TRUE);
  END_ENTITY; -- datum_target

  ENTITY definitional_representation
    SUBTYPE OF (representation);
    WHERE
      wr1:
('FEATURE_BASED_PROCESS_PLANNING.PARAMETRIC_REPRESENTATION_CONTEXT'
               IN TYPEOF(SELF\representation.context_of_items));
  END_ENTITY; -- definitional_representation

  ENTITY degenerate_toroidal_surface
    SUBTYPE OF (toroidal_surface);
      select_outer : BOOLEAN;
    WHERE
      wr1: (major_radius < minor_radius);
  END_ENTITY; -- degenerate_toroidal_surface

  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 (
```

©ISO 2000 - All rights reserved

```
              elements[1].exponent <> 1)));
    wr2: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
              'NAME_ATTRIBUTE.NAMED_ITEM')) <= 1);
END_ENTITY; -- derived_unit

ENTITY derived_unit_element;
    unit     : named_unit;
    exponent : REAL;
END_ENTITY; -- derived_unit_element

ENTITY description_attribute;
    attribute_value : text;
    described_item  : description_attribute_select;
END_ENTITY; -- description_attribute

ENTITY descriptive_representation_item
  SUBTYPE OF (representation_item);
    description : text;
END_ENTITY; -- descriptive_representation_item

ENTITY dimensional_characteristic_representation;
    dimension      : dimensional_characteristic;
    representation : shape_dimension_representation;
END_ENTITY; -- dimensional_characteristic_representation

ENTITY dimensional_exponents;
    length_exponent                    : REAL;
    mass_exponent                      : REAL;
    time_exponent                      : REAL;
    electric_current_exponent          : REAL;
    thermodynamic_temperature_exponent : REAL;
    amount_of_substance_exponent       : REAL;
    luminous_intensity_exponent        : REAL;
END_ENTITY; -- dimensional_exponents

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

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

ENTITY dimensional_size
  SUPERTYPE OF (ONEOF (angular_size,dimensional_size_with_path));
    applies_to : shape_aspect;
    name       : label;
  WHERE
    wr1: (applies_to.product_definitional = TRUE);
END_ENTITY; -- dimensional_size

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

ENTITY directed_action
  SUBTYPE OF (executed_action);
    directive : action_directive;
```

 ©ISO 2000 - All rights reserved

```
END_ENTITY; -- directed_action

ENTITY directed_dimensional_location
  SUBTYPE OF (dimensional_location);
END_ENTITY; -- directed_dimensional_location

ENTITY direction
  SUBTYPE OF (geometric_representation_item);
    direction_ratios : LIST [2:3] OF REAL;
  WHERE
    wr1: (SIZEOF(QUERY ( tmp <* direction_ratios | (tmp <> 0) )) > 0);
END_ENTITY; -- direction

ENTITY direction_shape_representation
  SUBTYPE OF (shape_representation);
  WHERE
    wr1: (SIZEOF(SELF.items) = 1);
    wr2: (SIZEOF(QUERY ( it <* SELF.items | (NOT (
           'FEATURE_BASED_PROCESS_PLANNING.DIRECTION' IN TYPEOF(it))) ))
             = 0);
END_ENTITY; -- direction_shape_representation

ENTITY document;
    id          : identifier;
    name        : label;
    description : OPTIONAL text;
    kind        : document_type;
  INVERSE
    representation_types : SET [0:?] OF document_representation_type FOR
                                  represented_document;
END_ENTITY; -- document

ENTITY document_file
  SUBTYPE OF (characterized_object, document);
  WHERE
    wr1: ((SIZEOF(QUERY ( adr <* QUERY ( dr <* USEDIN(SELF,
   'FEATURE_BASED_PROCESS_PLANNING.DOCUMENT_REFERENCE.ASSIGNED_DOCUMENT')
              | (
             'FEATURE_BASED_PROCESS_PLANNING.APPLIED_DOCUMENT_REFERENCE'
             IN TYPEOF(dr)) ) |
  ('FEATURE_BASED_PROCESS_PLANNING.EXTERNALLY_DEFINED_FEATURE_DEFINITION'
             IN TYPEOF(adr.items)) )) = 1) OR (SIZEOF(QUERY ( duc <*
             USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.DOCUMENT_USAGE_CONSTRAINT.SOURCE')
              | (NOT (SIZEOF(QUERY ( aduc <* QUERY ( duca <* USEDIN(duc,
   'FEATURE_BASED_PROCESS_PLANNING.DOCUMENT_USAGE_CONSTRAINT_ASSIGNMENT.'
             + 'ASSIGNED_DOCUMENT_USAGE') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'APPLIED_DOCUMENT_USAGE_CONSTRAINT_ASSIGNMENT') IN TYPEOF(
             duca)) ) |
  ('FEATURE_BASED_PROCESS_PLANNING.EXTERNALLY_DEFINED_FEATURE_DEFINITION'
             IN TYPEOF(aduc.items)) )) = 1)) )) = 0));
    wr2: (SIZEOF(QUERY ( drt <* USEDIN(SELF,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'DOCUMENT_REPRESENTATION_TYPE.REPRESENTED_DOCUMENT') | (drt.
             name = 'physical') )) = 1);
END_ENTITY; -- document_file

ENTITY document_reference
  ABSTRACT SUPERTYPE;
    assigned_document : document;
```

©ISO 2000 - All rights reserved

```
    source            : label;
  DERIVE
    role : object_role := get_role(SELF);
  WHERE
    wr1: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
             'ROLE_ASSOCIATION.ITEM_WITH_ROLE')) <= 1);
END_ENTITY; -- document_reference

ENTITY document_relationship;
    name              : label;
    description       : OPTIONAL text;
    relating_document : document;
    related_document  : document;
END_ENTITY; -- document_relationship

ENTITY document_representation_type;
    name                  : label;
    represented_document : document;
END_ENTITY; -- document_representation_type

ENTITY document_type;
    product_data_type : label;
END_ENTITY; -- document_type

ENTITY document_usage_constraint;
    source                : document;
    subject_element       : label;
    subject_element_value : text;
END_ENTITY; -- document_usage_constraint

ENTITY document_usage_constraint_assignment
  ABSTRACT SUPERTYPE;
    assigned_document_usage : document_usage_constraint;
    role                    : document_usage_role;
END_ENTITY; -- document_usage_constraint_assignment

ENTITY document_usage_role;
    name        : label;
    description : OPTIONAL text;
END_ENTITY; -- document_usage_role

ENTITY document_with_class
  SUBTYPE OF (document);
    class : identifier;
END_ENTITY; -- document_with_class

ENTITY edge
  SUPERTYPE OF (ONEOF (edge_curve,oriented_edge))
  SUBTYPE OF (topological_representation_item);
    edge_start : vertex;
    edge_end   : vertex;
END_ENTITY; -- edge

ENTITY edge_curve
  SUBTYPE OF (edge, geometric_representation_item);
    edge_geometry : curve;
    same_sense    : BOOLEAN;
END_ENTITY; -- edge_curve

ENTITY edge_loop
  SUBTYPE OF (loop, path);
```

 ©ISO 2000 - All rights reserved

```
  DERIVE
    ne : INTEGER := SIZEOF(SELF\path.edge_list);
  WHERE
    wr1: (SELF\path.edge_list[1].edge_start :=: SELF\path.edge_list[ne].
              edge_end);
END_ENTITY; -- edge_loop

ENTITY edge_round
  SUBTYPE OF (transition_feature);
  WHERE
    wr1: ((NOT (SELF\shape_aspect.description = 'constant radius')) OR (
              SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
   ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
              IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0));
    wr2: ((NOT (SELF\shape_aspect.description = 'constant radius')) OR (
              SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
   ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
              IN TYPEOF(pdr.used_representation)) ) | ((NOT (SIZEOF(
              impl_rep.used_representation.items) >= 1)) AND (SIZEOF(
              impl_rep.used_representation.items) <= 3)) )) = 0)) ))
              = 0));
    wr3: ((NOT (SELF.description = 'constant radius')) OR (SIZEOF(
              QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
   ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
              IN TYPEOF(pdr.used_representation)) ) | (NOT (SIZEOF(
              QUERY ( it <* impl_rep.used_representation.items | ((SIZEOF(
            ['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
              TYPEOF(it)) = 2) AND (it.name = 'radius')) )) = 1)) ))
              = 0)) ))
              = 0));
    wr4: ((NOT (SELF.description = 'constant radius')) OR (SIZEOF(
              QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
         | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
   ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
              IN TYPEOF(pdr.used_representation)) ) | (NOT (SIZEOF(
              QUERY ( it <* impl_rep.used_representation.items | ((SIZEOF(
            ['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
             TYPEOF(it)) = 2) AND (it.name = 'first offset')) )) <= 1)) ))
              = 0)) )) = 0));
    wr5: ((NOT (SELF.description = 'constant radius')) OR (SIZEOF(
              QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
```

©ISO 2000 - All rights reserved

```
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
    ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
              IN TYPEOF(pdr.used_representation)) ) | (NOT (SIZEOF(
              QUERY ( it <* impl_rep.used_representation.items | ((SIZEOF(
            ['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
            TYPEOF(it)) = 2) AND (it.name = 'second offset')) )) <= 1)) ))
              = 0)) )) = 0));
     wr6: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
              IN TYPEOF(pdr.used_representation)) AND (pdr.
              used_representation.name = 'edge round face')) )) = 1)) )) =
               0);
     wr7: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
              IN TYPEOF(pdr.used_representation)) AND (pdr.
              used_representation.name = 'first face shape')) )) = 1)) ))
              = 0);
     wr8: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
              IN TYPEOF(pdr.used_representation)) AND (pdr.
              used_representation.name = 'second face shape')) )) = 1)) ))
              = 0);
END_ENTITY; -- edge_round

ENTITY elementary_surface
  SUPERTYPE OF (ONEOF (plane,cylindrical_surface,conical_surface,
      spherical_surface,toroidal_surface))
  SUBTYPE OF (surface);
    position : axis2_placement_3d;
END_ENTITY; -- elementary_surface

ENTITY ellipse
  SUBTYPE OF (conic);
    semi_axis_1 : positive_length_measure;
    semi_axis_2 : positive_length_measure;
END_ENTITY; -- ellipse

ENTITY executed_action
  SUBTYPE OF (action);
END_ENTITY; -- executed_action

ENTITY expanded_uncertainty
  SUBTYPE OF (standard_uncertainty);
    coverage_factor : REAL;
END_ENTITY; -- expanded_uncertainty
```

 ©ISO 2000 - All rights reserved

```
ENTITY external_source;
    source_id : source_item;
  DERIVE
    description : text := get_description_value(SELF);
  WHERE
    wr1: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
              'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1);
END_ENTITY; -- external_source

ENTITY externally_defined_feature_definition
  SUBTYPE OF (feature_definition, externally_defined_item);
  WHERE
    wr1 : (((SELF\characterized_object.description = 'thread') AND (SELF
              \externally_defined_item.item_id = 'external thread') AND (
              SELF\externally_defined_item.source.source_id =
              'external feature specification')) OR ((SELF\
              characterized_object.description = 'marking') AND (SELF\
              externally_defined_item.item_id = 'external marking') AND (
              SELF\externally_defined_item.source.source_id =
              'external feature specification')) OR ((SELF\
              characterized_object.description = 'knurl') AND (SELF\
              externally_defined_item.item_id = 'external knurl') AND (
              SELF\externally_defined_item.source.source_id =
              'external feature specification')));
    wr2 : ((NOT (SELF\characterized_object.description = 'thread')) OR (
              SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) AND (8 <= SIZEOF(pdr.
              used_representation.items)) AND (SIZEOF(pdr.
              used_representation.items) <= 10)) )) = 1) )) = 1));
    wr3 : ((NOT (SELF\characterized_object.description = 'marking')) OR
              (SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) AND (SIZEOF(pdr.used_representation.
              items) = 2)) )) = 1) )) = 1));
    wr4 : ((NOT (SELF\characterized_object.description = 'knurl')) OR (
              SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) AND (SIZEOF(pdr.used_representation.
              items) = 1)) )) = 1) )) = 1));
    wr5 : ((NOT (SELF\characterized_object.description IN ['knurl',
              'thread'])) OR (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(
              SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (
              'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
```

```
            IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
            pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
             | ((sa_occ.description = 'partial area occurrence') AND (
            SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
          'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
            + 'RELATED_SHAPE_ASPECT') | ((sar.description =
            'applied area usage') AND
          ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
            IN TYPEOF(sar))) ) | (
            'FEATURE_BASED_PROCESS_PLANNING.APPLIED_AREA' IN TYPEOF(sdr
            .relating_shape_aspect)) )) = 1)) )) <= 1)) )) = 0));
wr6 : ((NOT (SIZEOF(QUERY ( doc <* QUERY ( adr <* USEDIN(SELF,
   'FEATURE_BASED_PROCESS_PLANNING.APPLIED_DOCUMENT_REFERENCE.ITEMS')
             | ('FEATURE_BASED_PROCESS_PLANNING.DOCUMENT' IN TYPEOF(adr
            .assigned_document)) ) | (
            'FEATURE_BASED_PROCESS_PLANNING.DOCUMENT_FILE' IN TYPEOF(
            doc)) )) = 1)) OR (NOT (SIZEOF(QUERY ( doc <*
            QUERY ( aduc <* USEDIN(SELF,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'APPLIED_DOCUMENT_USAGE_CONSTRAINT_ASSIGNMENT.ITEMS') | (
            'FEATURE_BASED_PROCESS_PLANNING.DOCUMENT' IN TYPEOF(aduc\
            document_usage_constraint_assignment.
            assigned_document_usage.source)) ) | (
            'FEATURE_BASED_PROCESS_PLANNING.DOCUMENT_FILE' IN TYPEOF(
            doc)) )) >= 1)));
wr7 : ((NOT (SELF\characterized_object.description = 'marking')) OR
            (SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
            pd,'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
            impl_rep.used_representation.items |
   (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
            IN TYPEOF(it)) AND (it.name = 'marking text')) )) = 1)) ))
            = 0)) )) = 0));
wr8 : ((NOT (SELF\characterized_object.description = 'thread')) OR (
            SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
            pd,'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
            impl_rep.used_representation.items |
   (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
            IN TYPEOF(it)) AND (it.name = 'removal direction') AND ((it
            .description = 'internal') OR (it.description
            = 'external'))) ))
            = 1)) )) = 0)) )) = 0));
wr9 : ((NOT (SELF\characterized_object.description = 'thread')) OR (
            SIZEOF(QUERY ( pd <* USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
            pd,'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
```

 ©ISO 2000 - All rights reserved

```
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |
  (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
          IN TYPEOF(it)) AND (it.name = 'qualifier')) )) <= 1)) )) =
          0)) )) = 0));
wr10: ((NOT (SELF\characterized_object.description = 'thread')) OR (
          SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |
   (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
          IN TYPEOF(it)) AND (it.name = 'hand')) )) = 1)) )) = 0)) ))
          = 0));
wr11: ((NOT (SELF\characterized_object.description = 'thread')) OR (
          SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |
   (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
          IN TYPEOF(it)) AND (it.name = 'fit class')) )) = 1)) ))
          = 0)) ))
          = 0));
wr12: ((NOT (SELF\characterized_object.description = 'thread')) OR (
          SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |
   (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
          IN TYPEOF(it)) AND (it.name = 'form')) )) = 1)) )) = 0)) ))
          = 0));
wr13: ((NOT (SELF\characterized_object.description = 'thread')) OR (
          SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
         * TYPEOF(it)) = 2) AND (it.name = 'major diameter')) ))
         = 1)) ))
         = 0)) )) = 0));
wr14: ((NOT (SELF\characterized_object.description = 'thread')) OR (
```

```
                SIZEOF(QUERY ( pd <*USEDIN(SELF,
            'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                  | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
                pd,'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items |((SIZEOF([
              'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
                'FEATURE_BASED_PROCESS_PLANNING.RATIO_MEASURE_WITH_UNIT'] *
                 TYPEOF(it)) = 2) AND (it.name = 'number of threads')) )) =
                 1)) )) = 0)) )) = 0));
     wr15: ((NOT (SELF\characterized_object.description IN ['knurl',
                'thread'])) OR (SIZEOF(QUERY ( pds <* QUERY ( pd <*
                USEDIN( SELF,
            'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                  | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
                IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
                pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
                 | (SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'applied shape') AND
              ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
                IN TYPEOF(sar))) ) | (
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT' IN TYPEOF(sdr
                .relating_shape_aspect)) )) = 1) )) <= 1)) )) = 0));
END_ENTITY; -- externally_defined_feature_definition

ENTITY externally_defined_item;
    item_id : source_item;
    source  : external_source;
END_ENTITY; -- externally_defined_item

ENTITY face
  SUPERTYPE OF (ONEOF (face_surface,oriented_face))
  SUBTYPE OF (topological_representation_item);
    bounds : SET [1:?] OF face_bound;
  WHERE
    wr1: (NOT mixed_loop_type_set(list_to_set(list_face_loops(SELF))));
    wr2: (SIZEOF(QUERY ( temp <* bounds | (
              'FEATURE_BASED_PROCESS_PLANNING.FACE_OUTER_BOUND' IN TYPEOF(
              temp)) )) <= 1);
END_ENTITY; -- face

ENTITY face_bound
  SUBTYPE OF (topological_representation_item);
    bound       : loop;
    orientation : BOOLEAN;
END_ENTITY; -- face_bound

ENTITY face_outer_bound
  SUBTYPE OF (face_bound);
END_ENTITY; -- face_outer_bound

ENTITY face_shape_representation
  SUBTYPE OF (shape_representation);
  WHERE
    wr1: (SIZEOF(SELF.items) >= 1);
```

 ©ISO 2000 - All rights reserved

```
    wr2: (SIZEOF(QUERY ( it <* SELF.items | (NOT ((
             'FEATURE_BASED_PROCESS_PLANNING.FACE_SURFACE' IN TYPEOF(it))
             OR ('FEATURE_BASED_PROCESS_PLANNING.ORIENTED_FACE' IN
             TYPEOF(it)))) )) = 0);
END_ENTITY; -- face_shape_representation

ENTITY face_shape_representation_relationship
  SUBTYPE OF (representation_relationship);
  WHERE
    wr1: ('FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION' IN
             TYPEOF(SELF.rep_1));
    wr2: ('FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION' IN
             TYPEOF(SELF.rep_2));
END_ENTITY; -- face_shape_representation_relationship

ENTITY face_surface
  SUBTYPE OF (face, geometric_representation_item);
    face_geometry : surface;
    same_sense    : BOOLEAN;
  WHERE
    wr1: (NOT ('FEATURE_BASED_PROCESS_PLANNING.ORIENTED_SURFACE' IN
             TYPEOF(face_geometry)));
END_ENTITY; -- face_surface

ENTITY feature_based_pp_action_assignment
  SUBTYPE OF (action_assignment);
    items : SET [1:?] OF feature_based_pp_action_item;
END_ENTITY; -- feature_based_pp_action_assignment

ENTITY feature_based_pp_action_request_assignment
  SUBTYPE OF (action_request_assignment);
    items : SET [1:?] OF feature_based_pp_action_request_item;
END_ENTITY; -- feature_based_pp_action_request_assignment

ENTITY feature_based_pp_approval_assignment
  SUBTYPE OF (approval_assignment);
    items : SET [1:?] OF feature_based_pp_approved_item;
END_ENTITY; -- feature_based_pp_approval_assignment

ENTITY feature_based_pp_date_assignment
  SUBTYPE OF (date_assignment);
    items : SET [1:?] OF feature_based_pp_dated_item;
END_ENTITY; -- feature_based_pp_date_assignment

ENTITY feature_based_pp_organization_assignment
  SUBTYPE OF (organization_assignment);
    items : SET [1:?] OF feature_based_pp_organization_item;
END_ENTITY; -- feature_based_pp_organization_assignment

ENTITY feature_based_pp_person_and_organization_assignment
  SUBTYPE OF (person_and_organization_assignment);
    items : SET [1:?] OF feature_based_pp_person_and_organization_item;
END_ENTITY; -- feature_based_pp_person_and_organization_assignment

ENTITY feature_based_pp_security_classification_assignment
  SUBTYPE OF (security_classification_assignment);
    items : SET [1:?] OF feature_based_pp_classified_item;
END_ENTITY; -- feature_based_pp_security_classification_assignment

ENTITY feature_component_definition
  SUBTYPE OF (characterized_object);
```

©ISO 2000 - All rights reserved

```
    WHERE
      wr1: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
            | (NOT (SIZEOF(USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')) =
            1)) ))
            = 0);
  END_ENTITY; -- feature_component_definition

  ENTITY feature_component_relationship
    SUPERTYPE OF (ONEOF
    (pattern_omit_membership,pattern_offset_membership))
    SUBTYPE OF (shape_aspect_relationship);
    WHERE
      wr1: ((SIZEOF([
                'FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_SHAPE_ASPECT',
                'FEATURE_BASED_PROCESS_PLANNING.REPLICATE_FEATURE',
                'FEATURE_BASED_PROCESS_PLANNING.TRANSITION_FEATURE',
                'FEATURE_BASED_PROCESS_PLANNING.MODIFIED_PATTERN'] * TYPEOF(
                SELF.relating_shape_aspect)) = 1) OR (
                'FEATURE_BASED_PROCESS_PLANNING.FEATURE_DEFINITION' IN
                TYPEOF(SELF.relating_shape_aspect.of_shape.definition)) OR
              ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
                IN TYPEOF(SELF.relating_shape_aspect.of_shape.definition)));
  END_ENTITY; -- feature_component_relationship

  ENTITY feature_definition
    SUBTYPE OF (characterized_object);
    WHERE
      wr1: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) )) = 1)) )) = 0);
      wr2: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
                USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items | ((
                'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
                AND (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
      wr3: (SIZEOF(['FEATURE_BASED_PROCESS_PLANNING.BOSS',
                'FEATURE_BASED_PROCESS_PLANNING.TURNED_KNURL',
                'FEATURE_BASED_PROCESS_PLANNING.THREAD',
                'FEATURE_BASED_PROCESS_PLANNING.MARKING',
                'FEATURE_BASED_PROCESS_PLANNING.RIB_TOP',
                'FEATURE_BASED_PROCESS_PLANNING.ROUND_HOLE',
                'FEATURE_BASED_PROCESS_PLANNING.OUTSIDE_PROFILE',
                'FEATURE_BASED_PROCESS_PLANNING.POCKET',
                'FEATURE_BASED_PROCESS_PLANNING.REMOVAL_VOLUME',
                'FEATURE_BASED_PROCESS_PLANNING.REVOLVED_PROFILE',
                'FEATURE_BASED_PROCESS_PLANNING.OUTER_ROUND',
                'FEATURE_BASED_PROCESS_PLANNING.FLAT_FACE',
```

 ©ISO 2000 - All rights reserved

```
              'FEATURE_BASED_PROCESS_PLANNING.PROTRUSION',
              'FEATURE_BASED_PROCESS_PLANNING.ROUNDED_END',
              'FEATURE_BASED_PROCESS_PLANNING.SLOT',
              'FEATURE_BASED_PROCESS_PLANNING.SPHERICAL_CAP',
              'FEATURE_BASED_PROCESS_PLANNING.STEP',
              'FEATURE_BASED_PROCESS_PLANNING.COMPOUND_FEATURE',
              'FEATURE_BASED_PROCESS_PLANNING.REPLICATE_FEATURE',
 'FEATURE_BASED_PROCESS_PLANNING.EXTERNALLY_DEFINED_FEATURE_DEFINITION']
              * TYPEOF(SELF)) = 1);
    wr4: ((NOT (SIZEOF(['FEATURE_BASED_PROCESS_PLANNING.ROUND_HOLE',
              'FEATURE_BASED_PROCESS_PLANNING.BOSS',
              'FEATURE_BASED_PROCESS_PLANNING.OUTSIDE_PROFILE',
              'FEATURE_BASED_PROCESS_PLANNING.REMOVAL_VOLUME',
              'FEATURE_BASED_PROCESS_PLANNING.FLAT_FACE',
              'FEATURE_BASED_PROCESS_PLANNING.POCKET',
              'FEATURE_BASED_PROCESS_PLANNING.PROTRUSION',
              'FEATURE_BASED_PROCESS_PLANNING.RIB_TOP',
              'FEATURE_BASED_PROCESS_PLANNING.ROUNDED_END',
              'FEATURE_BASED_PROCESS_PLANNING.SLOT',
              'FEATURE_BASED_PROCESS_PLANNING.STEP'] * TYPEOF(SELF)) = 1))
              OR (SIZEOF(QUERY ( pd <* USEDIN(SELF,
              'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
              'DEFINITION') | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
            (('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
              IN TYPEOF(pdr.used_representation)) AND (pdr.
              used_representation.name = 'maximum feature limit')) )) =
              1)) )) <= 1));
END_ENTITY; -- feature_definition

ENTITY feature_pattern
  SUBTYPE OF (replicate_feature);
  WHERE
    wr1: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) AND (SIZEOF(QUERY ( srwp_i <* pdr.
              used_representation.items | (NOT (
              'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN
              TYPEOF(srwp_i))) ))
              > 0)) )) > 0) )) = 0);
    wr2: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items | ((
              'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
              AND (it.name = 'base feature placement')) )) > 1)) )) = 0))
              )) = 0);
END_ENTITY; -- feature_pattern
```

©ISO 2000 - All rights reserved

```
ENTITY fillet
  SUBTYPE OF (transition_feature);
  WHERE
    wr1: ((NOT (SELF\shape_aspect.description = 'constant radius')) OR (
             SIZEOF(QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) )) = 1)) )) = 0));
    wr2: ((NOT (SELF\shape_aspect.description = 'constant radius')) OR (
             SIZEOF(QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
             USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) ) | ((NOT (SIZEOF(impl_rep.
             used_representation.items) >= 1)) AND (SIZEOF(impl_rep.
             used_representation.items) <= 3)) )) = 0)) )) = 0));
    wr3: ((NOT (SELF.description = 'constant radius')) OR (SIZEOF(
             QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
             USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
             impl_rep.used_representation.items |
             ((SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
             'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
              TYPEOF(it)) = 2) AND (it.name = 'radius')) )) = 1)) )) =
             0)) ))  = 0));
    wr4: ((NOT (SELF.description = 'constant radius')) OR (SIZEOF(
             QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
             USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
             impl_rep.used_representation.items |((SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
             'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
              TYPEOF(it)) = 2) AND (it.name = 'first offset')) )) <= 1))
             )) = 0)) )) = 0));
    wr5: ((NOT (SELF.description = 'constant radius')) OR (SIZEOF(
             QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
             USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
```

 ©ISO 2000 - All rights reserved

```
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items |((SIZEOF([
             'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
               TYPEOF(it)) = 2) AND (it.name = 'second offset')) )) <= 1))
              )) = 0)) )) = 0));
    wr6: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
              'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
              'DEFINITION') | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
              IN TYPEOF(pdr.used_representation)) AND (pdr.
              used_representation.name = 'fillet face')) )) = 1)) )) = 0);
    wr7: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
              'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
              'DEFINITION') | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
              IN TYPEOF(pdr.used_representation)) AND (pdr.
              used_representation.name = 'first face shape')) )) = 1)) ))
              = 0);
    wr8: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
              'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
              'DEFINITION') | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
              IN TYPEOF(pdr.used_representation)) AND (pdr.
              used_representation.name = 'second face shape')) )) = 1)) ))
              = 0);
END_ENTITY; -- fillet

ENTITY flat_face
  SUBTYPE OF (feature_definition);
  WHERE
    wr1: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
              IN TYPEOF(pdr.used_representation)) AND (pdr.
              used_representation.name = 'removal direction')) )) = 1)) ))
              = 0);
    wr2: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
              'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
               TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
              sa_occ.description = 'course of travel occurrence') AND (
              SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
              + 'RELATED_SHAPE_ASPECT') | ((sar.description =
              'path feature component usage') AND
             ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
```

©ISO 2000 - All rights reserved

```
                IN TYPEOF(sar))) ) | ((
                'FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT' IN
                TYPEOF(sdr.relating_shape_aspect)) AND (sdr.
                relating_shape_aspect.description = 'linear') AND (sdr.name
                = 'course of travel')) )) = 1)) )) = 1)) )) = 0);
      wr3: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
                 TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
                sa_occ.description = 'removal boundary occurrence') AND (
                SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'profile usage') AND
               ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
                IN TYPEOF(sar))) ) | ((
                'FEATURE_BASED_PROCESS_PLANNING.LINEAR_PROFILE' IN TYPEOF(
                sdr.relating_shape_aspect)) AND (sdr.name =
                'removal boundary')) )) = 1)) )) = 1)) )) = 0);
      wr4: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
                 TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
                sa_occ.description = 'enclosed boundary occurrence') AND (
                SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'profile usage') AND
               ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
                IN TYPEOF(sar))) ) | ((SIZEOF([
                'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE',
                'FEATURE_BASED_PROCESS_PLANNING.NGON_CLOSED_PROFILE',
                'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE',
                'FEATURE_BASED_PROCESS_PLANNING.CLOSED_PATH_PROFILE'] *
                TYPEOF(sdr.relating_shape_aspect)) = 1) AND (sdr.
                relating_shape_aspect.description = 'boundary')) )) = 1)) ))
                <= 1)) )) = 0);
  END_ENTITY; -- flat_face

  ENTITY flatness_tolerance
    SUBTYPE OF (geometric_tolerance);
    WHERE
      wr1: (NOT (('FEATURE_BASED_PROCESS_PLANNING.' +
                'GEOMETRIC_TOLERANCE_WITH_DATUM_REFERENCE') IN
TYPEOF(SELF)));
  END_ENTITY; -- flatness_tolerance

  ENTITY founded_item;
  END_ENTITY; -- founded_item

  ENTITY functionally_defined_transformation;
      name        : label;
      description : OPTIONAL text;
  END_ENTITY; -- functionally_defined_transformation

  ENTITY geometric_representation_context
    SUBTYPE OF (representation_context);
```

 ©ISO 2000 - All rights reserved

```
      coordinate_space_dimension : dimension_count;
  END_ENTITY; -- geometric_representation_context

  ENTITY geometric_representation_item
    SUPERTYPE OF (ONEOF (point,direction,vector,placement,
        cartesian_transformation_operator,curve,surface,edge_curve,
        face_surface,vertex_point,solid_model))
    SUBTYPE OF (representation_item);
    DERIVE
      dim : dimension_count := dimension_of(SELF);
    WHERE
      wr1: (SIZEOF(QUERY ( using_rep <* using_representations(SELF) | (
               NOT
('FEATURE_BASED_PROCESS_PLANNING.GEOMETRIC_REPRESENTATION_CONTEXT'
               IN TYPEOF(using_rep.context_of_items))) )) = 0);
  END_ENTITY; -- geometric_representation_item

  ENTITY geometric_tolerance;
      name                    : label;
      description             : text;
      magnitude               : measure_with_unit;
      toleranced_shape_aspect : shape_aspect;
    WHERE
      wr1: (magnitude.value_component >= 0);
  END_ENTITY; -- geometric_tolerance

  ENTITY geometric_tolerance_relationship;
      name                          : label;
      description                   : text;
      relating_geometric_tolerance : geometric_tolerance;
      related_geometric_tolerance  : geometric_tolerance;
  END_ENTITY; -- geometric_tolerance_relationship

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

  ENTITY geometric_tolerance_with_defined_unit
    SUBTYPE OF (geometric_tolerance);
      unit_size : measure_with_unit;
    WHERE
      wr1: (unit_size.value_component > 0);
  END_ENTITY; -- geometric_tolerance_with_defined_unit

  ENTITY global_uncertainty_assigned_context
    SUBTYPE OF (representation_context);
      uncertainty : SET [1:?] OF uncertainty_measure_with_unit;
  END_ENTITY; -- global_uncertainty_assigned_context

  ENTITY global_unit_assigned_context
    SUBTYPE OF (representation_context);
      units : SET [1:?] OF unit;
  END_ENTITY; -- global_unit_assigned_context

  ENTITY group;
      name        : label;
      description : OPTIONAL text;
  END_ENTITY; -- group

  ENTITY group_assignment
```

©ISO 2000 - All rights reserved

```
    ABSTRACT SUPERTYPE;
      assigned_group : group;
    DERIVE
      role : object_role := get_role(SELF);
    WHERE
      wr1: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
               'ROLE_ASSOCIATION.ITEM_WITH_ROLE')) <= 1);
  END_ENTITY; -- group_assignment

  ENTITY group_relationship;
      name           : label;
      description    : OPTIONAL text;
      relating_group : group;
      related_group  : group;
  END_ENTITY; -- group_relationship

  ENTITY hole_bottom
    SUBTYPE OF (shape_aspect);
    WHERE
      wr1 : ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
                 IN TYPEOF(SELF.of_shape.definition));
      wr2 : (SELF.description IN ['through','flat','flat with radius',
                 'flat with taper','spherical','conical']);
      wr3 : ((NOT (SELF.description = 'through')) OR (SIZEOF(
                 QUERY ( pd <*USEDIN(SELF,
            'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                  | (NOT (SIZEOF(USEDIN(pd,'FEATURE_BASED_PROCESS_PLANNING.'
                 + 'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION')) = 0))
                 ))  = 0));
      wr4 : ((NOT (SELF.description IN ['flat with radius',
                 'flat with taper','spherical','conical'])) OR (SIZEOF(
                 QUERY ( pd <*USEDIN(SELF,
            'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                  | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
                 'FEATURE_BASED_PROCESS_PLANNING.' +
                 'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                 'FEATURE_BASED_PROCESS_PLANNING.' +
                 'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                 used_representation)) )) = 1)) )) = 0));
      wr5 : ((NOT (SELF.description = 'flat')) OR (SIZEOF(QUERY ( pd <*
                USEDIN(SELF,
            'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                  | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
                 pd,'FEATURE_BASED_PROCESS_PLANNING.' +
                 'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                 'FEATURE_BASED_PROCESS_PLANNING.' +
                 'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                 used_representation)) ) | (NOT (SIZEOF(impl_rep.
                 used_representation.items) = 0)) )) = 0)) )) = 0));
      wr6 : ((NOT (SELF.description IN ['flat with radius','spherical']))
                 OR (SIZEOF(QUERY ( pd <*USEDIN(SELF,
            'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                  | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
                 pd,'FEATURE_BASED_PROCESS_PLANNING.' +
                 'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                 'FEATURE_BASED_PROCESS_PLANNING.' +
                 'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                 used_representation)) ) | (NOT (SIZEOF(impl_rep.
                 used_representation.items) = 1)) )) = 0)) )) = 0));
      wr7 : ((NOT (SELF.description = 'flat with taper')) OR (SIZEOF(
                 QUERY ( pd <*USEDIN(SELF,
```

 ©ISO 2000 - All rights reserved

```
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(impl_rep.
          used_representation.items) = 2)) )) = 0)) )) = 0));
wr8 : ((NOT (SELF.description = 'conical')) OR (SIZEOF(
          QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | ((NOT (SIZEOF(impl_rep.
          used_representation.items) >= 1)) AND (SIZEOF(impl_rep.
          used_representation.items) <= 2)) )) = 0)) )) = 0));
wr9 : ((SELF.description = 'through') OR (SIZEOF(QUERY ( fcr <*
          QUERY ( sar <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
          + 'RELATING_SHAPE_ASPECT') | ((sar.description =
          'hole bottom usage') AND
     ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
          IN TYPEOF(sar))) ) | (fcr.name IN ['hole depth start',
          'hole depth end']) )) >= 1));
wr10: ((NOT (SELF.description = 'flat with radius')) OR (SIZEOF(
          QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'corner radius')) )) =
           1)) )) = 0)) )) = 0));
wr11: ((NOT (SELF.description = 'spherical')) OR (SIZEOF(
          QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'radius')) )) = 1)) )) =
          0)) )) = 0));
wr12: ((NOT (SELF.description = 'conical')) OR (SIZEOF(
          QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
```

```
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items |((SIZEOF([
             'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
                'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
                * TYPEOF(it)) = 2) AND (it.name = 'tip radius')) )) <= 1))
                )) = 0)) )) = 0));
      wr13: ((NOT (SELF.description = 'conical')) OR (SIZEOF(
                QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
                pd,'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items |((SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
            'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
                * TYPEOF(it)) = 2) AND (it.name = 'tip angle')) )) = 1)) ))
                = 0)) )) = 0));
      wr14: (SIZEOF(QUERY ( fcr <* QUERY ( sar <*USEDIN(SELF,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATING_SHAPE_ASPECT') | ((sar.description =
                'hole bottom usage') AND
           ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
                IN TYPEOF(sar))) ) | ((fcr.related_shape_aspect.description
                = 'bottom condition occurrence') AND (
                'FEATURE_BASED_PROCESS_PLANNING.ROUND_HOLE' IN TYPEOF(fcr.
                related_shape_aspect.of_shape.definition))) )) >= 1);
      wr15: ((NOT (SELF.description = 'flat with taper')) OR (SIZEOF(
                QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
                pd,'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items |((SIZEOF([
              'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
                'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
                * TYPEOF(it)) = 2) AND (it.name = 'final diameter')) )) =
                1)) )) = 0)) )) = 0));
      wr16: ((NOT (SELF.description = 'flat with taper')) OR (SIZEOF(
                QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
                pd,'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items |((SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
            'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
                * TYPEOF(it)) = 2) AND (it.name = 'taper diameter')) )) =
                1)) )) = 0)) )) = 0));
END_ENTITY; -- hole_bottom
```

 ©ISO 2000 - All rights reserved

```
ENTITY hyperbola
  SUBTYPE OF (conic);
    semi_axis      : positive_length_measure;
    semi_imag_axis : positive_length_measure;
END_ENTITY; -- hyperbola

ENTITY id_attribute;
    attribute_value : identifier;
    identified_item : id_attribute_select;
END_ENTITY; -- id_attribute

ENTITY instanced_feature
  SUBTYPE OF (feature_definition, shape_aspect);
  WHERE
    wr1: ('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION' IN TYPEOF(
             SELF.of_shape.definition));
    wr2: SELF.product_definitional;
END_ENTITY; -- instanced_feature

ENTITY length_measure_with_unit
  SUBTYPE OF (measure_with_unit);
  WHERE
    wr1: ('FEATURE_BASED_PROCESS_PLANNING.LENGTH_UNIT' IN TYPEOF(SELF\
             measure_with_unit.unit_component));
END_ENTITY; -- length_measure_with_unit

ENTITY length_unit
  SUBTYPE OF (named_unit);
  WHERE
    wr1: ((SELF\named_unit.dimensions.length_exponent = 1) AND (SELF\
             named_unit.dimensions.mass_exponent = 0) AND (SELF\
             named_unit.dimensions.time_exponent = 0) AND (SELF\
             named_unit.dimensions.electric_current_exponent = 0) AND (
             SELF\named_unit.dimensions.
             thermodynamic_temperature_exponent = 0) AND (SELF\named_unit
             .dimensions.amount_of_substance_exponent = 0) AND (SELF\
             named_unit.dimensions.luminous_intensity_exponent = 0));
END_ENTITY; -- length_unit

ENTITY limits_and_fits;
    form_variance : label;
    zone_variance : label;
    grade         : label;
    source        : text;
END_ENTITY; -- limits_and_fits

ENTITY line
  SUBTYPE OF (curve);
    pnt : cartesian_point;
    dir : vector;
  WHERE
    wr1: (dir.dim = pnt.dim);
END_ENTITY; -- line

ENTITY line_profile_tolerance
  SUBTYPE OF (geometric_tolerance);
  WHERE
    wr1: ((NOT (('FEATURE_BASED_PROCESS_PLANNING.' +
             'GEOMETRIC_TOLERANCE_WITH_DATUM_REFERENCE') IN
             TYPEOF(SELF)))
             OR (SIZEOF(SELF\geometric_tolerance_with_datum_reference.
```

©ISO 2000 - All rights reserved

```
              datum_system) <= 3));
    wr2: (SIZEOF(QUERY ( sar <* USEDIN(SELF\geometric_tolerance.
              toleranced_shape_aspect,'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_ASPECT_RELATIONSHIP.RELATING_SHAPE_ASPECT') | (sar.
              name IN ['affected plane association',
              'resulting intersection curve association']) )) = 1);
END_ENTITY; -- line_profile_tolerance

ENTITY linear_profile
  SUBTYPE OF (shape_aspect);
  WHERE
    wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
              IN TYPEOF(SELF.of_shape.definition));
    wr2: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) )) = 1)) )) = 0);
    wr3: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(impl_rep.
              used_representation.items) = 2)) )) = 0)) )) = 0);
    wr4: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items | ((
              'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
              AND (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
    wr5: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items |((SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
               TYPEOF(it)) = 2) AND (it.name = 'profile length')) )) = 1))
              )) = 0)) )) = 0);
END_ENTITY; -- linear_profile

ENTITY location_shape_representation
  SUBTYPE OF (shape_representation);
```

 ©ISO 2000 - All rights reserved

```
  WHERE
    wr1: (SIZEOF(SELF.items) = 1);
    wr2: (SIZEOF(QUERY ( it <* SELF.items | (NOT (
         'FEATURE_BASED_PROCESS_PLANNING.POINT' IN TYPEOF(it))) )) =  0);
END_ENTITY; -- location_shape_representation

ENTITY loop
  SUPERTYPE OF (ONEOF (vertex_loop,edge_loop))
  SUBTYPE OF (topological_representation_item);
END_ENTITY; -- loop

ENTITY make_from_usage_option
  SUBTYPE OF (product_definition_usage);
    ranking           : INTEGER;
    ranking_rationale : text;
    quantity          : measure_with_unit;
  WHERE
    wr1: ((NOT ('NUMBER' IN TYPEOF(quantity.value_component))) OR (
             quantity.value_component > 0));
END_ENTITY; -- make_from_usage_option

ENTITY manifold_solid_brep
  SUBTYPE OF (solid_model);
    outer : closed_shell;
END_ENTITY; -- manifold_solid_brep

ENTITY mapped_item
  SUBTYPE OF (representation_item);
    mapping_source : representation_map;
    mapping_target : representation_item;
  WHERE
    wr1:
       acyclic_mapped_representation(using_representations(SELF),[SELF]);
END_ENTITY; -- mapped_item

ENTITY marking
  SUBTYPE OF (feature_definition);
  WHERE
    wr1: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) AND (2 <= SIZEOF(pdr.
             used_representation.items)) AND (SIZEOF(pdr.
             used_representation.items) <= 6)) )) = 1) )) = 1);
    wr2: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
             USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
             impl_rep.used_representation.items |
       (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
             IN TYPEOF(it)) AND (it.name = 'marking text')) )) = 1)) )) =
              0)) )) = 0);
```

©ISO 2000 - All rights reserved

```
wr3: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
         USEDIN(pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
         used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
         impl_rep.used_representation.items |
  (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
         IN TYPEOF(it)) AND (it.name = 'special instructions')) )) <=
          1)) )) = 0)) )) = 0);
wr4: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
         USEDIN(pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
         used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
         impl_rep.used_representation.items |
   (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
         IN TYPEOF(it)) AND (it.name = 'font name')) )) <= 1)) )) =
         0)) )) = 0);
wr5: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
         USEDIN(pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
         used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
         impl_rep.used_representation.items |((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
         'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
          TYPEOF(it)) = 2) AND (it.name = 'character height')) )) <=
         1)) )) = 0)) )) = 0);
wr6: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
         USEDIN(pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
         used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
         impl_rep.used_representation.items |((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
         'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
          TYPEOF(it)) = 2) AND (it.name = 'character spacing')) )) <=
          1)) )) = 0)) )) = 0);
wr7: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (
         'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
          TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
         'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | (
         SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
```

 ©ISO 2000 - All rights reserved

```
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
              + 'RELATED_SHAPE_ASPECT') | ((sar.description =
              'applied shape') AND
             ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
              IN TYPEOF(sar))) ) | (
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT' IN TYPEOF(sdr.
              relating_shape_aspect)) )) = 1) )) = 1)) )) = 0);
END_ENTITY; -- marking

ENTITY material_designation;
    name        : label;
    definitions : SET [1:?] OF characterized_definition;
END_ENTITY; -- material_designation

ENTITY material_property
  SUBTYPE OF (property_definition);
  UNIQUE
    ur1 : name, definition;
  WHERE
    wr1: (('FEATURE_BASED_PROCESS_PLANNING.CHARACTERIZED_OBJECT' IN
              TYPEOF(SELF\property_definition.definition)) OR (SIZEOF(
              bag_to_set(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION')) -
              QUERY ( temp <* bag_to_set(USEDIN(SELF,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION')) | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'MATERIAL_PROPERTY_REPRESENTATION') IN TYPEOF(temp)) )) =
               0));
END_ENTITY; -- material_property

ENTITY material_property_representation
  SUBTYPE OF (property_definition_representation);
    dependent_environment : data_environment;
END_ENTITY; -- material_property_representation

ENTITY measure_qualification;
    name              : label;
    description       : text;
    qualified_measure : measure_with_unit;
    qualifiers        : SET [1:?] OF value_qualifier;
  WHERE
    wr1: (SIZEOF(QUERY ( temp <* qualifiers | (
              'FEATURE_BASED_PROCESS_PLANNING.PRECISION_QUALIFIER' IN
              TYPEOF(temp)) )) < 2);
END_ENTITY; -- measure_qualification

ENTITY measure_representation_item
  SUBTYPE OF (representation_item, measure_with_unit);
END_ENTITY; -- measure_representation_item

ENTITY measure_with_unit
  SUPERTYPE OF (ONEOF (length_measure_with_unit,
      plane_angle_measure_with_unit,solid_angle_measure_with_unit,
      ratio_measure_with_unit));
    value_component : measure_value;
    unit_component  : unit;
  WHERE
    wr1: valid_units(SELF);
END_ENTITY; -- measure_with_unit
```

©ISO 2000 - All rights reserved

```
ENTITY modified_geometric_tolerance
  SUBTYPE OF (geometric_tolerance);
    modifier : limit_condition;
END_ENTITY; -- modified_geometric_tolerance

ENTITY modified_pattern
  SUBTYPE OF (shape_aspect);
  WHERE
    wr1: (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
              + 'RELATING_SHAPE_ASPECT') |
         ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
              IN TYPEOF(sar)) ) | ((SIZEOF([
              'FEATURE_BASED_PROCESS_PLANNING.REPLICATE_FEATURE',
              'FEATURE_BASED_PROCESS_PLANNING.INSTANCED_FEATURE'] *
              TYPEOF(fcr.related_shape_aspect.of_shape.definition)) >= 1)
              AND (fcr.description = 'base shape')) )) = 1);
    wr2: (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
              + 'RELATING_SHAPE_ASPECT') |
         ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
              IN TYPEOF(sar)) ) | ((SIZEOF([
              'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN',
              'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN'] *
              TYPEOF(fcr.related_shape_aspect.of_shape.definition)) = 1)
              AND (fcr.description = 'base pattern')) )) = 1);
    wr3: (SIZEOF(QUERY ( sar <* USEDIN(SELF,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
              + 'RELATING_SHAPE_ASPECT') | (SIZEOF(QUERY ( msar <* USEDIN(
              sar.related_shape_aspect,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
              + 'RELATED_SHAPE_ASPECT') | ((SIZEOF([
              'FEATURE_BASED_PROCESS_PLANNING.PATTERN_OFFSET_MEMBERSHIP',
              'FEATURE_BASED_PROCESS_PLANNING.PATTERN_OMIT_MEMBERSHIP'] *
              TYPEOF(sar)) = 1) AND (sar.description = 'modified pattern')
              AND (sar :<>: msar)) )) >= 1) )) = 0);
END_ENTITY; -- modified_pattern

ENTITY name_attribute;
    attribute_value : label;
    named_item      : name_attribute_select;
END_ENTITY; -- name_attribute

ENTITY named_unit
  SUPERTYPE OF (ONEOF (si_unit,conversion_based_unit,
      context_dependent_unit) ANDOR ONEOF (length_unit,plane_angle_unit,
      solid_angle_unit,ratio_unit));
    dimensions : dimensional_exponents;
END_ENTITY; -- named_unit

ENTITY next_assembly_usage_occurrence
  SUBTYPE OF (assembly_component_usage);
END_ENTITY; -- next_assembly_usage_occurrence

ENTITY ngon_closed_profile
  SUBTYPE OF (shape_aspect);
  WHERE
    wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
              IN TYPEOF(SELF.of_shape.definition));
    wr2: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
```

 ©ISO 2000 - All rights reserved

```
          | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
         used_representation)) )) = 1)) )) = 0);
wr3: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
         USEDIN(pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
         used_representation)) ) | (NOT ((SIZEOF(impl_rep.
         used_representation.items) >= 3) AND (SIZEOF(impl_rep.
         used_representation.items) <= 4))) )) = 0)) )) = 0);
wr4: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (((
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
         used_representation)) AND (SIZEOF(QUERY ( srwp_i <* pdr.
         used_representation.items | ((srwp_i.name = 'orientation')
         OR (srwp_i.name = 'number of sides') OR (srwp_i.name =
         'circumscribed diameter') OR (srwp_i.name = 'corner radius')
         OR (srwp_i.name = 'diameter across flats')) )) = SIZEOF(pdr.
         used_representation.items))) )) = 1) )) = 1);
wr5: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
         USEDIN(pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
         used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
         impl_rep.used_representation.items | ((
         'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
         AND (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
wr6: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
         USEDIN(pd,
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
         'FEATURE_BASED_PROCESS_PLANNING.' +
         'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
         used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
         impl_rep.used_representation.items |
       (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
         IN TYPEOF(it)) AND (
         'FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE' IN TYPEOF(it\
         measure_with_unit.value_component)) AND (it.name =
         'number of sides')) )) = 1)) )) = 0)) )) = 0);
wr7: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
          | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
          USEDIN(pd,
```

```
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items |((SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
               TYPEOF(it)) = 2) AND (it.name IN ['circumscribed diameter',
              'diameter across flats'])) )) = 1)) )) = 0)) )) = 0);
    wr8: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items |((SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
               TYPEOF(it)) = 2) AND (it.name = 'corner radius')) )) <= 1))
              )) = 0)) )) = 0);
END_ENTITY; -- ngon_closed_profile

ENTITY ngon_shape_representation
  SUBTYPE OF (shape_representation_with_parameters);
  WHERE
    wr1: (SIZEOF(SELF.items) = 5);
    wr2: (SIZEOF(QUERY ( it <* SELF.items | ((
              'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
              AND (it.name = 'orientation')) )) = 1);
    wr3: (SIZEOF(QUERY ( it <* SELF.items |((SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
               TYPEOF(it)) = 2) AND (it.name = 'length')) )) = 1);
    wr4: (SIZEOF(QUERY ( it <* SELF.items |((SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
               TYPEOF(it)) = 2) AND (it.name = 'corner radius')) )) = 1);
    wr5: (SIZEOF(QUERY ( it <* SELF.items |((SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
               TYPEOF(it)) = 2) AND (it.name IN ['circumscribed diameter',
              'diameter across flats'])) )) = 1);
    wr6: (SIZEOF(QUERY ( it <* SELF.items |
            (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
              IN TYPEOF(it)) AND (
              'FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE' IN TYPEOF(it\
              measure_with_unit.value_component)) AND (it.name =
              'number of sides')) )) = 1);
END_ENTITY; -- ngon_shape_representation

ENTITY object_role;
    name        : label;
    description : OPTIONAL text;
END_ENTITY; -- object_role

ENTITY open_path_profile
  SUBTYPE OF (shape_aspect);
```

 ©ISO 2000 - All rights reserved

```
  WHERE
    wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
              IN TYPEOF(SELF.of_shape.definition));
    wr2: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) )) = 1)) )) = 0);
    wr3: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(impl_rep.
              used_representation.items) = 1)) )) = 0)) )) = 0);
    wr4: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items | ((
              'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
              AND (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
    wr5: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (
              'FEATURE_BASED_PROCESS_PLANNING.PATH_SHAPE_REPRESENTATION'
              IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0);
    wr6: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
            (('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
              IN TYPEOF(pdr.used_representation)) AND (pdr.
              used_representation.name = 'profile limit')) )) <= 1)) )) =
              0);
END_ENTITY; -- open_path_profile

ENTITY open_shell
  SUBTYPE OF (connected_face_set);
END_ENTITY; -- open_shell

ENTITY ordered_part
  SUBTYPE OF (action_assignment, characterized_object);
    items : SET [1:?] OF feature_based_pp_ordered_item;
  WHERE
    wr1:(SIZEOF(USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION'))
```

```
              = 1);
    wr2: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(USEDIN(pd,'FEATURE_BASED_PROCESS_PLANNING.'
              + 'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION')) = 1)) ))
              = 0);
    wr3: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (NOT (
              SIZEOF(items) = 1)) )) = 0)) )) = 0);
    wr4: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (NOT (
              SIZEOF(QUERY ( it <* pdr.used_representation.items |
            (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
              IN TYPEOF(it)) AND (it.name = 'quantity')) )) = 1)) )) = 0))
              )) = 0);
END_ENTITY; -- ordered_part

ENTITY ordinal_date
  SUBTYPE OF (date);
    day_component : day_in_year_number;
  WHERE
    wr1: (((NOT leap_year(SELF.year_component)) AND (1 <= day_component)
              AND (day_component <= 365)) OR (leap_year(SELF.
              year_component) AND (1 <= day_component) AND (day_component
              <= 366)));
END_ENTITY; -- ordinal_date

ENTITY organization;
    id          : OPTIONAL identifier;
    name        : label;
    description : OPTIONAL text;
END_ENTITY; -- organization

ENTITY organization_assignment
  ABSTRACT SUPERTYPE;
    assigned_organization : organization;
    role                  : organization_role;
END_ENTITY; -- organization_assignment

ENTITY organization_role;
    name : label;
  DERIVE
    description : text := get_description_value(SELF);
  WHERE
    wr1: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
              'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1);
END_ENTITY; -- organization_role

ENTITY organizational_address
  SUBTYPE OF (address);
    organizations : SET [1:?] OF organization;
    description   : OPTIONAL text;
END_ENTITY; -- organizational_address

ENTITY oriented_closed_shell
```

 ©ISO 2000 - All rights reserved

```
  SUBTYPE OF (closed_shell);
    closed_shell_element : closed_shell;
    orientation          : BOOLEAN;
  DERIVE
    SELF\connected_face_set.cfs_faces : SET [1:?] OF face :=
                                        conditional_reverse(SELF.
                                        orientation,SELF.
                                        closed_shell_element.cfs_faces);
  WHERE
    wr1: (NOT ('FEATURE_BASED_PROCESS_PLANNING.ORIENTED_CLOSED_SHELL' IN
              TYPEOF(SELF.closed_shell_element)));
END_ENTITY; -- oriented_closed_shell

ENTITY oriented_edge
  SUBTYPE OF (edge);
    edge_element : edge;
    orientation  : BOOLEAN;
  DERIVE
    SELF\edge.edge_start : vertex := boolean_choose(SELF.orientation,
                               SELF.edge_element.edge_start,SELF.
                               edge_element.edge_end);
    SELF\edge.edge_end   : vertex := boolean_choose(SELF.orientation,
                               SELF.edge_element.edge_end,SELF.
                               edge_element.edge_start);
  WHERE
    wr1: (NOT ('FEATURE_BASED_PROCESS_PLANNING.ORIENTED_EDGE' IN TYPEOF(
             SELF.edge_element)));
END_ENTITY; -- oriented_edge

ENTITY oriented_face
  SUBTYPE OF (face);
    face_element : face;
    orientation  : BOOLEAN;
  DERIVE
    SELF\face.bounds : SET [1:?] OF face_bound := conditional_reverse(
                          SELF.orientation,SELF.face_element.bounds);
  WHERE
    wr1: (NOT ('FEATURE_BASED_PROCESS_PLANNING.ORIENTED_FACE' IN TYPEOF(
             SELF.face_element)));
END_ENTITY; -- oriented_face

ENTITY oriented_open_shell
  SUBTYPE OF (open_shell);
    open_shell_element : open_shell;
    orientation        : BOOLEAN;
  DERIVE
    SELF\connected_face_set.cfs_faces : SET [1:?] OF face :=
                                            conditional_reverse(SELF.
                                            orientation,SELF.
                                            open_shell_element.cfs_faces);
  WHERE
    wr1: (NOT ('FEATURE_BASED_PROCESS_PLANNING.ORIENTED_OPEN_SHELL' IN
             TYPEOF(SELF.open_shell_element)));
END_ENTITY; -- oriented_open_shell

ENTITY oriented_path
  SUBTYPE OF (path);
    path_element : path;
    orientation  : BOOLEAN;
  DERIVE
    SELF\path.edge_list : LIST [1:?] OF UNIQUE oriented_edge :=
```

©ISO 2000 - All rights reserved

```
                                conditional_reverse(SELF.orientation,SELF.
                                path_element.edge_list);
    WHERE
      wr1: (NOT ('FEATURE_BASED_PROCESS_PLANNING.ORIENTED_PATH' IN TYPEOF(
                SELF.path_element)));
  END_ENTITY; -- oriented_path

  ENTITY outer_round
    SUBTYPE OF (feature_definition);
    WHERE
    wr1: ((NOT (SELF\characterized_object.description = 'outer diameter'))
                OR (SIZEOF(QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
    (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
                IN TYPEOF(pdr.used_representation)) AND (SIZEOF(pdr.
                used_representation.items) = 3)) )) = 1) )) = 1));
      wr2: ((NOT (SELF\characterized_object.description =
                'outer diameter to shoulder')) OR (SIZEOF(QUERY ( pd <*
               USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
    (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
                IN TYPEOF(pdr.used_representation)) AND (SIZEOF(pdr.
                used_representation.items) = 2)) )) = 1) )) = 1));
    wr3: (SELF\characterized_object.description IN ['outer diameter',
                'outer diameter to shoulder']);
    wr4: ((NOT (SELF\characterized_object.description = 'outer diameter'))
                OR (SIZEOF(QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
                USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items |((SIZEOF([
             'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
                'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
                 TYPEOF(it)) = 2) AND (it.name = 'length')) )) = 1)) )) =
                0)) ))  = 0));
      wr5: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
                USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items |
    ((SIZEOF(['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
                'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
                 TYPEOF(it)) = 2) AND (it.name = 'diameter')) )) = 1)) )) =
                0)) )) = 0);
      wr6: ((NOT (SELF\characterized_object.description =
```

 ©ISO 2000 - All rights reserved

```
            'outer diameter to shoulder')) OR (SIZEOF(QUERY ( pds <*
            QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (
            'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
             TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
            'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
            sa_occ.description = 'v-shape boundary occurrence') AND (
            SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
            'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
            + 'RELATED_SHAPE_ASPECT') | ((sar.description =
            'profile usage') AND
           ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
            IN TYPEOF(sar))) ) | ((
            'FEATURE_BASED_PROCESS_PLANNING.VEE_PROFILE' IN TYPEOF(sdr.
            relating_shape_aspect)) AND (sdr.relating_shape_aspect.
            description = 'v-shape')) )) = 1)) )) = 1)) )) = 0));
  wr7: ((NOT (SELF\characterized_object.description = 'outer diameter'))
            OR (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (
            'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
             TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
            'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
            sa_occ.description = 'reduced size occurrence') AND (SIZEOF(
            QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
            'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
            + 'RELATED_SHAPE_ASPECT') | ((sar.description =
            'taper usage') AND
        ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
            IN TYPEOF(sar))) ) | ((
            'FEATURE_BASED_PROCESS_PLANNING.TAPER' IN TYPEOF(sdr.
            relating_shape_aspect)) AND (sdr.name = 'reduced size')) ))
            = 1)) )) <= 1)) )) = 0));
END_ENTITY; -- outer_round

ENTITY outside_profile
  SUBTYPE OF (feature_definition);
  WHERE
    wr1 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
  (('FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
             IN TYPEOF(pdr.used_representation)) AND (SIZEOF(pdr.
             used_representation.items) = 1)) )) = 1) )) = 1);
    wr2 : (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (
             'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
             IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
             pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
              | (sa_occ.description IN ['boundary occurrence',
             'non-planar boundary occurrence',
             'partial circular boundary occurrence',
             'closed circular boundary occurrence',
             'open rectangular boundary occurrence',
             'closed rectangular boundary occurrence']) )) = 1)) )) =
             0);
    wr3 : ((NOT (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
```

```
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
             'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
             IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
             pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
              | (sa_occ.description = 'boundary occurrence') )) = 1)) ))
             = 0)) OR (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
             'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
             IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
             pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
              | (SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
             'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
             + 'RELATED_SHAPE_ASPECT') | ((sar.description =
             'profile usage') AND (('FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_DEFINING_RELATIONSHIP') IN TYPEOF(sar))) ) | ((
             SIZEOF([
             'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE',
             'FEATURE_BASED_PROCESS_PLANNING.NGON_CLOSED_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE',
             'FEATURE_BASED_PROCESS_PLANNING.CLOSED_PATH_PROFILE',
             'FEATURE_BASED_PROCESS_PLANNING.SQUARE_U_PROFILE',
             'FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE',
             'FEATURE_BASED_PROCESS_PLANNING.ROUNDED_U_PROFILE',
             'FEATURE_BASED_PROCESS_PLANNING.LINEAR_PROFILE',
             'FEATURE_BASED_PROCESS_PLANNING.VEE_PROFILE',
             'FEATURE_BASED_PROCESS_PLANNING.TEE_PROFILE',
             'FEATURE_BASED_PROCESS_PLANNING.OPEN_PATH_PROFILE'] *
             TYPEOF(sdr.relating_shape_aspect)) = 1) AND (sdr.
             relating_shape_aspect.description = 'outside boundary')) ))
             = 1) )) = 1)) )) = 0));
    wr4 : ((NOT (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
             'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
             IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
             pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
              | (sa_occ.description IN ['complex boundary occurrence',
             'partial circular boundary occurrence',
             'closed circular boundary occurrence',
             'open rectangular boundary occurrence',
             'closed rectangular boundary occurrence']) )) = 1)) )) =
             0))
             OR (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
             'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
             IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
             pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
              | (NOT (SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
             'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
             + 'RELATED_SHAPE_ASPECT') | ((sar.description =
             'profile floor usage') AND
        ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
             IN TYPEOF(sar))) ) | (
             'FEATURE_BASED_PROCESS_PLANNING.PROFILE_FLOOR' IN TYPEOF(
             sdr.relating_shape_aspect)) )) = 1)) )) = 0)) )) = 0));
    wr5 : ((NOT (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
```

 ©ISO 2000 - All rights reserved

```
            'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
            IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
            pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
             | (sa_occ.description IN ['outside boundary',
            'complex boundary occurrence',
            'partial circular boundary occurrence',
            'closed circular boundary occurrence',
            'open rectangular boundary occurrence',
            'closed rectangular boundary occurrence']) )) = 1)) )) =
            0))
            OR (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (
            'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
            IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
            pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
             | (SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
           'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
            + 'RELATED_SHAPE_ASPECT') | ((sar.description =
            'path feature component usage') AND ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_DEFINING_RELATIONSHIP') IN TYPEOF(sar))) ) | ((
            SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT'] *
            TYPEOF(sdr.relating_shape_aspect)) = 1) AND (sdr.name =
            'profile swept shape') AND (sdr.relating_shape_aspect.
            description = 'linear')) )) = 1) )) = 1)) )) = 0));
wr6 : ((NOT (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (
            'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
            IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
            pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
             | (sa_occ.description = 'complex boundary occurrence') ))
            = 1)) )) = 0)) OR (SIZEOF(QUERY ( pds <* QUERY ( pd <*
           USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (
            'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
            IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
            pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
             | (SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
            'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
            + 'RELATED_SHAPE_ASPECT') | ((sar.description =
            'profile usage') AND
          ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
            IN TYPEOF(sar))) ) | (SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE',
           'FEATURE_BASED_PROCESS_PLANNING.NGON_CLOSED_PROFILE',
           'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.CLOSED_PATH_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.SQUARE_U_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.ROUNDED_U_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.VEE_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.TEE_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.LINEAR_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.OPEN_PATH_PROFILE'] *
            TYPEOF(sdr.relating_shape_aspect)) = 1) )) = 1) )) = 1)) ))
            = 0));
wr7 : ((NOT (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
```

©ISO 2000 - All rights reserved

```
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
             'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
             IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
             pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
              | (sa_occ.description =
             'partial circular boundary occurrence') )) = 1)) )) = 0))
             OR (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
             'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
             IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
             pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
              | (SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
             'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
             + 'RELATED_SHAPE_ASPECT') | ((sar.description =
             'profile usage') AND
           ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
             IN TYPEOF(sar))) ) | (
             'FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE'
             IN TYPEOF(sdr.relating_shape_aspect)) )) = 1) )) = 1)) )) =
              0));
wr8 : ((NOT (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
             'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
             IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
             pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
              | (sa_occ.description =
             'closed circular boundary occurrence') )) = 1)) )) = 0)) OR
              (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
             'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
             IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
             pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
              | (SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
             'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
             + 'RELATED_SHAPE_ASPECT') | ((sar.description =
             'profile usage') AND
           ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
             IN TYPEOF(sar))) ) | (
             'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE' IN
              TYPEOF(sdr.relating_shape_aspect)) )) = 1) )) = 1)) )) =
              0));
wr9 : ((NOT (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
             'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
             IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
             pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
              | (sa_occ.description =
             'open rectangular boundary occurrence') )) = 1)) )) = 0))
             OR (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
             'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
             IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
             pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
              | (SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
            'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
```

 ©ISO 2000 - All rights reserved

```
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'profile usage') AND
              ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
                IN TYPEOF(sar))) ) | (
                'FEATURE_BASED_PROCESS_PLANNING.SQUARE_U_PROFILE' IN
                TYPEOF(sdr.relating_shape_aspect)) )) = 1) )) = 1)) )) =
                0));
     wr10: ((NOT (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
                IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
                pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
                 | (sa_occ.description =
                'closed rectangular boundary occurrence') )) = 1)) )) = 0))
                OR (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
                IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
                pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
                 | (SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'profile usage') AND
              ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
                IN TYPEOF(sar))) ) |
              ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE'
                IN TYPEOF(sdr.relating_shape_aspect)) )) = 1) )) = 1)) )) =
                 0));
     wr11: ((SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
                IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
                pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
                 | (sa_occ.description IN ['boundary occurrence',
                'complex boundary occurrence',
                'partial circular boundary occurrence',
                'closed circular boundary occurrence',
                'open rectangular boundary occurrence',
                'closed rectangular boundary occurrence']) )) = 1)) )) = 0)
                OR (SIZEOF(QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
         (('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
                IN TYPEOF(pdr.used_representation)) AND (pdr.
                used_representation.name = 'removal direction')) )) = 1))
                )) = 0));
END_ENTITY; -- outside_profile

ENTITY parabola
  SUBTYPE OF (conic);
    focal_dist : length_measure;
  WHERE
    wr1: (focal_dist <> 0);
END_ENTITY; -- parabola
```

```
ENTITY parallelism_tolerance
  SUBTYPE OF (geometric_tolerance_with_datum_reference);
  WHERE
    wr1: (SIZEOF(SELF\geometric_tolerance_with_datum_reference.
             datum_system) < 3);
END_ENTITY; -- parallelism_tolerance

ENTITY parametric_representation_context
  SUBTYPE OF (representation_context);
END_ENTITY; -- parametric_representation_context

ENTITY partial_circular_profile
  SUBTYPE OF (shape_aspect);
  WHERE
    wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
             IN TYPEOF(SELF.of_shape.definition));
    wr2: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) )) = 1)) )) = 0);
    wr3: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
             USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) ) | (NOT (SIZEOF(impl_rep.
             used_representation.items) >= 3)) )) = 0)) )) = 0);
    wr4: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
             USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
             impl_rep.used_representation.items | ((
             'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
             AND (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
    wr5: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
             USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
             impl_rep.used_representation.items |((SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
             'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
              TYPEOF(it)) = 2) AND (it.name = 'radius')) )) = 1)) )) =
             0)) )) = 0);
    wr6: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
```

 ©ISO 2000 - All rights reserved

```
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items |((SIZEOF([
           'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
           'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
              * TYPEOF(it)) = 2) AND (it.name = 'sweep angle')) )) = 1))
              )) = 0)) )) = 0);
    wr7: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
            (('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
              IN TYPEOF(pdr.used_representation)) AND (pdr.
              used_representation.name = 'profile limit')) )) <= 1)) )) =
              0);
END_ENTITY; -- partial_circular_profile

ENTITY path
  SUPERTYPE OF (ONEOF (edge_loop,oriented_path))
  SUBTYPE OF (topological_representation_item);
    edge_list : LIST [1:?] OF UNIQUE oriented_edge;
  WHERE
    wr1: path_head_to_tail(SELF);
END_ENTITY; -- path

ENTITY path_feature_component
  SUBTYPE OF (shape_aspect);
  WHERE
    wr1 : ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
              IN TYPEOF(SELF.of_shape.definition));
    wr2 : (SELF.description IN ['partial circular','complete circular',
              'linear','complex']);
    wr3 : ((NOT (SELF.description = 'complex')) OR (SIZEOF(
              QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) )) = 1)) )) = 0));
    wr4 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
              pd,'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items | ((
              'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
              AND (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
    wr5 : ((NOT (SELF.description = 'partial circular')) OR (SIZEOF(
              QUERY ( pd <*USEDIN(SELF,
```

©ISO 2000 - All rights reserved

```
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
            pd,'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(impl_rep.
            used_representation.items) = 3)) )) = 0)) )) = 0));
wr6 : ((NOT (SELF.description = 'partial circular')) OR (SIZEOF(
            QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
            pd,'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
            impl_rep.used_representation.items |((SIZEOF([
          'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
            'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
            * TYPEOF(it)) = 2) AND (it.name = 'radius')) )) = 1)) )) =
            0)) )) = 0));
wr7 : ((NOT (SELF.description = 'partial circular')) OR (SIZEOF(
            QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
            pd,'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
            impl_rep.used_representation.items |((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
            * TYPEOF(it)) = 2) AND (it.name = 'sweep angle')) )) = 1))
            ))  = 0)) )) = 0));
wr8 : ((NOT (SELF.description = 'complete circular')) OR (SIZEOF(
            QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
            pd,'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(impl_rep.
            used_representation.items) = 2)) )) = 0)) )) = 0));
wr9 : ((NOT (SELF.description = 'complete circular')) OR (SIZEOF(
            QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
            pd,'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
            impl_rep.used_representation.items |((SIZEOF([
          'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
            'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
            * TYPEOF(it)) = 2) AND (it.name = 'radius')) )) = 1)) )) =
            0)) )) = 0));
wr10: ((NOT (SELF.description = 'linear')) OR (SIZEOF(QUERY ( pd <*
```

 ©ISO 2000 - All rights reserved

```
              USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
              pd,'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(impl_rep.
              used_representation.items) = 2)) )) = 0)) )) = 0));
    wr11: ((NOT (SELF.description = 'linear')) OR (SIZEOF(QUERY ( pd <*
              USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
              pd,'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items |((SIZEOF(
           ['FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
              * TYPEOF(it)) = 2) AND (it.name = 'distance')) )) = 1)) ))
              = 0)) )) = 0));
    wr12: ((NOT (SELF.description = 'linear')) OR (SIZEOF(QUERY ( pd <*
              USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
          ('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
              IN TYPEOF(pdr.used_representation)) )) = 1)) )) = 0));
    wr13: ((NOT (SELF.description = 'complex')) OR (SIZEOF(
              QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.PATH_SHAPE_REPRESENTATION'
              IN TYPEOF(pdr.used_representation)) AND (pdr.
              used_representation.name = 'sweep path') AND (SIZEOF(
              QUERY ( srwp_i <* pdr.used_representation.items | (srwp_i.
              name = 'profile shape') )) = 1)) )) = 1)) )) = 0));
END_ENTITY; -- path_feature_component

ENTITY path_shape_representation
  SUBTYPE OF (shape_representation);
  WHERE
    wr1: (SIZEOF(SELF.items) >= 1);
    wr2: (SIZEOF(QUERY ( i <* SELF.items | (SIZEOF([
              'FEATURE_BASED_PROCESS_PLANNING.BOUNDED_CURVE',
              'FEATURE_BASED_PROCESS_PLANNING.EDGE_CURVE',
              'FEATURE_BASED_PROCESS_PLANNING.PATH'] * TYPEOF(i)) = 1) ))
              >= 1);
END_ENTITY; -- path_shape_representation

ENTITY pattern_offset_membership
  SUBTYPE OF (feature_component_relationship);
  WHERE
    wr1 : (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF.
              relating_shape_aspect,
            'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
```

```
            + 'RELATING_SHAPE_ASPECT') |
     (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
            IN TYPEOF(sar)) AND (sar :<>: SELF)) ) | (SIZEOF(
            QUERY ( pdr <* QUERY ( pd <* USEDIN(fcr.
           related_shape_aspect.of_shape,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (
            'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
            IN TYPEOF(pd)) ) | (SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN',
            'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN'] *
            TYPEOF(pdr.definition)) = 1) )) = 0) )) = 0);
wr2 : (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF.
           related_shape_aspect,
           'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
            + 'RELATED_SHAPE_ASPECT') |
     (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
            IN TYPEOF(sar)) AND (sar :<>: SELF)) ) | ((fcr.description
            = 'modified pattern') AND (
            'FEATURE_BASED_PROCESS_PLANNING.MODIFIED_PATTERN' IN
            TYPEOF(fcr.relating_shape_aspect))) )) >= 1);
wr3 : (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF.
           related_shape_aspect,
           'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
            + 'RELATED_SHAPE_ASPECT') |
     (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
            IN TYPEOF(sar)) AND (sar :<>: SELF)) ) | ((
            'FEATURE_BASED_PROCESS_PLANNING.MODIFIED_PATTERN' IN
            TYPEOF(fcr.relating_shape_aspect)) AND (NOT (SIZEOF(
            QUERY ( modfcr <* QUERY ( modsar <* USEDIN(fcr.
            relating_shape_aspect,'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_ASPECT_RELATIONSHIP.RELATING_SHAPE_ASPECT') | ((
            SIZEOF(['FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN',
            'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN'] *
            TYPEOF(modsar.related_shape_aspect.of_shape.definition)) =
            1) AND (modsar :<>: fcr)) ) | (NOT (modfcr.
            related_shape_aspect.of_shape.definition :=: SELF.
            relating_shape_aspect.of_shape.definition)) )) = 0))) )) =
            0);
wr4 : ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN' IN
             TYPEOF(SELF.relating_shape_aspect.of_shape.definition)))
            OR (SIZEOF(QUERY ( pd <*USEDIN(SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(USEDIN(pd,'FEATURE_BASED_PROCESS_PLANNING.'
            + 'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION')) = 2))
            ))  = 0));
wr5 : ((NOT ('FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN' IN
            TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
             (SIZEOF(QUERY ( pd <*USEDIN(SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(USEDIN(pd,'FEATURE_BASED_PROCESS_PLANNING.'
            + 'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION')) = 1))
            )) = 0));
wr6 : ((NOT ('FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN' IN
            TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
             (SIZEOF(QUERY ( pd <*USEDIN(SELF.related_shape_aspect,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (NOT (
            SIZEOF(pdr.used_representation.items) = 2)) )) = 0)) )) =
```

 ©ISO 2000 - All rights reserved

```
        0));
wr7 : ((NOT ('FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN' IN
          TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
           (SIZEOF(QUERY ( pd <*USEDIN(SELF.related_shape_aspect,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |
        (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
          IN TYPEOF(it)) AND (
          'FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE' IN TYPEOF(it
          \measure_with_unit.value_component)) AND (it.name =
          'index number')) )) = 1)) )) = 0)) )) = 0));
wr8 : ((NOT ('FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN' IN
          TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
           (SIZEOF(QUERY ( pd <*USEDIN(SELF.related_shape_aspect,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |
     (('FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT'
          IN TYPEOF(it)) AND (it.name = 'offset')) )) = 1)) )) = 0))
          ))  = 0));
wr9 : ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN' IN
           TYPEOF(SELF.relating_shape_aspect.of_shape.definition)))
          OR (SIZEOF(QUERY ( pd <*USEDIN(SELF.related_shape_aspect,
    'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(impl_rep.
          used_representation.items) = 3)) )) = 0)) )) = 0));
wr10: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN' IN
           TYPEOF(SELF.relating_shape_aspect.of_shape.definition)))
          OR (SIZEOF(QUERY ( pd <*USEDIN(SELF.related_shape_aspect,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |
        (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
          IN TYPEOF(it)) AND (
          'FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE' IN TYPEOF(it
          \measure_with_unit.value_component)) AND (it.name =
          'row index')) )) = 1)) )) = 0)) )) = 0));
wr11: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN' IN
           TYPEOF(SELF.relating_shape_aspect.of_shape.definition)))
          OR (SIZEOF(QUERY ( pd <*USEDIN(SELF.related_shape_aspect,
```

©ISO 2000 - All rights reserved

```
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
              pd,'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items |
            (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
              IN TYPEOF(it)) AND (
              'FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE' IN TYPEOF(it
              \measure_with_unit.value_component)) AND (it.name =
              'column index')) )) = 1)) )) = 0)) )) = 0));
    wr12: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN' IN
               TYPEOF(SELF.relating_shape_aspect.of_shape.definition)))
              OR (SIZEOF(QUERY ( pd <*USEDIN(SELF.related_shape_aspect,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
              pd,'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items | ((
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'
              IN TYPEOF(it)) AND (it.name = 'offset distance')) )) = 1))
              )) = 0)) )) = 0));
    wr13: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN' IN
               TYPEOF(SELF.relating_shape_aspect.of_shape.definition)))
              OR (SIZEOF(QUERY ( pd <*USEDIN(SELF.related_shape_aspect,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
         (('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
              IN TYPEOF(pdr.used_representation)) AND (pdr.
              used_representation.name = 'offset direction')) )) = 1)) ))
              = 0));
END_ENTITY; -- pattern_offset_membership

ENTITY pattern_omit_membership
  SUBTYPE OF (feature_component_relationship);
  WHERE
    wr1: (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF.
              relating_shape_aspect,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
              + 'RELATING_SHAPE_ASPECT') |
         (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
              IN TYPEOF(sar)) AND (sar :<>: SELF)) ) | (SIZEOF(
              QUERY ( pdr <* QUERY ( pd <*
              USEDIN(fcr.related_shape_aspect.of_shape,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (
              'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
               TYPEOF(pd)) ) | (SIZEOF([
              'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN',
              'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN'] *
              TYPEOF(pdr.definition)) = 1) )) = 0) )) = 0);
    wr2: (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF.
              related_shape_aspect,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
```

 ©ISO 2000 - All rights reserved

```
          + 'RELATED_SHAPE_ASPECT') |
     (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
          IN TYPEOF(sar)) AND (sar :<>: SELF)) ) | ((fcr.description =
           'modified pattern') AND (
          'FEATURE_BASED_PROCESS_PLANNING.MODIFIED_PATTERN' IN TYPEOF(
          fcr.relating_shape_aspect))) )) >= 1);
wr3: (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF.
          related_shape_aspect,
          'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
          + 'RELATED_SHAPE_ASPECT') |
     (('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
          IN TYPEOF(sar)) AND (sar :<>: SELF)) ) | ((
          'FEATURE_BASED_PROCESS_PLANNING.MODIFIED_PATTERN' IN TYPEOF(
          fcr.relating_shape_aspect)) AND (NOT (SIZEOF(
          QUERY ( modfcr <* QUERY ( modsar <* USEDIN(fcr.
          relating_shape_aspect,'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_ASPECT_RELATIONSHIP.RELATING_SHAPE_ASPECT') | ((
          SIZEOF(['FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN',
          'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN'] *
          TYPEOF(modsar.related_shape_aspect.of_shape.definition)) =
          1)
          AND (modsar :<>: fcr)) ) | (NOT (modfcr.related_shape_aspect
          .of_shape.definition :=: SELF.relating_shape_aspect.of_shape
          .definition)) )) = 0))) )) = 0);
wr4: (SIZEOF(QUERY ( pd <*USEDIN(SELF.related_shape_aspect,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(USEDIN(pd,'FEATURE_BASED_PROCESS_PLANNING.'
          + 'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION')) = 1)) ))
          = 0);
wr5: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN' IN
          TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
          (SIZEOF(QUERY ( pd <*USEDIN(SELF.related_shape_aspect,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (NOT (
          SIZEOF(pdr.used_representation.items) = 1)) )) = 0)) )) =
          0));
wr6: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_PATTERN' IN
          TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
          (SIZEOF(QUERY ( pd <*USEDIN(SELF.related_shape_aspect,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
          USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |
        (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
          IN TYPEOF(it)) AND (
          'FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE' IN TYPEOF(it\
          measure_with_unit.value_component)) AND (it.name =
          'index number')) )) = 1)) )) = 0)) )) = 0));
wr7: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN' IN
          TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
          (SIZEOF(QUERY ( pd <*USEDIN(SELF.related_shape_aspect,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
```

```
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (NOT (
                SIZEOF(pdr.used_representation.items) = 2)) )) = 0)) )) =
                0));
     wr8: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN' IN
                TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
                (SIZEOF(QUERY ( pd <*USEDIN(SELF.related_shape_aspect,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
                USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items |
              (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
                IN TYPEOF(it)) AND (
                'FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE' IN TYPEOF(it\
                measure_with_unit.value_component)) AND (it.name =
                'row index')) )) = 1)) )) = 0)) )) = 0));
     wr9: ((NOT ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_PATTERN' IN
                TYPEOF(SELF.relating_shape_aspect.of_shape.definition))) OR
                (SIZEOF(QUERY ( pd <*USEDIN(SELF.related_shape_aspect,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
                USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items |
              (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
                IN TYPEOF(it)) AND (
                'FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE' IN TYPEOF(it\
                measure_with_unit.value_component)) AND (it.name =
                'column index')) )) = 1)) )) = 0)) )) = 0));
END_ENTITY; -- pattern_omit_membership

ENTITY pcurve
  SUBTYPE OF (curve);
    basis_surface      : surface;
    reference_to_curve : definitional_representation;
  WHERE
    wr1: (SIZEOF(reference_to_curve\representation.items) = 1);
    wr2: ('FEATURE_BASED_PROCESS_PLANNING.CURVE' IN TYPEOF(
                reference_to_curve\representation.items[1]));
    wr3: (reference_to_curve\representation.items[1]\
                geometric_representation_item.dim = 2);
END_ENTITY; -- pcurve

ENTITY perpendicularity_tolerance
  SUBTYPE OF (geometric_tolerance_with_datum_reference);
  WHERE
    wr1: (SIZEOF(SELF\geometric_tolerance_with_datum_reference.
                datum_system) <= 3);
END_ENTITY; -- perpendicularity_tolerance

ENTITY person;
    id            : identifier;
    last_name     : OPTIONAL label;
```

 ©ISO 2000 - All rights reserved

```
    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; -- person

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,'FEATURE_BASED_PROCESS_PLANNING.' +
             'NAME_ATTRIBUTE.NAMED_ITEM')) <= 1);
    wr2: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
             'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1);
END_ENTITY; -- person_and_organization

ENTITY person_and_organization_assignment
  ABSTRACT SUPERTYPE;
    assigned_person_and_organization : person_and_organization;
    role                             : person_and_organization_role;
END_ENTITY; -- person_and_organization_assignment

ENTITY person_and_organization_role;
    name : label;
  DERIVE
    description : text := get_description_value(SELF);
  WHERE
    wr1: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
             'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1);
END_ENTITY; -- person_and_organization_role

ENTITY personal_address
  SUBTYPE OF (address);
    people      : SET [1:?] OF person;
    description : OPTIONAL text;
END_ENTITY; -- personal_address

ENTITY placed_datum_target_feature
  SUBTYPE OF (datum_target);
  WHERE
    wr1 : (SELF.description IN ['point','line','rectangle','circle']);
    wr2 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) )) = 1)) )) = 0);
    wr3 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( dtm_rep <* QUERY ( pdr <*
             USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
```

```
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
            dtm_rep.used_representation.items | ((
            'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
            AND (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
wr4 : ((NOT (SELF.description = 'point')) OR (SIZEOF(QUERY ( pd <*
           USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( dtm_rep <* QUERY ( pdr <*
            USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(dtm_rep.
            used_representation.items) = 1)) )) = 0)) )) = 0));
wr5 : ((NOT (SELF.description IN ['line','circle'])) OR (SIZEOF(
            QUERY ( pd <*USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( dtm_rep <* QUERY ( pdr <*
            USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(dtm_rep.
            used_representation.items) = 2)) )) = 0)) )) = 0));
wr6 : ((NOT (SELF.description = 'rectangle')) OR (SIZEOF(
            QUERY ( pd <*USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( dtm_rep <* QUERY ( pdr <*
USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(dtm_rep.
            used_representation.items) = 3)) )) = 0)) )) = 0));
wr7 : ((NOT (SELF.description = 'circle')) OR (SIZEOF(QUERY ( pd <*
           USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( dtm_rep <* QUERY ( pdr <*
            USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
            dtm_rep.used_representation.items |((SIZEOF([
          'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
            'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
            * TYPEOF(it)) = 2) AND (it.name = 'target diameter')) )) =
            1)) )) = 0)) )) = 0));
wr8 : ((NOT (SELF.description = 'line')) OR (SIZEOF(QUERY ( pd <*
           USEDIN(SELF,
      'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( dtm_rep <* QUERY ( pdr <*
            USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
```

 ©ISO 2000 - All rights reserved

```
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                dtm_rep.used_representation.items |((SIZEOF([
              'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
                'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
                * TYPEOF(it)) = 2) AND (it.name = 'target length')) )) =
                1)) )) = 0)) )) = 0));
     wr9 : ((NOT (SELF.description = 'rectangle')) OR (SIZEOF(
                QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( dtm_rep <* QUERY ( pdr <*
                USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                dtm_rep.used_representation.items |((SIZEOF([
              'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
                'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
                * TYPEOF(it)) = 2) AND (it.name = 'target length')) )) =
                1)) )) = 0)) )) = 0));
     wr10: ((NOT (SELF.description = 'rectangle')) OR (SIZEOF(
                QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( dtm_rep <* QUERY ( pdr <*
                USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                dtm_rep.used_representation.items |((SIZEOF([
              'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
                'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
                * TYPEOF(it)) = 2) AND (it.name = 'target width')) )) = 1))
                )) = 0)) )) = 0));
END_ENTITY; -- placed_datum_target_feature

ENTITY placement
  SUPERTYPE OF (ONEOF (axis1_placement,axis2_placement_2d,
      axis2_placement_3d))
  SUBTYPE OF (geometric_representation_item);
    location : cartesian_point;
END_ENTITY; -- placement

ENTITY planar_shape_representation
  SUBTYPE OF (shape_representation);
  WHERE
    wr1: (SIZEOF(SELF.items) = 1);
    wr2: (SIZEOF(QUERY ( it <* SELF.items | (NOT (
              'FEATURE_BASED_PROCESS_PLANNING.PLANE' IN TYPEOF(it))) )) =
             1);
END_ENTITY; -- planar_shape_representation

ENTITY plane
  SUBTYPE OF (elementary_surface);
END_ENTITY; -- plane

ENTITY plane_angle_measure_with_unit
  SUBTYPE OF (measure_with_unit);
```

©ISO 2000 - All rights reserved

```
    WHERE
      wr1: ('FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_UNIT' IN TYPEOF(
               SELF\measure_with_unit.unit_component));
  END_ENTITY; -- plane_angle_measure_with_unit

  ENTITY plane_angle_unit
    SUBTYPE OF (named_unit);
    WHERE
      wr1: ((SELF\named_unit.dimensions.length_exponent = 0) AND (SELF\
               named_unit.dimensions.mass_exponent = 0) AND (SELF\
               named_unit.dimensions.time_exponent = 0) AND (SELF\
               named_unit.dimensions.electric_current_exponent = 0) AND (
               SELF\named_unit.dimensions.
               thermodynamic_temperature_exponent = 0) AND (SELF\named_unit
               .dimensions.amount_of_substance_exponent = 0) AND (SELF\
               named_unit.dimensions.luminous_intensity_exponent = 0));
  END_ENTITY; -- plane_angle_unit

  ENTITY plus_minus_tolerance;
      range                : tolerance_method_definition;
      toleranced_dimension : dimensional_characteristic;
    UNIQUE
      ur1 : toleranced_dimension;
  END_ENTITY; -- plus_minus_tolerance

  ENTITY pocket
    SUBTYPE OF (feature_definition);
    WHERE
      wr1 : (SELF\characterized_object.description IN [
                 'closed rectangular','open rectangular','complex',
                 'circular cutout','complex cutout','recess']);
      wr2 : (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
            'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                  | (
                 'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
                 IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
                 pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
                  | ((sa_occ.description = 'pocket depth occurrence') AND (
                 SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
                 'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                 + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                 'path feature component usage') AND (sar.name =
                 'pocket depth') AND
               ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
                 IN TYPEOF(sar))) ) | ((
                 'FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT' IN
                 TYPEOF(sdr.relating_shape_aspect)) AND (sdr.
                 relating_shape_aspect.description = 'linear')) )) = 1)) ))
                 = 1)) )) = 0);
      wr3 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
            'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                  | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
                 'FEATURE_BASED_PROCESS_PLANNING.' +
                 'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                 'FEATURE_BASED_PROCESS_PLANNING.' +
                 'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                 used_representation)) )) = 1)) )) = 0);
      wr4 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
            'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                  | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
                 'FEATURE_BASED_PROCESS_PLANNING.' +
```

 ©ISO 2000 - All rights reserved

```
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) AND (1 <= SIZEOF(pdr.
          used_representation.items)) AND (SIZEOF(pdr.
          used_representation.items) <= 2)) )) = 1) )) = 1);
wr5 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) AND (SIZEOF(QUERY ( srwp_i <* pdr.
          used_representation.items | ((srwp_i.name = 'orientation')
          OR (srwp_i.name = 'fillet radius')) )) = SIZEOF(pdr.
          used_representation.items))) )) = 1) )) = 1);
wr6 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'fillet radius')) )) <=
          1)) )) = 0)) )) = 0);
wr7 : ((NOT (SELF\characterized_object.description IN ['complex',
          'non-circular cutout','recess'])) OR (SIZEOF(
          QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (
          'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
          IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
          pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
           | ((sa_occ.description = 'boundary occurrence') AND (
          SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
          'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
          + 'RELATED_SHAPE_ASPECT') | ((sar.description =
          'profile usage') AND
        ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
          IN TYPEOF(sar))) ) | (SIZEOF([
          'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE',
         'FEATURE_BASED_PROCESS_PLANNING.NGON_CLOSED_PROFILE',
         'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE',
          'FEATURE_BASED_PROCESS_PLANNING.CLOSED_PATH_PROFILE',
          'FEATURE_BASED_PROCESS_PLANNING.SQUARE_U_PROFILE',
          'FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE',
          'FEATURE_BASED_PROCESS_PLANNING.ROUNDED_U_PROFILE',
          'FEATURE_BASED_PROCESS_PLANNING.VEE_PROFILE',
          'FEATURE_BASED_PROCESS_PLANNING.TEE_PROFILE',
          'FEATURE_BASED_PROCESS_PLANNING.OPEN_PATH_PROFILE'] *
          TYPEOF(sdr.relating_shape_aspect)) = 1) )) = 1)) )) = 1))
          ))  = 0));
wr8 : ((NOT (SELF\characterized_object.description =
          'closed rectangular')) OR (SIZEOF(QUERY ( pds <*
          QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
```

```
                | (
               'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
               IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
               pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
                | ((sa_occ.description = 'closed boundary occurrence') AND
                (SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
               + 'RELATED_SHAPE_ASPECT') | ((sar.description =
               'profile usage') AND
             ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
               IN TYPEOF(sar))) ) |
              ('FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE'
               IN TYPEOF(sdr.relating_shape_aspect)) )) = 1)) )) = 1)) ))
               = 0));
wr9 : ((NOT (SELF\characterized_object.description =
               'open rectangular')) OR (SIZEOF(QUERY ( pds <*
               QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (
               'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
               IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
               pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
                | ((sa_occ.description = 'open boundary occurrence') AND (
               SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
               + 'RELATED_SHAPE_ASPECT') | ((sar.description =
               'profile usage') AND
             ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
               IN TYPEOF(sar))) ) | (
               'FEATURE_BASED_PROCESS_PLANNING.SQUARE_U_PROFILE' IN
               TYPEOF(sdr.relating_shape_aspect)) )) = 1)) )) = 1)) )) =
               0));
wr10: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (
               'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
               IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
               pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
                | ((sa_occ.description = 'bottom condition occurrence')
               AND (SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
               + 'RELATED_SHAPE_ASPECT') | ((sar.description =
               'pocket bottom usage') AND
        ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
               IN TYPEOF(sar))) ) | (
               'FEATURE_BASED_PROCESS_PLANNING.POCKET_BOTTOM' IN TYPEOF(
               sdr.relating_shape_aspect)) )) = 1)) )) = 1)) )) = 0);
wr11: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
        'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (
               'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
               IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
               pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
                | ((sa_occ.description = 'change in boundary occurrence')
               AND (SIZEOF(QUERY ( fcr <* QUERY ( sar <*USEDIN(sa_occ,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
               + 'RELATING_SHAPE_ASPECT') | ((sar.description =
               'taper usage') AND
        ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
               IN TYPEOF(sar))) ) | ((
               'FEATURE_BASED_PROCESS_PLANNING.TAPER' IN TYPEOF(fcr.
```

 ©ISO 2000 - All rights reserved

```
                related_shape_aspect)) AND (fcr.related_shape_aspect.
                description IN ['angle taper','directed taper'])) )) = 1))
                ))  <= 1)) )) = 0);
     wr12: ((NOT (SELF\characterized_object.description =
                'circular cutout')) OR (SIZEOF(QUERY ( pds <*
                QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
                IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
                pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
                 | ((sa_occ.description = 'enclosed boundary occurrence')
                AND (SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'profile usage') AND
              ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
                IN TYPEOF(sar))) ) | (
                'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE' IN
                 TYPEOF(sdr.relating_shape_aspect)) )) = 1)) )) = 1)) )) =
                0));
     wr13: ((NOT (SELF\characterized_object.description IN [
                'circular cutout','complex cutout'])) OR (SIZEOF(
                QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
                IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
                pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
                 | ((sa_occ.description = 'bottom condition occurrence')
                AND (SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'pocket bottom usage') AND
          ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
                IN TYPEOF(sar))) ) | ((
                'FEATURE_BASED_PROCESS_PLANNING.POCKET_BOTTOM' IN TYPEOF(
                sdr.relating_shape_aspect)) AND (sdr.relating_shape_aspect.
                description = 'through')) )) = 1)) )) = 1)) )) = 0));
     wr14: ((NOT (SELF\characterized_object.description = 'recess')) OR (
                SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
                IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
                pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
                 | ((sa_occ.description = 'bottom condition occurrence')
                AND (SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'pocket bottom usage') AND
          ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
                IN TYPEOF(sar))) ) | ((
                'FEATURE_BASED_PROCESS_PLANNING.POCKET_BOTTOM' IN TYPEOF(
                sdr.relating_shape_aspect)) AND (sdr.relating_shape_aspect.
                description IN ['planar','complex'])) )) = 1)) )) = 1)) ))
                = 0));
END_ENTITY; -- pocket

ENTITY pocket_bottom
  SUBTYPE OF (shape_aspect);
```

©ISO 2000 - All rights reserved

```
WHERE
  wr1 : ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
            IN TYPEOF(SELF.of_shape.definition));
  wr2 : (SELF.description IN ['planar','complex','through']);
  wr3 : ((NOT (SELF.description = 'planar')) OR (SIZEOF(QUERY ( pd <*
           USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
      (('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
            IN TYPEOF(pdr.used_representation)) AND (pdr.
            used_representation.name = 'floor normal')) )) = 1)) )) =
            0));
  wr4 : ((NOT (SELF.description = 'planar')) OR (SIZEOF(QUERY ( pd <*
           USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       (('FEATURE_BASED_PROCESS_PLANNING.LOCATION_SHAPE_REPRESENTATION'
            IN TYPEOF(pdr.used_representation)) AND (pdr.
            used_representation.name = 'floor location')) )) = 1)) )) =
             0));
  wr5 : ((NOT (SELF.description = 'complex')) OR (SIZEOF(
            QUERY ( pd <*USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
            IN TYPEOF(pdr.used_representation)) AND (pdr.
            used_representation.name = 'floor face')) )) = 1)) )) =
            0));
  wr6 : ((NOT (SELF.description IN ['planar','complex'])) OR (SIZEOF(
            QUERY ( pd <*USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) )) = 1)) )) = 0));
  wr7 : ((NOT (SELF.description IN ['planar','complex'])) OR (SIZEOF(
            QUERY ( pd <*USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
            pd,'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(impl_rep.
            used_representation.items) <= 1)) )) = 0)) )) = 0));
  wr8 : ((NOT (SELF.description = 'through')) OR (SIZEOF(
            QUERY ( pd <*USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
```

 ©ISO 2000 - All rights reserved

```
                used_representation)) )) = 0)) )) = 0));
     wr9 : ((NOT (SELF.description IN ['planar','complex'])) OR (SIZEOF(
                QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
                pd,'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items |((SIZEOF([
              'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
                'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
                * TYPEOF(it)) = 2) AND (it.name = 'radius')) )) <= 1)) )) =
                 0)) )) = 0));
     wr10: (SIZEOF(QUERY ( fcr <* QUERY ( sar <*USEDIN(SELF,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATING_SHAPE_ASPECT') | ((sar.description =
                'pocket bottom usage') AND
           ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
                IN TYPEOF(sar))) ) | ((fcr.related_shape_aspect.description
                = 'bottom condition occurrence') AND (
                'FEATURE_BASED_PROCESS_PLANNING.POCKET' IN TYPEOF(fcr.
                related_shape_aspect.of_shape.definition))) )) >= 1);
     wr11: ((NOT (SELF.description IN ['planar','complex'])) OR (SIZEOF(
                QUERY ( fcr <* QUERY ( sar <*USEDIN(SELF,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATING_SHAPE_ASPECT') | ((sar.description =
                'pocket bottom usage') AND
           ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
                IN TYPEOF(sar))) ) | ((fcr.related_shape_aspect.description
                = 'bottom condition occurrence') AND (fcr.
                related_shape_aspect.name IN ['pocket depth start',
                'pocket depth end'])) )) = 0));
END_ENTITY; -- pocket_bottom

ENTITY point
  SUPERTYPE OF (cartesian_point)
  SUBTYPE OF (geometric_representation_item);
END_ENTITY; -- point

ENTITY polyline
  SUBTYPE OF (bounded_curve);
    points : LIST [2:?] OF cartesian_point;
END_ENTITY; -- polyline

ENTITY position_tolerance
  SUBTYPE OF (geometric_tolerance_with_datum_reference);
  WHERE
    wr1: (SIZEOF(SELF\geometric_tolerance_with_datum_reference.
              datum_system) <= 3);
END_ENTITY; -- position_tolerance

ENTITY precision_qualifier;
    precision_value : INTEGER;
END_ENTITY; -- precision_qualifier

ENTITY product;
    id                 : identifier;
    name               : label;
    description        : OPTIONAL text;
```

©ISO 2000 - All rights reserved

```
    frame_of_reference : SET [1:?] OF product_context;
END_ENTITY; -- product

ENTITY product_context
  SUBTYPE OF (application_context_element);
    discipline_type : label;
END_ENTITY; -- product_context

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,'FEATURE_BASED_PROCESS_PLANNING.' +
             'NAME_ATTRIBUTE.NAMED_ITEM')) <= 1);
END_ENTITY; -- product_definition

ENTITY product_definition_context
  SUBTYPE OF (application_context_element);
    life_cycle_stage : label;
END_ENTITY; -- product_definition_context

ENTITY product_definition_formation;
    id          : identifier;
    description : OPTIONAL text;
    of_product  : product;
  UNIQUE
    ur1 : id, of_product;
END_ENTITY; -- product_definition_formation

ENTITY product_definition_relationship;
    id                          : identifier;
    name                        : label;
    description                 : OPTIONAL text;
    relating_product_definition : product_definition;
    related_product_definition  : product_definition;
END_ENTITY; -- product_definition_relationship

ENTITY product_definition_shape
  SUBTYPE OF (property_definition);
  UNIQUE
    ur1 : definition;
  WHERE
    wr1:(SIZEOF([
    'FEATURE_BASED_PROCESS_PLANNING.CHARACTERIZED_PRODUCT_DEFINITION',
             'FEATURE_BASED_PROCESS_PLANNING.CHARACTERIZED_OBJECT'] *
             TYPEOF(SELF\property_definition.definition)) > 0);
END_ENTITY; -- product_definition_shape

ENTITY product_definition_usage
  SUPERTYPE OF (ONEOF (make_from_usage_option,assembly_component_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],
             'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_USAGE');
```

 ©ISO 2000 - All rights reserved

```
END_ENTITY; -- product_definition_usage

ENTITY product_definition_with_associated_documents
  SUBTYPE OF (product_definition);
    documentation_ids : SET [1:?] OF document;
END_ENTITY; -- product_definition_with_associated_documents

ENTITY profile_floor
  SUBTYPE OF (shape_aspect);
  WHERE
    wr1 : ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
              IN TYPEOF(SELF.of_shape.definition));
    wr2 : (SELF.description IN ['planar','complex','through']);
    wr3 : ((NOT (SELF.description IN ['planar','complex'])) OR (SIZEOF(
              QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) )) = 1)) )) = 0));
    wr4 : ((NOT (SELF.description IN ['planar','complex'])) OR (SIZEOF(
              QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
              pd,'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | ((NOT (SIZEOF(impl_rep.
              used_representation.items) >= 1)) AND (SIZEOF(impl_rep.
              used_representation.items) <= 2)) )) = 0)) )) = 0));
    wr5 : ((NOT (SELF.description = 'through')) OR (SIZEOF(
              QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) )) = 0)) )) = 0));
    wr6 : ((NOT (SELF.description IN ['planar','complex'])) OR (SIZEOF(
              QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
              pd,'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items |((SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
              * TYPEOF(it)) = 2) AND (it.name = 'radius')) )) = 1)) )) =
              0)) )) = 0));
    wr7 : (SIZEOF(QUERY ( fcr <* QUERY ( sar <*USEDIN(SELF,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
              + 'RELATING_SHAPE_ASPECT') | ((sar.description =
              'profile floor usage') AND
         ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
```

```
                IN TYPEOF(sar))) ) | (
                'FEATURE_BASED_PROCESS_PLANNING.OUTSIDE_PROFILE' IN TYPEOF(
                fcr.related_shape_aspect.of_shape.definition)) )) >= 1);
     wr8 : ((NOT (SELF.description IN ['planar','complex'])) OR (SIZEOF(
                QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
                pd,'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items |
         (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
                IN TYPEOF(it)) AND (it.name =
                'shape profile floor orientation') AND (it.description IN [
                'shape profile start','shape profile end'])) )) = 1)) )) =
                0)) )) = 0));
     wr9 : ((NOT (SELF.description = 'complex')) OR (SIZEOF(
                QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
                IN TYPEOF(pdr.used_representation)) AND (pdr.
                used_representation.name = 'floor')) )) = 1)) )) = 1));
     wr10: ((NOT (SELF.description = 'planar')) OR (SIZEOF(QUERY ( pd <*
                USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
             (('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
                IN TYPEOF(pdr.used_representation)) AND (pdr.
                used_representation.name = 'floor')) )) = 1)) )) = 1));
END_ENTITY; -- profile_floor

ENTITY projected_zone_definition
  SUBTYPE OF (tolerance_zone_definition);
    projection_end   : shape_aspect;
    projected_length : measure_with_unit;
  WHERE
    wr1: (projected_length.value_component > 0);
END_ENTITY; -- projected_zone_definition

ENTITY property_definition;
    name        : label;
    description : OPTIONAL text;
    definition  : characterized_definition;
END_ENTITY; -- property_definition

ENTITY property_definition_relationship;
    name                        : label;
    description                 : text;
    relating_property_definition : property_definition;
    related_property_definition  : property_definition;
END_ENTITY; -- property_definition_relationship

ENTITY property_definition_representation;
    definition          : represented_definition;
```

 ©ISO 2000 - All rights reserved

```
    used_representation : representation;
  DERIVE
    description : text := get_description_value(SELF);
    name        : label := get_name_value(SELF);
  WHERE
    wr1: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
              'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1);
    wr2: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
              'NAME_ATTRIBUTE.NAMED_ITEM')) <= 1);
END_ENTITY; -- property_definition_representation

ENTITY protrusion
  SUBTYPE OF (feature_definition);
  WHERE
    wr1: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
             USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) ) | (NOT (SIZEOF(impl_rep.
             used_representation.items) = 1)) )) = 0)) )) = 0);
    wr2: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
             'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
              TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
             'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
             sa_occ.description = 'shape volume occurrence') AND (SIZEOF(
             QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
             'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
             + 'RELATED_SHAPE_ASPECT') | ((sar.description =
             'volume shape usage') AND
            ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
             IN TYPEOF(sar))) ) | (sdr.relating_shape_aspect.description
             = 'volume shape') )) = 1)) )) = 1)) )) = 0);
END_ENTITY; -- protrusion

ENTITY qualified_representation_item
  SUBTYPE OF (representation_item);
    qualifiers : SET [1:?] OF value_qualifier;
  WHERE
    wr1: (SIZEOF(QUERY ( temp <* qualifiers | (
             'FEATURE_BASED_PROCESS_PLANNING.PRECISION_QUALIFIER' IN
             TYPEOF(temp)) )) < 2);
END_ENTITY; -- qualified_representation_item

ENTITY qualitative_uncertainty
  SUBTYPE OF (uncertainty_qualifier);
    uncertainty_value : text;
END_ENTITY; -- qualitative_uncertainty

ENTITY quasi_uniform_curve
  SUBTYPE OF (b_spline_curve);
END_ENTITY; -- quasi_uniform_curve

ENTITY quasi_uniform_surface
  SUBTYPE OF (b_spline_surface);
END_ENTITY; -- quasi_uniform_surface
```

©ISO 2000 - All rights reserved

```
ENTITY ratio_measure_with_unit
  SUBTYPE OF (measure_with_unit);
  WHERE
    wr1: ('FEATURE_BASED_PROCESS_PLANNING.RATIO_UNIT' IN TYPEOF(SELF\
             measure_with_unit.unit_component));
END_ENTITY; -- ratio_measure_with_unit

ENTITY ratio_unit
  SUBTYPE OF (named_unit);
  WHERE
    wr1: ((SELF\named_unit.dimensions.length_exponent = 0) AND (SELF\
             named_unit.dimensions.mass_exponent = 0) AND (SELF\
             named_unit.dimensions.time_exponent = 0) AND (SELF\
             named_unit.dimensions.electric_current_exponent = 0) AND (
             SELF\named_unit.dimensions.
             thermodynamic_temperature_exponent = 0) AND (SELF\named_unit
             .dimensions.amount_of_substance_exponent = 0) AND (SELF\
             named_unit.dimensions.luminous_intensity_exponent = 0));
END_ENTITY; -- ratio_unit

ENTITY rational_b_spline_curve
  SUBTYPE OF (b_spline_curve);
    weights_data : LIST [2:?] OF REAL;
  DERIVE
    weights : ARRAY [0:upper_index_on_control_points] OF REAL :=
                 list_to_array(weights_data,0,
                 upper_index_on_control_points);
  WHERE
    wr1: (SIZEOF(weights_data) = SIZEOF(SELF\b_spline_curve.
             control_points_list));
    wr2: curve_weights_positive(SELF);
END_ENTITY; -- rational_b_spline_curve

ENTITY rational_b_spline_surface
  SUBTYPE OF (b_spline_surface);
    weights_data : LIST [2:?] OF LIST [2:?] OF REAL;
  DERIVE
    weights : ARRAY [0:u_upper] OF ARRAY [0:v_upper] OF REAL :=
                 make_array_of_array(weights_data,0,u_upper,0,v_upper);
  WHERE
    wr1: ((SIZEOF(weights_data) = SIZEOF(SELF\b_spline_surface.
             control_points_list)) AND (SIZEOF(weights_data[1]) = SIZEOF(
             SELF\b_spline_surface.control_points_list[1])));
    wr2: surface_weights_positive(SELF);
END_ENTITY; -- rational_b_spline_surface

ENTITY rectangular_closed_profile
  SUBTYPE OF (shape_aspect);
  WHERE
    wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
             IN TYPEOF(SELF.of_shape.definition));
    wr2: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) )) = 1)) )) = 0);
    wr3: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
```

 ©ISO 2000 - All rights reserved

```
            | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
           USEDIN(pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
           used_representation)) ) | (NOT ((SIZEOF(impl_rep.
           used_representation.items) >= 3) AND (SIZEOF(impl_rep.
           used_representation.items) <= 4))) )) = 0)) )) = 0);
  wr4: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION.DEFINITION') | (SIZEOF(QUERY ( pdr <*
           USEDIN(pd,'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
(('FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS'
           IN TYPEOF(pdr.used_representation)) AND (SIZEOF(
           QUERY ( srwp_i <* pdr.used_representation.items | ((srwp_i.
           name = 'orientation') OR (srwp_i.name = 'length') OR (srwp_i
           .name = 'width') OR (srwp_i.name = 'corner radius')) )) =
           SIZEOF(pdr.used_representation.items))) )) = 1) )) = 1);
  wr5: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
            | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
           USEDIN(pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
           used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
           impl_rep.used_representation.items | ((
           'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
           AND (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
  wr6: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
            | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
            USEDIN(pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
           used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
           impl_rep.used_representation.items |((SIZEOF([
         'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
           'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
            TYPEOF(it)) = 2) AND (it.name = 'width')) )) = 1)) )) = 0))
            ))  = 0);
  wr7: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
       'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
            | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
           USEDIN(pd,
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
           'FEATURE_BASED_PROCESS_PLANNING.' +
           'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
           used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
           impl_rep.used_representation.items |((SIZEOF([
          'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
           'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
            TYPEOF(it)) = 2) AND (it.name = 'length')) )) = 1)) )) =
            0)) )) = 0);
  wr8: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
```

```
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items |((SIZEOF([
             'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
               TYPEOF(it)) = 2) AND (it.name = 'corner radius')) )) <= 1))
              ))  = 0)) )) = 0);
END_ENTITY; -- rectangular_closed_profile

ENTITY rectangular_pattern
  SUBTYPE OF (replicate_feature);
  WHERE
    wr1 : (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (
               'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
               IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
               pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
                | (SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
               + 'RELATING_SHAPE_ASPECT') |
          ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
               IN TYPEOF(sar)) ) | (
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT' IN TYPEOF(sdr
               .related_shape_aspect)) )) = 1) )) <= 5)) )) = 0);
    wr2 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
         (('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
               IN TYPEOF(pdr.used_representation)) AND (pdr.
               used_representation.name = 'row layout direction')) )) =
               1)) )) = 0);
    wr3 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
         (('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
               IN TYPEOF(pdr.used_representation)) AND (pdr.
               used_representation.name = 'column layout direction')) )) =
                1)) )) = 0);
    wr4 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
               'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
               'DEFINITION') | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
               used_representation)) )) = 1)) )) = 0);
    wr5 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
               pd,'FEATURE_BASED_PROCESS_PLANNING.' +
```

 ©ISO 2000 - All rights reserved

```
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(impl_rep.
          used_representation.items) = 5)) )) = 0)) )) = 0);
wr6 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |
        (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
          IN TYPEOF(it)) AND (
          'FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE' IN TYPEOF(it
          \measure_with_unit.value_component)) AND (it.name =
          'number of rows')) )) = 1)) )) = 0)) )) = 0);
wr7 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |
        (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
          IN TYPEOF(it)) AND (
          'FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE' IN TYPEOF(it
          \measure_with_unit.value_component)) AND (it.name =
          'number of columns')) )) = 1)) )) = 0)) )) = 0);
wr8 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
         'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'row spacing')) )) = 1))
          )) = 0)) )) = 0);
wr9 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
         'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'column spacing')) )) =
          1)) )) = 0)) )) = 0);
wr10: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
```

```
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
               pd,'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
               used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
               impl_rep.used_representation.items | ((
               'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
               AND (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
END_ENTITY; -- rectangular_pattern

ENTITY referenced_modified_datum
  SUBTYPE OF (datum_reference);
    modifier : limit_condition;
END_ENTITY; -- referenced_modified_datum

ENTITY removal_volume
  SUBTYPE OF (feature_definition);
  WHERE
    wr1: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) ) | (NOT (SIZEOF(impl_rep.
             used_representation.items) = 1)) )) = 0)) )) = 0);
    wr2: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
             'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
              TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
             'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
             sa_occ.description = 'shape volume occurrence') AND (SIZEOF(
             QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
             'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
             + 'RELATED_SHAPE_ASPECT') | ((sar.description =
             'volume shape usage') AND ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_DEFINING_RELATIONSHIP') IN TYPEOF(sar))) ) | (sdr.
             relating_shape_aspect.description = 'volume shape') )) = 1))
             )) = 1)) )) = 0);
END_ENTITY; -- removal_volume

ENTITY replicate_feature
  SUPERTYPE OF (ONEOF (circular_pattern,rectangular_pattern,
      feature_pattern))
  SUBTYPE OF (feature_definition);
  WHERE
    wr1: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) )) = 1)) )) = 0);
    wr2: (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF,
             'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
```

 ©ISO 2000 - All rights reserved

```
            + 'RELATING_SHAPE_ASPECT') |
        ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
            IN TYPEOF(sar)) ) | ((SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.REPLICATE_FEATURE',
            'FEATURE_BASED_PROCESS_PLANNING.INSTANCED_FEATURE'] *
            TYPEOF(fcr.related_shape_aspect)) >= 1) AND (fcr.name =
            'pattern basis')) )) = 1);
    wr3: ((SIZEOF(QUERY ( sar <* USEDIN(SELF,
            'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
            + 'RELATING_SHAPE_ASPECT') | (NOT
        ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
            IN TYPEOF(sar))) )) + SIZEOF(QUERY ( sar <* USEDIN(SELF,
            'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
            + 'RELATED_SHAPE_ASPECT') | (NOT
        ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
            IN TYPEOF(sar))) ))) = 0);
END_ENTITY; -- replicate_feature

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,'FEATURE_BASED_PROCESS_PLANNING.' +
            'ID_ATTRIBUTE.IDENTIFIED_ITEM')) <= 1);
    wr2: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
            'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1);
END_ENTITY; -- representation

ENTITY representation_context;
    context_identifier : identifier;
    context_type       : text;
  INVERSE
    representations_in_context : SET [1:?] OF representation FOR
                                    context_of_items;
END_ENTITY; -- representation_context

ENTITY representation_item;
    name : label;
  WHERE
    wr1: (SIZEOF(using_representations(SELF)) > 0);
END_ENTITY; -- representation_item

ENTITY representation_map;
    mapping_origin        : representation_item;
    mapped_representation : representation;
  INVERSE
    map_usage : SET [1:?] OF mapped_item FOR mapping_source;
  WHERE
    wr1: item_in_context(SELF.mapping_origin,SELF.mapped_representation.
            context_of_items);
END_ENTITY; -- representation_map

ENTITY representation_relationship;
    name        : label;
    description : OPTIONAL text;
    rep_1       : representation;
    rep_2       : representation;
```

©ISO 2000 - All rights reserved

```
END_ENTITY; -- representation_relationship

ENTITY revolved_profile
  SUBTYPE OF (feature_definition);
  WHERE
    wr1: (SELF\characterized_object.description IN ['groove','flat',
             'round','open profile']);
    wr2: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
             USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) ) | (NOT (SIZEOF(impl_rep.
             used_representation.items) = 2)) )) = 0)) )) = 0);
    wr3: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
             USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
             impl_rep.used_representation.items |((SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
             'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
              TYPEOF(it)) = 2) AND (it.name = 'radius')) )) = 1)) )) =
             0)) )) = 0);
    wr4: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
             IN TYPEOF(pdr.used_representation)) AND (pdr.
             used_representation.name = 'removal direction')) )) = 1)) ))
             = 0);
    wr5: ((NOT (SELF\characterized_object.description = 'open profile'))
             OR (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
             'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
              TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
             'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
             sa_occ.description = 'outer edge shape occurrence') AND (
             SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
             'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
             + 'RELATED_SHAPE_ASPECT') | ((sar.description =
             'profile usage') AND
            ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
             IN TYPEOF(sar))) ) | ((
             'FEATURE_BASED_PROCESS_PLANNING.OPEN_PATH_PROFILE' IN
             TYPEOF(sdr.relating_shape_aspect)) AND (sdr.
             relating_shape_aspect.description = 'outer edge shape')) ))
             = 1)) )) = 1)) )) = 0));
    wr6: ((NOT (SELF\characterized_object.description = 'flat')) OR (
             SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
```

 ©ISO 2000 - All rights reserved

```
                 | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
                 TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
                sa_occ.description = 'flat edge shape occurrence') AND (
                SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'profile usage') AND
               ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
                IN TYPEOF(sar))) ) | ((
                'FEATURE_BASED_PROCESS_PLANNING.LINEAR_PROFILE' IN TYPEOF(
                sdr.relating_shape_aspect)) AND (sdr.relating_shape_aspect.
                description = 'flat edge shape')) )) = 1)) )) = 1)) )) =
                0));
     wr7: ((NOT (SELF\characterized_object.description = 'round')) OR (
                SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
                 TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
                sa_occ.description = 'rounded edge shape occurrence') AND (
                SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'profile usage') AND
               ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
                IN TYPEOF(sar))) ) | ((
                'FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE' IN
                 TYPEOF(sdr.relating_shape_aspect)) AND (sdr.
                relating_shape_aspect.description = 'rounded edge shape'))
                )) = 1)) )) = 1)) )) = 0));
     wr8: ((NOT (SELF\characterized_object.description = 'groove')) OR (
                SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
                 TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
                sa_occ.description = 'sweep occurrence') AND (SIZEOF(
                QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'profile usage') AND
               ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
                IN TYPEOF(sar))) ) | ((SIZEOF([
                'FEATURE_BASED_PROCESS_PLANNING.SQUARE_U_PROFILE',
                'FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE',
                'FEATURE_BASED_PROCESS_PLANNING.ROUNDED_U_PROFILE',
                'FEATURE_BASED_PROCESS_PLANNING.VEE_PROFILE',
                'FEATURE_BASED_PROCESS_PLANNING.TEE_PROFILE',
                'FEATURE_BASED_PROCESS_PLANNING.OPEN_PATH_PROFILE'] *
                TYPEOF(sdr.relating_shape_aspect)) = 1) AND (sdr.
                relating_shape_aspect.description = 'sweep')) )) = 1)) )) =
                1)) )) = 0));
END_ENTITY; -- revolved_profile

ENTITY rib_top
  SUBTYPE OF (feature_definition);
  WHERE
```

```
      wr1: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (
               'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
                TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
               sa_occ.description = 'rib top condition occurrence') AND (
               SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
               + 'RELATED_SHAPE_ASPECT') | ((sar.description =
               'rib top usage') AND
           ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
               IN TYPEOF(sar))) ) | (
               'FEATURE_BASED_PROCESS_PLANNING.RIB_TOP_FLOOR' IN TYPEOF(sdr
               .relating_shape_aspect)) )) = 1)) )) = 1)) )) = 0);
      wr2: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
          (('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
               IN TYPEOF(pdr.used_representation)) AND (pdr.
               used_representation.name = 'removal direction')) )) = 1)) ))
               = 1);
  END_ENTITY; -- rib_top

  ENTITY rib_top_floor
    SUBTYPE OF (shape_aspect);
    WHERE
      wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
               IN TYPEOF(SELF.of_shape.definition));
      wr2: (SELF.description IN ['planar','complex']);
      wr3: (SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(SELF,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
               + 'RELATING_SHAPE_ASPECT') | ((sar.description =
               'rib top usage') AND
           ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
               IN TYPEOF(sar))) ) | ((fcr.related_shape_aspect.description
               = 'rib top condition occurrence') AND (
               'FEATURE_BASED_PROCESS_PLANNING.RIB_TOP' IN TYPEOF(fcr.
               related_shape_aspect.of_shape.definition))) )) >= 1);
      wr4: ((NOT (SELF.description = 'complex')) OR (SIZEOF(QUERY ( pd <*
              USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
               'FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION'
               IN TYPEOF(pdr.used_representation)) AND (pdr.
               used_representation.name = 'rib top face')) )) = 1)) )) =
               0));
      wr5: ((NOT (SELF.description = 'planar')) OR (SIZEOF(QUERY ( pd <*
              USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
             (('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
               IN TYPEOF(pdr.used_representation)) AND (pdr.
               used_representation.name = 'rib top face')) )) = 1)) )) =
               0));
```

 ©ISO 2000 - All rights reserved

```
    wr6: ((NOT (SELF.description = 'planar')) OR (SIZEOF(QUERY ( pds <*
              QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (
              'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
               TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
              sa_occ.description = 'boundary occurrence') AND (SIZEOF(
              QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
              + 'RELATED_SHAPE_ASPECT') | ((sar.description =
              'profile usage') AND
             ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
              IN TYPEOF(sar))) ) | ((SIZEOF([
              'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE',
              'FEATURE_BASED_PROCESS_PLANNING.NGON_CLOSED_PROFILE',
              'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE',
              'FEATURE_BASED_PROCESS_PLANNING.CLOSED_PATH_PROFILE'] *
              TYPEOF(sdr.relating_shape_aspect)) = 1) AND (sdr.
            relating_shape_aspect.description = 'rib top floor boundary'))
               )) = 1)) )) = 1)) )) = 0));
END_ENTITY; -- rib_top_floor

ENTITY role_association;
    role           : object_role;
    item_with_role : role_select;
END_ENTITY; -- role_association

ENTITY round_hole
  SUBTYPE OF (feature_definition);
  WHERE
    wr1: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (
              'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
               TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
              sa_occ.description = 'diameter occurrence') AND (SIZEOF(
              QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
              + 'RELATED_SHAPE_ASPECT') | ((sar.description =
              'profile usage') AND
             ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
              IN TYPEOF(sar))) ) | ((
              'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE' IN
              TYPEOF(sdr.relating_shape_aspect)) AND (sdr.name =
              'diameter')) )) = 1)) )) = 1)) )) = 0);
    wr2: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (
              'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
               TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
              sa_occ.description = 'hole depth occurrence') AND (SIZEOF(
              QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
              + 'RELATED_SHAPE_ASPECT') | ((sar.description =
              'path feature component usage') AND
             ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
              IN TYPEOF(sar))) ) | ((
              'FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT' IN
```

©ISO 2000 - All rights reserved

```
                TYPEOF(sdr.relating_shape_aspect)) AND (sdr.name =
                'hole depth') AND (sdr.relating_shape_aspect.description =
                'linear')) )) = 1)) )) = 1)) )) = 0);
     wr3: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
                 TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
                sa_occ.description = 'bottom condition occurrence') AND (
                SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(sa_occ,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'hole bottom usage') AND
          ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
                IN TYPEOF(sar))) ) | (
                'FEATURE_BASED_PROCESS_PLANNING.HOLE_BOTTOM' IN TYPEOF(fcr.
                relating_shape_aspect)) )) = 1)) )) = 1)) )) = 0);
     wr4: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
                 TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
                sa_occ.description = 'change in diameter occurrence') AND (
                SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(sa_occ,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'taper usage') AND
          ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
                IN TYPEOF(sar))) ) | ('FEATURE_BASED_PROCESS_PLANNING.TAPER'
                IN TYPEOF(fcr.relating_shape_aspect)) )) = 1)) )) <= 1)) ))
                = 0);
     wr5: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) )) = 1)) )) = 0);
END_ENTITY; -- round_hole

ENTITY rounded_end
  SUBTYPE OF (feature_definition);
  WHERE
    wr1: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
                USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(impl_rep.
                used_representation.items) = 1)) )) = 0)) )) = 0);
     wr2: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
                 TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
```

 ©ISO 2000 - All rights reserved

```
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
                sa_occ.description = 'partial circular boundary occurrence')
                AND (SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'profile usage') AND
('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
                IN TYPEOF(sar))) ) | (
                'FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE' IN
                 TYPEOF(sdr.relating_shape_aspect)) )) = 1)) )) = 1)) )) =
                 0);
      wr3: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
                 TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
                sa_occ.description = 'course of travel occurrence') AND (
                SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'path feature component usage') AND
               ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
                IN TYPEOF(sar))) ) | ((
                'FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT' IN
                TYPEOF(sdr.relating_shape_aspect)) AND (sdr.
                relating_shape_aspect.description = 'linear')) )) = 1)) )) =
                 1)) )) = 0);
  END_ENTITY; -- rounded_end

  ENTITY rounded_u_profile
    SUBTYPE OF (shape_aspect);
    WHERE
      wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
                IN TYPEOF(SELF.of_shape.definition));
      wr2: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) )) = 1)) )) = 0);
      wr3: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
                USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(impl_rep.
                used_representation.items) = 2)) )) = 0)) )) = 0);
      wr4: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
                USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
```

©ISO 2000 - All rights reserved

```
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items | ((
                'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
                AND (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
      wr5: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
                USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items |((SIZEOF([
              'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
                'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
                 TYPEOF(it)) = 2) AND (it.name = 'width')) )) = 1)) )) = 0))
                ))  = 0);
      wr6: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
              (('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
                IN TYPEOF(pdr.used_representation)) AND (pdr.
                used_representation.name = 'profile limit')) )) <= 1)) )) =
                0);
  END_ENTITY; -- rounded_u_profile

  ENTITY roundness_tolerance
    SUBTYPE OF (geometric_tolerance);
    WHERE
      wr1: (NOT (('FEATURE_BASED_PROCESS_PLANNING.' +
                'GEOMETRIC_TOLERANCE_WITH_DATUM_REFERENCE') IN
TYPEOF(SELF)));
  END_ENTITY; -- roundness_tolerance

  ENTITY runout_zone_definition
    SUBTYPE OF (tolerance_zone_definition);
      orientation : runout_zone_orientation;
  END_ENTITY; -- runout_zone_definition

  ENTITY runout_zone_orientation;
      angle : measure_with_unit;
  END_ENTITY; -- runout_zone_orientation

  ENTITY security_classification;
      name           : label;
      purpose        : text;
      security_level : security_classification_level;
  END_ENTITY; -- security_classification

  ENTITY security_classification_assignment
    ABSTRACT SUPERTYPE;
      assigned_security_classification : security_classification;
    DERIVE
      role : object_role := get_role(SELF);
    WHERE
      wr1: (SIZEOF(USEDIN(SELF,'FEATURE_BASED_PROCESS_PLANNING.' +
                'ROLE_ASSOCIATION.ITEM_WITH_ROLE')) <= 1);
  END_ENTITY; -- security_classification_assignment
```

 ©ISO 2000 - All rights reserved

```
ENTITY security_classification_level;
    name : label;
END_ENTITY; -- security_classification_level

ENTITY shape_aspect;
    name                 : label;
    description          : OPTIONAL text;
    of_shape             : product_definition_shape;
    product_definitional : LOGICAL;
END_ENTITY; -- shape_aspect

ENTITY shape_aspect_relationship;
    name                   : label;
    description            : OPTIONAL text;
    relating_shape_aspect : shape_aspect;
    related_shape_aspect  : shape_aspect;
END_ENTITY; -- shape_aspect_relationship

ENTITY shape_defining_relationship
  SUBTYPE OF (shape_aspect_relationship);
END_ENTITY; -- shape_defining_relationship

ENTITY shape_definition_representation
  SUBTYPE OF (property_definition_representation);
  WHERE
    wr1: (('FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
             TYPEOF(SELF.definition)) OR (
             'FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINITION' IN TYPEOF(
             SELF.definition.definition)));
    wr2: ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION' IN
             TYPEOF(SELF.used_representation));
END_ENTITY; -- shape_definition_representation

ENTITY shape_dimension_representation
  SUBTYPE OF (shape_representation);
  WHERE
    wr1: (SIZEOF(QUERY ( temp <* SELF.items | (NOT
        ('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
             IN TYPEOF(temp))) )) = 0);
    wr2: (SIZEOF(SELF.items) <= 3);
    wr3: (SIZEOF(QUERY ( pos_mri <* QUERY ( real_mri <* SELF.items | (
             'REAL' IN
              TYPEOF(real_mri\measure_with_unit.value_component)) )
              | (NOT (pos_mri\measure_with_unit.value_component > 0)) ))
             = 0);
END_ENTITY; -- shape_dimension_representation

ENTITY shape_representation
  SUBTYPE OF (representation);
END_ENTITY; -- shape_representation

ENTITY shape_representation_with_parameters
  SUBTYPE OF (shape_representation);
  WHERE
    wr1: (SIZEOF(QUERY ( it <* SELF.items | (NOT (SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT',
            'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
        'FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM']
             * TYPEOF(it)) = 1)) )) = 0);
END_ENTITY; -- shape_representation_with_parameters
```

©ISO 2000 - All rights reserved

```
  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; -- si_unit

  ENTITY slot
    SUBTYPE OF (feature_definition);
    WHERE
      wr1: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
                USEDIN(pd,
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
               used_representation)) ) | (NOT (SIZEOF(impl_rep.
               used_representation.items) = 1)) )) = 0)) )) = 0);
      wr2: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (
               'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
                TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
               sa_occ.description = 'swept shape occurrence') AND (SIZEOF(
               QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
               + 'RELATED_SHAPE_ASPECT') | ((sar.description =
               'profile usage') AND
              ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
               IN TYPEOF(sar))) ) | (SIZEOF([
               'FEATURE_BASED_PROCESS_PLANNING.SQUARE_U_PROFILE',
               'FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE',
               'FEATURE_BASED_PROCESS_PLANNING.ROUNDED_U_PROFILE',
               'FEATURE_BASED_PROCESS_PLANNING.VEE_PROFILE',
               'FEATURE_BASED_PROCESS_PLANNING.TEE_PROFILE',
               'FEATURE_BASED_PROCESS_PLANNING.OPEN_PATH_PROFILE'] *
               TYPEOF(sdr.relating_shape_aspect)) = 1) )) = 1)) )) = 1)) ))
               = 0);
      wr3: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
           'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (
               'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
                TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
               sa_occ.description = 'course of travel occurrence') AND (
               SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
               'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
               + 'RELATED_SHAPE_ASPECT') | ((sar.description =
               'path feature component usage') AND (sar.name =
               'course of travel') AND
('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
               IN TYPEOF(sar))) ) | (
               'FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT' IN
               TYPEOF(sdr.relating_shape_aspect)) )) = 1)) )) = 1)) )) =
               0);
      wr4: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
```

 ©ISO 2000 - All rights reserved

```
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (
              'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
               TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
              sa_occ.description = 'end condition occurrence') AND (
              SIZEOF(QUERY ( fcr <* QUERY ( sar <* USEDIN(sa_occ,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
              + 'RELATED_SHAPE_ASPECT') | ((sar.description =
              'slot end usage') AND (sar.name IN
              ['course of travelstart',
              'course of travel end']) AND
          ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
              IN TYPEOF(sar))) ) | (
              'FEATURE_BASED_PROCESS_PLANNING.SLOT_END' IN TYPEOF(fcr.
              relating_shape_aspect)) )) = 1)) )) = 2)) )) = 0);
END_ENTITY; -- slot

ENTITY slot_end
  SUBTYPE OF (shape_aspect);
  WHERE
    wr1 : ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
               IN TYPEOF(SELF.of_shape.definition));
    wr2 : (SELF.description IN ['open','radiused','flat','woodruff']);
    wr3 : ((NOT (SELF.description IN ['open','radiused'])) OR (SIZEOF(
               QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
               used_representation)) )) = 0)) )) = 0));
    wr4 : ((NOT (SELF.description IN ['flat','woodruff'])) OR (SIZEOF(
               QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
               used_representation)) )) = 1)) )) = 0));
    wr5 : ((NOT (SELF.description IN ['flat'])) OR (SIZEOF(
               QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
               pd,'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
               used_representation)) ) | (NOT (SIZEOF(impl_rep.
               used_representation.items) = 2)) )) = 0)) )) = 0));
    wr6 : ((NOT (SELF.description = 'flat')) OR (SIZEOF(QUERY ( pd <*
              USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
               pd,'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
               used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
```

```
                impl_rep.used_representation.items |((SIZEOF([
              'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
                'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
                * TYPEOF(it)) = 2) AND (it.name = 'first radius')) )) = 1))
                )) = 0)) )) = 0));
    wr7 : ((NOT (SELF.description = 'flat')) OR (SIZEOF(QUERY ( pd <*
              USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
                pd,'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items |((SIZEOF([
              'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
                'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
                * TYPEOF(it)) = 2) AND (it.name = 'second radius')) )) =
                1)) ))  = 0)) )) = 0));
    wr8 : ((NOT (SELF.description = 'woodruff')) OR (SIZEOF(
                QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
                pd,'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items | ((
                'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'
                IN TYPEOF(it)) AND (it.name = 'radius')) )) = 1)) )) = 0))
                ))  = 0));
    wr9 : ((NOT (SELF.description IN ['woodruff'])) OR (SIZEOF(
                QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
                pd,'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(impl_rep.
                used_representation.items) = 1)) )) = 0)) )) = 0));
    wr10: (SIZEOF(QUERY ( fcr <* QUERY ( sar <*USEDIN(SELF,
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATING_SHAPE_ASPECT') | ((sar.description =
                'slot end usage') AND (sar.name IN [
                'course of travel start','course of travel end']) AND
         ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP'
                IN TYPEOF(sar))) ) | ((fcr.related_shape_aspect.description
                = 'end condition occurrence') AND (
                'FEATURE_BASED_PROCESS_PLANNING.SLOT' IN TYPEOF(fcr.
                related_shape_aspect.of_shape.definition))) )) >= 1);
END_ENTITY; -- slot_end

ENTITY solid_angle_measure_with_unit
  SUBTYPE OF (measure_with_unit);
  WHERE
    wr1: ('FEATURE_BASED_PROCESS_PLANNING.SOLID_ANGLE_UNIT' IN TYPEOF(
              SELF\measure_with_unit.unit_component));
END_ENTITY; -- solid_angle_measure_with_unit
```

 ©ISO 2000 - All rights reserved

```
ENTITY solid_angle_unit
  SUBTYPE OF (named_unit);
  WHERE
    wr1: ((SELF\named_unit.dimensions.length_exponent = 0) AND (SELF\
              named_unit.dimensions.mass_exponent = 0) AND (SELF\
              named_unit.dimensions.time_exponent = 0) AND (SELF\
              named_unit.dimensions.electric_current_exponent = 0) AND (
              SELF\named_unit.dimensions.
              thermodynamic_temperature_exponent = 0) AND (SELF\named_unit
              .dimensions.amount_of_substance_exponent = 0) AND (SELF\
              named_unit.dimensions.luminous_intensity_exponent = 0));
END_ENTITY; -- solid_angle_unit

ENTITY solid_model
  SUPERTYPE OF (manifold_solid_brep)
  SUBTYPE OF (geometric_representation_item);
END_ENTITY; -- solid_model

ENTITY spherical_cap
  SUBTYPE OF (feature_definition);
  WHERE
    wr1: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(impl_rep.
              used_representation.items) = 3)) )) = 0)) )) = 0);
    wr2: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items |((SIZEOF([
             'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
              TYPEOF(it)) = 2) AND (it.name = 'radius')) )) = 1)) )) =
              0)) ))
              = 0);
    wr3: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items |((SIZEOF([
             'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
              * TYPEOF(it)) = 2) AND (it.name = 'internal angle')) )) =
              1)) )) = 0)) )) = 0);
END_ENTITY; -- spherical_cap
```

©ISO 2000 - All rights reserved

```
ENTITY spherical_surface
  SUBTYPE OF (elementary_surface);
    radius : positive_length_measure;
END_ENTITY; -- spherical_surface

ENTITY square_u_profile
  SUBTYPE OF (shape_aspect);
  WHERE
    wr1 : ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
              IN TYPEOF(SELF.of_shape.definition));
    wr2 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) )) = 1)) )) = 0);
    wr3 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
              pd,'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | ((NOT (SIZEOF(impl_rep.
              used_representation.items) >= 4)) AND (SIZEOF(impl_rep.
              used_representation.items) <= 6)) )) = 0)) )) = 0);
    wr4 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) AND (SIZEOF(QUERY ( srwp_i <* pdr.
              used_representation.items | ((srwp_i.name = 'orientation')
              OR (srwp_i.name = 'width') OR (srwp_i.name = 'first angle')
              OR (srwp_i.name = 'second angle') OR (srwp_i.name =
              'first radius') OR (srwp_i.name = 'second radius') OR (
              srwp_i.name = 'profile limit')) )) = SIZEOF(pdr.
              used_representation.items))) )) = 1) )) = 1);
    wr5 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
              pd,'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items | ((
              'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
              AND (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
    wr6 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
               | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
              pd,'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
```

 ©ISO 2000 - All rights reserved

```
          impl_rep.used_representation.items |((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'width')) )) = 1)) )) =
          0)) )) = 0);
wr7 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'first radius')) )) <=
          1)) ))  = 0)) )) = 0);
wr8 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'second radius')) )) <=
        1)) ))  = 0)) )) = 0);
wr9 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'first angle')) )) = 1))
          )) = 0)) )) = 0);
wr10: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'second angle')) )) = 1))
           )) = 0)) )) = 0);
wr11: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
          'DEFINITION') | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
          'FEATURE_BASED_PROCESS_PLANNING.' +
```

```
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
            (('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
               IN TYPEOF(pdr.used_representation)) AND (pdr.
               used_representation.name = 'profile limit')) )) <= 1)) )) =
                0);
END_ENTITY; -- square_u_profile

ENTITY standard_uncertainty
  SUPERTYPE OF (expanded_uncertainty)
  SUBTYPE OF (uncertainty_qualifier);
    uncertainty_value : REAL;
END_ENTITY; -- standard_uncertainty

ENTITY step
  SUBTYPE OF (feature_definition);
  WHERE
    wr1: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
             USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) ) | (NOT (SIZEOF(impl_rep.
             used_representation.items) = 1)) )) = 0)) )) = 0);
    wr2: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
             'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
              TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
             'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
             sa_occ.description = 'course of travel occurrence') AND (
             SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
             'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
             + 'RELATED_SHAPE_ASPECT') | ((sar.description =
             'path feature component usage') AND
            ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
             IN TYPEOF(sar))) ) | ((
             'FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT' IN
             TYPEOF(sdr.relating_shape_aspect)) AND (sdr.
             relating_shape_aspect.description = 'linear')) )) = 1)) )) =
              1)) )) = 0);
    wr3: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (
             'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE' IN
              TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(pds,
             'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE') | ((
             sa_occ.description = 'removal boundary occurrence') AND (
             SIZEOF(QUERY ( sdr <* QUERY ( sar <* USEDIN(sa_occ,
             'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
             + 'RELATED_SHAPE_ASPECT') | ((sar.description =
             'profile usage') AND
            ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
             IN TYPEOF(sar))) ) | (
             'FEATURE_BASED_PROCESS_PLANNING.VEE_PROFILE' IN TYPEOF(sdr.
             relating_shape_aspect)) )) = 1)) )) = 1)) )) = 0);
END_ENTITY; -- step
```

 ©ISO 2000 - All rights reserved

```
  ENTITY straightness_tolerance
    SUBTYPE OF (geometric_tolerance);
    WHERE
      wr1: (NOT (('FEATURE_BASED_PROCESS_PLANNING.' +
               'GEOMETRIC_TOLERANCE_WITH_DATUM_REFERENCE') IN
TYPEOF(SELF)));
  END_ENTITY; -- straightness_tolerance

  ENTITY surface
    SUPERTYPE OF (ONEOF (elementary_surface,swept_surface,bounded_surface))
    SUBTYPE OF (geometric_representation_item);
  END_ENTITY; -- surface

  ENTITY surface_curve
    SUBTYPE OF (curve);
      curve_3d              : curve;
      associated_geometry   : LIST [1:2] OF pcurve_or_surface;
      master_representation : preferred_surface_curve_representation;
    DERIVE
      basis_surface : SET [1:2] OF surface := get_basis_surface(SELF);
    WHERE
      wr1: (curve_3d.dim = 3);
      wr2: (('FEATURE_BASED_PROCESS_PLANNING.PCURVE' IN TYPEOF(
               associated_geometry[1])) OR (master_representation <>
               pcurve_s1));
      wr3: (('FEATURE_BASED_PROCESS_PLANNING.PCURVE' IN TYPEOF(
               associated_geometry[2])) OR (master_representation <>
               pcurve_s2));
      wr4: (NOT ('FEATURE_BASED_PROCESS_PLANNING.PCURVE' IN TYPEOF(
               curve_3d)));
  END_ENTITY; -- surface_curve

  ENTITY surface_of_linear_extrusion
    SUBTYPE OF (swept_surface);
      extrusion_axis : vector;
  END_ENTITY; -- surface_of_linear_extrusion

  ENTITY surface_of_revolution
    SUBTYPE OF (swept_surface);
      axis_position : axis1_placement;
    DERIVE
      axis_line : line := dummy_gri || line(axis_position.location,
                     dummy_gri || vector(axis_position.z,1));
  END_ENTITY; -- surface_of_revolution

  ENTITY surface_profile_tolerance
    SUBTYPE OF (geometric_tolerance);
    WHERE
      wr1: ((NOT (('FEATURE_BASED_PROCESS_PLANNING.' +
               'GEOMETRIC_TOLERANCE_WITH_DATUM_REFERENCE') IN
TYPEOF(SELF)))
               OR (SIZEOF(SELF\geometric_tolerance_with_datum_reference.
               datum_system) <= 3));
  END_ENTITY; -- surface_profile_tolerance

  ENTITY swept_surface
    SUPERTYPE OF (ONEOF
(surface_of_linear_extrusion,surface_of_revolution))
    SUBTYPE OF (surface);
      swept_curve : curve;
  END_ENTITY; -- swept_surface
```

©ISO 2000 - All rights reserved

```
ENTITY symmetry_tolerance
  SUBTYPE OF (geometric_tolerance_with_datum_reference);
  WHERE
    wr1: (SIZEOF(SELF\geometric_tolerance_with_datum_reference.
              datum_system) <= 3);
END_ENTITY; -- symmetry_tolerance

ENTITY taper
  SUBTYPE OF (shape_aspect);
  WHERE
    wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
              IN TYPEOF(SELF.of_shape.definition));
    wr2: (SELF.description IN ['angle taper','diameter taper',
              'directed taper']);
    wr3: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) )) = 1)) )) = 0);
    wr4: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(impl_rep.
              used_representation.items) = 1)) )) = 0)) )) = 0);
    wr5: ((NOT (SELF.description = 'angle taper')) OR (SIZEOF(
              QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items |((SIZEOF([
             'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
           'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
              * TYPEOF(it)) = 2) AND (it.name = 'taper angle')) )) = 1))
              ))  = 0)) )) = 0));
    wr6: ((NOT (SELF.description = 'diameter taper')) OR (SIZEOF(
              QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items |((SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT'] *
               TYPEOF(it)) = 2) AND (it.name = 'final diameter')) )) = 1))
```

 ©ISO 2000 - All rights reserved

```
            ))  = 0)) )) = 0));
    wr7: ((NOT (SELF.description = 'directed taper')) OR (SIZEOF(
            QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
            USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
            impl_rep.used_representation.items |((SIZEOF([
           'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
         'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
            * TYPEOF(it)) = 2) AND (it.name = 'angle')) )) = 1)) )) =
         0)) ))  = 0));
    wr8: ((NOT (SELF.description = 'directed taper')) OR (SIZEOF(
            QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
        (('FEATURE_BASED_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION'
            IN TYPEOF(pdr.used_representation)) AND (pdr.
            used_representation.name = 'direction')) )) = 1)) )) = 0));
END_ENTITY; -- taper

ENTITY tee_profile
  SUBTYPE OF (shape_aspect);
  WHERE
    wr1 : ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
             IN TYPEOF(SELF.of_shape.definition));
    wr2 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) )) = 1)) )) = 0);
    wr3 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
             pd,'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) ) | ((NOT (SIZEOF(impl_rep.
             used_representation.items) >= 9)) AND (SIZEOF(impl_rep.
             used_representation.items) <= 10)) )) = 0)) )) = 0);
    wr4 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (((
             'FEATURE_BASED_PROCESS_PLANNING.' +
             'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
             used_representation)) AND (SIZEOF(QUERY ( srwp_i <* pdr.
             used_representation.items | ((srwp_i.name = 'orientation')
             OR (srwp_i.name = 'width') OR (srwp_i.name = 'depth') OR (
             srwp_i.name = 'cross bar width') OR (srwp_i.name =
```

```
            'cross bar depth') OR (srwp_i.name = 'first offset') OR (
            srwp_i.name = 'second offset') OR (srwp_i.name =
            'first angle') OR (srwp_i.name = 'second angle') OR (srwp_i
            .name = 'radius')) )) = SIZEOF(pdr.used_representation.
            items))) )) = 1) )) = 1);
wr5 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
            pd,'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
            impl_rep.used_representation.items | ((
            'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
            AND (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
wr6 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
            pd,'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
            impl_rep.used_representation.items |((SIZEOF([
          'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
            'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
            * TYPEOF(it)) = 2) AND (it.name = 'width')) )) = 1)) )) =
            0)) ))  = 0);
wr7 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
            pd,'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
            impl_rep.used_representation.items |((SIZEOF([
          'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
            'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
            * TYPEOF(it)) = 2) AND (it.name = 'depth')) )) = 1)) )) =
            0)) ))  = 0);
wr8 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
            pd,'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
            'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
            used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
            impl_rep.used_representation.items |((SIZEOF([
          'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
            'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
            * TYPEOF(it)) = 2) AND (it.name = 'cross bar width')) )) =
            1)) )) = 0)) )) = 0);
wr9 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
             | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
            pd,'FEATURE_BASED_PROCESS_PLANNING.' +
            'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
            'FEATURE_BASED_PROCESS_PLANNING.' +
```

 ©ISO 2000 - All rights reserved

```
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items | ((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'cross bar depth')) )) =
          1)) )) = 0)) )) = 0);
wr10: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'first offset')) )) = 1))
          )) = 0)) )) = 0);
wr11: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'second offset')) )) =
          1)) )) = 0)) )) = 0);
wr12: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'first angle')) )) = 1))
           ))  = 0)) )) = 0);
wr13: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
      'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
      'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'second angle')) )) = 1))
           )) = 0)) )) = 0);
wr14: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
```

©ISO 2000 - All rights reserved

```
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
               pd,'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
               used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
               impl_rep.used_representation.items |((SIZEOF([
             'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
               'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
               * TYPEOF(it)) = 2) AND (it.name = 'radius')) )) <= 1)) )) =
                0)) )) = 0);
    wr15: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
               'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
               'DEFINITION') | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
            (('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
               IN TYPEOF(pdr.used_representation)) AND (pdr.
               used_representation.name = 'profile limit')) )) <= 1)) )) =
                0);
END_ENTITY; -- tee_profile

ENTITY thread
  SUBTYPE OF (feature_definition);
  WHERE
    wr1 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (((
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
               used_representation)) AND (8 <= SIZEOF(pdr.
               used_representation.items)) AND (SIZEOF(pdr.
               used_representation.items) <= 11)) )) = 1) )) = 1);
    wr2 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
               pd,'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
               used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
               impl_rep.used_representation.items |((SIZEOF([
             'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
               'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
               * TYPEOF(it)) = 2) AND (it.name = 'major diameter')) )) =
               1)) )) = 0)) )) = 0);
    wr3 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
               pd,'FEATURE_BASED_PROCESS_PLANNING.' +
               'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
               'FEATURE_BASED_PROCESS_PLANNING.' +
               'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
               used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
               impl_rep.used_representation.items |((SIZEOF([
             'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
               'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
               * TYPEOF(it)) = 2) AND (it.name = 'minor diameter')) )) <=
               1)) )) = 0)) )) = 0);
```

 ©ISO 2000 - All rights reserved

```
wr4 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'pitch diameter')) )) =
          1)) )) = 0)) )) = 0);
wr5 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.RATIO_MEASURE_WITH_UNIT'] *
           TYPEOF(it)) = 2) AND (it.name = 'number of threads')) )) =
           1)) )) = 0)) )) = 0);
wr6 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |
   (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
          IN TYPEOF(it)) AND (it.name = 'fit class')) )) = 1)) )) =
          0)) )) = 0);
wr7 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |
   (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
          IN TYPEOF(it)) AND (it.name = 'form')) )) = 1)) )) = 0)) ))
          = 0);
wr8 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |
   (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
          IN TYPEOF(it)) AND (it.name = 'hand') AND (it.description
```

©ISO 2000 - All rights reserved

```
          IN ['left','right'])) )) = 1)) )) = 0)) )) = 0);
wr9 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |
   (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
          IN TYPEOF(it)) AND (it.name = 'qualifier')) )) <= 1)) )) =
          0)) )) = 0);
wr10: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |
   (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
          IN TYPEOF(it)) AND (it.name = 'thread side') AND ((it.
          description = 'internal') OR
          (it.description = 'external'))) ))
          = 1)) )) = 0)) )) = 0);
wr11: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'crest')) )) <= 1)) )) =
          0)) )) = 0);
wr12: (SIZEOF(QUERY ( pds <* QUERY ( pd <* USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (
          'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
          IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
          pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
           | ((sa_occ.description = 'partial area occurrence') AND (
          SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
          'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
          + 'RELATED_SHAPE_ASPECT') | ((sar.description =
          'applied area usage') AND
        ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
          IN TYPEOF(sar))) ) | (
          'FEATURE_BASED_PROCESS_PLANNING.APPLIED_AREA' IN TYPEOF(sdr
          .relating_shape_aspect)) )) = 1)) )) = 1)) )) = 0);
wr13: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (
          'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
          IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
          pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
```

 ©ISO 2000 - All rights reserved

```
                 | (SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'applied shape') AND
              ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
                IN TYPEOF(sar))) ) | (
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT' IN TYPEOF(sdr
                .relating_shape_aspect)) )) = 1) )) = 1)) )) = 0);
  END_ENTITY; -- thread

  ENTITY tolerance_value;
      lower_bound : measure_with_unit;
      upper_bound : measure_with_unit;
    WHERE
      wr1: (upper_bound.value_component > lower_bound.value_component);
      wr2: (upper_bound.unit_component = lower_bound.unit_component);
  END_ENTITY; -- tolerance_value

  ENTITY tolerance_zone
    SUBTYPE OF (shape_aspect);
      defining_tolerance : SET [1:?] OF geometric_tolerance;
      form               : tolerance_zone_form;
  END_ENTITY; -- tolerance_zone

  ENTITY tolerance_zone_definition
    SUPERTYPE OF (ONEOF
(projected_zone_definition,runout_zone_definition));
      zone       : tolerance_zone;
      boundaries : SET [1:?] OF shape_aspect;
  END_ENTITY; -- tolerance_zone_definition

  ENTITY tolerance_zone_form;
      name : label;
  END_ENTITY; -- tolerance_zone_form

  ENTITY topological_representation_item
    SUPERTYPE OF (ONEOF (vertex,edge,face_bound,face,connected_face_set,
        loop ANDOR path))
    SUBTYPE OF (representation_item);
  END_ENTITY; -- topological_representation_item

  ENTITY toroidal_surface
    SUBTYPE OF (elementary_surface);
      major_radius : positive_length_measure;
      minor_radius : positive_length_measure;
  END_ENTITY; -- toroidal_surface

  ENTITY total_runout_tolerance
    SUBTYPE OF (geometric_tolerance_with_datum_reference);
    WHERE
      wr1: (SIZEOF(SELF\geometric_tolerance_with_datum_reference.
                datum_system) <= 2);
  END_ENTITY; -- total_runout_tolerance

  ENTITY transition_feature
    SUPERTYPE OF (ONEOF (chamfer,edge_round,fillet))
    SUBTYPE OF (shape_aspect);
    WHERE
      wr1: (SIZEOF(['FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION',
                'FEATURE_BASED_PROCESS_PLANNING.COMPOUND_FEATURE'] * TYPEOF(
                SELF.of_shape.definition)) = 1);
```

©ISO 2000 - All rights reserved

```
    wr2: (SIZEOF(['FEATURE_BASED_PROCESS_PLANNING.CHAMFER',
           'FEATURE_BASED_PROCESS_PLANNING.EDGE_ROUND',
           'FEATURE_BASED_PROCESS_PLANNING.FILLET'] * TYPEOF(SELF)) = 1);
END_ENTITY; -- transition_feature

ENTITY turned_knurl
  SUBTYPE OF (feature_definition);
  WHERE
    wr1 : (SELF\characterized_object.description IN ['diamond',
              'diagonal','straight']);
    wr2 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) AND (6 <= SIZEOF(pdr.
              used_representation.items)) AND (SIZEOF(pdr.
              used_representation.items) <= 9)) )) = 1) )) = 1);
    wr3 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
              pd,'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items |
            (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
              IN TYPEOF(it)) AND (
              'FEATURE_BASED_PROCESS_PLANNING.COUNT_MEASURE' IN TYPEOF(it
              \measure_with_unit.value_component)) AND (it.name =
              'number of teeth')) )) <= 1)) )) = 0)) )) = 0);
    wr4 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
              pd,'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items |((SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
              * TYPEOF(it)) = 2) AND (it.name = 'major diameter')) )) =
              1)) )) = 0)) )) = 0);
    wr5 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
              pd,'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items |((SIZEOF([
            'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
              'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
              * TYPEOF(it)) = 2) AND (it.name = 'nominal diameter')) )) =
               1)) )) = 0)) )) = 0);
    wr6 : (SIZEOF(QUERY ( pd <* USEDIN(SELF,
```

 ©ISO 2000 - All rights reserved

```
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.' +
          'DEFINITION') | (NOT (SIZEOF(QUERY ( impl_rep <*
          QUERY ( pdr <* USEDIN(pd,'FEATURE_BASED_PROCESS_PLANNING.'
          + 'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'tooth depth')) )) <= 1))
          )) = 0)) )) = 0);
wr7 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'root fillet')) )) <= 1))
          )) = 0)) )) = 0);
wr8 : (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
        'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
          'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'diametral pitch')) )) =
          1)) )) = 0)) )) = 0);
wr9 : ((NOT (SELF\characterized_object.description IN ['diamond',
          'diagonal'])) OR (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
          impl_rep.used_representation.items |((SIZEOF([
       'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
       'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
          * TYPEOF(it)) = 2) AND (it.name = 'helix angle')) )) = 1))
          )) = 0)) )) = 0));
wr10: ((NOT (SELF\characterized_object.description = 'diagonal')) OR
           (SIZEOF(QUERY ( pd <*USEDIN(SELF,
     'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
           | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <* USEDIN(
          pd,'FEATURE_BASED_PROCESS_PLANNING.' +
          'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
          'FEATURE_BASED_PROCESS_PLANNING.' +
          'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
          used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
```

```
                impl_rep.used_representation.items |
          (('FEATURE_BASED_PROCESS_PLANNING.DESCRIPTIVE_REPRESENTATION_ITEM'
                IN TYPEOF(it)) AND (it.name = 'helix hand')) )) = 1)) )) =
                0)) )) = 0));
     wr11: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
                IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
                pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
                 | ((sa_occ.description = 'partial area occurrence') AND (
                SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'applied area usage') AND ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_DEFINING_RELATIONSHIP') IN TYPEOF(sar))) ) | (
                'FEATURE_BASED_PROCESS_PLANNING.APPLIED_AREA' IN TYPEOF(sdr
                .relating_shape_aspect)) )) = 1)) )) = 1)) )) = 0);
     wr12: (SIZEOF(QUERY ( pds <* QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (
                'FEATURE_BASED_PROCESS_PLANNING.PRODUCT_DEFINITION_SHAPE'
                IN TYPEOF(pd)) ) | (NOT (SIZEOF(QUERY ( sa_occ <* USEDIN(
                pds,'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT.OF_SHAPE')
                 | (SIZEOF(QUERY ( sdr <* QUERY ( sar <*USEDIN(sa_occ,
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT_RELATIONSHIP.'
                + 'RELATED_SHAPE_ASPECT') | ((sar.description =
                'applied shape') AND
              ('FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'
                IN TYPEOF(sar))) ) | (
                'FEATURE_BASED_PROCESS_PLANNING.SHAPE_ASPECT' IN TYPEOF(sdr
                .relating_shape_aspect)) )) = 1) )) = 1)) )) = 0);
END_ENTITY; -- turned_knurl

ENTITY type_qualifier;
    name : label;
END_ENTITY; -- type_qualifier

ENTITY uncertainty_measure_with_unit
  SUBTYPE OF (measure_with_unit);
    name        : label;
    description : OPTIONAL text;
  WHERE
    wr1: valid_measure_value(SELF\measure_with_unit.value_component);
END_ENTITY; -- uncertainty_measure_with_unit

ENTITY uncertainty_qualifier
  SUPERTYPE OF (ONEOF (standard_uncertainty,qualitative_uncertainty));
    measure_name : label;
    description  : text;
END_ENTITY; -- uncertainty_qualifier

ENTITY uniform_curve
  SUBTYPE OF (b_spline_curve);
END_ENTITY; -- uniform_curve

ENTITY uniform_surface
  SUBTYPE OF (b_spline_surface);
END_ENTITY; -- uniform_surface
```

 ©ISO 2000 - All rights reserved

```
ENTITY vector
  SUBTYPE OF (geometric_representation_item);
    orientation : direction;
    magnitude   : length_measure;
  WHERE
    wr1: (magnitude >= 0);
END_ENTITY; -- vector

ENTITY vee_profile
  SUBTYPE OF (shape_aspect);
  WHERE
    wr1: ('FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION'
              IN TYPEOF(SELF.of_shape.definition));
    wr2: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) )) = 1)) )) = 0);
    wr3: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | ((NOT (SIZEOF(impl_rep.
              used_representation.items) >= 3)) AND (SIZEOF(impl_rep.
              used_representation.items) <= 4)) )) = 0)) )) = 0);
    wr4: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (SIZEOF(QUERY ( pdr <* USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) AND (SIZEOF(QUERY ( srwp_i <* pdr.
              used_representation.items | ((srwp_i.name = 'orientation')
              OR (srwp_i.name = 'profile angle') OR (srwp_i.name =
              'tilt angle') OR (srwp_i.name = 'profile radius')) )) =
              SIZEOF(pdr.used_representation.items))) )) = 1) )) = 1);
    wr5: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
              'FEATURE_BASED_PROCESS_PLANNING.' +
              'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
              used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
              impl_rep.used_representation.items | ((
              'FEATURE_BASED_PROCESS_PLANNING.PLACEMENT' IN TYPEOF(it))
              AND (it.name = 'orientation')) )) = 1)) )) = 0)) )) = 0);
    wr6: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
              | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
              USEDIN(pd,
              'FEATURE_BASED_PROCESS_PLANNING.' +
```

©ISO 2000 - All rights reserved

```
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items |
              (('FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM'
                IN TYPEOF(it)) AND (
                'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE_WITH_UNIT' IN
                 TYPEOF(it\measure_with_unit.value_component)) AND (it.name
                = 'profile radius')) )) <= 1)) )) = 0)) )) = 0);
     wr7: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
                USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items |((SIZEOF([
               'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
               'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
                * TYPEOF(it)) = 2) AND (it.name = 'profile angle')) )) = 1))
                )) = 0)) )) = 0);
     wr8: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( impl_rep <* QUERY ( pdr <*
                 USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | ((
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'SHAPE_REPRESENTATION_WITH_PARAMETERS') IN TYPEOF(pdr.
                used_representation)) ) | (NOT (SIZEOF(QUERY ( it <*
                impl_rep.used_representation.items |((SIZEOF([
               'FEATURE_BASED_PROCESS_PLANNING.MEASURE_REPRESENTATION_ITEM',
               'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE_WITH_UNIT']
                * TYPEOF(it)) = 2) AND (it.name = 'tilt angle')) )) = 1)) ))
                = 0)) )) = 0);
     wr9: (SIZEOF(QUERY ( pd <*USEDIN(SELF,
          'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
                 | (NOT (SIZEOF(QUERY ( pdr <* USEDIN(pd,
                'FEATURE_BASED_PROCESS_PLANNING.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
              (('FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION'
                IN TYPEOF(pdr.used_representation)) AND (pdr.
                used_representation.name = 'profile limit')) )) <= 1)) )) =
                0);
END_ENTITY; -- vee_profile

ENTITY versioned_action_request;
    id          : identifier;
    version     : label;
    purpose     : text;
    description : OPTIONAL text;
END_ENTITY; -- versioned_action_request

ENTITY vertex
  SUBTYPE OF (topological_representation_item);
END_ENTITY; -- vertex

ENTITY vertex_loop
```

 ©ISO 2000 - All rights reserved

```
    SUBTYPE OF (loop);
      loop_vertex : vertex;
  END_ENTITY; -- vertex_loop

  ENTITY vertex_point
    SUBTYPE OF (vertex, geometric_representation_item);
      vertex_geometry : point;
  END_ENTITY; -- vertex_point

  ENTITY week_of_year_and_day_date
    SUBTYPE OF (date);
      week_component : week_in_year_number;
      day_component  : OPTIONAL day_in_week_number;
  END_ENTITY; -- week_of_year_and_day_date

  RULE application_context_requires_ap_definition FOR (application_context,
                          application_protocol_definition);

  WHERE
    wr1: (SIZEOF(QUERY ( ac <* application_context | (NOT (SIZEOF(
              QUERY ( apd <* application_protocol_definition | ((ac :=: apd.
              application) AND (apd.
              application_interpreted_model_schema_name =
              'feature_based_process_planning')) )) = 1)) )) = 0);

  END_RULE; -- application_context_requires_ap_definition

  RULE approval_requires_approval_date_time FOR (approval,
              approval_date_time);

  WHERE
    wr1: (SIZEOF(QUERY ( app <* approval | (NOT (SIZEOF(QUERY ( adt <*
              approval_date_time | (app :=: adt.dated_approval) )) = 1)) ))
              = 0);

  END_RULE; -- approval_requires_approval_date_time

  RULE approval_requires_approval_person_organization FOR (approval,
              approval_person_organization);

  WHERE
    wr1: (SIZEOF(QUERY ( app <* approval | (NOT (SIZEOF(QUERY ( apo <*
              approval_person_organization | (app :=: apo.
              authorized_approval) )) >= 1)) )) = 0);

  END_RULE; -- approval_requires_approval_person_organization

  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);
```

©ISO 2000 - All rights reserved

```
  END_RULE; -- compatible_dimension

  RULE dependent_instantiable_action_request_status FOR (
            action_request_status);

  WHERE
    wr1: (SIZEOF(QUERY ( arst <* action_request_status | (NOT (SIZEOF(
            USEDIN(arst,'')) >= 1)) )) = 0);

  END_RULE; -- dependent_instantiable_action_request_status

  RULE dependent_instantiable_approval_status FOR (approval_status);

  WHERE
    wr1: (SIZEOF(QUERY ( ast <* approval_status | (NOT (SIZEOF(USEDIN(ast,
            '')) >= 1)) )) = 0);

  END_RULE; -- dependent_instantiable_approval_status

  RULE dependent_instantiable_date FOR (date);

  WHERE
    wr1: (SIZEOF(QUERY ( dt <* date | (NOT (SIZEOF(USEDIN(dt,'')) >= 1)) ))
            = 0);

  END_RULE; -- dependent_instantiable_date

  RULE dependent_instantiable_named_unit FOR (named_unit);

  WHERE
    wr1: (SIZEOF(QUERY ( nu <* named_unit | (NOT (SIZEOF(USEDIN(nu,'')) >=
             1)) )) = 0);

  END_RULE; -- dependent_instantiable_named_unit

  RULE dependent_instantiable_precision_qualifier FOR
(precision_qualifier);

  WHERE
    wr1: (SIZEOF(QUERY ( pq <* precision_qualifier | (NOT (SIZEOF(USEDIN(
            pq,'')) >= 1)) )) = 0);

  END_RULE; -- dependent_instantiable_precision_qualifier

  RULE dependent_instantiable_security_classification_level FOR (
            security_classification_level);

  WHERE
    wr1: (SIZEOF(QUERY ( scl <* security_classification_level | (NOT (
            SIZEOF(USEDIN(scl,'')) >= 1)) )) = 0);

  END_RULE; -- dependent_instantiable_security_classification_level

  RULE dependent_instantiable_shape_representation FOR (
            shape_representation);

  WHERE
    wr1: (SIZEOF(QUERY ( sr <* shape_representation | (NOT (SIZEOF(USEDIN(
            sr,'')) >= 1)) )) = 0);

  END_RULE; -- dependent_instantiable_shape_representation
```

 ©ISO 2000 - All rights reserved

```
RULE dependent_instantiable_type_qualifier FOR (type_qualifier);

WHERE
  wr1: (SIZEOF(QUERY ( tq <* type_qualifier | (NOT (SIZEOF(USEDIN(tq,'')))
           >= 1)) )) = 0);

END_RULE; -- dependent_instantiable_type_qualifier

RULE dependent_instantiable_uncertainty_qualifier FOR (
           uncertainty_qualifier);

WHERE
  wr1: (SIZEOF(QUERY ( uq <* uncertainty_qualifier | (NOT (SIZEOF(
           USEDIN(uq,'')) >= 1)) )) = 0);

END_RULE; -- dependent_instantiable_uncertainty_qualifier

RULE geometric_tolerance_subtype_exclusiveness FOR (geometric_tolerance);

WHERE
  wr1: (SIZEOF(QUERY ( gt <* geometric_tolerance | (NOT (SIZEOF(TYPEOF(
           gt) * ['FEATURE_BASED_PROCESS_PLANNING.ANGULARITY_TOLERANCE',
           'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_RUNOUT_TOLERANCE',
           'FEATURE_BASED_PROCESS_PLANNING.CONCENTRICITY_TOLERANCE',
           'FEATURE_BASED_PROCESS_PLANNING.CYLINDRICITY_TOLERANCE',
           'FEATURE_BASED_PROCESS_PLANNING.FLATNESS_TOLERANCE',
           'FEATURE_BASED_PROCESS_PLANNING.LINE_PROFILE_TOLERANCE',
           'FEATURE_BASED_PROCESS_PLANNING.PARALLELISM_TOLERANCE',
           'FEATURE_BASED_PROCESS_PLANNING.PERPENDICULARITY_TOLERANCE',
           'FEATURE_BASED_PROCESS_PLANNING.POSITION_TOLERANCE',
           'FEATURE_BASED_PROCESS_PLANNING.ROUNDNESS_TOLERANCE',
           'FEATURE_BASED_PROCESS_PLANNING.STRAIGHTNESS_TOLERANCE',
           'FEATURE_BASED_PROCESS_PLANNING.SURFACE_PROFILE_TOLERANCE',
           'FEATURE_BASED_PROCESS_PLANNING.SYMMETRY_TOLERANCE',
           'FEATURE_BASED_PROCESS_PLANNING.TOTAL_RUNOUT_TOLERANCE']) <=
           2)) ))  = 0);

END_RULE; -- geometric_tolerance_subtype_exclusiveness

RULE machining_feature_life_cycle FOR (instanced_feature);

WHERE
  wr1: (SIZEOF(QUERY ( mf <* instanced_feature | (NOT (mf.of_shape.
           definition.frame_of_reference.life_cycle_stage =
           'manufacturing planning')) )) = 0);

END_RULE; -- machining_feature_life_cycle

RULE material_is_specified_for_part FOR (product_definition,
           make_from_usage_option);

WHERE
  wr1: (SIZEOF(QUERY ( nmpd <* QUERY ( pd <* product_definition | (
           SIZEOF(USEDIN(pd,'FEATURE_BASED_PROCESS_PLANNING.' +
           'MATERIAL_DESIGNATION.DEFINITIONS')) = 0) ) | (NOT (SIZEOF(
           QUERY ( mfuo <* make_from_usage_option | (NOT (nmpd :=: mfuo.
           relating_product_definition)) )) >= 1)) )) = 0);

END_RULE; -- material_is_specified_for_part

RULE non_instantiable_description_attribute FOR (description_attribute);
```

©ISO 2000 - All rights reserved

```
  WHERE
    wr1: (SIZEOF(QUERY ( da <* description_attribute | (NOT (SIZEOF(
             USEDIN(da,'')) = 1)) )) = 0);

  END_RULE; -- non_instantiable_description_attribute

  RULE non_instantiable_object_role FOR (object_role);

  WHERE
    wr1: (SIZEOF(QUERY ( da <* object_role | (NOT (SIZEOF(USEDIN(da,'')) =
              1)) )) = 0);

  END_RULE; -- non_instantiable_object_role

  RULE non_instantiable_role_association FOR (role_association);

  WHERE
    wr1: (SIZEOF(QUERY ( da <* role_association | (NOT (SIZEOF(USEDIN(da,
             '')) = 1)) )) = 0);

  END_RULE; -- non_instantiable_role_association

  RULE part_requires_project_order FOR (product_definition_formation,
             feature_based_pp_action_assignment);

  WHERE
    wr1: (SIZEOF(QUERY ( pdf <* product_definition_formation | (NOT (
             SIZEOF(QUERY ( fbppaa <* feature_based_pp_action_assignment |
             ((pdf IN fbppaa.items) AND (fbppaa.assigned_action.name =
             'project order')) )) = 1)) )) = 0);

  END_RULE; -- part_requires_project_order

  RULE part_to_approval FOR (product_definition_formation,
             feature_based_pp_approval_assignment);

  WHERE
    wr1: (SIZEOF(QUERY ( pdf <* product_definition_formation | (NOT (
             SIZEOF(QUERY ( fbppa <* feature_based_pp_approval_assignment
              | (pdf IN fbppa.items) )) <= 1)) )) = 0);

  END_RULE; -- part_to_approval

  RULE product_definition_formation_requires_security_classification FOR (
             product_definition_formation,
             feature_based_pp_security_classification_assignment);

  WHERE
    wr1: (SIZEOF(QUERY ( pdf <* product_definition_formation | (NOT (
             SIZEOF(QUERY ( fbppsca <*
             feature_based_pp_security_classification_assignment | (pdf IN
             fbppsca.items) )) = 1)) )) = 0);

 END_RULE; --product_definition_formation_requires_security_classification

  RULE product_requires_version FOR (product,
product_definition_formation);

  WHERE
    wr1: (SIZEOF(QUERY ( prod <* product | (NOT (SIZEOF(QUERY ( pdf <*
             product_definition_formation | (prod :=: pdf.of_product) )) >=
```

 ©ISO 2000 - All rights reserved

```
            1)) )) = 0);

END_RULE; -- product_requires_version

RULE project_order_requires_approval FOR (directed_action,
           feature_based_pp_approval_assignment);

WHERE
  wr1: (SIZEOF(QUERY ( po <* QUERY ( da <* directed_action | (da.name =
           'project order') ) | (NOT (SIZEOF(QUERY ( fbppapp <*
           feature_based_pp_approval_assignment | (po IN fbppapp.items)
           ))   = 1)) )) = 0);

END_RULE; -- project_order_requires_approval

RULE project_order_tracking_relationships FOR (directed_action,
           action_relationship);

WHERE
  wr1: (SIZEOF(QUERY ( da <* directed_action | ((da.name IN [
           'shop work order','resource acquisition order',
           'digital technical data package work order',
           'pedigree creation order']) AND (NOT (SIZEOF(QUERY ( ar <*
           action_relationship | ((da :=: ar.related_action) AND (
           'FEATURE_BASED_PROCESS_PLANNING.DIRECTED_ACTION' IN TYPEOF(ar.
           relating_action)) AND (ar.relating_action.name =
           'project order')) )) = 1))) )) = 0);
  wr2: (SIZEOF(QUERY ( da <* directed_action | ((da.name =
           'project order') AND (NOT (SIZEOF(QUERY ( ar <*
           action_relationship | ((da :=: ar.relating_action) AND (
           'FEATURE_BASED_PROCESS_PLANNING.DIRECTED_ACTION' IN TYPEOF(ar.
           related_action)) AND (ar.related_action.name =
           'shop work order')) )) <= 1))) )) = 0);
  wr3: (SIZEOF(QUERY ( da <* directed_action | ((da.name =
           'project order') AND (NOT (SIZEOF(QUERY ( ar <*
           action_relationship | ((da :=: ar.relating_action) AND (
           'FEATURE_BASED_PROCESS_PLANNING.DIRECTED_ACTION' IN TYPEOF(ar.
           related_action)) AND (ar.related_action.name =
           'resource acquisition order')) )) <= 1))) )) = 0);
  wr4: (SIZEOF(QUERY ( da <* directed_action | ((da.name =
           'project order') AND (NOT (SIZEOF(QUERY ( ar <*
           action_relationship | ((da :=: ar.relating_action) AND (
           'FEATURE_BASED_PROCESS_PLANNING.DIRECTED_ACTION' IN TYPEOF(ar.
           related_action)) AND (ar.related_action.name =
           'digital technical data package work order')) )) <= 1))) )) =
           0);
  wr5: (SIZEOF(QUERY ( da <* directed_action | ((da.name =
           'project order') AND (NOT (SIZEOF(QUERY ( ar <*
           action_relationship | ((da :=: ar.relating_action) AND (
           'FEATURE_BASED_PROCESS_PLANNING.DIRECTED_ACTION' IN TYPEOF(ar.
           related_action)) AND (ar.related_action.name =
           'pedigree creation order')) )) <= 1))) )) = 0);
  wr6: (SIZEOF(QUERY ( da <* directed_action | ((da.name =
           'customer order') AND (NOT (SIZEOF(QUERY ( ar <*
           action_relationship | ((da :=: ar.related_action) AND (
           'FEATURE_BASED_PROCESS_PLANNING.DIRECTED_ACTION' IN TYPEOF(ar.
           relating_action)) AND (ar.relating_action.name =
           'project order')) )) >= 1))) )) = 0);
  wr7: (SIZEOF(QUERY ( da <* directed_action | ((da.name =
           'project order') AND (NOT (SIZEOF(QUERY ( ar <*
           action_relationship | ((da :=: ar.relating_action) AND (
```

©ISO 2000 - All rights reserved

```
              'FEATURE_BASED_PROCESS_PLANNING.DIRECTED_ACTION' IN TYPEOF(ar.
              related_action)) AND (ar.related_action.name =
              'customer order')) )) <= 1))) )) = 0);

  END_RULE; -- project_order_tracking_relationships

  RULE representation_subtype_exclusiveness FOR (representation);

  WHERE
    wr1: (SIZEOF(QUERY ( rep <* representation | (NOT (SIZEOF(TYPEOF(rep)
              * ['FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION',

'FEATURE_BASED_PROCESS_PLANNING.DEFINITIONAL_REPRESENTATION','FEATURE_BASED
_PROCESS_PLANNING.DIRECTION_SHAPE_REPRESENTATION',
              'FEATURE_BASED_PROCESS_PLANNING.FACE_SHAPE_REPRESENTATION',

'FEATURE_BASED_PROCESS_PLANNING.PLANAR_SHAPE_REPRESENTATION','FEATURE_BASED
_PROCESS_PLANNING.LOCATION_SHAPE_REPRESENTATION',
              'FEATURE_BASED_PROCESS_PLANNING.PATH_SHAPE_REPRESENTATION'])
              <= 2)) )) = 0);

  END_RULE; -- representation_subtype_exclusiveness

  RULE restrict_approval_status FOR (approval_status);

  WHERE
    wr1: (SIZEOF(QUERY ( ast <* approval_status | (NOT (ast.name IN [
              'approved','not yet approved','disapproved','withdrawn'])) ))
              = 0);

  END_RULE; -- restrict_approval_status

  RULE restrict_security_classification_level FOR (
              security_classification_level);

  WHERE
    wr1: (SIZEOF(QUERY ( scl <* security_classification_level | (NOT (scl.
              name IN ['unclassified','classified','proprietary',
              'confidential','secret','top_secret'])) )) = 0);

  END_RULE; -- restrict_security_classification_level

  RULE shape_aspect_relationship_subtype_exclusiveness FOR (
              shape_aspect_relationship);

  WHERE
    wr1: (SIZEOF(QUERY ( sr <* shape_aspect_relationship | (NOT (SIZEOF(
              TYPEOF(sr) * [
             'FEATURE_BASED_PROCESS_PLANNING.DIMENSIONAL_LOCATION',
          'FEATURE_BASED_PROCESS_PLANNING.GEOMETRIC_TOLERANCE_RELATIONSHIP',
          'FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_RELATIONSHIP',
              'FEATURE_BASED_PROCESS_PLANNING.SHAPE_DEFINING_RELATIONSHIP'])
              <= 2)) )) = 0);

  END_RULE; -- shape_aspect_relationship_subtype_exclusiveness

  RULE shape_aspect_subtype_exclusiveness FOR (shape_aspect);

  WHERE
    wr1: (SIZEOF(QUERY ( sr <* shape_aspect | (NOT (SIZEOF(TYPEOF(sr) * [
              'FEATURE_BASED_PROCESS_PLANNING.PATH_FEATURE_COMPONENT',
```

 ©ISO 2000 - All rights reserved

```
            'FEATURE_BASED_PROCESS_PLANNING.SLOT_END',
            'FEATURE_BASED_PROCESS_PLANNING.POCKET_BOTTOM',
            'FEATURE_BASED_PROCESS_PLANNING.PROFILE_FLOOR',
            'FEATURE_BASED_PROCESS_PLANNING.RIB_TOP_FLOOR',
            'FEATURE_BASED_PROCESS_PLANNING.BOSS_TOP',
            'FEATURE_BASED_PROCESS_PLANNING.HOLE_BOTTOM',
            'FEATURE_BASED_PROCESS_PLANNING.APPLIED_AREA',
            'FEATURE_BASED_PROCESS_PLANNING.TAPER',
            'FEATURE_BASED_PROCESS_PLANNING.CHAMFER_OFFSET',
            'FEATURE_BASED_PROCESS_PLANNING.CIRCULAR_CLOSED_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.NGON_CLOSED_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.CLOSED_PATH_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.SQUARE_U_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.LINEAR_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.TEE_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.VEE_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.RECTANGULAR_CLOSED_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.PARTIAL_CIRCULAR_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.ROUNDED_U_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.OPEN_PATH_PROFILE',
            'FEATURE_BASED_PROCESS_PLANNING.INSTANCED_FEATURE',
            'FEATURE_BASED_PROCESS_PLANNING.REPLICATE_FEATURE',
            'FEATURE_BASED_PROCESS_PLANNING.TRANSITION_FEATURE',
            'FEATURE_BASED_PROCESS_PLANNING.DATUM',
            'FEATURE_BASED_PROCESS_PLANNING.DATUM_FEATURE',
            'FEATURE_BASED_PROCESS_PLANNING.DATUM_TARGET']) <= 2)) )) =
             0);

END_RULE; -- shape_aspect_subtype_exclusiveness

RULE shape_representation_subtype_exclusiveness FOR (
           shape_representation);

WHERE
  wr1: (SIZEOF(QUERY ( sr <* shape_representation | (NOT (SIZEOF(TYPEOF(
           sr) *
   ['FEATURE_BASED_PROCESS_PLANNING.ADVANCED_BREP_SHAPE_REPRESENTATION',
  'FEATURE_BASED_PROCESS_PLANNING.SHAPE_REPRESENTATION_WITH_PARAMETERS',
  'FEATURE_BASED_PROCESS_PLANNING.SHAPE_DIMENSION_REPRESENTATION'])
           <= 2)) )) = 0);

END_RULE; -- shape_representation_subtype_exclusiveness

RULE subtype_mandatory_characterized_object FOR (characterized_object);

WHERE
  wr1: (SIZEOF(QUERY ( csa <* characterized_object | ((NOT (SIZEOF([
           'FEATURE_BASED_PROCESS_PLANNING.DOCUMENT_FILE',
           'FEATURE_BASED_PROCESS_PLANNING.FEATURE_DEFINITION',
           'FEATURE_BASED_PROCESS_PLANNING.FEATURE_COMPONENT_DEFINITION',
           'FEATURE_BASED_PROCESS_PLANNING.ORDERED_PART'] * TYPEOF(csa))
           = 1)) AND (NOT (SIZEOF(QUERY ( pd <*USEDIN(csa,
         'FEATURE_BASED_PROCESS_PLANNING.PROPERTY_DEFINITION.DEFINITION')
            | ('FEATURE_BASED_PROCESS_PLANNING.MATERIAL_PROPERTY' IN
           TYPEOF(pd)) )) = 1))) )) = 0);

END_RULE; -- subtype_mandatory_characterized_object

RULE transition_feature_life_cycle FOR (transition_feature);

WHERE
```

```
    wr1: (SIZEOF(QUERY ( tf <* transition_feature | (NOT (tf.of_shape.
             definition.frame_of_reference.life_cycle_stage =
             'manufacturing planning')) )) = 0);

  END_RULE; -- transition_feature_life_cycle

  RULE transition_feature_on_part_boundary FOR (transition_feature);

  WHERE
    wr1: (SIZEOF(QUERY ( tf <* transition_feature | (NOT tf.
             product_definitional) )) = 0);

  END_RULE; -- transition_feature_on_part_boundary

  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 | (
        'FEATURE_BASED_PROCESS_PLANNING.MAPPED_ITEM' IN TYPEOF(z)) );
    IF SIZEOF(x) > 0 THEN
      REPEAT i := 1 TO HIINDEX(x) BY 1;
        IF x[i]\mapped_item.mapping_source.mapped_representation IN
            parent_set THEN
          RETURN(FALSE);
        END_IF;
        IF NOT acyclic_mapped_representation(parent_set + x[i]\mapped_item
            .mapping_source.mapped_representation,x[i]\mapped_item.
            mapping_source.mapped_representation.items) THEN
          RETURN(FALSE);
        END_IF;
      END_REPEAT;
    END_IF;
    x := children_set - x;
    IF SIZEOF(x) > 0 THEN
      REPEAT i := 1 TO HIINDEX(x) BY 1;
        y := QUERY ( z <* bag_to_set(USEDIN(x[i],'')) | (
            'FEATURE_BASED_PROCESS_PLANNING.REPRESENTATION_ITEM' IN TYPEOF(
            z)) );
        IF NOT acyclic_mapped_representation(parent_set,y) THEN
          RETURN(FALSE);
        END_IF;
      END_REPEAT;
    END_IF;
    RETURN(TRUE);

  END_FUNCTION; -- acyclic_mapped_representation

  FUNCTION acyclic_product_definition_relationship(
               relation: product_definition_relationship;
               relatives: SET [1:?] OF product_definition;
               specific_relation: STRING
      ): BOOLEAN;

    LOCAL
      x : SET OF product_definition_relationship;
```

 ©ISO 2000 - All rights reserved

```
  END_LOCAL;
  IF relation.relating_product_definition IN relatives THEN
    RETURN(FALSE);
  END_IF;
  x := QUERY ( pd <* bag_to_set(USEDIN(relation.
      relating_product_definition,'FEATURE_BASED_PROCESS_PLANNING.' +
      'PRODUCT_DEFINITION_RELATIONSHIP.' + 'RELATED_PRODUCT_DEFINITION'))
       | (specific_relation IN TYPEOF(pd)) );
  REPEAT i := 1 TO HIINDEX(x) BY 1;
    IF NOT acyclic_product_definition_relationship(x[i],relatives +
        relation.relating_product_definition,specific_relation) THEN
      RETURN(FALSE);
    END_IF;
  END_REPEAT;
  RETURN(TRUE);

END_FUNCTION; -- acyclic_product_definition_relationship

FUNCTION associated_surface(
             arg: pcurve_or_surface
    ): surface;

  LOCAL
    surf : surface;
  END_LOCAL;
  IF 'FEATURE_BASED_PROCESS_PLANNING.PCURVE' IN TYPEOF(arg) THEN
    surf := arg.basis_surface;
  ELSE
    surf := arg;
  END_IF;
  RETURN(surf);

END_FUNCTION; -- associated_surface

FUNCTION bag_to_set(
             the_bag: BAG OF GENERIC:intype
    ): SET OF GENERIC:intype;

  LOCAL
    the_set : SET OF GENERIC:intype := [];
  END_LOCAL;
  IF SIZEOF(the_bag) > 0 THEN
    REPEAT i := 1 TO HIINDEX(the_bag) BY 1;
      the_set := the_set + the_bag[i];
    END_REPEAT;
  END_IF;
  RETURN(the_set);

END_FUNCTION; -- bag_to_set

FUNCTION base_axis(
             dim: INTEGER;
             axis1, axis2, axis3: direction
    ): LIST [2:3] OF direction;

  LOCAL
    u      : LIST [2:3] OF direction;
    d1     : direction;
    d2     : direction;
    factor : REAL;
  END_LOCAL;
```

©ISO 2000 - All rights reserved

```
    IF dim = 3 THEN
      d1 := NVL(normalise(axis3),dummy_gri || direction([0,0,1]));
      d2 := first_proj_axis(d1,axis1);
      u := [d2,second_proj_axis(d1,d2,axis2),d1];
    ELSE
      IF EXISTS(axis1) THEN
        d1 := normalise(axis1);
        u := [d1,orthogonal_complement(d1)];
        IF EXISTS(axis2) THEN
          factor := dot_product(axis2,u[2]);
          IF factor < 0 THEN
            u[2].direction_ratios[1] := -u[2].direction_ratios[1];
            u[2].direction_ratios[2] := -u[2].direction_ratios[2];
          END_IF;
        END_IF;
      ELSE
        IF EXISTS(axis2) THEN
          d1 := normalise(axis2);
          u := [orthogonal_complement(d1),d1];
          u[1].direction_ratios[1] := -u[1].direction_ratios[1];
          u[1].direction_ratios[2] := -u[1].direction_ratios[2];
        ELSE
          u := [dummy_gri || direction([1,0]),dummy_gri ||
                direction([0,1])];
        END_IF;
      END_IF;
    END_IF;
    RETURN(u);

  END_FUNCTION; -- base_axis

  FUNCTION boolean_choose(
                b: BOOLEAN;
                choice1, choice2: GENERIC:item
      ): GENERIC:item;
    IF b THEN
      RETURN(choice1);
    ELSE
      RETURN(choice2);
    END_IF;

  END_FUNCTION; -- boolean_choose

  FUNCTION build_2axes(
                ref_direction: direction
      ): LIST [2:2] OF direction;

    LOCAL
      d : direction := NVL(normalise(ref_direction),dummy_gri ||
           direction([1,0]));
    END_LOCAL;
    RETURN([d,orthogonal_complement(d)]);

  END_FUNCTION; -- build_2axes

  FUNCTION build_axes(
                axis, ref_direction: direction
      ): LIST [3:3] OF direction;

    LOCAL
      d1 : direction;
```

 ©ISO 2000 - All rights reserved

```
    d2 : direction;
  END_LOCAL;
  d1 := NVL(normalise(axis),dummy_gri || direction([0,0,1]));
  d2 := first_proj_axis(d1,ref_direction);
  RETURN([d2,normalise(cross_product(d1,d2)).orientation,d1]);

END_FUNCTION; -- build_axes

FUNCTION closed_shell_reversed(
             a_shell: closed_shell
    ): oriented_closed_shell;

  LOCAL
    the_reverse : oriented_closed_shell;
  END_LOCAL;
  IF 'FEATURE_BASED_PROCESS_PLANNING.ORIENTED_CLOSED_SHELL' IN TYPEOF(
      a_shell) THEN
    the_reverse := dummy_tri || connected_face_set(a_shell\
        connected_face_set.cfs_faces) || closed_shell() ||
        oriented_closed_shell(a_shell\oriented_closed_shell.
        closed_shell_element,NOT a_shell\oriented_closed_shell.
        orientation);
  ELSE
    the_reverse := dummy_tri || connected_face_set(a_shell\
        connected_face_set.cfs_faces) || closed_shell() ||
        oriented_closed_shell(a_shell,FALSE);
  END_IF;
  RETURN(the_reverse);

END_FUNCTION; -- closed_shell_reversed

FUNCTION conditional_reverse(
             p: BOOLEAN;
             an_item: reversible_topology
    ): reversible_topology;
  IF p THEN
    RETURN(an_item);
  ELSE
    RETURN(topology_reversed(an_item));
  END_IF;

END_FUNCTION; -- conditional_reverse

FUNCTION constraints_param_b_spline(
             degree, up_knots, up_cp: INTEGER;
             knot_mult: LIST OF INTEGER;
             knots: LIST OF parameter_value
    ): BOOLEAN;

  LOCAL
    k      : INTEGER;
    sum    : INTEGER;
    result : BOOLEAN := TRUE;
  END_LOCAL;
  sum := knot_mult[1];
  REPEAT i := 2 TO up_knots BY 1;
    sum := sum + knot_mult[i];
  END_REPEAT;
  IF (degree < 1) OR (up_knots < 2) OR (up_cp < degree) OR (sum <> (
      degree + up_cp + 2)) THEN
    result := FALSE;
```

©ISO 2000 - All rights reserved

```
      RETURN(result);
    END_IF;
    k := knot_mult[1];
    IF (k < 1) OR (k > (degree + 1)) THEN
      result := FALSE;
      RETURN(result);
    END_IF;
    REPEAT i := 2 TO up_knots BY 1;
      IF (knot_mult[i] < 1) OR (knots[i] <= knots[i - 1]) THEN
        result := FALSE;
        RETURN(result);
      END_IF;
      k := knot_mult[i];
      IF (i < up_knots) AND (k > degree) THEN
        result := FALSE;
        RETURN(result);
      END_IF;
      IF (i = up_knots) AND (k > (degree + 1)) THEN
        result := FALSE;
        RETURN(result);
      END_IF;
    END_REPEAT;
    RETURN(result);

  END_FUNCTION; -- constraints_param_b_spline

  FUNCTION cross_product(
               arg1, arg2: direction
      ): vector;

    LOCAL
      v2     : LIST [3:3] OF REAL;
      v1     : LIST [3:3] OF REAL;
      mag    : REAL;
      res    : direction;
      result : vector;
    END_LOCAL;
    IF (NOT EXISTS(arg1)) OR (arg1.dim = 2) OR (NOT EXISTS(arg2)) OR (arg2
        .dim = 2) THEN
      RETURN(?);
    ELSE
      BEGIN
        v1 := normalise(arg1).direction_ratios;
        v2 := normalise(arg2).direction_ratios;
        res := dummy_gri || direction([(v1[2] * v2[3]) - (v1[3] * v2[2]),(
            v1[3] * v2[1]) - (v1[1] * v2[3]),(v1[1] * v2[2]) - (v1[2] * v2[
            1])]);
        mag := 0;
        REPEAT i := 1 TO 3 BY 1;
          mag := mag + (res.direction_ratios[i] * res.direction_ratios[i]);
        END_REPEAT;
        IF mag > 0 THEN
          result := dummy_gri || vector(res,SQRT(mag));
        ELSE
          result := dummy_gri || vector(arg1,0);
        END_IF;
        RETURN(result);
      END;
    END_IF;

  END_FUNCTION; -- cross_product
```

 ©ISO 2000 - All rights reserved

```
FUNCTION curve_weights_positive(
             b: rational_b_spline_curve
    ): BOOLEAN;

  LOCAL
    result : BOOLEAN := TRUE;
  END_LOCAL;
  REPEAT i := 0 TO b.upper_index_on_control_points BY 1;
    IF b.weights[i] <= 0 THEN
      result := FALSE;
      RETURN(result);
    END_IF;
  END_REPEAT;
  RETURN(result);

END_FUNCTION; -- curve_weights_positive

FUNCTION derive_dimensional_exponents(
             x: unit
    ): dimensional_exponents;

  LOCAL
    result : dimensional_exponents := dimensional_exponents(0,0,0,0,0,0,
              0);
  END_LOCAL;
  IF 'FEATURE_BASED_PROCESS_PLANNING.DERIVED_UNIT' IN TYPEOF(x) THEN
    REPEAT i := LOINDEX(x.elements) TO HIINDEX(x.elements) BY 1;
      result.length_exponent := result.length_exponent + (x.elements[i].
          exponent * x.elements[i].unit.dimensions.length_exponent);
      result.mass_exponent := result.mass_exponent + (x.elements[i].
          exponent * x.elements[i].unit.dimensions.mass_exponent);
      result.time_exponent := result.time_exponent + (x.elements[i].
          exponent * x.elements[i].unit.dimensions.time_exponent);
      result.electric_current_exponent := result.
          electric_current_exponent + (x.elements[i].exponent * x.
          elements[i].unit.dimensions.electric_current_exponent);
      result.thermodynamic_temperature_exponent := result.
          thermodynamic_temperature_exponent + (x.elements[i].exponent *
          x.elements[i].unit.dimensions.
          thermodynamic_temperature_exponent);
      result.amount_of_substance_exponent := result.
          amount_of_substance_exponent + (x.elements[i].exponent * x.
          elements[i].unit.dimensions.amount_of_substance_exponent);
      result.luminous_intensity_exponent := result.
          luminous_intensity_exponent + (x.elements[i].exponent * x.
          elements[i].unit.dimensions.luminous_intensity_exponent);
    END_REPEAT;
  ELSE
    result := x.dimensions;
  END_IF;
  RETURN(result);

END_FUNCTION; -- derive_dimensional_exponents

FUNCTION dimension_of(
             item: geometric_representation_item
    ): dimension_count;

  LOCAL
    x : SET OF representation;
    y : representation_context;
```

©ISO 2000 - All rights reserved

```
    END_LOCAL;
    x := using_representations(item);
    y := x[1].context_of_items;
    RETURN(y\geometric_representation_context.coordinate_space_dimension);

  END_FUNCTION; -- dimension_of

  FUNCTION dimensions_for_si_unit(
               n: si_unit_name
      ): dimensional_exponents;
    CASE n OF
      metre         :         RETURN(dimensional_exponents(1,0,0,0,0,0,0));
      gram          :         RETURN(dimensional_exponents(0,1,0,0,0,0,0));
      second        :         RETURN(dimensional_exponents(0,0,1,0,0,0,0));
      ampere        :         RETURN(dimensional_exponents(0,0,0,1,0,0,0));
      kelvin        :         RETURN(dimensional_exponents(0,0,0,0,1,0,0));
      mole          :         RETURN(dimensional_exponents(0,0,0,0,0,1,0));
      candela       :         RETURN(dimensional_exponents(0,0,0,0,0,0,1));
      radian        :         RETURN(dimensional_exponents(0,0,0,0,0,0,0));
      steradian     :         RETURN(dimensional_exponents(0,0,0,0,0,0,0));
      hertz         :
RETURN(dimensional_exponents(0,0,-1,0,0,0,0));
      newton        :
RETURN(dimensional_exponents(1,1,-2,0,0,0,0));
      pascal        :
RETURN(dimensional_exponents(-1,1,-2,0,0,0,0));
      joule         :
RETURN(dimensional_exponents(2,1,-2,0,0,0,0));
      watt          :
RETURN(dimensional_exponents(2,1,-3,0,0,0,0));
      coulomb       :         RETURN(dimensional_exponents(0,0,1,1,0,0,0));
      volt          :
RETURN(dimensional_exponents(2,1,-3,-1,0,0,0));
      farad         :
RETURN(dimensional_exponents(-2,-1,4,1,0,0,0));
      ohm           :
RETURN(dimensional_exponents(2,1,-3,-2,0,0,0));
      siemens       :
RETURN(dimensional_exponents(-2,-1,3,2,0,0,0));
      weber         :
RETURN(dimensional_exponents(2,1,-2,-1,0,0,0));
      tesla         :
RETURN(dimensional_exponents(0,1,-2,-1,0,0,0));
      henry         :
RETURN(dimensional_exponents(2,1,-2,-2,0,0,0));
      degree_celsius  :
RETURN(dimensional_exponents(0,0,0,0,1,0,0));
      lumen         :
RETURN(dimensional_exponents(0,0,0,0,0,0,1));
      lux           :
RETURN(dimensional_exponents(-2,0,0,0,0,0,1));
      becquerel     :
RETURN(dimensional_exponents(0,0,-1,0,0,0,0));
      gray          :
RETURN(dimensional_exponents(2,0,-2,0,0,0,0));
      sievert       :
RETURN(dimensional_exponents(2,0,-2,0,0,0,0));
      OTHERWISE     :         RETURN(?);
      END_CASE;

  END_FUNCTION; -- dimensions_for_si_unit
```

 ©ISO 2000 - All rights reserved

```
FUNCTION dot_product(
             arg1, arg2: direction
    ): REAL;

  LOCAL
    ndim   : INTEGER;
    scalar : REAL;
    vec1   : direction;
    vec2   : direction;
  END_LOCAL;
  IF (NOT EXISTS(arg1)) OR (NOT EXISTS(arg2)) THEN
    scalar := ?;
  ELSE
    IF arg1.dim <> arg2.dim THEN
      scalar := ?;
    ELSE
      BEGIN
        vec1 := normalise(arg1);
        vec2 := normalise(arg2);
        ndim := arg1.dim;
        scalar := 0;
        REPEAT i := 1 TO ndim BY 1;
          scalar := scalar + (vec1.direction_ratios[i] * vec2.
              direction_ratios[i]);
        END_REPEAT;
      END;
    END_IF;
  END_IF;
  RETURN(scalar);

END_FUNCTION; -- dot_product

FUNCTION edge_reversed(
             an_edge: edge
    ): oriented_edge;

  LOCAL
    the_reverse : oriented_edge;
  END_LOCAL;
  IF 'FEATURE_BASED_PROCESS_PLANNING.ORIENTED_EDGE' IN TYPEOF(an_edge)
       THEN
    the_reverse := dummy_tri || edge(an_edge.edge_end,an_edge.edge_start)
        || oriented_edge(an_edge\oriented_edge.edge_element,NOT an_edge\
        oriented_edge.orientation);
  ELSE
    the_reverse := dummy_tri || edge(an_edge.edge_end,an_edge.edge_start)
        || oriented_edge(an_edge,FALSE);
  END_IF;
  RETURN(the_reverse);

END_FUNCTION; -- edge_reversed

FUNCTION face_bound_reversed(
             a_face_bound: face_bound
    ): face_bound;

  LOCAL
    the_reverse : face_bound;
  END_LOCAL;
  IF 'FEATURE_BASED_PROCESS_PLANNING.FACE_OUTER_BOUND' IN TYPEOF(
      a_face_bound) THEN
```

©ISO 2000 - All rights reserved

```
      the_reverse := dummy_tri || face_bound(a_face_bound\face_bound.bound,
          NOT a_face_bound\face_bound.orientation) || face_outer_bound();
    ELSE
      the_reverse := dummy_tri || face_bound(a_face_bound.bound,NOT
          a_face_bound.orientation);
    END_IF;
    RETURN(the_reverse);

  END_FUNCTION; -- face_bound_reversed

  FUNCTION face_reversed(
               a_face: face
      ): oriented_face;

    LOCAL
      the_reverse : oriented_face;
    END_LOCAL;
    IF 'FEATURE_BASED_PROCESS_PLANNING.ORIENTED_FACE' IN TYPEOF(a_face)
         THEN
      the_reverse := dummy_tri || face(set_of_topology_reversed(a_face.
          bounds)) || oriented_face(a_face\oriented_face.face_element,NOT
          a_face\oriented_face.orientation);
    ELSE
      the_reverse := dummy_tri || face(set_of_topology_reversed(a_face.
          bounds)) || oriented_face(a_face,FALSE);
    END_IF;
    RETURN(the_reverse);

  END_FUNCTION; -- face_reversed

  FUNCTION first_proj_axis(
               z_axis, arg: direction
      ): direction;

    LOCAL
      x_vec  : vector;
      v      : direction;
      z      : direction;
      x_axis : direction;
    END_LOCAL;
    IF NOT EXISTS(z_axis) THEN
      RETURN(?);
    ELSE
      z := normalise(z_axis);
      IF NOT EXISTS(arg) THEN
        IF z.direction_ratios <> [1,0,0] THEN
          v := dummy_gri || direction([1,0,0]);
        ELSE
          v := dummy_gri || direction([0,1,0]);
        END_IF;
      ELSE
        IF arg.dim <> 3 THEN
          RETURN(?);
        END_IF;
        IF cross_product(arg,z).magnitude = 0 THEN
          RETURN(?);
        ELSE
          v := normalise(arg);
        END_IF;
      END_IF;
      x_vec := scalar_times_vector(dot_product(v,z),z);
```

 ©ISO 2000 - All rights reserved

```
    x_axis := vector_difference(v,x_vec).orientation;
    x_axis := normalise(x_axis);
  END_IF;
  RETURN(x_axis);

END_FUNCTION; -- first_proj_axis

FUNCTION get_basis_surface(
             c: curve_on_surface
    ): SET [0:2] OF surface;

  LOCAL
    surfs : SET [0:2] OF surface;
    n     : INTEGER;
  END_LOCAL;
  surfs := [];
  IF 'FEATURE_BASED_PROCESS_PLANNING.PCURVE' IN TYPEOF(c) THEN
    surfs := [c\pcurve.basis_surface];
  ELSE
    IF 'FEATURE_BASED_PROCESS_PLANNING.SURFACE_CURVE' IN TYPEOF(c) THEN
      n := SIZEOF(c\surface_curve.associated_geometry);
      REPEAT i := 1 TO n BY 1;
        surfs := surfs + associated_surface(c\surface_curve.
            associated_geometry[i]);
      END_REPEAT;
    END_IF;
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.COMPOSITE_CURVE_ON_SURFACE' IN
      TYPEOF(c) THEN
    n := SIZEOF(c\composite_curve.segments);
    surfs := get_basis_surface(c\composite_curve.segments[1].
        parent_curve);
    IF n > 1 THEN
      REPEAT i := 2 TO n BY 1;
        surfs := surfs * get_basis_surface(c\composite_curve.segments[i]
            .parent_curve);
      END_REPEAT;
    END_IF;
  END_IF;
  RETURN(surfs);

END_FUNCTION; -- get_basis_surface

FUNCTION get_description_value(
             obj: description_attribute_select
    ): text;

  LOCAL
    description_bag : BAG OF description_attribute := USEDIN(obj,
                        'FEATURE_BASED_PROCESS_PLANNING.' +
                        'DESCRIPTION_ATTRIBUTE.' + 'DESCRIBED_ITEM');
  END_LOCAL;
  IF SIZEOF(description_bag) = 1 THEN
    RETURN(description_bag[1].attribute_value);
  ELSE
    RETURN(?);
  END_IF;

END_FUNCTION; -- get_description_value

FUNCTION get_id_value(
```

©ISO 2000 - All rights reserved

```
              obj: id_attribute_select
    ): identifier;

  LOCAL
    id_bag : BAG OF id_attribute := USEDIN(obj,
               'FEATURE_BASED_PROCESS_PLANNING.' + 'ID_ATTRIBUTE.' +
               'IDENTIFIED_ITEM');
  END_LOCAL;
  IF SIZEOF(id_bag) = 1 THEN
    RETURN(id_bag[1].attribute_value);
  ELSE
    RETURN(?);
  END_IF;

END_FUNCTION; -- get_id_value

FUNCTION get_name_value(
              obj: name_attribute_select
    ): label;

  LOCAL
    name_bag : BAG OF name_attribute := USEDIN(obj,
                 'FEATURE_BASED_PROCESS_PLANNING.' + 'NAME_ATTRIBUTE.' +
                 'NAMED_ITEM');
  END_LOCAL;
  IF SIZEOF(name_bag) = 1 THEN
    RETURN(name_bag[1].attribute_value);
  ELSE
    RETURN(?);
  END_IF;

END_FUNCTION; -- get_name_value

FUNCTION get_role(
              obj: role_select
    ): object_role;

  LOCAL
    role_bag : BAG OF role_association := USEDIN(obj,
                 'FEATURE_BASED_PROCESS_PLANNING.' + 'ROLE_ASSOCIATION.' +
                  'ITEM_WITH_ROLE');
  END_LOCAL;
  IF SIZEOF(role_bag) = 1 THEN
    RETURN(role_bag[1].role);
  ELSE
    RETURN(?);
  END_IF;

END_FUNCTION; -- get_role

FUNCTION item_in_context(
              item: representation_item;
              cntxt: representation_context
    ): BOOLEAN;

  LOCAL
    y : BAG OF representation_item;
  END_LOCAL;
  IF SIZEOF(USEDIN(item,
      'FEATURE_BASED_PROCESS_PLANNING.REPRESENTATION.ITEMS') * cntxt.
      representations_in_context) > 0 THEN
```

 ©ISO 2000 - All rights reserved

```
      RETURN(TRUE);
    ELSE
      y := QUERY ( z <* USEDIN(item,'') | (
          'FEATURE_BASED_PROCESS_PLANNING.REPRESENTATION_ITEM' IN
TYPEOF(z)) );
      IF SIZEOF(y) > 0 THEN
        REPEAT i := 1 TO HIINDEX(y) BY 1;
          IF item_in_context(y[i],cntxt) THEN
            RETURN(TRUE);
          END_IF;
        END_REPEAT;
      END_IF;
    END_IF;
    RETURN(FALSE);

  END_FUNCTION; -- item_in_context

  FUNCTION leap_year(
              year: year_number
      ): BOOLEAN;
    IF (((year MOD 4) = 0) AND ((year MOD 100) <> 0)) OR ((year MOD 400) =
        0) THEN
      RETURN(TRUE);
    ELSE
      RETURN(FALSE);
    END_IF;

  END_FUNCTION; -- leap_year

  FUNCTION list_face_loops(
              f: face
      ): LIST [0:?] OF loop;

    LOCAL
      loops : LIST [0:?] OF loop := [];
    END_LOCAL;
    REPEAT i := 1 TO SIZEOF(f.bounds) BY 1;
      loops := loops + f.bounds[i].bound;
    END_REPEAT;
    RETURN(loops);

  END_FUNCTION; -- list_face_loops

  FUNCTION list_of_topology_reversed(
              a_list: list_of_reversible_topology_item
      ): list_of_reversible_topology_item;

    LOCAL
      the_reverse : list_of_reversible_topology_item;
    END_LOCAL;
    the_reverse := [];
    REPEAT i := 1 TO SIZEOF(a_list) BY 1;
      the_reverse := topology_reversed(a_list[i]) + the_reverse;
    END_REPEAT;
    RETURN(the_reverse);

  END_FUNCTION; -- list_of_topology_reversed

  FUNCTION list_to_array(
              lis: LIST [0:?] OF GENERIC:t;
              low, u: INTEGER
```

©ISO 2000 - All rights reserved

```
      ): ARRAY OF GENERIC:t;

    LOCAL
      n   : INTEGER;
      res : ARRAY [low:u] OF GENERIC:t;
    END_LOCAL;
    n := SIZEOF(lis);
    IF n <> ((u - low) + 1) THEN
      RETURN(?);
    ELSE
      res := [lis[1],n];
      REPEAT i := 2 TO n BY 1;
        res[(low + i) - 1] := lis[i];
      END_REPEAT;
      RETURN(res);
    END_IF;

  END_FUNCTION; -- list_to_array

  FUNCTION list_to_set(
               l: LIST [0:?] OF GENERIC:t
      ): SET OF GENERIC:t;

    LOCAL
      s : SET OF GENERIC:t := [];
    END_LOCAL;
    REPEAT i := 1 TO SIZEOF(l) BY 1;
      s := s + l[i];
    END_REPEAT;
    RETURN(s);

  END_FUNCTION; -- list_to_set

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

    LOCAL
      res : ARRAY [low1:u1] OF ARRAY [low2:u2] OF GENERIC:t;
    END_LOCAL;
    IF ((u1 - low1) + 1) <> SIZEOF(lis) THEN
      RETURN(?);
    END_IF;
    IF ((u2 - low2) + 1) <> SIZEOF(lis[1]) THEN
      RETURN(?);
    END_IF;
    res := [list_to_array(lis[1],low2,u2),(u1 - low1) + 1];
    REPEAT i := 2 TO HIINDEX(lis) BY 1;
      IF ((u2 - low2) + 1) <> SIZEOF(lis[i]) THEN
        RETURN(?);
      END_IF;
      res[(low1 + i) - 1] := list_to_array(lis[i],low2,u2);
    END_REPEAT;
    RETURN(res);

  END_FUNCTION; -- make_array_of_array

  FUNCTION mixed_loop_type_set(
               l: SET [0:?] OF loop
      ): LOGICAL;
```

 ©ISO 2000 - All rights reserved

```
  LOCAL
    poly_loop_type : LOGICAL;
  END_LOCAL;
  IF SIZEOF(l) <= 1 THEN
    RETURN(FALSE);
  END_IF;
  poly_loop_type := 'FEATURE_BASED_PROCESS_PLANNING.POLY_LOOP' IN
      TYPEOF(l[1]);
  REPEAT i := 2 TO SIZEOF(l) BY 1;
    IF ('FEATURE_BASED_PROCESS_PLANNING.POLY_LOOP' IN TYPEOF(l[i])) <>
        poly_loop_type THEN
      RETURN(TRUE);
    END_IF;
  END_REPEAT;
  RETURN(FALSE);

END_FUNCTION; -- mixed_loop_type_set

FUNCTION msb_shells(
             brep: manifold_solid_brep
    ): SET [1:?] OF closed_shell;
  IF SIZEOF(QUERY ( msbtype <* TYPEOF(brep) | (msbtype LIKE
      '*BREP_WITH_VOIDS') )) >= 1 THEN
    RETURN(brep\brep_with_voids.voids + brep.outer);
  ELSE
    RETURN([brep.outer]);
  END_IF;

END_FUNCTION; -- msb_shells

FUNCTION normalise(
             arg: vector_or_direction
    ): vector_or_direction;

  LOCAL
    ndim   : INTEGER;
    v      : direction;
    vec    : vector;
    mag    : REAL;
    result : vector_or_direction;
  END_LOCAL;
  IF NOT EXISTS(arg) THEN
    result := ?;
  ELSE
    ndim := arg.dim;
    IF 'FEATURE_BASED_PROCESS_PLANNING.VECTOR' IN TYPEOF(arg) THEN
      BEGIN
        v := dummy_gri || direction(arg.orientation.direction_ratios);
        IF arg.magnitude = 0 THEN
          RETURN(?);
        ELSE
          vec := dummy_gri || vector(v,1);
        END_IF;
      END;
    ELSE
      v := dummy_gri || direction(arg.direction_ratios);
    END_IF;
    mag := 0;
    REPEAT i := 1 TO ndim BY 1;
      mag := mag + (v.direction_ratios[i] * v.direction_ratios[i]);
    END_REPEAT;
```

©ISO 2000 - All rights reserved

```
      IF mag > 0 THEN
        mag := SQRT(mag);
        REPEAT i := 1 TO ndim BY 1;
          v.direction_ratios[i] := v.direction_ratios[i] / mag;
        END_REPEAT;
        IF 'FEATURE_BASED_PROCESS_PLANNING.VECTOR' IN TYPEOF(arg) THEN
          vec.orientation := v;
          result := vec;
        ELSE
          result := v;
        END_IF;
      ELSE
        RETURN(?);
      END_IF;
    END_IF;
    RETURN(result);

  END_FUNCTION; -- normalise

  FUNCTION open_shell_reversed(
                a_shell: open_shell
      ): oriented_open_shell;

    LOCAL
      the_reverse : oriented_open_shell;
    END_LOCAL;
    IF 'FEATURE_BASED_PROCESS_PLANNING.ORIENTED_OPEN_SHELL' IN TYPEOF(
        a_shell) THEN
      the_reverse := dummy_tri || connected_face_set(a_shell\
          connected_face_set.cfs_faces) || open_shell() ||
          oriented_open_shell(a_shell\oriented_open_shell.
          open_shell_element,NOT a_shell\oriented_open_shell.orientation);
    ELSE
      the_reverse := dummy_tri || connected_face_set(a_shell\
          connected_face_set.cfs_faces) || open_shell() ||
          oriented_open_shell(a_shell,FALSE);
    END_IF;
    RETURN(the_reverse);

  END_FUNCTION; -- open_shell_reversed

  FUNCTION orthogonal_complement(
                vec: direction
      ): direction;

    LOCAL
      result : direction;
    END_LOCAL;
    IF (vec.dim <> 2) OR (NOT EXISTS(vec)) THEN
      RETURN(?);
    ELSE
      result := dummy_gri || direction([-vec.direction_ratios[2],vec.
          direction_ratios[1]]);
      RETURN(result);
    END_IF;

  END_FUNCTION; -- orthogonal_complement

  FUNCTION path_head_to_tail(
                a_path: path
      ): BOOLEAN;
```

 ©ISO 2000 - All rights reserved

```
  LOCAL
    n : INTEGER;
    p : BOOLEAN := TRUE;
  END_LOCAL;
  n := SIZEOF(a_path.edge_list);
  REPEAT i := 2 TO n BY 1;
    p := p AND (a_path.edge_list[i - 1].edge_end :=: a_path.edge_list[i]
        .edge_start);
  END_REPEAT;
  RETURN(p);

END_FUNCTION; -- path_head_to_tail

FUNCTION path_reversed(
             a_path: path
    ): oriented_path;

  LOCAL
    the_reverse : oriented_path;
  END_LOCAL;
  IF 'FEATURE_BASED_PROCESS_PLANNING.ORIENTED_PATH' IN TYPEOF(a_path)
       THEN
    the_reverse := dummy_tri || path(list_of_topology_reversed(a_path.
        edge_list)) || oriented_path(a_path\oriented_path.path_element,
        NOT a_path\oriented_path.orientation);
  ELSE
    the_reverse := dummy_tri || path(list_of_topology_reversed(a_path.
        edge_list)) || oriented_path(a_path,FALSE);
  END_IF;
  RETURN(the_reverse);

END_FUNCTION; -- path_reversed

FUNCTION scalar_times_vector(
             scalar: REAL;
             vec: vector_or_direction
    ): vector;

  LOCAL
    v      : direction;
    mag    : REAL;
    result : vector;
  END_LOCAL;
  IF (NOT EXISTS(scalar)) OR (NOT EXISTS(vec)) THEN
    RETURN(?);
  ELSE
    IF 'FEATURE_BASED_PROCESS_PLANNING.VECTOR' IN TYPEOF(vec) THEN
      v := dummy_gri || direction(vec.orientation.direction_ratios);
      mag := scalar * vec.magnitude;
    ELSE
      v := dummy_gri || direction(vec.direction_ratios);
      mag := scalar;
    END_IF;
    IF mag < 0 THEN
      REPEAT i := 1 TO SIZEOF(v.direction_ratios) BY 1;
        v.direction_ratios[i] := -v.direction_ratios[i];
      END_REPEAT;
      mag := -mag;
    END_IF;
    result := dummy_gri || vector(normalise(v),mag);
  END_IF;
```

©ISO 2000 - All rights reserved

```
    RETURN(result);

  END_FUNCTION; -- scalar_times_vector

  FUNCTION second_proj_axis(
               z_axis, x_axis, arg: direction
      ): direction;

    LOCAL
      temp   : vector;
      v      : direction;
      y_axis : vector;
    END_LOCAL;
    IF NOT EXISTS(arg) THEN
      v := dummy_gri || direction([0,1,0]);
    ELSE
      v := arg;
    END_IF;
    temp := scalar_times_vector(dot_product(v,z_axis),z_axis);
    y_axis := vector_difference(v,temp);
    temp := scalar_times_vector(dot_product(v,x_axis),x_axis);
    y_axis := vector_difference(y_axis,temp);
    y_axis := normalise(y_axis);
    RETURN(y_axis.orientation);

  END_FUNCTION; -- second_proj_axis

  FUNCTION set_of_topology_reversed(
               a_set: set_of_reversible_topology_item
      ): set_of_reversible_topology_item;

    LOCAL
      the_reverse : set_of_reversible_topology_item;
    END_LOCAL;
    the_reverse := [];
    REPEAT i := 1 TO SIZEOF(a_set) BY 1;
      the_reverse := the_reverse + topology_reversed(a_set[i]);
    END_REPEAT;
    RETURN(the_reverse);

  END_FUNCTION; -- set_of_topology_reversed

  FUNCTION shell_reversed(
               a_shell: shell
      ): shell;
    IF 'FEATURE_BASED_PROCESS_PLANNING.OPEN_SHELL' IN TYPEOF(a_shell)
         THEN
      RETURN(open_shell_reversed(a_shell));
    ELSE
      IF 'FEATURE_BASED_PROCESS_PLANNING.CLOSED_SHELL' IN TYPEOF(a_shell)
           THEN
        RETURN(closed_shell_reversed(a_shell));
      ELSE
        RETURN(?);
      END_IF;
    END_IF;

  END_FUNCTION; -- shell_reversed

  FUNCTION surface_weights_positive(
               b: rational_b_spline_surface
```

 ©ISO 2000 - All rights reserved

```
    ): BOOLEAN;

  LOCAL
    result : BOOLEAN := TRUE;
  END_LOCAL;
  REPEAT i := 0 TO b.u_upper BY 1;
    REPEAT j := 0 TO b.v_upper BY 1;
      IF b.weights[i][j] <= 0 THEN
        result := FALSE;
        RETURN(result);
      END_IF;
    END_REPEAT;
  END_REPEAT;
  RETURN(result);

END_FUNCTION; -- surface_weights_positive

FUNCTION topology_reversed(
             an_item: reversible_topology
    ): reversible_topology;
  IF 'FEATURE_BASED_PROCESS_PLANNING.EDGE' IN TYPEOF(an_item) THEN
    RETURN(edge_reversed(an_item));
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.PATH' IN TYPEOF(an_item) THEN
    RETURN(path_reversed(an_item));
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.FACE_BOUND' IN TYPEOF(an_item)
       THEN
    RETURN(face_bound_reversed(an_item));
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.FACE' IN TYPEOF(an_item) THEN
    RETURN(face_reversed(an_item));
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.SHELL' IN TYPEOF(an_item) THEN
    RETURN(shell_reversed(an_item));
  END_IF;
  IF 'SET' IN TYPEOF(an_item) THEN
    RETURN(set_of_topology_reversed(an_item));
  END_IF;
  IF 'LIST' IN TYPEOF(an_item) THEN
    RETURN(list_of_topology_reversed(an_item));
  END_IF;
  RETURN(?);

END_FUNCTION; -- topology_reversed

FUNCTION using_items(
             item: founded_item_select;
             checked_items: SET OF founded_item_select
    ): SET OF founded_item_select;

  LOCAL
    next_items      : SET OF founded_item_select;
    new_check_items : SET OF founded_item_select;
    result_items    : SET OF founded_item_select;
  END_LOCAL;
  result_items := [];
  new_check_items := checked_items + item;
  next_items := QUERY ( z <* bag_to_set(USEDIN(item,'')) | ((
      'FEATURE_BASED_PROCESS_PLANNING.REPRESENTATION_ITEM' IN TYPEOF(z))
      OR ('FEATURE_BASED_PROCESS_PLANNING.FOUNDED_ITEM' IN TYPEOF(z))) );
```

©ISO 2000 - All rights reserved

```
    IF SIZEOF(next_items) > 0 THEN
      REPEAT i := 1 TO HIINDEX(next_items) BY 1;
        IF NOT (next_items[i] IN new_check_items) THEN
          result_items := result_items + next_items[i] + using_items(
              next_items[i],new_check_items);
        END_IF;
      END_REPEAT;
    END_IF;
    RETURN(result_items);

  END_FUNCTION; -- using_items

  FUNCTION using_representations(
                item: founded_item_select
      ): SET OF representation;

    LOCAL
      results            : SET OF representation;
      intermediate_items : SET OF founded_item_select;
      result_bag         : BAG OF representation;
    END_LOCAL;
    results := [];
    result_bag := USEDIN(item,
        'FEATURE_BASED_PROCESS_PLANNING.REPRESENTATION.ITEMS');
    IF SIZEOF(result_bag) > 0 THEN
      REPEAT i := 1 TO HIINDEX(result_bag) BY 1;
        results := results + result_bag[i];
      END_REPEAT;
    END_IF;
    intermediate_items := using_items(item,[]);
    IF SIZEOF(intermediate_items) > 0 THEN
      REPEAT i := 1 TO HIINDEX(intermediate_items) BY 1;
        result_bag := USEDIN(intermediate_items[i],
            'FEATURE_BASED_PROCESS_PLANNING.REPRESENTATION.ITEMS');
        IF SIZEOF(result_bag) > 0 THEN
          REPEAT j := 1 TO HIINDEX(result_bag) BY 1;
            results := results + result_bag[j];
          END_REPEAT;
        END_IF;
      END_REPEAT;
    END_IF;
    RETURN(results);

  END_FUNCTION; -- using_representations

  FUNCTION valid_calendar_date(
                date: calendar_date
      ): LOGICAL;
    CASE date.month_component OF
      1 :         RETURN((1 <= date.day_component) AND (date.day_component
            <= 31));
      2 :         BEGIN
          IF leap_year(date.year_component) THEN
            RETURN((1 <= date.day_component) AND (date.day_component <=
29));
          ELSE
            RETURN((1 <= date.day_component) AND (date.day_component <=
28));
          END_IF;
        END;
      3 :         RETURN((1 <= date.day_component) AND (date.day_component
```

 ©ISO 2000 - All rights reserved

```
          <= 31));
    4 :         RETURN((1 <= date.day_component) AND (date.day_component
          <= 30));
    5 :         RETURN((1 <= date.day_component) AND (date.day_component
          <= 31));
    6 :         RETURN((1 <= date.day_component) AND (date.day_component
          <= 30));
    7 :         RETURN((1 <= date.day_component) AND (date.day_component
          <= 31));
    8 :         RETURN((1 <= date.day_component) AND (date.day_component
          <= 31));
    9 :         RETURN((1 <= date.day_component) AND (date.day_component
          <= 30));
    10  :         RETURN((1 <= date.day_component) AND (date.
          day_component <= 31));
    11  :         RETURN((1 <= date.day_component) AND (date.
          day_component <= 30));
    12  :         RETURN((1 <= date.day_component) AND (date.
          day_component <= 31));
    END_CASE;
  RETURN(FALSE);

END_FUNCTION; -- valid_calendar_date

FUNCTION valid_measure_value(
             m: measure_value
    ): BOOLEAN;
  IF 'REAL' IN TYPEOF(m) THEN
    RETURN(m > 0);
  ELSE
    IF 'INTEGER' IN TYPEOF(m) THEN
      RETURN(m > 0);
    ELSE
      RETURN(TRUE);
    END_IF;
  END_IF;

END_FUNCTION; -- valid_measure_value

FUNCTION valid_units(
             m: measure_with_unit
    ): BOOLEAN;
  IF 'FEATURE_BASED_PROCESS_PLANNING.LENGTH_MEASURE' IN TYPEOF(m.
      value_component) THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(1,0,0,0,0,0,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.MASS_MEASURE' IN TYPEOF(m.
      value_component) THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(0,1,0,0,0,0,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.TIME_MEASURE' IN TYPEOF(m.
      value_component) THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(0,0,1,0,0,0,0) THEN
      RETURN(FALSE);
```

©ISO 2000 - All rights reserved

```
    END_IF;
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.ELECTRIC_CURRENT_MEASURE' IN
      TYPEOF(m.value_component) THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(0,0,0,1,0,0,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.THERMODYNAMIC_TEMPERATURE_MEASURE'
      IN TYPEOF(m.value_component) THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(0,0,0,0,1,0,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.AMOUNT_OF_SUBSTANCE_MEASURE' IN
      TYPEOF(m.value_component) THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(0,0,0,0,0,1,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.LUMINOUS_INTENSITY_MEASURE' IN
      TYPEOF(m.value_component) THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(0,0,0,0,0,0,1) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.PLANE_ANGLE_MEASURE' IN TYPEOF(m.
      value_component) THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(0,0,0,0,0,0,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.SOLID_ANGLE_MEASURE' IN TYPEOF(m.
      value_component) THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(0,0,0,0,0,0,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.AREA_MEASURE' IN TYPEOF(m.
      value_component) THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(2,0,0,0,0,0,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.VOLUME_MEASURE' IN TYPEOF(m.
      value_component) THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(3,0,0,0,0,0,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.RATIO_MEASURE' IN TYPEOF(m.
      value_component) THEN
    IF derive_dimensional_exponents(m.unit_component) <>
```

 ©ISO 2000 - All rights reserved

```
        dimensional_exponents(0,0,0,0,0,0,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.POSITIVE_LENGTH_MEASURE' IN TYPEOF(
      m.value_component) THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(1,0,0,0,0,0,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  IF 'FEATURE_BASED_PROCESS_PLANNING.POSITIVE_PLANE_ANGLE_MEASURE' IN
      TYPEOF(m.value_component) THEN
    IF derive_dimensional_exponents(m.unit_component) <>
        dimensional_exponents(0,0,0,0,0,0,0) THEN
      RETURN(FALSE);
    END_IF;
  END_IF;
  RETURN(TRUE);

END_FUNCTION; -- valid_units

FUNCTION vector_difference(
             arg1, arg2: vector_or_direction
    ): vector;

  LOCAL
    ndim   : INTEGER;
    mag2   : REAL;
    mag1   : REAL;
    mag    : REAL;
    res    : direction;
    vec1   : direction;
    vec2   : direction;
    result : vector;
  END_LOCAL;
  IF (NOT EXISTS(arg1)) OR (NOT EXISTS(arg2)) OR (arg1.dim <> arg2.dim)
       THEN
    RETURN(?);
  ELSE
    BEGIN
      IF 'FEATURE_BASED_PROCESS_PLANNING.VECTOR' IN TYPEOF(arg1) THEN
        mag1 := arg1.magnitude;
        vec1 := arg1.orientation;
      ELSE
        mag1 := 1;
        vec1 := arg1;
      END_IF;
      IF 'FEATURE_BASED_PROCESS_PLANNING.VECTOR' IN TYPEOF(arg2) THEN
        mag2 := arg2.magnitude;
        vec2 := arg2.orientation;
      ELSE
        mag2 := 1;
        vec2 := arg2;
      END_IF;
      vec1 := normalise(vec1);
      vec2 := normalise(vec2);
      ndim := SIZEOF(vec1.direction_ratios);
      mag := 0;
      res := dummy_gri || direction(vec1.direction_ratios);
      REPEAT i := 1 TO ndim BY 1;
```

©ISO 2000 - All rights reserved

```
          res.direction_ratios[i] := (mag1 * vec1.direction_ratios[i]) + (
              mag2 * vec2.direction_ratios[i]);
          mag := mag + (res.direction_ratios[i] * res.direction_ratios[i]);
        END_REPEAT;
        IF mag > 0 THEN
          result := dummy_gri || vector(res,SQRT(mag));
        ELSE
          result := dummy_gri || vector(vec1,0);
        END_IF;
      END;
    END_IF;
    RETURN(result);

  END_FUNCTION; -- vector_difference

END_SCHEMA; -- feature_based_process_planning
```

 ©ISO 2000 - All rights reserved

Annex B
(normative)

AIM short names

Table B.1 provides the short names of entities specified in the AIM of this part of ISO 10303. Requirements on the use of the short names are found in the implementation methods included in ISO 10303.

Table B.1 - AIM short names of entities

Entity Name	Short Name
ACTION	ACTION
ACTION_ASSIGNMENT	ACTASS
ACTION_DIRECTIVE	ACTDRC
ACTION_METHOD	ACTMTH
ACTION_RELATIONSHIP	ACTRLT
ACTION_REQUEST_ASSIGNMENT	ACRQAS
ACTION_REQUEST_SOLUTION	ACRQSL
ACTION_REQUEST_STATUS	ACRQST
ACTION_STATUS	ACTSTT
ADDRESS	ADDRSS
ADVANCED_BREP_SHAPE_REPRESENTATION	ABSR
ADVANCED_FACE	ADVFC
ANGULARITY_TOLERANCE	ANGTLR
ANGULAR_LOCATION	ANGLCT
ANGULAR_SIZE	ANGSZ
APPLICATION_CONTEXT	APPCNT
APPLICATION_CONTEXT_ELEMENT	APCNEL
APPLICATION_PROTOCOL_DEFINITION	APPRDF
APPLIED_AREA	APPAR
APPLIED_DOCUMENT_REFERENCE	APDCRF
APPLIED_DOCUMENT_USAGE_CONSTRAINT_ASSIGNMENT	ADUCA

©ISO 2000 - All rights reserved

Table B.1 - AIM short names of entities (continued)

APPLIED_GROUP_ASSIGNMENT	APGRAS
APPROVAL	APPRVL
APPROVAL_ASSIGNMENT	APPASS
APPROVAL_DATE_TIME	APDTTM
APPROVAL_PERSON_ORGANIZATION	APPROR
APPROVAL_ROLE	APPRL
APPROVAL_STATUS	APPSTT
ASSEMBLY_COMPONENT_USAGE	ASCMUS
AXIS1_PLACEMENT	AX1PLC
AXIS2_PLACEMENT_2D	A2PL2D
AXIS2_PLACEMENT_3D	A2PL3D
BEZIER_CURVE	BZRCRV
BEZIER_SURFACE	BZRSRF
BLOCK_SHAPE_REPRESENTATION	BLSHRP
BOSS	BOSS
BOSS_TOP	BSSTP
BOUNDED_CURVE	BNDCRV
BOUNDED_SURFACE	BNDSRF
BREP_WITH_VOIDS	BRWTVD
B_SPLINE_CURVE	BSPCR
B_SPLINE_CURVE_WITH_KNOTS	BSCWK
B_SPLINE_SURFACE	BSPSR
B_SPLINE_SURFACE_WITH_KNOTS	BSSWK
CALENDAR_DATE	CLNDT
CARTESIAN_POINT	CRTPNT
CARTESIAN_TRANSFORMATION_OPERATOR	CRTROP
CARTESIAN_TRANSFORMATION_OPERATOR_3D	CTO3
CHAMFER	CHMFR

 ©ISO 2000 - All rights reserved

Table B.1 - AIM short names of entities (continued)

CHAMFER_OFFSET	CHMOFF
CHARACTERIZED_OBJECT	CHROBJ
CIRCLE	CIRCLE
CIRCULAR_CLOSED_PROFILE	CRCLPR
CIRCULAR_PATTERN	CRCPTT
CIRCULAR_RUNOUT_TOLERANCE	CRRNTL
CLOSED_PATH_PROFILE	CLPTPR
CLOSED_SHELL	CLSSHL
COMPOSITE_CURVE	CMPCRV
COMPOSITE_CURVE_SEGMENT	CMCRSG
COMPOSITE_HOLE	CMPHL
COMPOSITE_SHAPE_ASPECT	CMSHAS
COMPOUND_FEATURE	CMPFTR
CONCENTRICITY_TOLERANCE	CNCTLR
CONIC	CONIC
CONICAL_SURFACE	CNCSRF
CONNECTED_FACE_SET	CNFCST
CONTEXT_DEPENDENT_UNIT	CNDPUN
CONVERSION_BASED_UNIT	CNBSUN
CURVE	CURVE
CYLINDRICAL_SHAPE_REPRESENTATION	CYSHRP
CYLINDRICAL_SURFACE	CYLSRF
CYLINDRICITY_TOLERANCE	CYLTLR
DATA_ENVIRONMENT	DTENV
DATE	DATE
DATE_ASSIGNMENT	DTASS
DATE_ROLE	DTRL
DATUM	DATUM

©ISO 2000 - All rights reserved

Table B.1 - AIM short names of entities (continued)

DATUM_FEATURE	DTMFTR
DATUM_REFERENCE	DTMRFR
DATUM_TARGET	DTMTRG
DEFINITIONAL_REPRESENTATION	DFNRPR
DEGENERATE_TOROIDAL_SURFACE	DGTRSR
DERIVED_UNIT	DRVUNT
DERIVED_UNIT_ELEMENT	DRUNEL
DESCRIPTION_ATTRIBUTE	DSCATT
DESCRIPTIVE_REPRESENTATION_ITEM	DSRPIT
DIMENSIONAL_CHARACTERISTIC_REPRESENTATION	DMCHRP
DIMENSIONAL_EXPONENTS	DMNEXP
DIMENSIONAL_LOCATION	DMNLCT
DIMENSIONAL_LOCATION_WITH_PATH	DLWP
DIMENSIONAL_SIZE	DMNSZ
DIMENSIONAL_SIZE_WITH_PATH	DSWP
DIRECTED_ACTION	DRCACT
DIRECTED_DIMENSIONAL_LOCATION	DRDMLC
DIRECTION	DRCTN
DIRECTION_SHAPE_REPRESENTATION	DRSHRP
DOCUMENT	DCMNT
DOCUMENT_FILE	DCMFL
DOCUMENT_REFERENCE	DCMRFR
DOCUMENT_RELATIONSHIP	DCMRLT
DOCUMENT_REPRESENTATION_TYPE	DCRPTY
DOCUMENT_TYPE	DCMTYP
DOCUMENT_USAGE_CONSTRAINT	DCUSCN
DOCUMENT_USAGE_CONSTRAINT_ASSIGNMENT	DUCA
DOCUMENT_USAGE_ROLE	DCUSRL

 ©ISO 2000 - All rights reserved

Table B.1 - AIM short names of entities (continued)

DOCUMENT_WITH_CLASS	DCWTCL
EDGE	EDGE
EDGE_CURVE	EDGCRV
EDGE_LOOP	EDGLP
EDGE_ROUND	EDGRND
ELEMENTARY_SURFACE	ELMSRF
ELLIPSE	ELLPS
EXECUTED_ACTION	EXCACT
EXPANDED_UNCERTAINTY	EXPUNC
EXTERNALLY_DEFINED_FEATURE_DEFINITION	EDFD
EXTERNALLY_DEFINED_ITEM	EXDFIT
EXTERNAL_SOURCE	EXTSRC
FACE	FACE
FACE_BOUND	FCBND
FACE_OUTER_BOUND	FCOTBN
FACE_SHAPE_REPRESENTATION	FCSHRP
FACE_SHAPE_REPRESENTATION_RELATIONSHIP	FSRR
FACE_SURFACE	FCSRF
FEATURE_BASED_PP_ACTION_ASSIGNMENT	FBPAA
FEATURE_BASED_PP_ACTION_REQUEST_ASSIGNMENT	FBPARA
FEATURE_BASED_PP_APPROVAL_ASSIGNMENT	FBP0
FEATURE_BASED_PP_DATE_ASSIGNMENT	FBPDA
FEATURE_BASED_PP_ORGANIZATION_ASSIGNMENT	FBPOA
FEATURE_BASED_PP_PERSON_AND_ORGANIZATION_ASSIGNMENT	FBPPAO
FEATURE_BASED_PP_SECURITY_CLASSIFICATION_ASSIGNMENT	FBPSCA
FEATURE_COMPONENT_DEFINITION	FTCMDF
FEATURE_COMPONENT_RELATIONSHIP	FTCMRL

©ISO 2000 - All rights reserved

Table B.1 - AIM short names of entities (continued)

FEATURE_DEFINITION	FTRDFN
FEATURE_PATTERN	FTRPTT
FILLET	FILLET
FLATNESS_TOLERANCE	FLTTLR
FLAT_FACE	FLTFC
FOUNDED_ITEM	FNDITM
FUNCTIONALLY_DEFINED_TRANSFORMATION	FNDFTR
GEOMETRIC_REPRESENTATION_CONTEXT	GMRPCN
GEOMETRIC_REPRESENTATION_ITEM	GMRPIT
GEOMETRIC_TOLERANCE	GMTTLR
GEOMETRIC_TOLERANCE_RELATIONSHIP	GMTLRL
GEOMETRIC_TOLERANCE_WITH_DATUM_REFERENCE	GTWDR
GEOMETRIC_TOLERANCE_WITH_DEFINED_UNIT	GTWDU
GLOBAL_UNCERTAINTY_ASSIGNED_CONTEXT	GC
GLOBAL_UNIT_ASSIGNED_CONTEXT	GUAC
GROUP	GROUP
GROUP_ASSIGNMENT	GRPASS
GROUP_RELATIONSHIP	GRPRLT
HOLE_BOTTOM	HLBTT
HYPERBOLA	HYPRBL
ID_ATTRIBUTE	IDATT
INSTANCED_FEATURE	INSFTR
LENGTH_MEASURE_WITH_UNIT	LMWU
LENGTH_UNIT	LNGUNT
LIMITS_AND_FITS	LMANFT
LINE	LINE
LINEAR_PROFILE	LNRPRF
LINE_PROFILE_TOLERANCE	LNP0

 ©ISO 2000 - All rights reserved

Table B.1 - AIM short names of entities (continued)

LOCATION_SHAPE_REPRESENTATION	LCSHRP
LOOP	LOOP
MAKE_FROM_USAGE_OPTION	MFUO
MANIFOLD_SOLID_BREP	MNSLBR
MAPPED_ITEM	MPPITM
MARKING	MRKNG
MATERIAL_DESIGNATION	MTRDSG
MATERIAL_PROPERTY	MTRPRP
MATERIAL_PROPERTY_REPRESENTATION	MTPRRP
MEASURE_QUALIFICATION	MSRQLF
MEASURE_REPRESENTATION_ITEM	MSRPIT
MEASURE_WITH_UNIT	MSWTUN
MODIFIED_GEOMETRIC_TOLERANCE	MDGMTL
MODIFIED_PATTERN	MDFPTT
NAMED_UNIT	NMDUNT
NAME_ATTRIBUTE	NMATT
NEXT_ASSEMBLY_USAGE_OCCURRENCE	NAUO
NGON_CLOSED_PROFILE	NGCLPR
NGON_SHAPE_REPRESENTATION	NGSHRP
OBJECT_ROLE	OBJRL
OPEN_PATH_PROFILE	OPPTPR
OPEN_SHELL	OPNSHL
ORDERED_PART	ORDPRT
ORDINAL_DATE	ORDDT
ORGANIZATION	ORGNZT
ORGANIZATIONAL_ADDRESS	ORGADD
ORGANIZATION_ASSIGNMENT	ORGASS
ORGANIZATION_ROLE	ORGRL

©ISO 2000 - All rights reserved

Table B.1 - AIM short names of entities (continued)

ORIENTED_CLOSED_SHELL	ORCLSH
ORIENTED_EDGE	ORNEDG
ORIENTED_FACE	ORNFC
ORIENTED_OPEN_SHELL	OROPSH
ORIENTED_PATH	ORNPTH
OUTER_ROUND	OTRRND
OUTSIDE_PROFILE	OTSPRF
PARABOLA	PRBL
PARALLELISM_TOLERANCE	PRLTLR
PARAMETRIC_REPRESENTATION_CONTEXT	PRRPCN
PARTIAL_CIRCULAR_PROFILE	PRCRPR
PATH	PATH
PATH_FEATURE_COMPONENT	PTFTCM
PATH_SHAPE_REPRESENTATION	PTSHRP
PATTERN_OFFSET_MEMBERSHIP	PTOFMM
PATTERN_OMIT_MEMBERSHIP	PTOMMM
PCURVE	PCURVE
PERPENDICULARITY_TOLERANCE	PRPTLR
PERSON	PERSON
PERSONAL_ADDRESS	PRSADD
PERSON_AND_ORGANIZATION	PRANOR
PERSON_AND_ORGANIZATION_ASSIGNMENT	PAOA
PERSON_AND_ORGANIZATION_ROLE	PAOR
PLACED_DATUM_TARGET_FEATURE	PDT0
PLACEMENT	PLCMNT
PLANAR_SHAPE_REPRESENTATION	PLSHRP
PLANE	PLANE
PLANE_ANGLE_MEASURE_WITH_UNIT	PAMWU

 ©ISO 2000 - All rights reserved

Table B.1 - AIM short names of entities (continued)

PLANE_ANGLE_UNIT	PLANUN
PLUS_MINUS_TOLERANCE	PLMNTL
POCKET	POCKET
POCKET_BOTTOM	PCKBTT
POINT	POINT
POLYLINE	PLYLN
POSITION_TOLERANCE	PSTTLR
PRECISION_QUALIFIER	PRCQLF
PRODUCT	PRDCT
PRODUCT_CONTEXT	PRDCNT
PRODUCT_DEFINITION	PRDDFN
PRODUCT_DEFINITION_CONTEXT	PRDFCN
PRODUCT_DEFINITION_FORMATION	PRDFFR
PRODUCT_DEFINITION_RELATIONSHIP	PRDFRL
PRODUCT_DEFINITION_SHAPE	PRDFSH
PRODUCT_DEFINITION_USAGE	PRDFUS
PRODUCT_DEFINITION_WITH_ASSOCIATED_DOCUMENTS	PDWAD
PROFILE_FLOOR	PRFFLR
PROJECTED_ZONE_DEFINITION	PRZNDF
PROPERTY_DEFINITION	PRPDFN
PROPERTY_DEFINITION_RELATIONSHIP	PRDFR
PROPERTY_DEFINITION_REPRESENTATION	PRDFRP
PROTRUSION	PRTRSN
QUALIFIED_REPRESENTATION_ITEM	QLRPIT
QUALITATIVE_UNCERTAINTY	QLTUNC
QUASI_UNIFORM_CURVE	QSUNCR
QUASI_UNIFORM_SURFACE	QSUNSR
RATIONAL_B_SPLINE_CURVE	RBSC

©ISO 2000 - All rights reserved

Table B.1 - AIM short names of entities (continued)

RATIONAL_B_SPLINE_SURFACE	RBSS
RATIO_MEASURE_WITH_UNIT	RMWU
RATIO_UNIT	RTUNT
RECTANGULAR_CLOSED_PROFILE	RCCLPR
RECTANGULAR_PATTERN	RCTPTT
REFERENCED_MODIFIED_DATUM	RFMDDT
REMOVAL_VOLUME	RMVVLM
REPLICATE_FEATURE	RPLFTR
REPRESENTATION	RPRSNT
REPRESENTATION_CONTEXT	RPRCNT
REPRESENTATION_ITEM	RPRITM
REPRESENTATION_MAP	RPRMP
REPRESENTATION_RELATIONSHIP	RPRRLT
REVOLVED_PROFILE	RVLPRF
RIB_TOP	RBTP
RIB_TOP_FLOOR	RBTPFL
ROLE_ASSOCIATION	RLASS
ROUNDED_END	RNDEND
ROUNDED_U_PROFILE	RNUPR
ROUNDNESS_TOLERANCE	RNDTLR
ROUND_HOLE	RNDHL
RUNOUT_ZONE_DEFINITION	RNZNDF
RUNOUT_ZONE_ORIENTATION	RNZNOR
SECURITY_CLASSIFICATION	SCRCLS
SECURITY_CLASSIFICATION_ASSIGNMENT	SCCLAS
SECURITY_CLASSIFICATION_LEVEL	SCCLLV
SHAPE_ASPECT	SHPASP
SHAPE_ASPECT_RELATIONSHIP	SHASRL

 ©ISO 2000 - All rights reserved

Table B.1 - AIM short names of entities (continued)

SHAPE_DEFINING_RELATIONSHIP	SHDFRL
SHAPE_DEFINITION_REPRESENTATION	SHDFRP
SHAPE_DIMENSION_REPRESENTATION	SHDMRP
SHAPE_REPRESENTATION	SHPRPR
SHAPE_REPRESENTATION_WITH_PARAMETERS	SRWP
SI_UNIT	SUNT
SLOT	SLOT
SLOT_END	SLTEND
SOLID_ANGLE_MEASURE_WITH_UNIT	SAMWU
SOLID_ANGLE_UNIT	SLANUN
SOLID_MODEL	SLDMDL
SPHERICAL_CAP	SPHCP
SPHERICAL_SURFACE	SPHSRF
SQUARE_U_PROFILE	SQUPR
STANDARD_UNCERTAINTY	STNUNC
STEP	STEP
STRAIGHTNESS_TOLERANCE	STRTLR
SURFACE	SRFC
SURFACE_CURVE	SRFCRV
SURFACE_OF_LINEAR_EXTRUSION	SL
SURFACE_OF_REVOLUTION	SROFRV
SURFACE_PROFILE_TOLERANCE	SRPRTL
SWEPT_SURFACE	SWPSRF
SYMMETRY_TOLERANCE	SYMTLR
TAPER	TAPER
TEE_PROFILE	TPRF
THREAD	THREAD
TOLERANCE_VALUE	TLRVL

©ISO 2000 - All rights reserved

Table B.1 - AIM short names of entities (concluded)

TOLERANCE_ZONE	TLRZN
TOLERANCE_ZONE_DEFINITION	TLZNDF
TOLERANCE_ZONE_FORM	TLZNFR
TOPOLOGICAL_REPRESENTATION_ITEM	TPRPIT
TOROIDAL_SURFACE	TRDSRF
TOTAL_RUNOUT_TOLERANCE	TTRNTL
TRANSITION_FEATURE	TRNFTR
TURNED_KNURL	TRNKNR
TYPE_QUALIFIER	TYPQLF
UNCERTAINTY_MEASURE_WITH_UNIT	UMWU
UNCERTAINTY_QUALIFIER	UNCQLF
UNIFORM_CURVE	UNFCRV
UNIFORM_SURFACE	UNFSRF
VECTOR	VECTOR
VEE_PROFILE	VPRF
VERSIONED_ACTION_REQUEST	VRACRQ
VERTEX	VERTEX
VERTEX_LOOP	VRTLP
VERTEX_POINT	VRTPNT
WEEK_OF_YEAR_AND_DAY_DATE	WOYADD

 ©ISO 2000 - All rights reserved

Annex C
(normative)

Implementation method specific requirements

The implementation method defines what type of exchange behavior is required with respect to this part of ISO 10303. Conformance to this part of ISO 10303 shall be realized in an exchange structure. The file format shall be encoded according to the syntax and EXPRESS language mapping defined in ISO 10303-21 and annotated listing defined in Annex A of this part of ISO 10303. The header of the exchange structure shall identify use of this part of ISO 10303 by the schema name 'feature_based_-process_planning'.

©ISO 2000 - All rights reserved

Annex D
(normative)

Protocol Implementation Conformance Statement (proforma)

This clause list the optional elements of this part of ISO 10303. An implementation may chose to support any combination of these optional elements. However, certain combinations of options are likely to be implemented together. These combinations are called conformance classes and are described in the subclauses of this annex.

This annex is in the form of a questionnaire. This questionnaire is intended to be filled out by the filled out by the implementor and may be used in preparation for conformance testing by a testing laboratory. The completed PICS proforma is referred to as a PICS.

The information contained in the PICS is used to configure as appropriate executable test suite for use by the client.

A single conformance class is identified in this part of ISO 10303. A conforming implementation shall support this one conformance class. This class is detailed in clause 6 of ISO 10303-224.

Question:

1. Please provide an identifier for the product or system for which conformance is claimed:

    Product name and current version number:_____

2. Please indicate the implementation method chosen:

    – ISO 10303-21 Exchange Structure -- preprocessor
    Preprocessor name and current version number:_____

    – ISO 10303-21 Exchange Structure -- postprocessor
    Postprocessor name and current version number:_____

 ©ISO 2000 - All rights reserved

Annex E
(normative)

Information object registration

E.1 Document identification

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

{iso standard 10303 part(224) version (3) }

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

E.2 Schema identification

To provide for unambiguous identification of the schema specification given in this application protocol feature-based-process-planning-schema in the open information system, the object identifier are assigned as follows:

{ iso standard 10303 part(224) version (3) object (1) feature-based-process-planning-schema(1) }

is assigned to the feature_based_process_planning_schema (see annex A).

{ iso standard 10303 part(224) version (3) object (1) feature-based-process-planning-schema(2) }

is assigned to the feature_based_process_planning schema short form schema (see 5.2).

The meaning of this value is defined in ISO 8824-1, and is described in ISO 10303-1.

©ISO 2000 - All rights reserved

Annex F
(informative)

Application activity model

The application activity model (AAM) is provided to aid in the understanding the scope and information requirements defined in this application protocol. The model is presented as a set of activity figures that contain the activity diagrams and a set of definitions of the activities and their data.

The AAM covers activities which go beyond the subject of this application protocol. The diagrams use a modified IDEF0 notation [2]. Figure F.1 gives the basic notation. Each activity may be decomposed to provide more detail. If an activity has been decomposed, a separate figure is included.

As with any IDEF0 model, the application activity model is dependent on a particular viewpoint and purpose. The viewpoint of the application activity model is from a manufacturing engineer. The purpose of the application activity model is to clarify the context and scope of this application protocol.

This is an activity model of an enterprise for manufacturing a part. There are several activity diagrams that have all activities out of scope but they are important in illustrating how the manufacture of a part process was developed and how the in-scope requirements were derived.

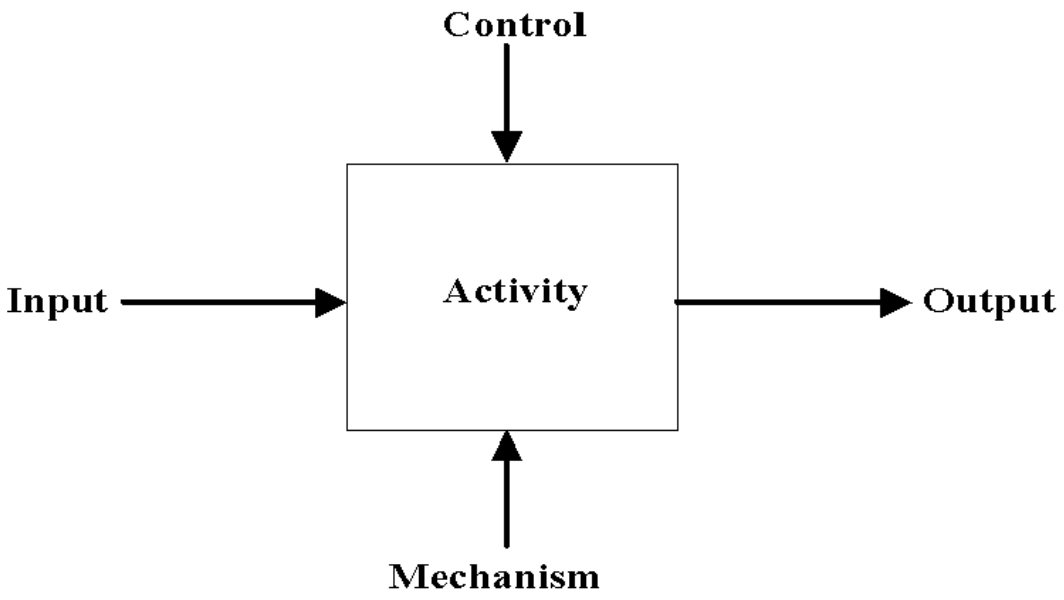


Figure F.1 - IDEF0 Basic notation

 ©ISO 2000 - All rights reserved

F.1 Application activity model definitions

The following terms are used in the application activity model. Terms marked with an asterisk are outside the scope of this application protocol.

The definitions in this annex do not supersede the definitions given in the main body of the text.

F.1.1 Application activity model activity definitions

The following terms are used to define the activities of the AAM diagrams.

F.1.1.1 Capture Digital Product Data Definition (A2): Executes the methods available, for transforming part technical data into ISO 10303 product data.

F.1.1.2 Capture Notes (A2141): Compiles non-graphical information contained in the part information packet and represents it in the digital part model.

F.1.1.3 Capture Part Data Definition (A21): Examines the part information packet, determines the required information for a complete part definition, and creates digital part data definition.

F.1.1.4 Capture Part Management Data (A211): Collects product life cycle information associated with the part design from the part information packet.

F.1.1.5 Capture Part Shape Data (A2142): Interprets the graphical shape representation of the part from the part information packet, and creates an equivalent shape representation in the digital part model.

F.1.1.6 Capture Part Tolerance (A2143): Defines dimensional and geometric tolerance information contained in the part information packet in the digital part model.

F.1.1.7 Compile Part Pedigree Data (A42)*: Tracks and compiles required historical information pertaining to a part throughout the entire manufacturing process as defined by a customer.

F.1.1.8 Control Workstations and Coordinate Transportation (A41)*: Provides all of the information processing required to support the execution of production tasks and to support the transport of items.

F.1.1.9 Create Digital Product Data Definition (A25): Creates digital product data definition for a specific part based on the part information packet.

F.1.1.10 Create Manufacturing Bill of Materials (A314)*: Generates a list of raw stock and indirect material items that must be available prior to manufacturing the part.

F.1.1.11 Create Part Model (A214): Develops a digital model of a part from the information contained in the part information packet.

F.1.1.12 Create Shop Work Order (A32)*: Prepares a shop work agenda and associated information required to route the part through the shop floor for manufacture.

©ISO 2000 - All rights reserved

F.1.1.13 Define Cutting Tools (A3111)*: Selects the machine tools required to perform machining operations on the part.

F.1.1.14 Define External Processes (A3114)*: Determines the required manufacturing processes that cannot be satisfied by resources available on the shop floor.

F.1.1.15 Define Gauges (A3115)*: Selects machine gauges required to perform inspection operations on the part.

F.1.1.16 Define Machine Instructions (A315)*: Produces the machine controller instructions required for the machines to perform operations on a given part.

F.1.1.17 Define Manufacturing Features (A2145): Captures design features of a part specified in the part information packet and represents them as manufacturing features in the digital part model.

F.1.1.18 Define Materials (A3116)*: Identifies the raw materials required to produce the part.

F.1.1.19 Define Modular and Dedicated Fixturing (A3112)*: Selects machine fixtures required to perform machining operations on the part.

F.1.1.20 Define Operator Data (A312)*: Produces graphical and textual instructions required by the operator to perform machining and inspection operations on a given part.

F.1.1.21 Define Part Properties (A2144): Captures non-graphical material and surface properties, and process information contained in the part information packet.

F.1.1.22 Define Resources (A311)*: Generates the list of items that must be available prior to the manufacturing process.

F.1.1.23 Define Route (A313)*: Produces the complete list of sequenced workstation types and cells including descriptions of the operations to be performed at each workstation.

F.1.1.24 Define Shop Floor Graphics (A3121)*: Creates graphical representation of the fixtures and tools necessary for visual instructions to perform machining operations on the part.

F.1.1.25 Define Shop Floor Operator Instructions (A3122)*: Describes the machine setup and machine tear-down instructions for a set of machining operations on a given part.

F.1.1.26 Define Workstations (A3113)*: Activity specifies candidate workstations machining operations for a given part are to be performed.

NOTE - The workstations chosen are determined from the workstation capabilities (defined by the types of components and materials that can be processed, workstation limitations and ranges of operations) and availability.

F.1.1.27 Determine Production Cost & Schedule (A12)*: Delineates all costs incurred in the production of a part and the detailed agenda of task assignments and resource allocations to execute the manufacturing process.

 ©ISO 2000 - All rights reserved

F.1.1.28 Establish Part Acquisition Order (A11)*: Compiles customer part requisition information and initiates an order for work to begin manufacturing a particular part.

F.1.1.29 Execute Process Plan (A43)*: Carries out instructions specified in the process plan to manufacture a part.

F.1.1.30 Generate Manufacturing Data (A3)*: Produces information for use on the shop floor, defining the method to be used to manufacture a particular part, and feedback about the digital product data definition if necessary.

F.1.1.31 Generate Process Plan (A31)*: Creates process plans, and determines resources required for the process plan.

F.1.1.32 Generate Quality Reports (A44)*: Collects part quality data and inspection results generated during the execution of the process plan operations and assembles reports.

F.1.1.33 Generate Test and Inspection Plan (A3123)*: Generates the instructions for any final inspection or test procedures to be executed after a part has been manufactured and before the part is shipped.

F.1.1.34 Identify & Evaluate Design Problem (A33)*: Determines design exceptions discovered during creation of the process plan for a part.

NOTE - Anomalies are evaluated for corrective action to be taken, if any; otherwise, the condition will warrant termination of the process plan generation.

F.1.1.35 Identify Base Shape (A212): Determines the basic part shape as the building block for part model creation.

EXAMPLE - Examples of basic part shapes may be cylinder, cube, or prism.

F.1.1.36 Identify Manufacturing Features (A213): Recognizes and classifies manufacturing features to be applied to the base shape for part model creation.

F.1.1.37 Import ISO 10303 Data (A22): Reads in a ISO 10303 part definition and creates digital product data definition.

F.1.1.38 Maintain Shop Floor Resources (A132)*: Generates internal preventative maintenance requirements, and schedules and requests internal and external preventive and outage maintenance.

F.1.1.39 Manage Equipment & Materials (A13)*: Tracks and coordinates material and equipment usage, and maintenance.

F.1.1.40 Manage Inventory (A131)*: Tracks and acquires necessary materials and equipment needed for manufacturing.

F.1.1.41 Manage Manufacturing Process (A1)*: Provides administrative support for manufacturing mechanical parts from receipt of a customer part requisition through part shipment.

©ISO 2000 - All rights reserved

F.1.1.42 Manufacture Mechanical Parts (A0): Defines and processes all data and interfaces necessary for manufacturing machined parts.

F.1.1.43 Monitor Cutting Tool Inventory (A1311)*: Tracks and obtains necessary tools needed to support manufacturing.

F.1.1.44 Monitor Dedicated Fixture Inventory (A1314)*: Tracks and obtains necessary dedicated fixtures needed to support manufacturing.

F.1.1.45 Monitor Indirect Inventory (A1316)*: Tracks and obtains necessary indirect material items needed to support manufacturing. Indirect materials include both consumable and deliverable items.

F.1.1.46 Monitor Machine Inventory (A1315)*: Tracks and obtains necessary machines needed to support manufacturing.

F.1.1.47 Monitor Material Inventory (A1312)*: Tracks and obtains necessary raw stock and components needed to support manufacturing.

F.1.1.48 Monitor Modular Fixture Inventory (A1313)*: Tracks and obtains necessary modular fixtures needed to support manufacturing.

F.1.1.49 Operate Shop Floor (A4)*: Executes the work plans required for manufacturing a part.

F.1.1.50 Pack & Ship Finished Parts (A45)*: Coordinates packing and shipment of finished parts to the customer and receipt of shipment notifications.

F.1.1.51 Process Design Exception (A15)*: Evaluates a design problem encountered during the creation of a process plan and determines a recommendation for design problem resolution.

F.1.1.52 Process Discrepant Part (A14)*: Determines services needed to investigate the causes of inspection failure, process plan failure, or process plan execution failure, and defines the corrective action to be taken.

F.1.1.53 Produce Machine Code (A3151)*: Compute the numerical control instructions for machine controllers.

F.1.1.54 Refine Product Data Definition (A23): Corrects data definition errors and/or modifies/adds the information necessary to completely define the part for manufacturing.

F.1.1.55 Release & Track Shop Work Order (A16)*: Releases the schedule and tracking information required for manufacturing a part on the shop floor.

F.1.1.56 Track Shop Floor Utilization (A133)*: Monitors machine work loads.

F.1.1.57 Validate Part Model (A215): Validates the accuracy and completeness of the digital part model with respect to the part data definition from the part information packet.

F.1.1.58 Verify Product Data Definition (A24): Verifies that the product data definition is syntactically and semantically correct, and validated for data completeness.

 ©ISO 2000 - All rights reserved

F.1.1.59 Verify Machine Tool Path (A3152)*: Simulates the operations of a workstation where the results are reviewed and verified to determine whether or not the machine programs will function correctly.

F.1.2 Application activity model ICOM definitions

The following terms are used to define the inputs, controls, outputs, and mechanisms of the AAM diagrams.

F.1.2.1 Application Interpreted Model Schema: An EXPRESS schema that specifies the ISO 10303 constructs used for the communication of the information specified in the application reference model of an AP.

F.1.2.2 Basic Part Shape: The categorization of the shape of a given part as determined from the part technical data that is used as the basis for machining feature application.

F.1.2.3 Bill of Materials:* A definition of all materials, parts, components and special tools necessary to manufacture a given part.

F.1.2.4 Computer Integrated Manufacturing: Hardware and software technology to aide in the manufacturing of a part.

EXAMPLE - Computer-aided process planning (CAPP), Computer-aided manufacturing(CAM), Computer-aided design(CAD) are examples of computer integrated manufacturing.

F.1.2.5 Corrective Action Plan:* Process adjustment instructions that are required to correct a shop floor operation problem.

F.1.2.6 Customer Order: Notification that data has been received from the customer necessary to process a request for the manufacture of a part.

F.1.2.7 Cutting Tools:* A resource required for material removal in machining operations as defined in the process plan.

F.1.2.8 Cutting Tool Requisition: An order for tools required to support part manufacturing.

F.1.2.9 Dedicated Fixture Requisition: An order for dedicated fixtures required to support part manufacturing.

F.1.2.10 Design Exception Notice: The notification of a design discrepancy discovered during the creation of the process plan for a given part.

EXAMPLE - A tolerance called out on a feature may not be held by a machine.

F.1.2.11 Design Specifications: Part design requirements that manufacturing process must follow or adhere.

F.1.2.12 Digital Product Data Definition: The set of data elements that completely defines a given part represented in a computer interpretable form.

©ISO 2000 - All rights reserved

F.1.2.13 Digital Product Data Schema: A view of a physical representation of product data on a computer. The schema encodes the semantics that completely define the product for manufacturing in some computer interpretable syntax.

F.1.2.14 Digital Product Data Work Order: Notification that the data needed to create the digital product data definition for a part has been collected.

F.1.2.15 Electrical Property Specifications: Document containing part electrical characteristic requirements that manufacturing process must follow or adhere.

F.1.2.16 Engineering Service Request:* A description of a problem encountered during the execution of the process plan and a request for determination of the problem cause and the development of corrective actions or corrective services.

F.1.2.17 Equipment and Materials Requisition: An order for items required to support part manufacturing.

F.1.2.18 Equipment Requirements: Needs for items to produce a part as determined by process planning.

F.1.2.19 Estimated Process Time:* Time measurements for specific operations at a workstation determined from approved standard operation times for that workstation.

F.1.2.20 External Processes:* Specified processes defined for a given part that are performed outside the shop that the part is being machined.

F.1.2.21 Featured Part Shape: The representation of the shape of a part defined by manufacturing form features.

F.1.2.22 Gauges:* A resource required for part measurements as defined in the process plan.

F.1.2.23 Geometric/Topological Shape Representation: Part shape as defined by geometric and topological data elements.

F.1.2.24 Indirect Stock Requisition: An order for indirect items required to support part manufacturing. Indirect materials include both consumable and deliverable items.

F.1.2.25 Inspection Results:* Part inspection information generated from the actual manufacturing execution of process plan operations.

F.1.2.26 Inspection Specifications: Document containing part inspection requirements that manufacturing process must follow or adhere.

F.1.2.27 Inventory Status:* An availability check of inventory on order and on hand.

F.1.2.28 ISO 10303 Physical File Definition: A definition of a file that conforms to ISO 10303-21 containing ISO 10303 product data that is to be exchanged between computer systems.

 ©ISO 2000 - All rights reserved

F.1.2.29 Machine Code:* Machine instructions required by a machine tool controller to automatically control the machining of a given part.

F.1.2.30 Machine Code Feedback:* Instructions that define how to modify numerical control (NC) code that did not pass verification analysis and requires further refinement.

F.1.2.31 Machine Instructions:*Commands that are sent to a machine to execute a task. Typically machine instructions are in the form of NC programs to be downloaded to the machine controller or a machine tool. Directions in a form understandable by a machine that produce the desired transformations of an input material into an intermediate or final form.

F.1.2.32 Machine Requisition: An order for machines required to support part manufacturing.

F.1.2.33 Machining Features: A shape that conforms to some preconceived pattern for an application purpose.

F.1.2.34 Maintenance Schedule:* An agenda of maintenance and repair activities for the shop floor.

F.1.2.35 Manufacturing Practices:* Procedures pertaining to the manufacturing process that provide intuitive knowledge about certain aspects of manufacturing a part that are generally accepted as standard within the manufacturing industry.

F.1.2.36 Materials:* A definition of the raw stock as called out in the process plan.

F.1.2.37 Material Properties: Characteristics of the part materials defined in the part technical data.

F.1.2.38 Material Requirements: Information about the material required to produce the part as interpreted from the technical data packet.

F.1.2.39 Material Requisition: An order for materials required to support part manufacturing.

F.1.2.40 Material Specifications: Document containing part material requirements that manufacturing process must follow or adhere.

F.1.2.41 Mechanical Part:* A thing of substance produced from raw materials. A mechanical part contains physical material characteristics.

F.1.2.42 Modular and Dedicated Fixtures:* A resource required to secure a given part for machining operations as defined in the process plan.

F.1.2.43 Modular Fixture Requisition: An order for modular fixtures required to support part manufacturing.

F.1.2.44 Operator Data:* Both the graphical and textual data supplied to an operator on the shop floor.

F.1.2.45 Optical Property Specifications: Document containing part optical characteristic requirements that the manufacturing process must follow or adhere.

F.1.2.46 Packaged Part:* A mechanical part that has been packaged and prepared for shipping to a customer.

©ISO 2000 - All rights reserved

F.1.2.47 Part Acquisition Order: Notification that Manufacturing data needed for shop floor processing has been collected.

NOTE - This data includes customer order information, part identification, quantity, and schedule dates.

F.1.2.48 Part Administration Data: Information about the part that is used to make decisions concerning the parts status within the manufacturing process from a management prospective.

F.1.2.49 Part Data Definition: Information that completely describes a part to be manufactured.

F.1.2.50 Part Dimensions: The measured values associated with a physical part that defines the shape and location characteristics of part features. There may be two types of measurement - size dimensions and location dimensions. Size dimensions give the size of a piece, component part, hole, slot or other feature. Location dimensions fix the relationship of a component part (projections, holes, slots, and other significant forms) or a piece or structure.

F.1.2.51 Part Exception Notice:* The notification of a discrepancy in part processing on the shop floor. This can be a failure of the process plan or its execution.

F.1.2.52 Part Information Packet:* Part technical data and customer order information.

F.1.2.53 Part Model: The complete representation of a given part that captures part design aspects (including all elements that define a part and their relationships) as interpreted from the technical data packet.

F.1.2.54 Part Model Feedback: Information about the part model that is a result of its verification.

F.1.2.55 Part Pedigree:* Complete history of processing of a part information packet through the manufacturing process to the creation of the final part.

F.1.2.56 Part Pedigree Data:* Any particular piece of data that comprises the complete history of the part.

F.1.2.57 Part Shape Form Feature Representation: The set of design form feature types as determined from the part technical data to be applied to the selected basic part shape in the creation of the part model.

F.1.2.58 Part Technical Data: The physical, material, and manufacturing process information that describes a given part.

F.1.2.59 Part Tolerances: Part dimension variances used in the manufacture of a given part.

NOTE - Tolerances impact the types of manufacturing processes capable of producing particular part features. Part tolerances are added to the part form features.

F.1.2.60 Part Workstation History:* Data that represents workstation part transactions resulting from process plan execution.

NOTE - Part workstation history reports machine, tool, and fixture usage, material identification, and material certification.

 ©ISO 2000 - All rights reserved

F.1.2.61 Pedigree Creation Request: A notice that signifies that a given part has been identified for part pedigree tracking.

F.1.2.62 Process Plan:* The data that provides the available machines with the engineering specifications required to manufacture a given part.

NOTE - This data includes resources, operator data, routing, process times, bill of materials, and machine instructions.

F.1.2.63 Process Properties: Characteristics of manufacturing processes for a given part as defined in the part technical data.

F.1.2.64 Process Specifications: Document containing part process requirements that manufacturing process must follow or adhere.

F.1.2.65 Product Data Definition Completion Notice:* A notification that the digital product data definition for a given part is complete.

F.1.2.66 Product Data Definition Deficiency Notice:* A notification that the digital product data definition for a given part is incomplete.

F.1.2.67 Product Data Definition Feedback: Information about the data elements defining a part that did not pass verification analysis and requires further refinement.

F.1.2.68 Product Data Definition Status Notice:* A notification that the digital product data definition for a given part is either complete or incomplete.

F.1.2.69 Production Cost & Schedule:* All the information that describes the cost to produce a given part, and the calendar for all events associated with the manufacture of that part.

F.1.2.70 Property Requirements: Part characteristics specified in the part technical data.

F.1.2.71 Quality Reports:* A formatted compilation of inspection results, resource usage and condition results, and quality data.

F.1.2.72 Raw Stock:* The unprocessed materials necessary to manufacture a given part or set of parts.

F.1.2.73 Released Shop Work Order: The shop work order that is formally released to the shop floor when all items and process plans are available, and shop capacity permits.

F.1.2.74 Request for Technical Data:* An inquiry made to obtain part technical data to manufacture a given part.

F.1.2.75 Resources:* All items required for the manufacture of a given part.

EXAMPLE - Such items may include tools, fixtures, workstations, gauges, materials, and processes.

©ISO 2000 - All rights reserved

F.1.2.76 Resource Availability:* A report containing the status of obtainable resources at particular point in time.

F.1.2.77 Resource Usage and Condition:* The status of those resources consumed and the condition of those resources utilized during the execution of the process plan.

F.1.2.78 Routing:* A sequential list of manufacturing processes that are required to manufacture a given part.

F.1.2.79 Shop Floor Graphics:* Pictorial aids that are useful in explaining the machining processes to shop floor personnel by providing a snapshot view of a given part.

F.1.2.80 Shop Floor Operator Instructions:* Detailed textual descriptions that explain the machining processes to shop floor personnel.

F.1.2.81 Shop Work Order: Notification to the shop floor that part acquisition order data and process plan data defining the manufacturing requirements for a part is ready for processing.

F.1.2.82 Specifications: Documents containing requirements that the manufacturing process must follow or adhere.

F.1.2.83 Standards: Documents that establish engineering and technical limitations and applications for items, materials, processes, methods, designs, and engineering practices that have achieved formal consensus within a particular organizational context.

F.1.2.84 Surface Finish Properties: Part surface characteristics that define some required value of surface roughness as contained in the part technical data.

F.1.2.85 Surface Finish Specifications: Document containing requirements for the state of part surfaces that the manufacturing process must follow or adhere.

F.1.2.86 Technical Recommendation: The recommended resolution to a design problem discovered during the creation of the process plan for a part.

F.1.2.87 Test and Inspection Plan:* Instructions for any final inspection or test procedures to be executed after a part has been manufactured and before the part is shipped.

F.1.2.88 Time Standards:* Historical data for specific manufacturing resource configurations containing times spent to complete various manufacturing processes that is used for future process time estimations.

F.1.2.89 Verified Digital Product Data Definition: The set of complete data elements defining a part that is syntactically and semantically correct.

F.1.2.90 Verified Machine Code:* NC code that is validated for a particular machine and tool.

F.1.2.91 Validated Part Model: A part model that has been checked against the original part technical data to insure correctness and completeness.

 ©ISO 2000 - All rights reserved

F.1.2.92 Updated Part Acquisition Order: A revised part acquisition order that reflects additional information necessary for discrepant part processing.

F.1.2.93 Workstations:* The machine center assigned to perform a particular operation on a part.

F.1.2.94 Workstation Control Commands:* All information describing operation, resource, process, and transportation requirements and instructions for a production operation to be performed on a part.

F.1.2.95 Work Load Report:* Information that identifies the utilization of shop floor resources.

F.2 Application activity model diagrams

The application activity model is given in figures F.2 through F.14. Activities and data flows tat are out of scope are marked with asterisks.

©ISO 2000 - All rights reserved

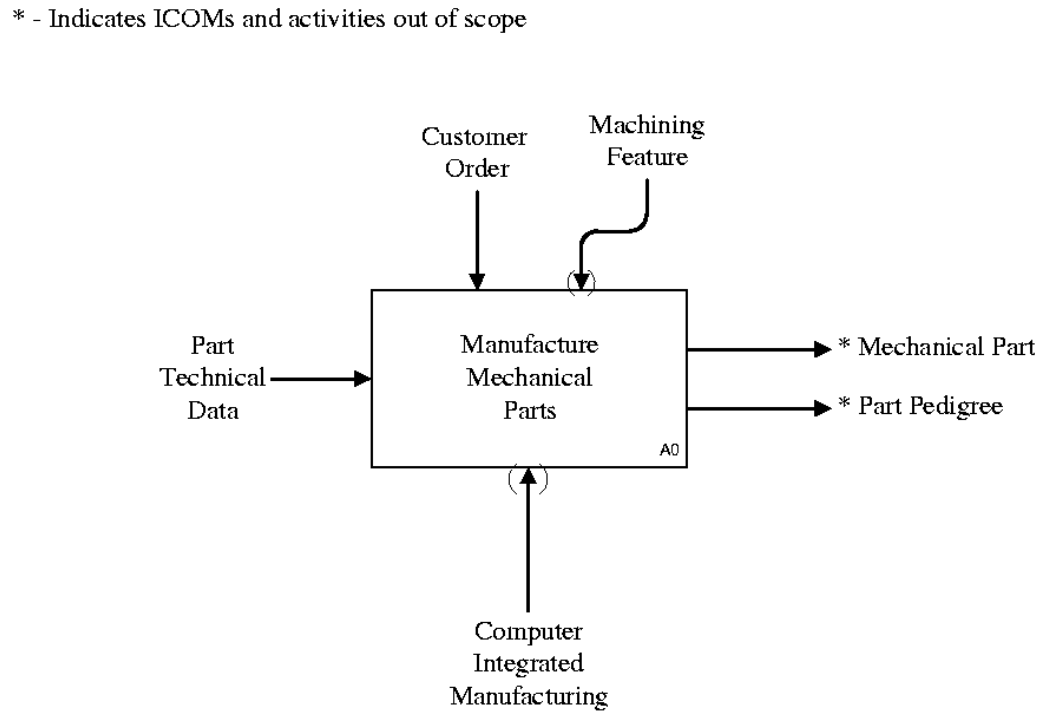


Figure F.2 - Mechanical products definition for process planning using machining features

©ISO 2000 - All rights reserved

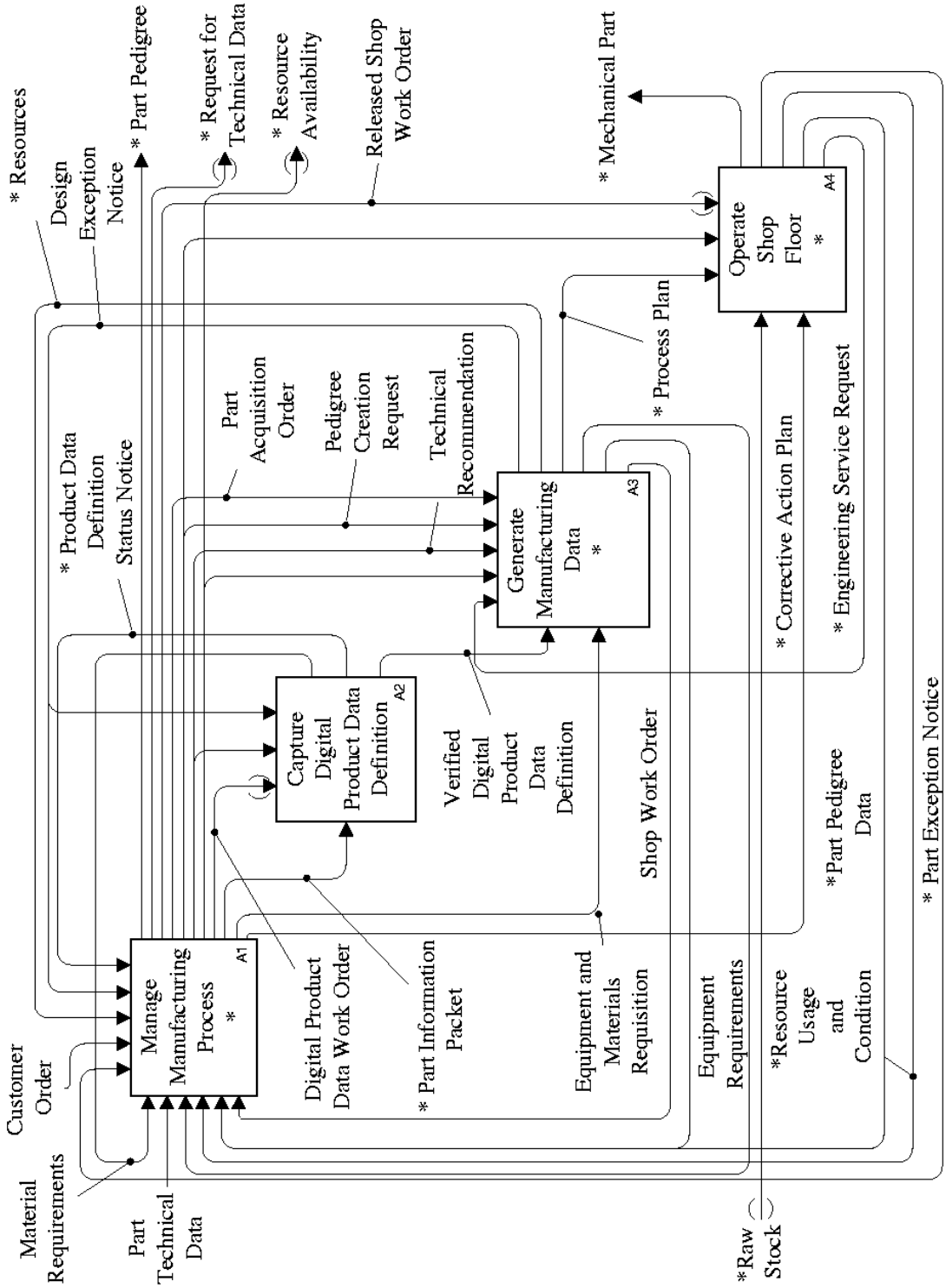


Figure F.3 - A0 manufacture mechanical parts

©ISO 2000 - All rights reserved

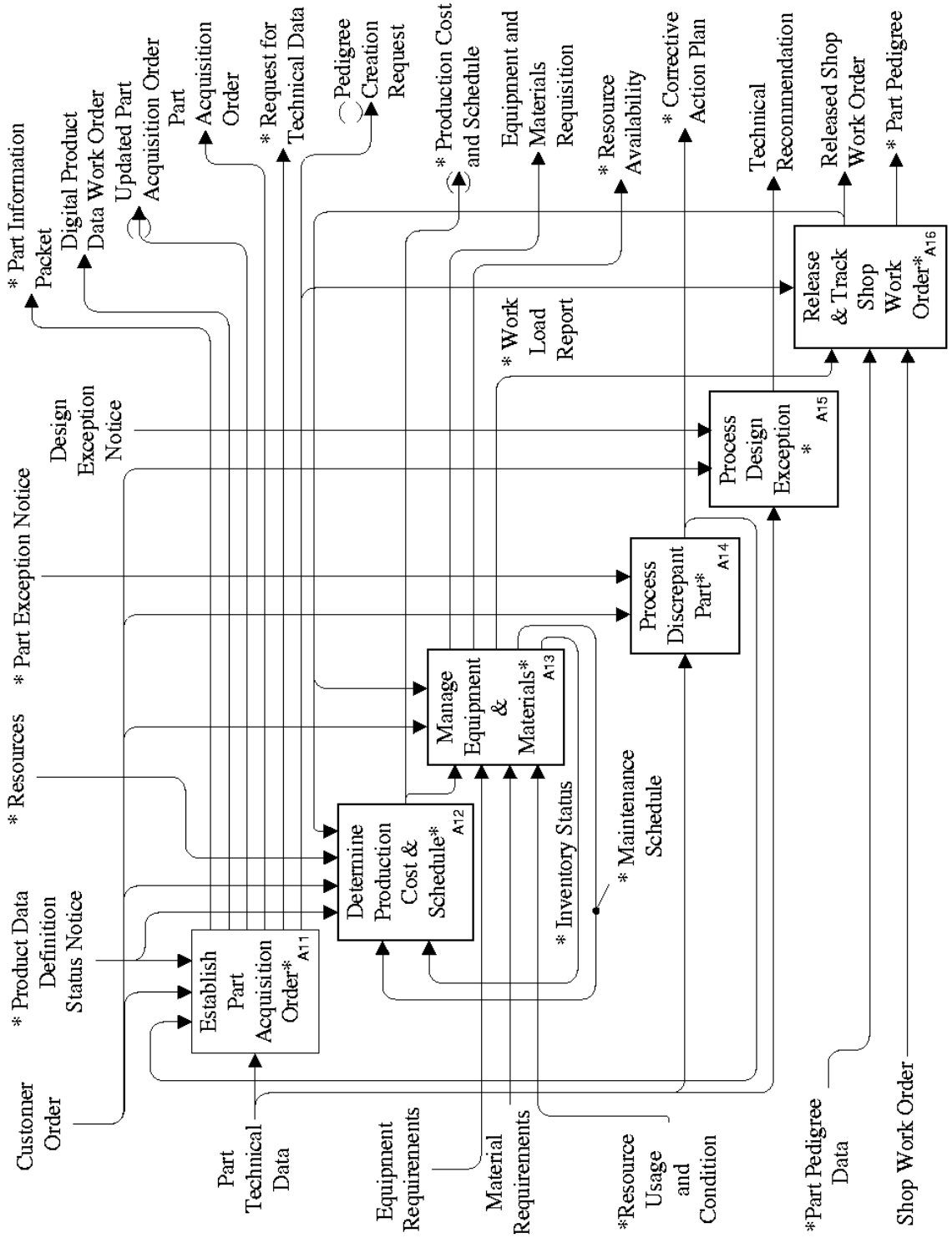


Figure F.4 - A1 manage manufacturing process

©ISO 2000 - All rights reserved

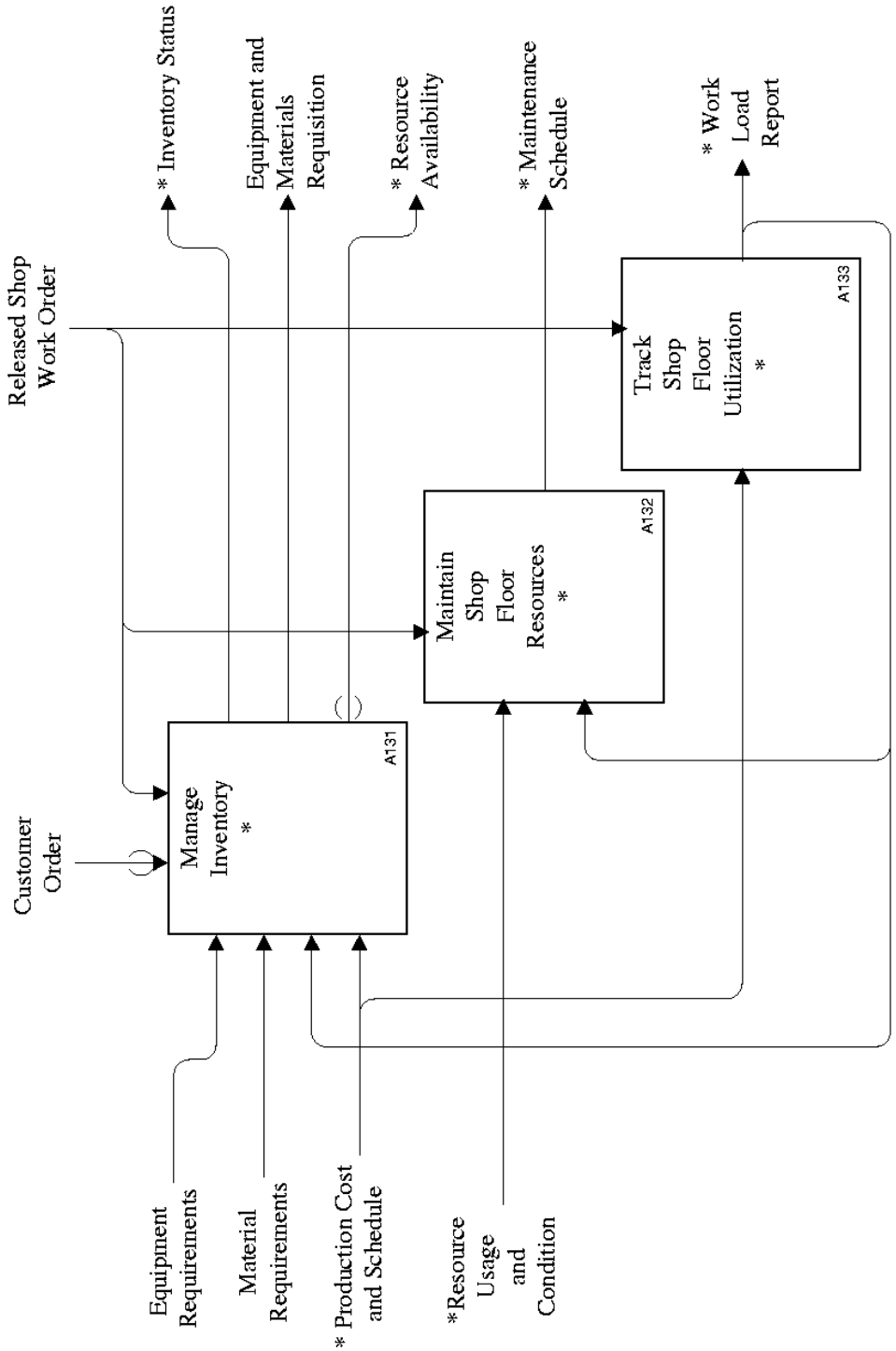


Figure F.5 - A13 manage equipment and materials

©ISO 2000 - All rights reserved

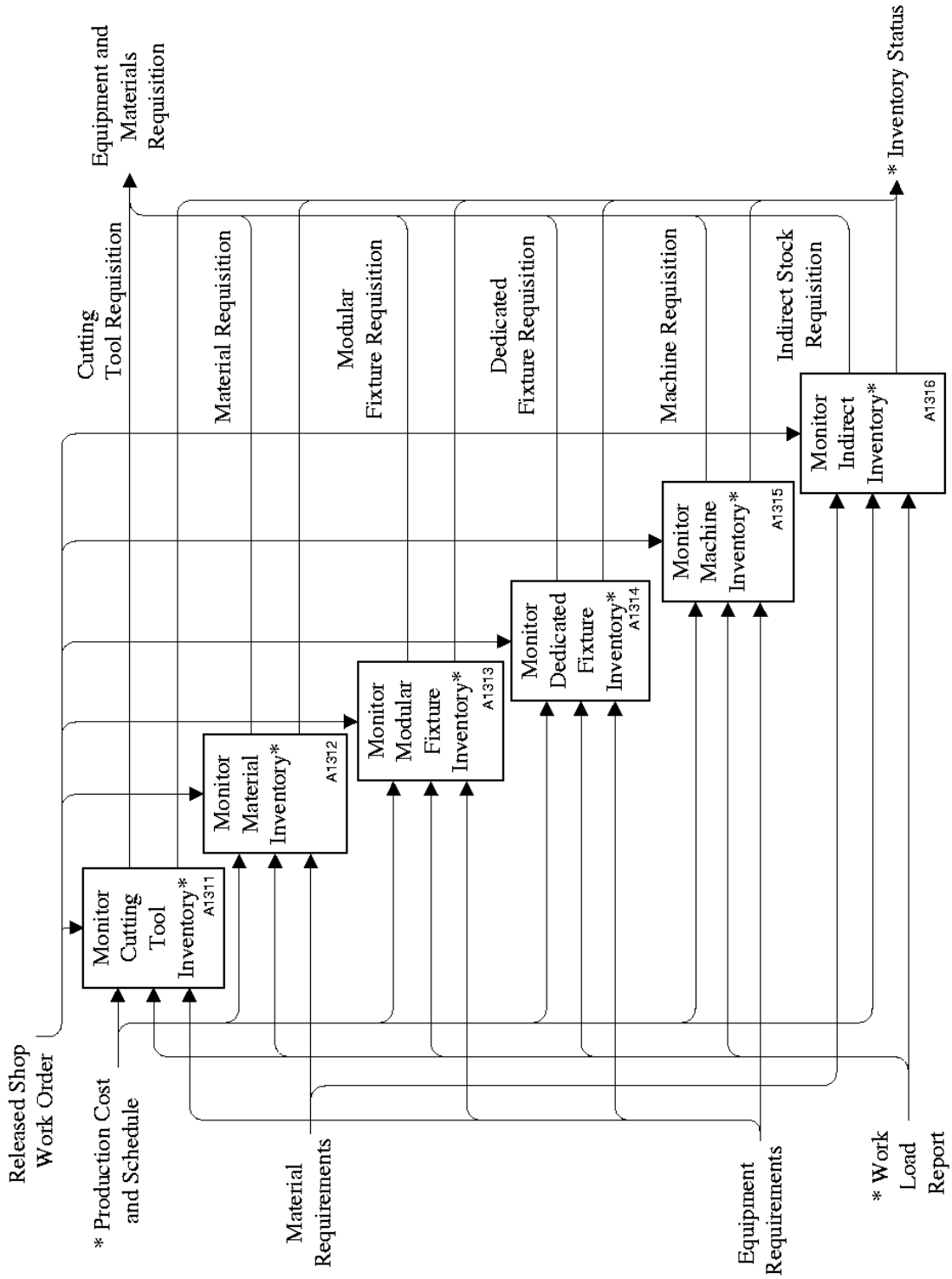


Figure F.6 - A131 manage inventory

 ©ISO 2000 - All rights reserved

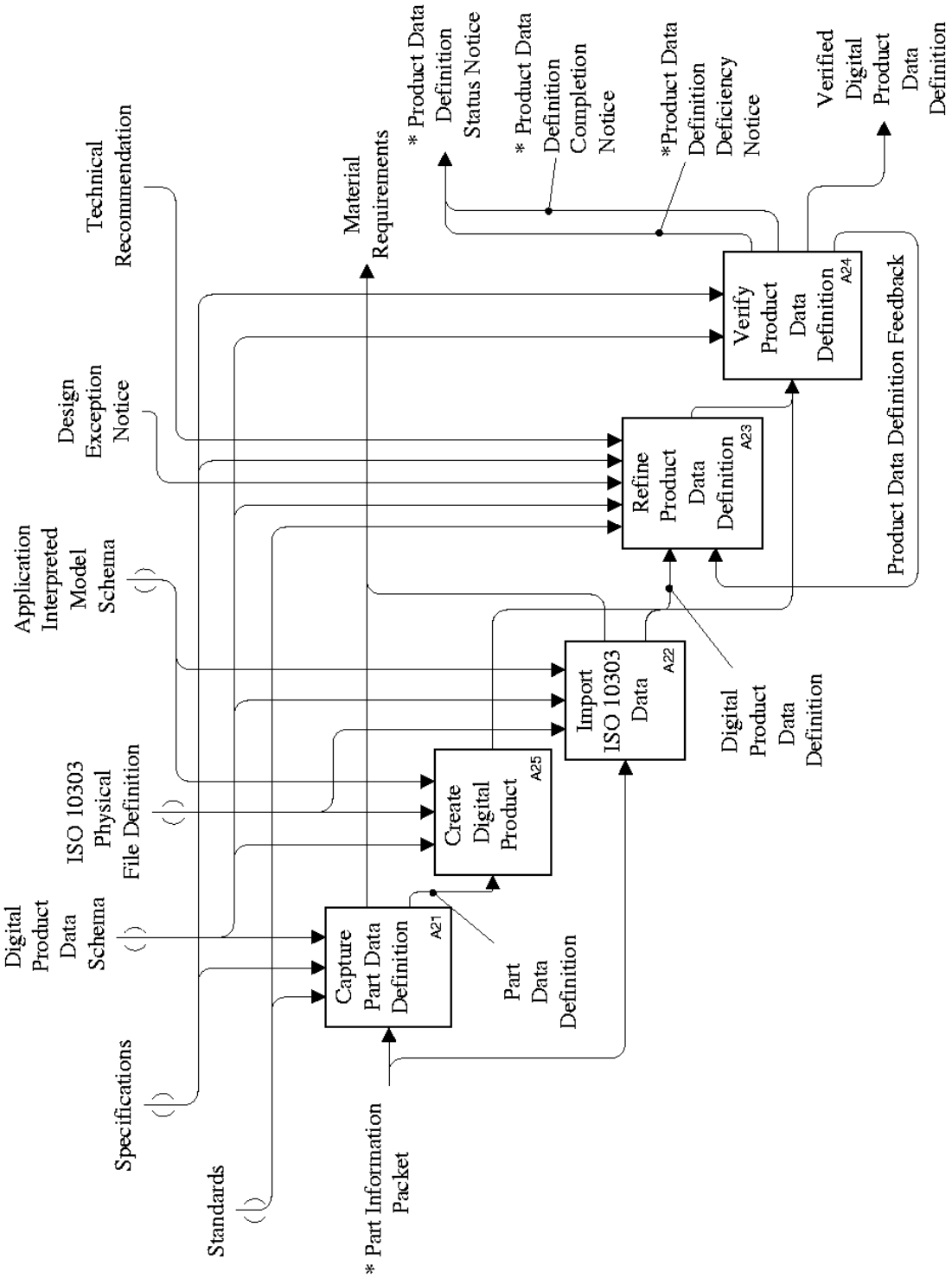


Figure F.7 - A2 capture digital product definition

©ISO 2000 - All rights reserved

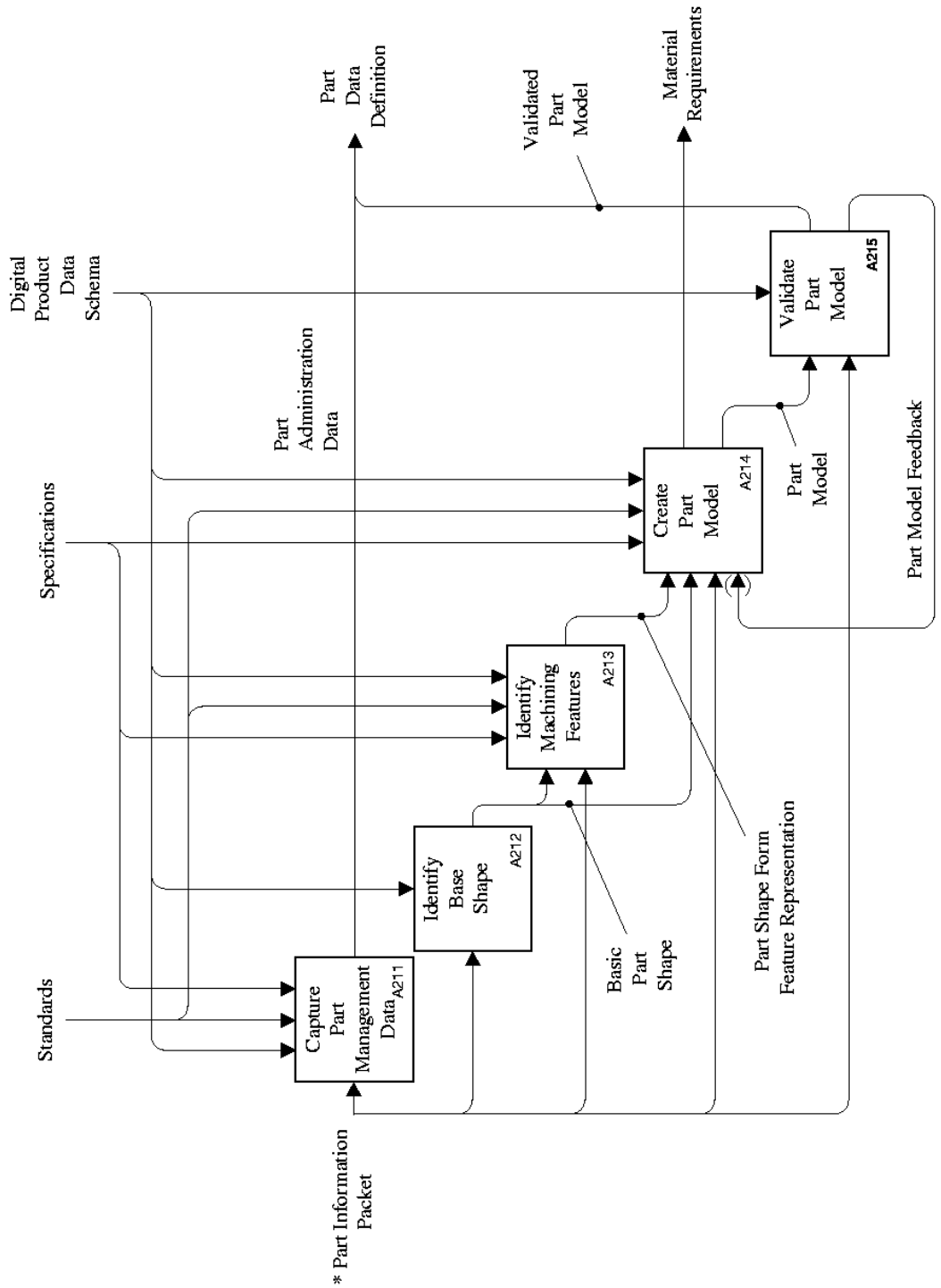


Figure F.8 - A21 capture part data definition

 ©ISO 2000 - All rights reserved

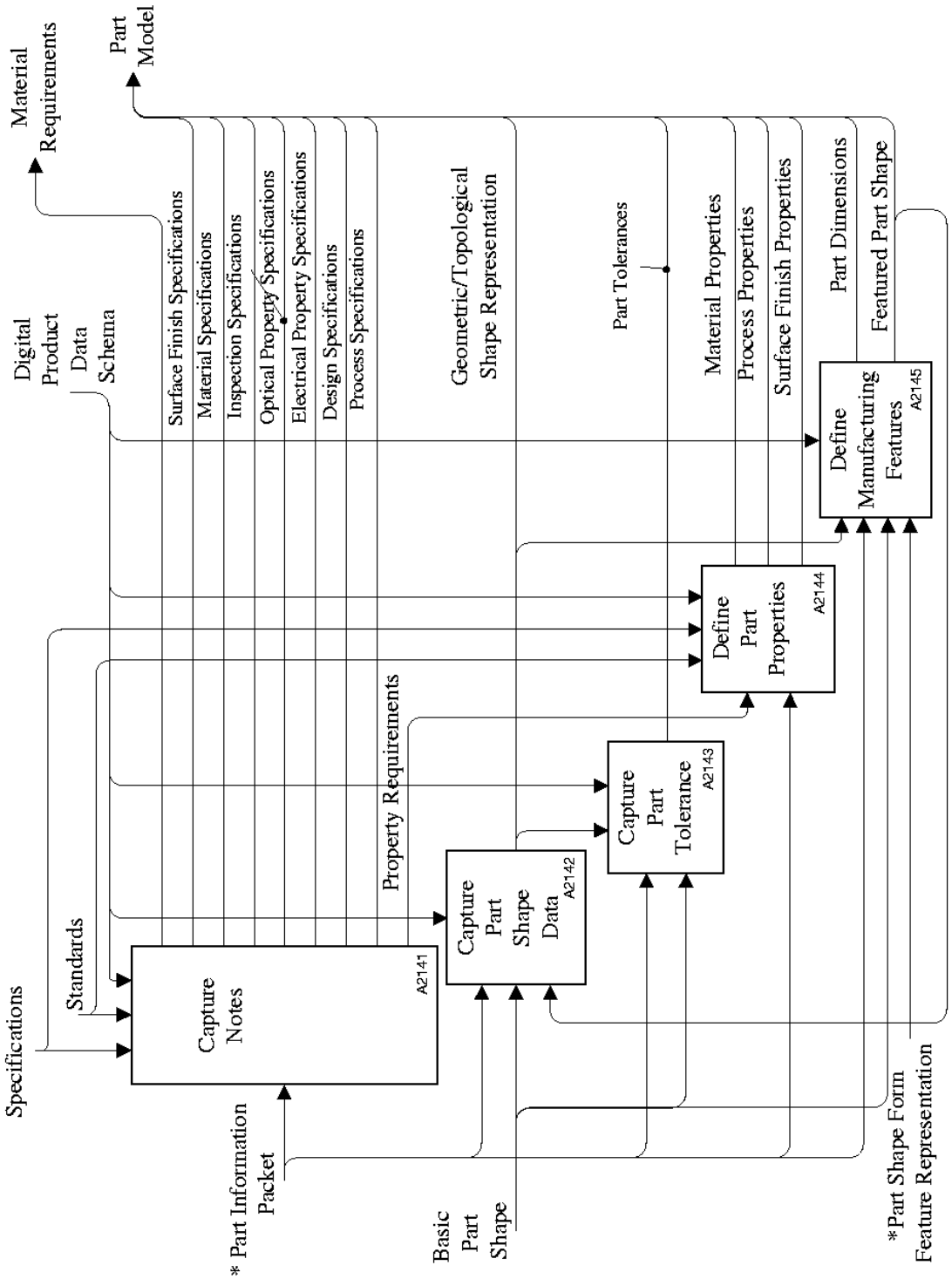


Figure F.9 - A214 create part model

©ISO 2000 - All rights reserved

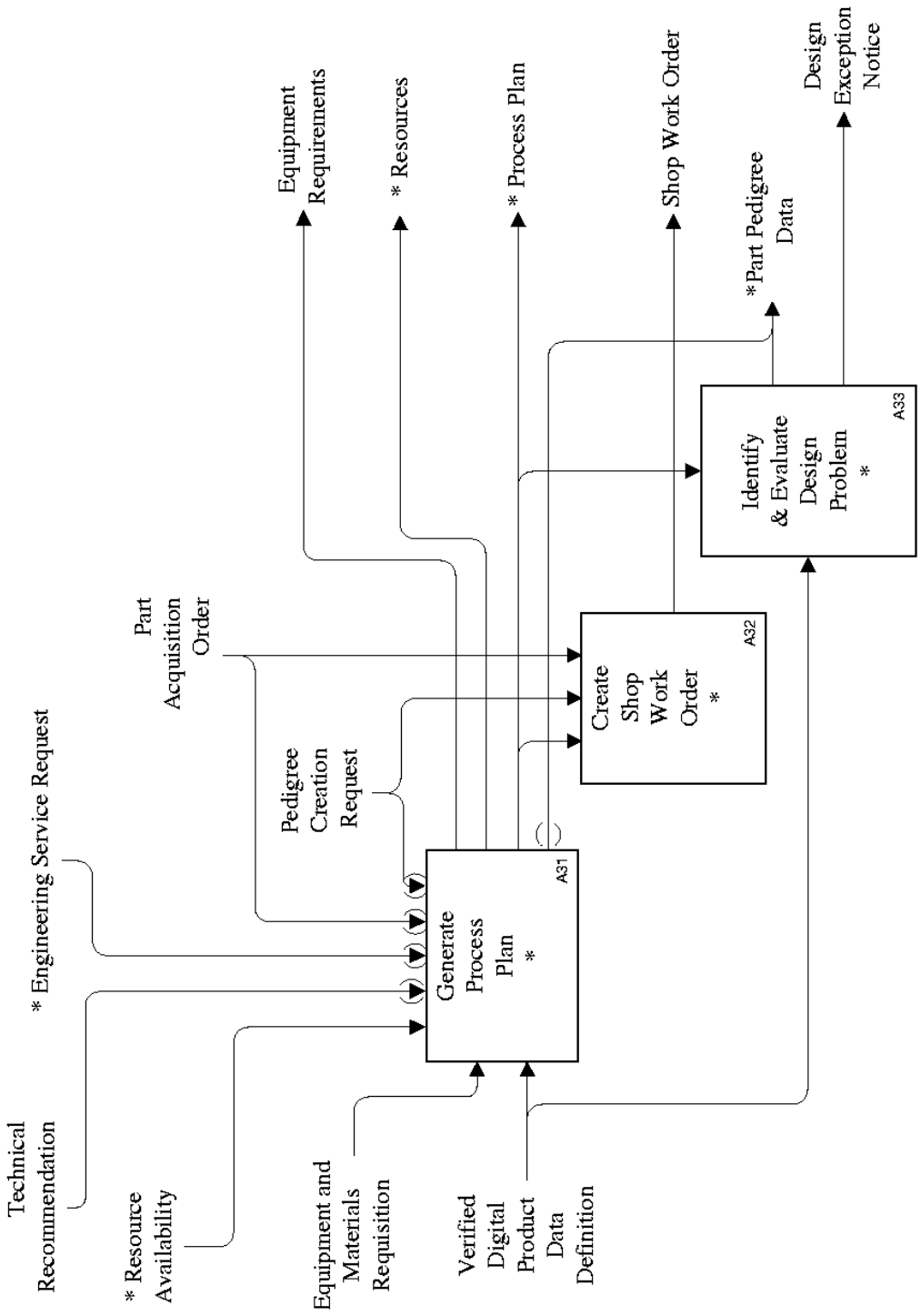


Figure F.10 - A3 generate manufacturing data

 ©ISO 2000 - All rights reserved

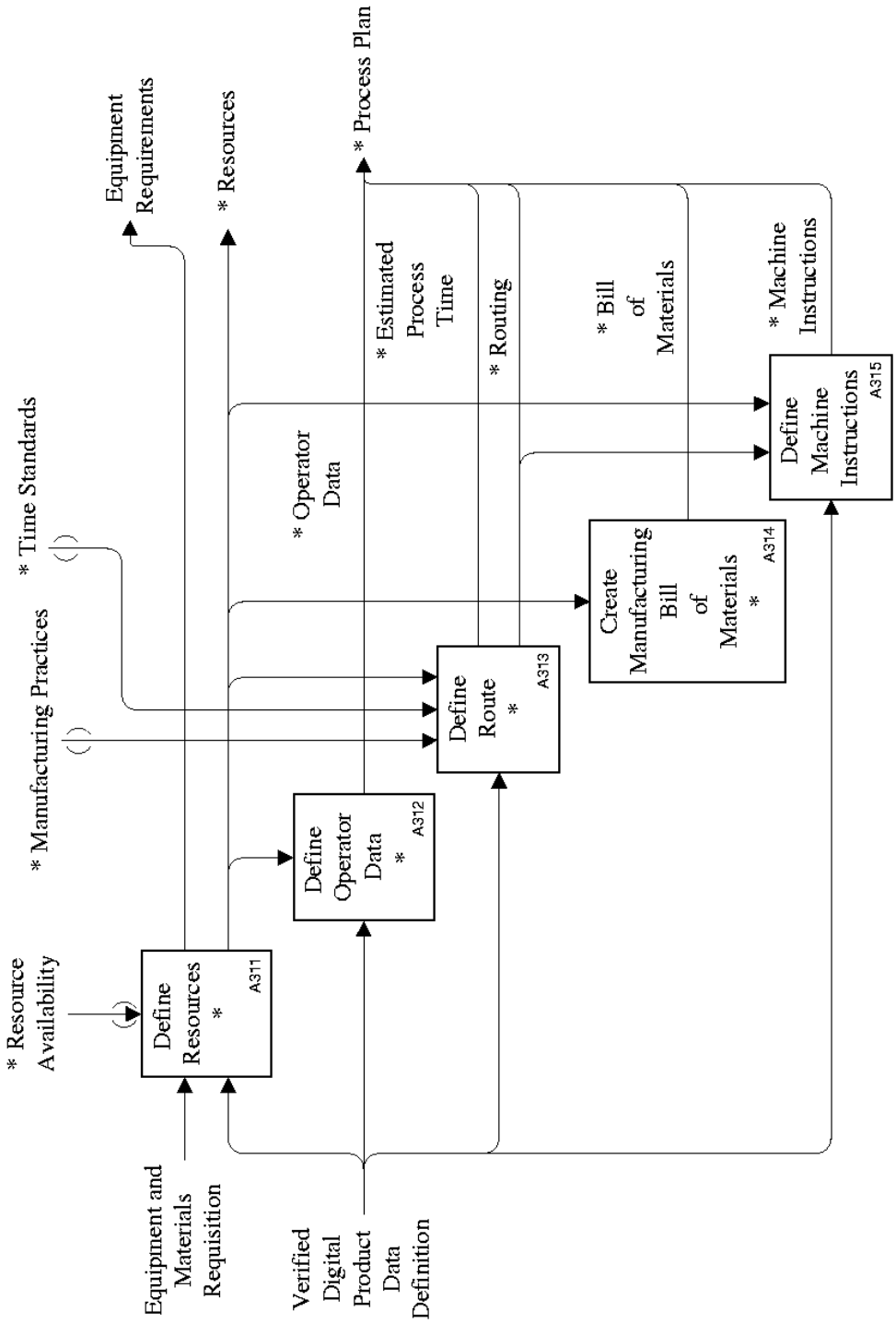


Figure F.11 - A31 generate process plan

©ISO 2000 - All rights reserved

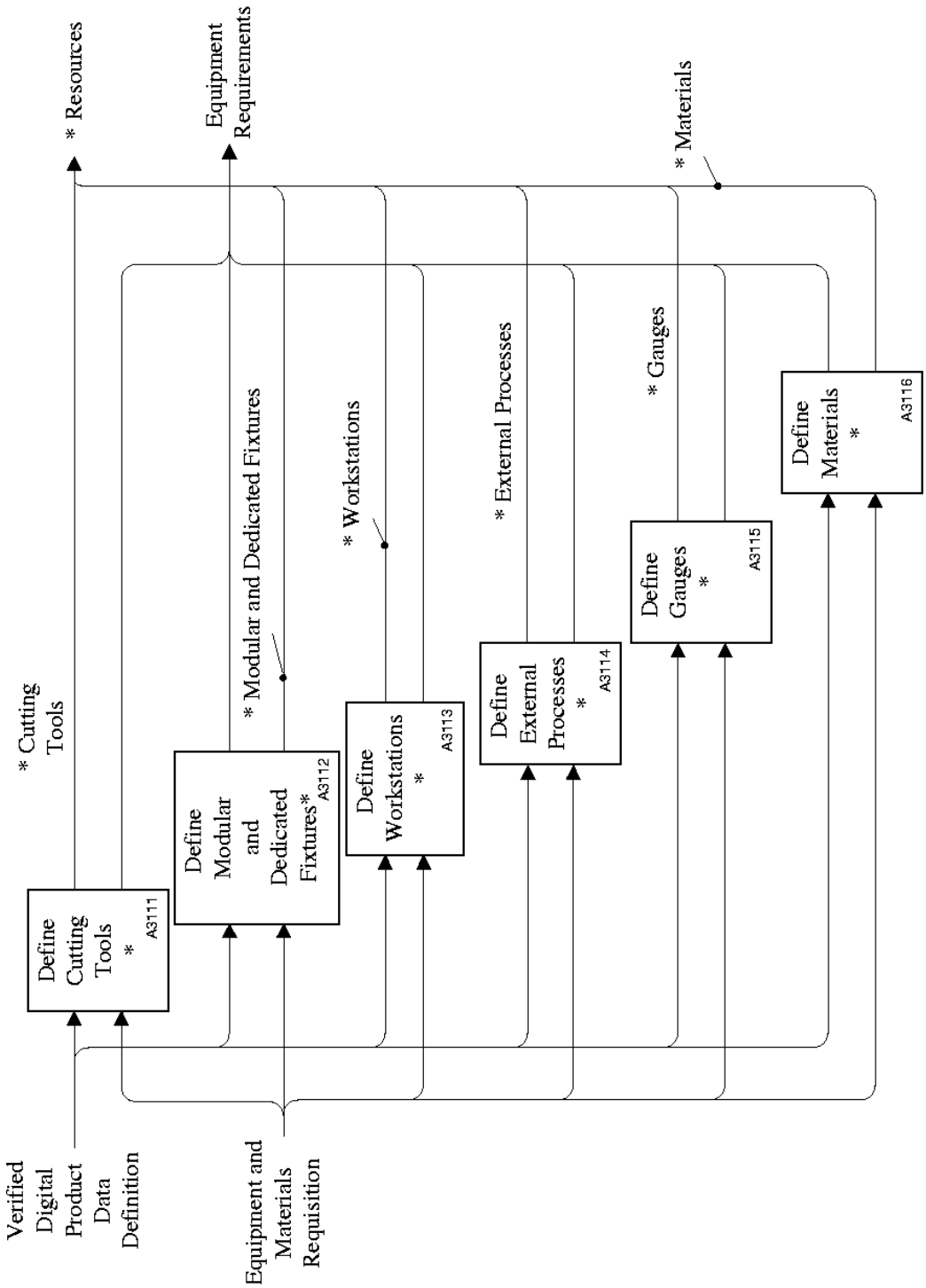


Figure F.12 - A311 define resources

 ©ISO 2000 - All rights reserved

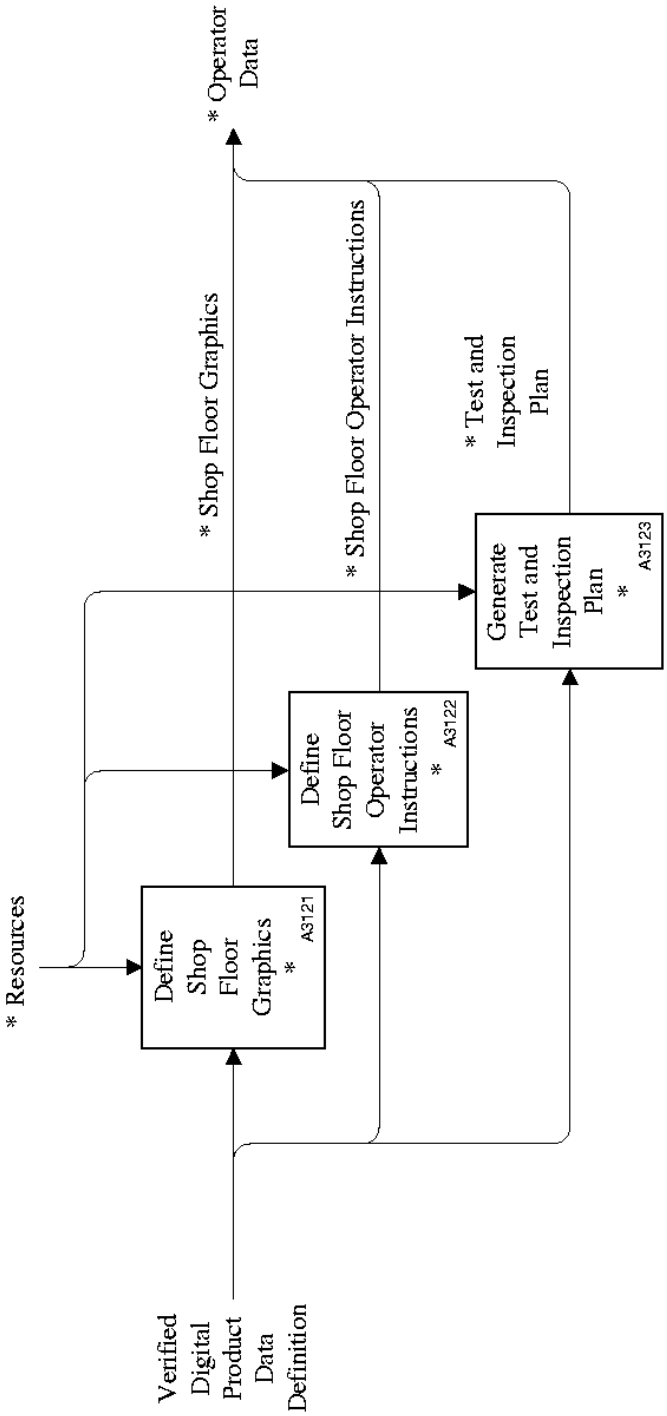


Figure F.13 - A312 define operator data

©ISO 2000 - All rights reserved

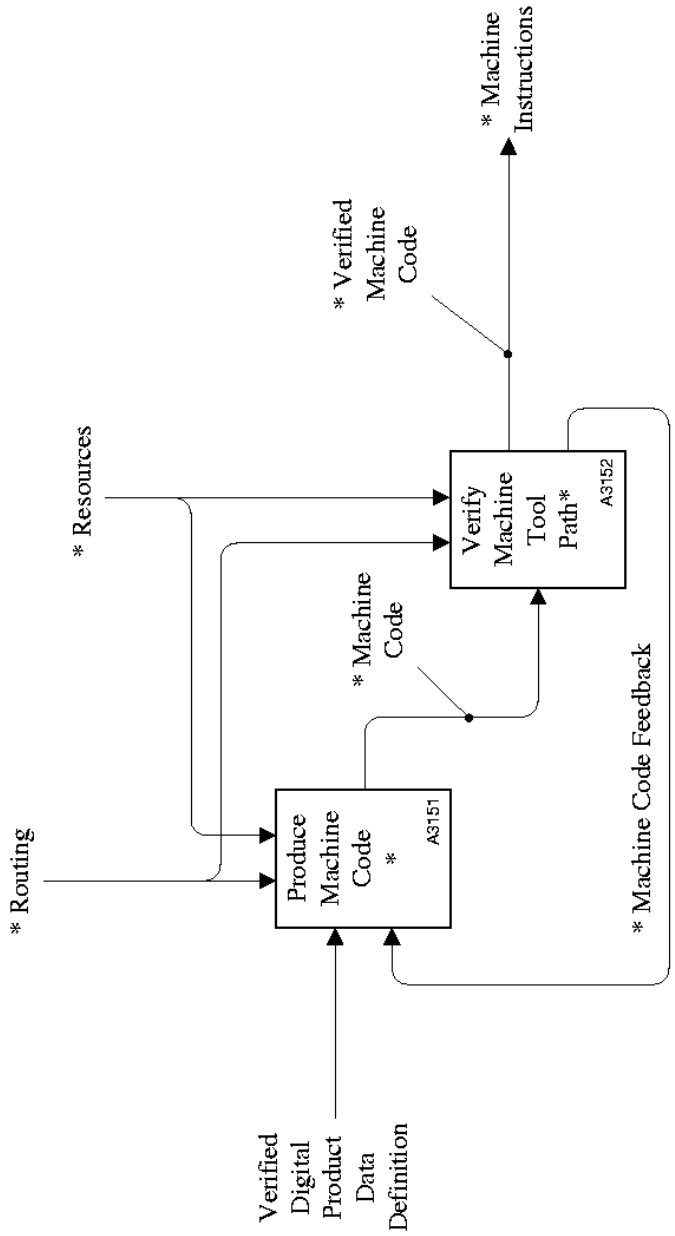


Figure F.14 - A315 define machine instructions

©ISO 2000 - All rights reserved

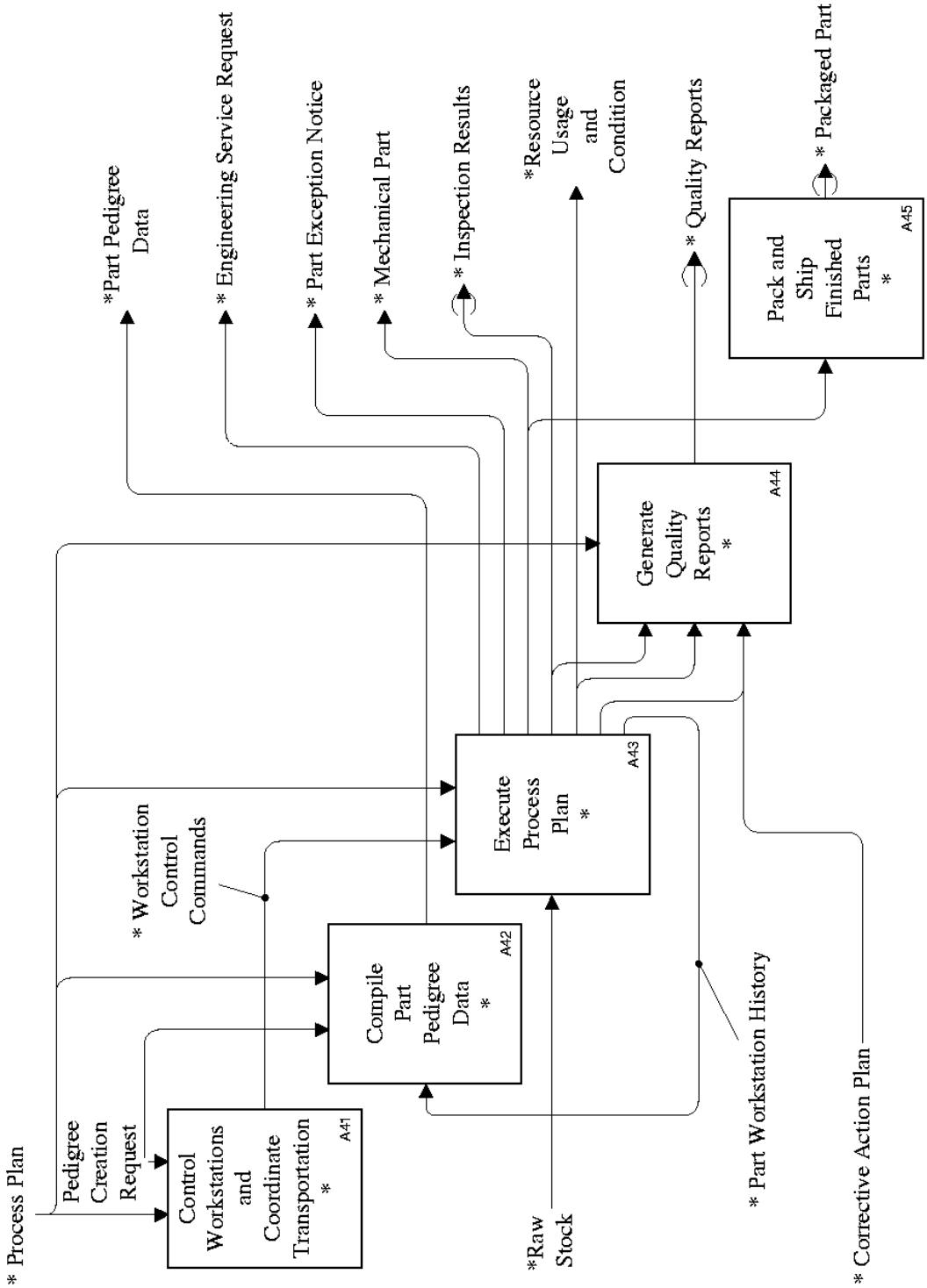


Figure F.15 - A4 operate shop floor

©ISO 2000 - All rights reserved

Annex G
(informative)

Application reference model

This annex provides the application reference model for this part of ISO 10303 and is given in figures G.1 through G.28. The application reference model id a graphical representation of the structure and constraints of the application objects specified in clause 4. The graphical form of the application reference model is presented in the EXPRESS-G. The application reference model is independent from any implementation method.

 ©ISO 2000 - All rights reserved

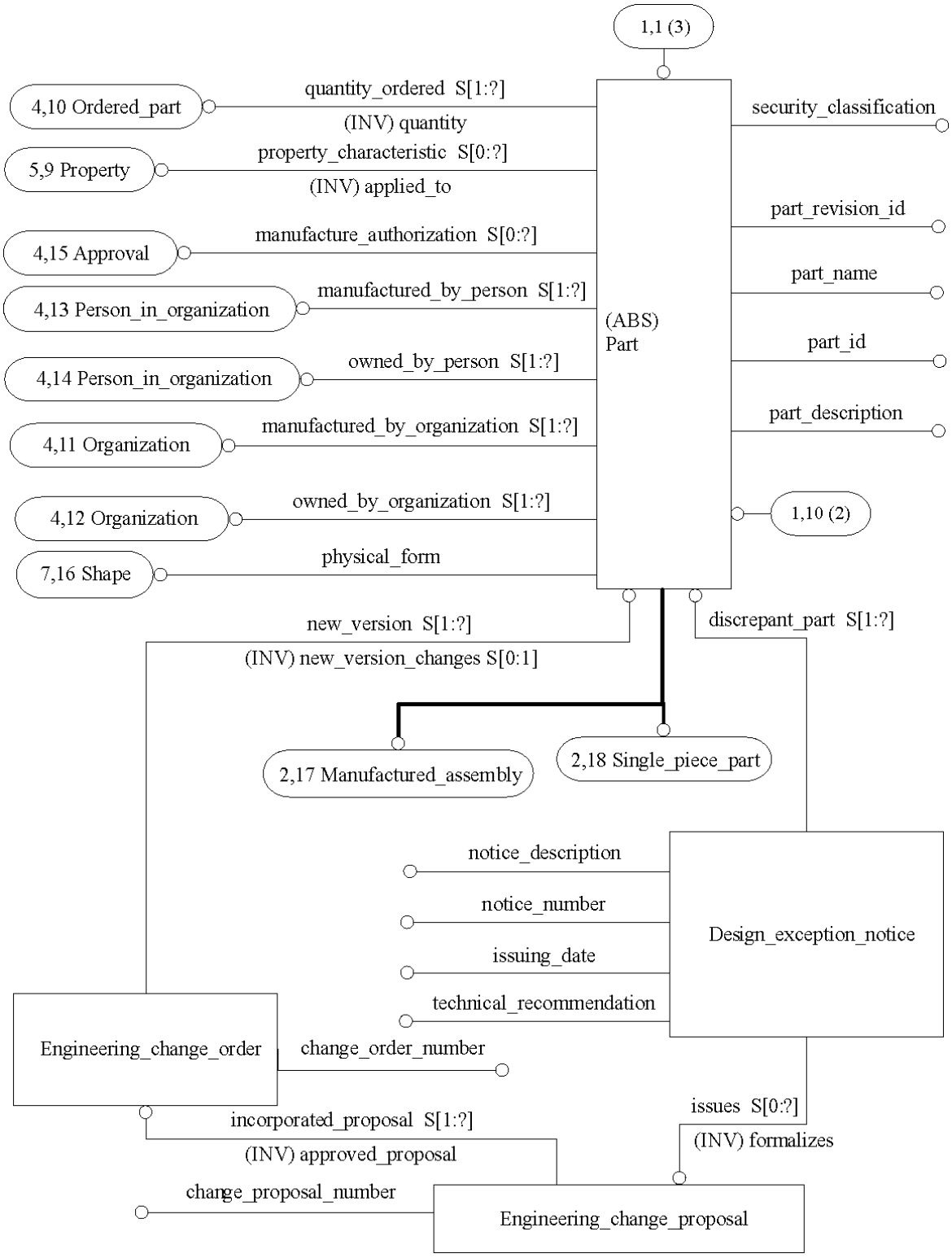


Figure G.1 - ARM EXPRESS-G diagram 1 of 28

©ISO 2000 - All rights reserved

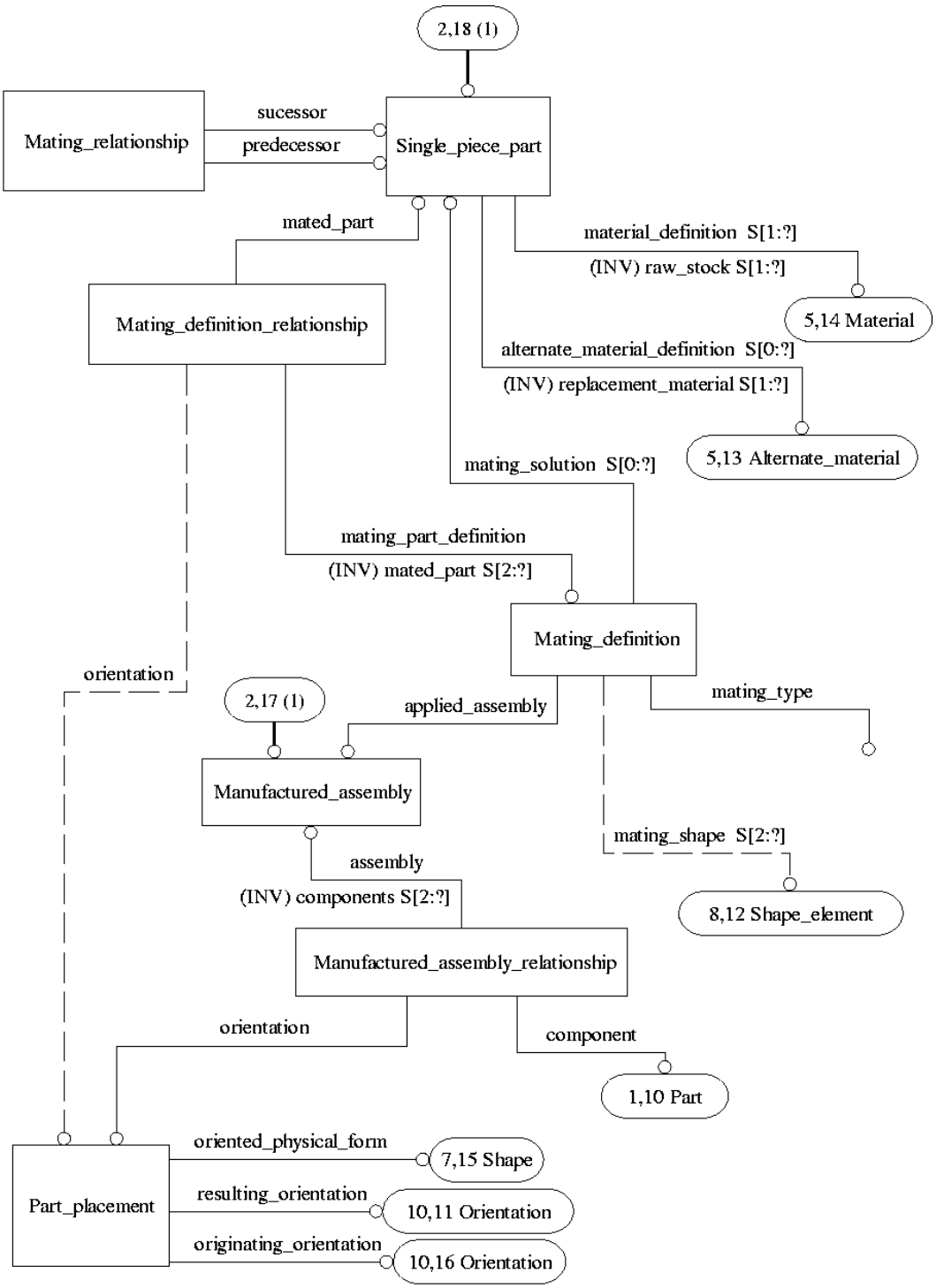


Figure G.2 - ARM EXPRESS-G diagram 2 of 28

 ©ISO 2000 - All rights reserved

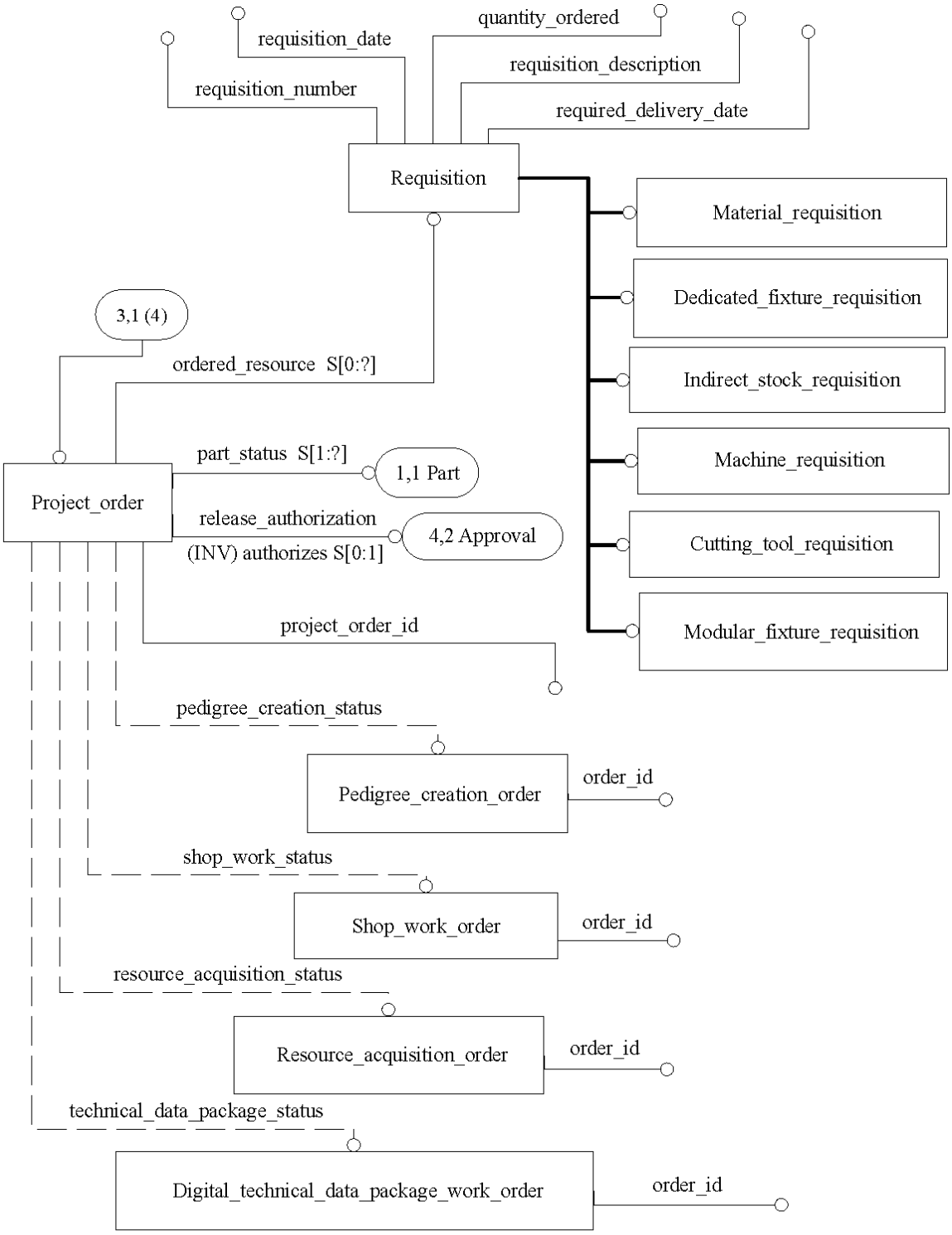


Figure G.3 - ARM EXPRESS-G diagram 3 of 28

©ISO 2000 - All rights reserved

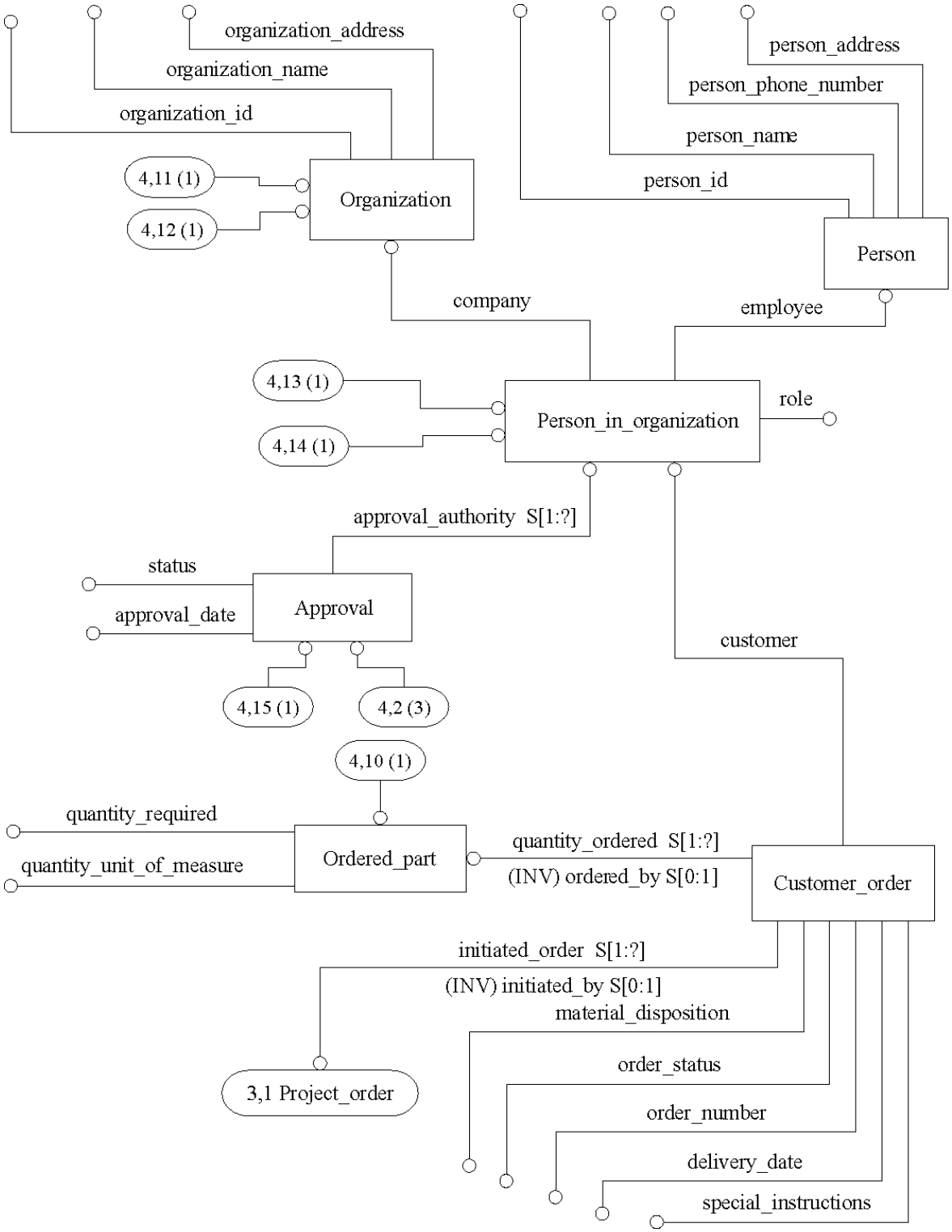


Figure G.4 - ARM EXPRESS-G diagram 4 of 28

 ©ISO 2000 - All rights reserved

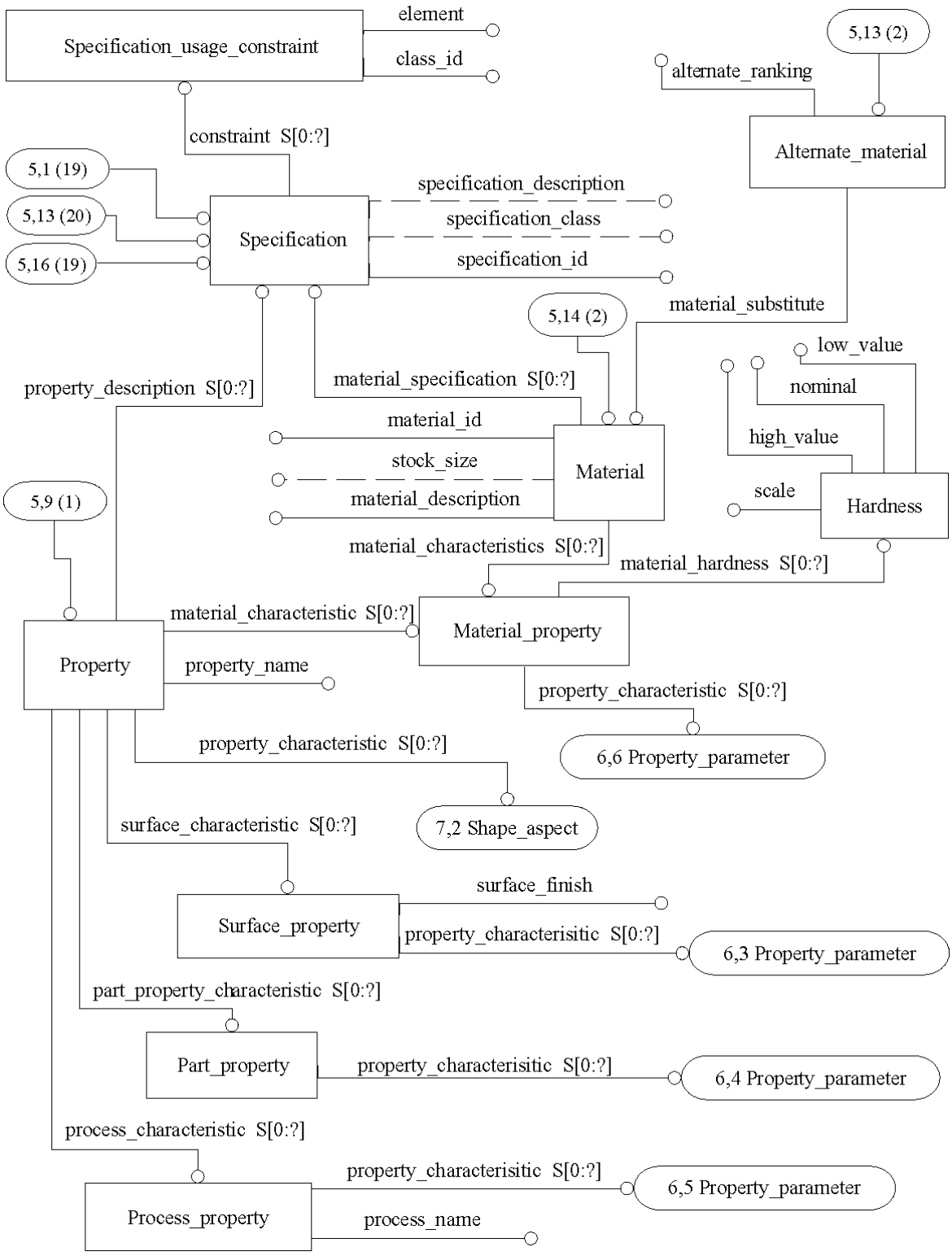


Figure G.5 - ARM EXPRESS-G diagram 5 of 28

©ISO 2000 - All rights reserved

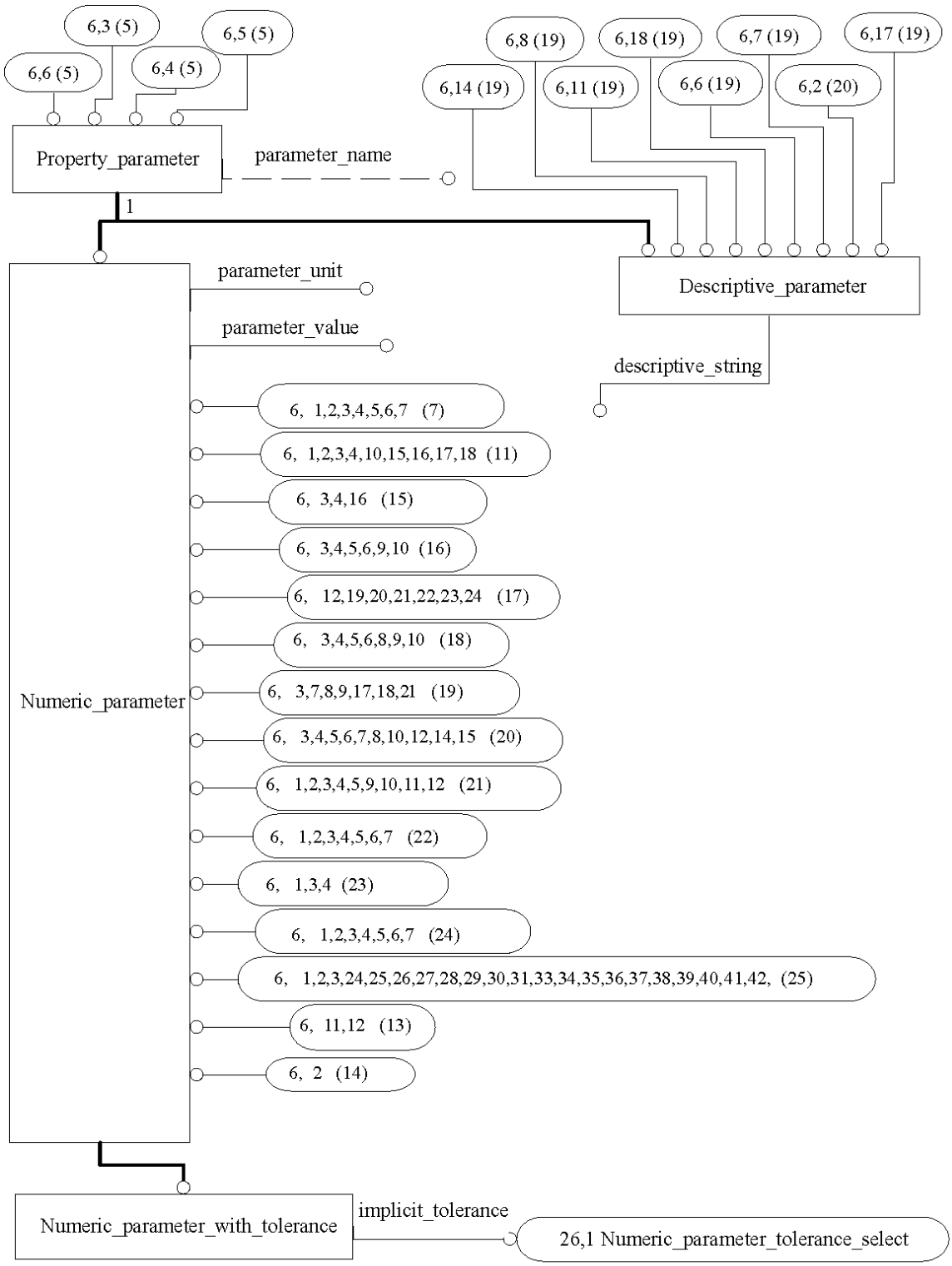


Figure G.6 - ARM EXPRESS-G diagram 6 of 28

 ©ISO 2000 - All rights reserved

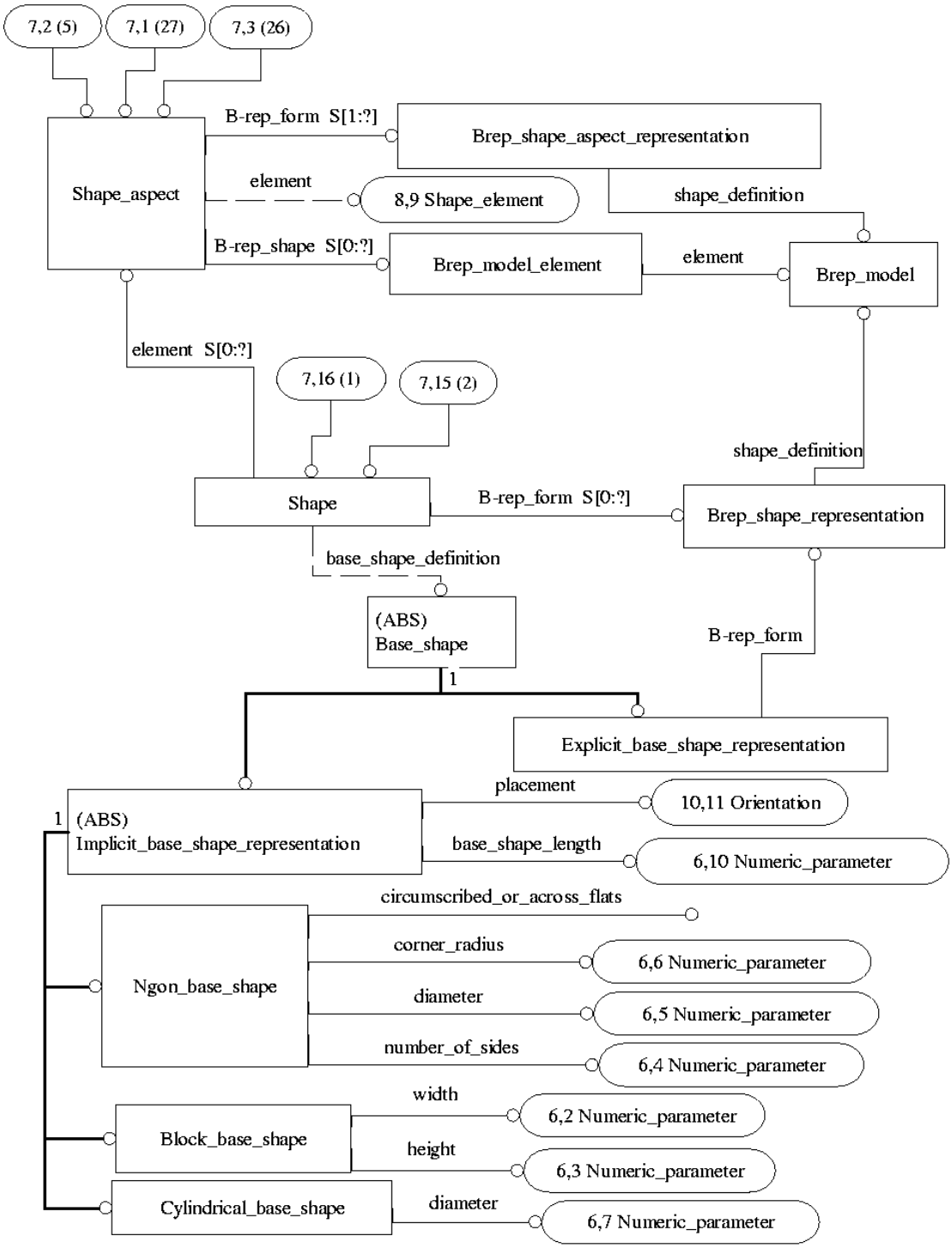


Figure G.7 - ARM EXPRESS-G diagram 7 of 28

©ISO 2000 - All rights reserved

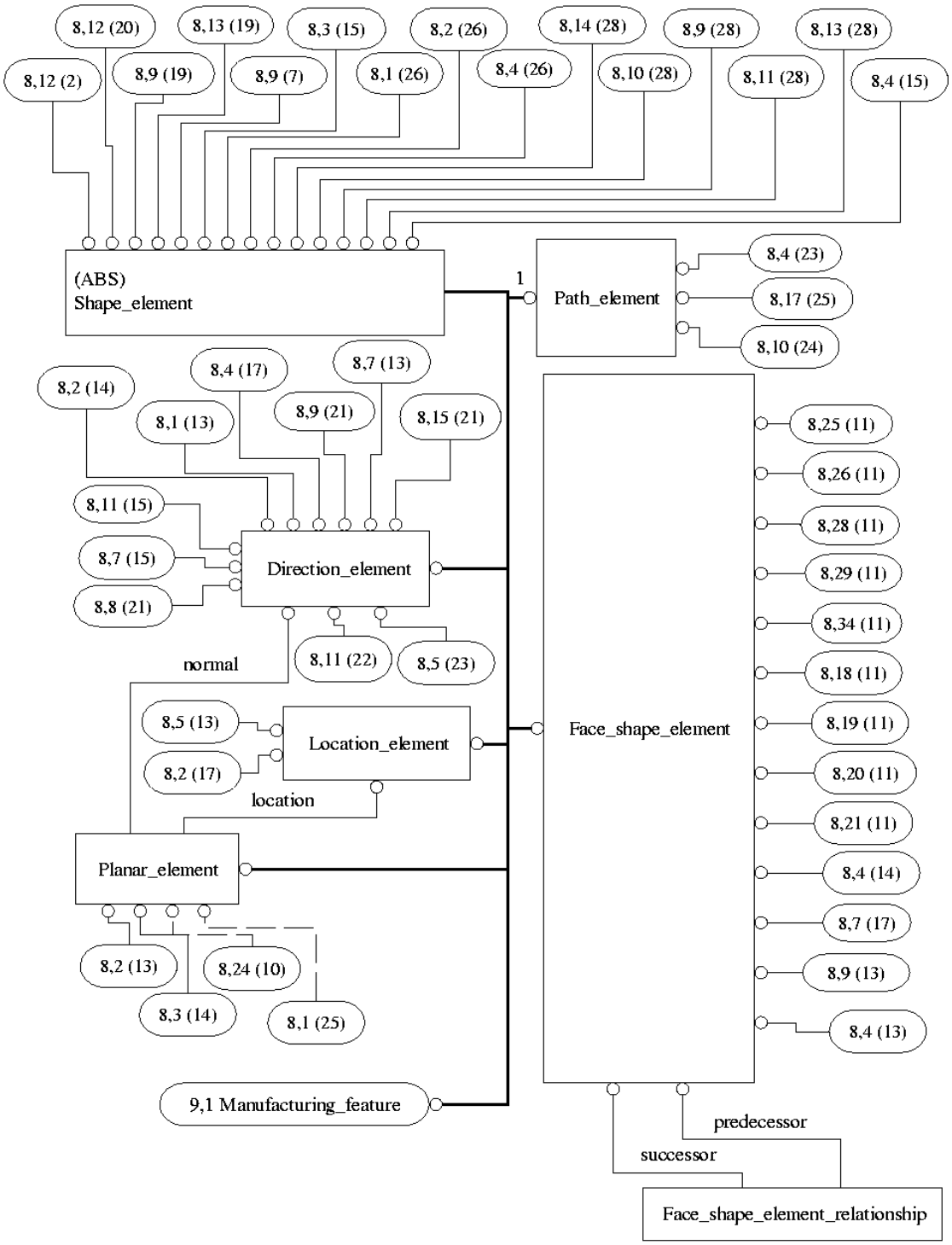


Figure G.8 - ARM EXPRESS-G diagram 8 of 28

 ©ISO 2000 - All rights reserved

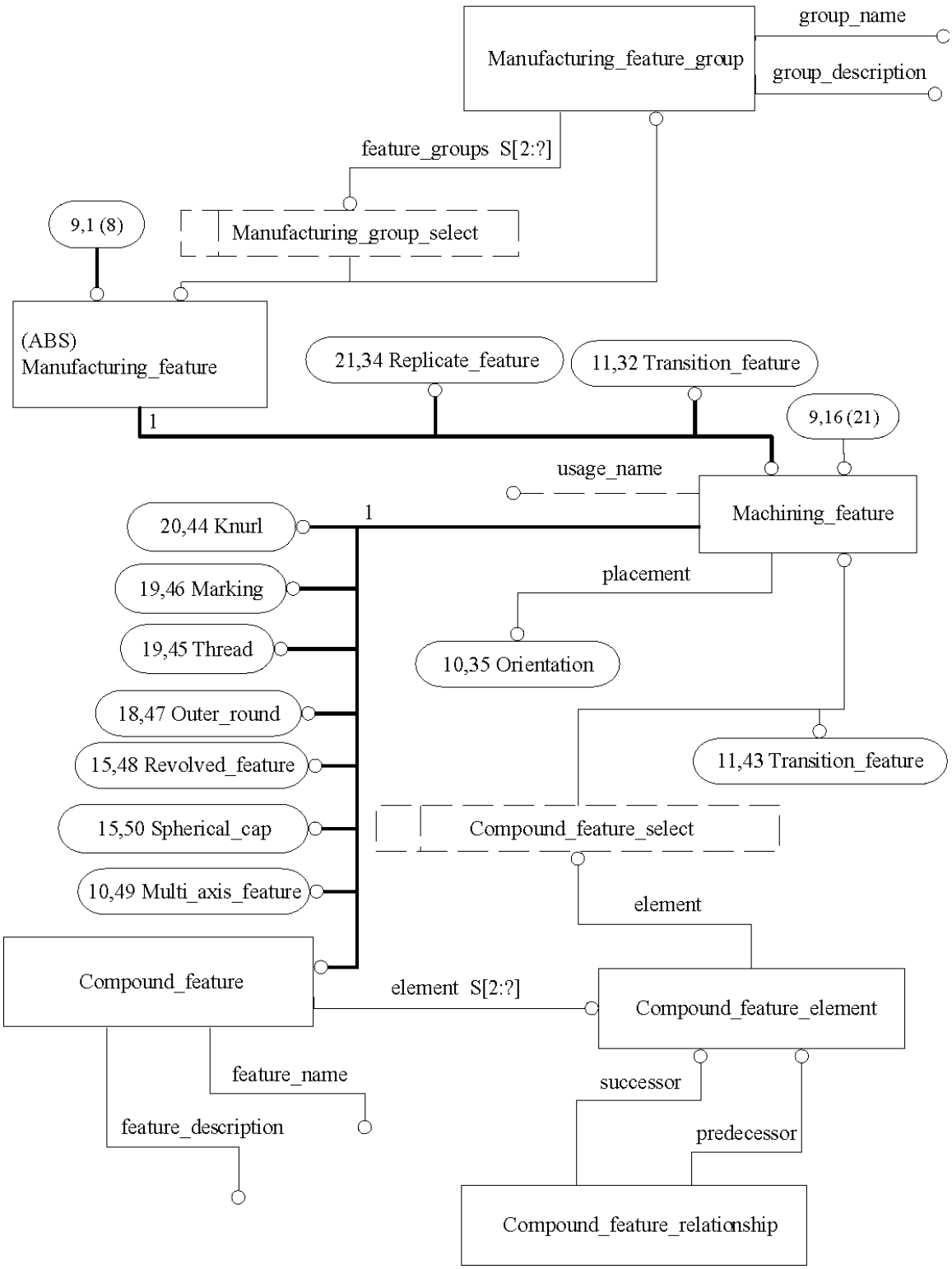


Figure G.9 - ARM EXPRESS-G diagram 9 of 28

©ISO 2000 - All rights reserved

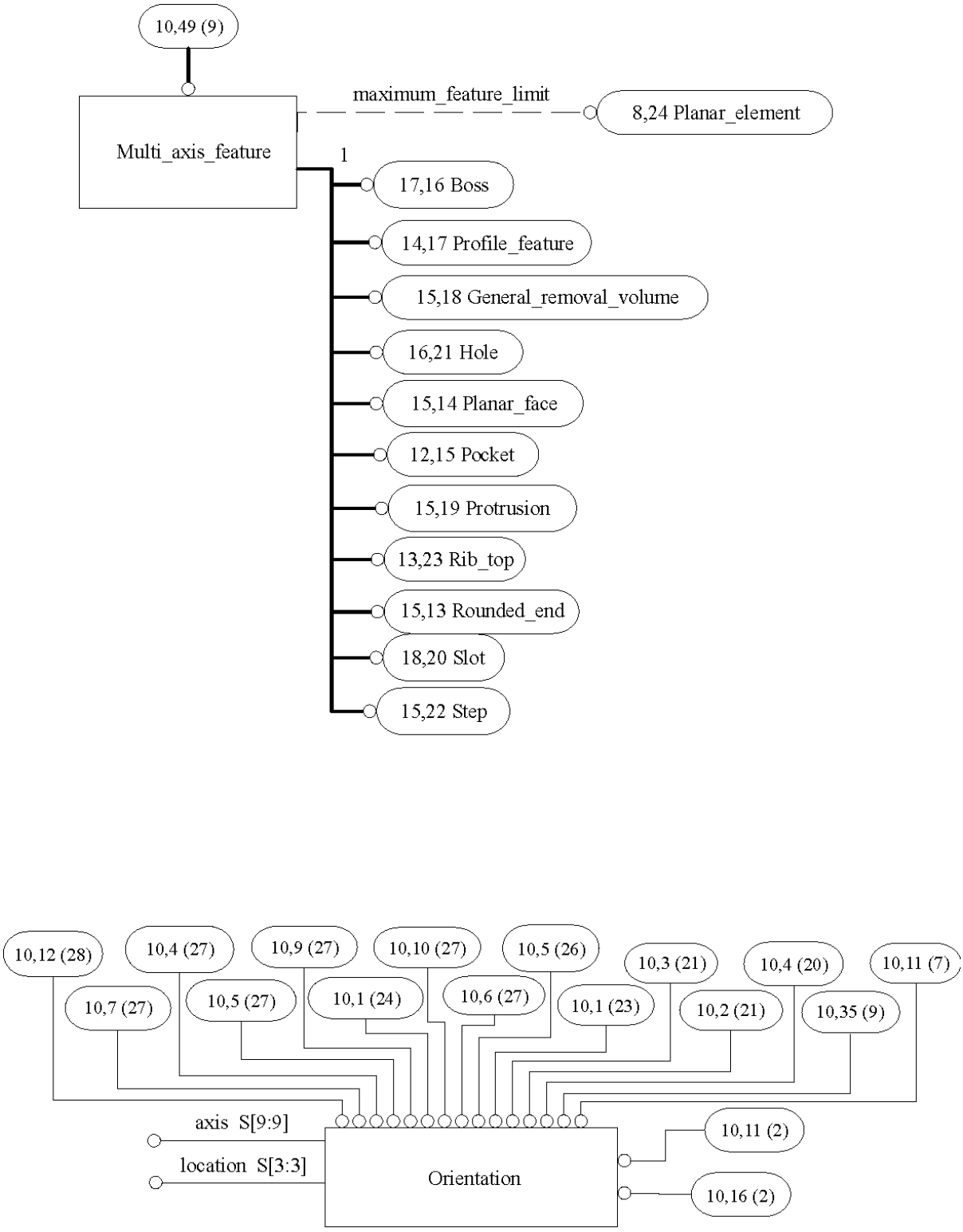


Figure G.10 - ARM EXPRESS-G diagram 10 of 28

 ©ISO 2000 - All rights reserved

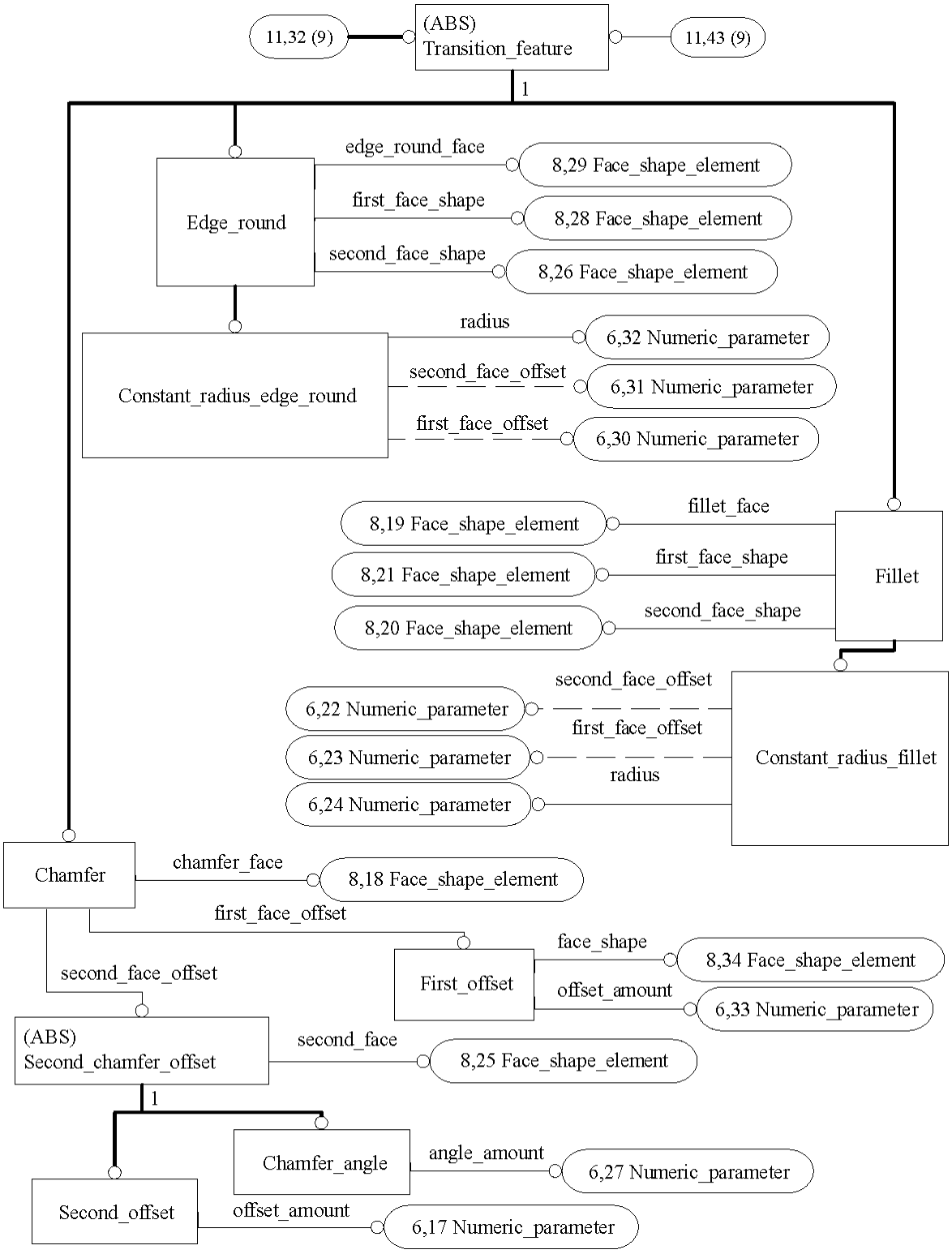


Figure G.11 - ARM EXPRESS-G diagram 11 of 28

©ISO 2000 - All rights reserved

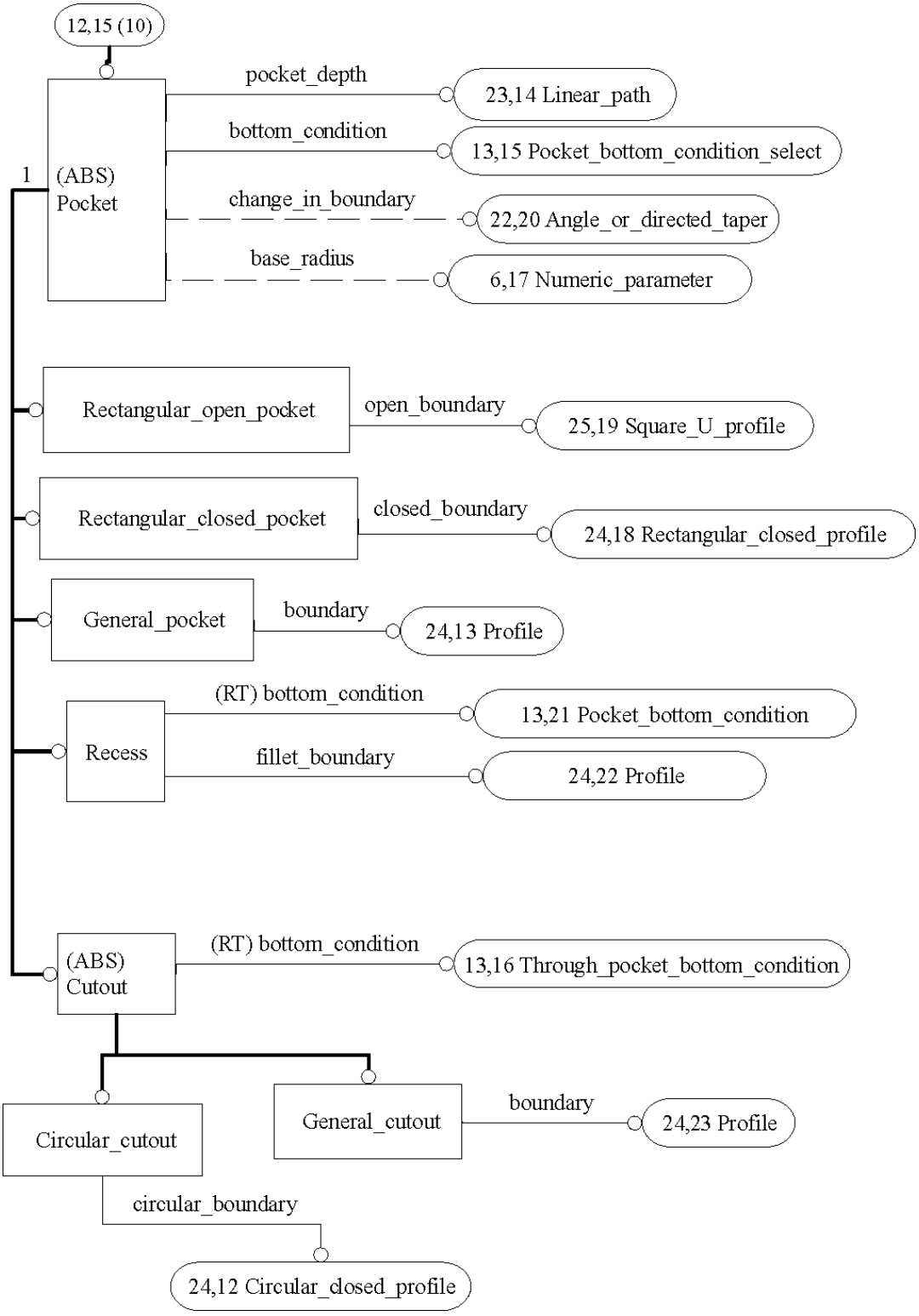
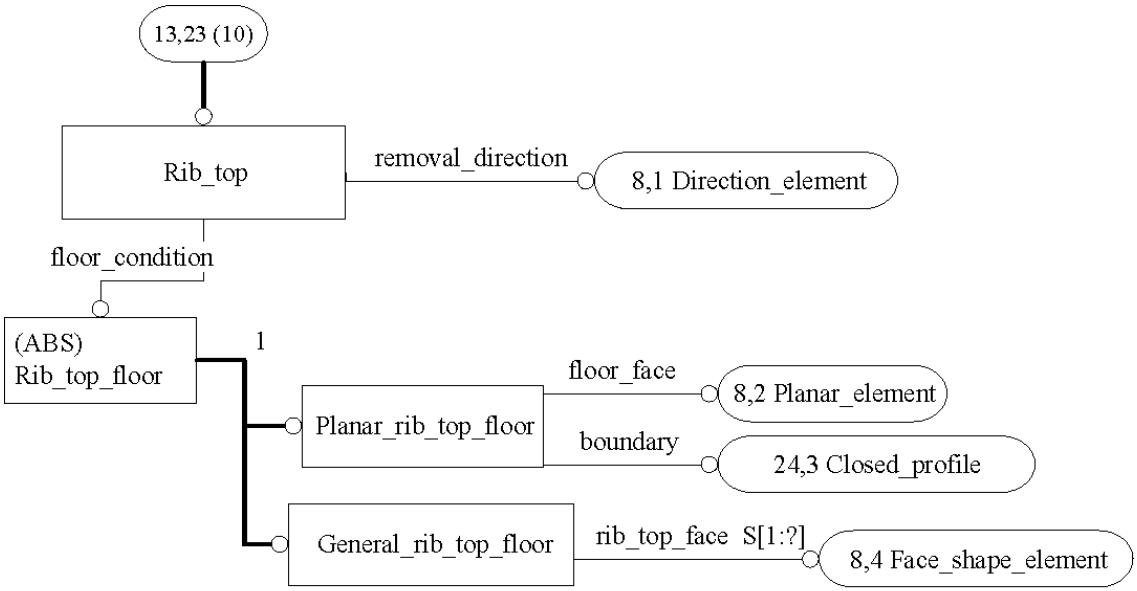


Figure G.12 - ARM EXPRESS-G diagram 12 of 28

 ©ISO 2000 - All rights reserved



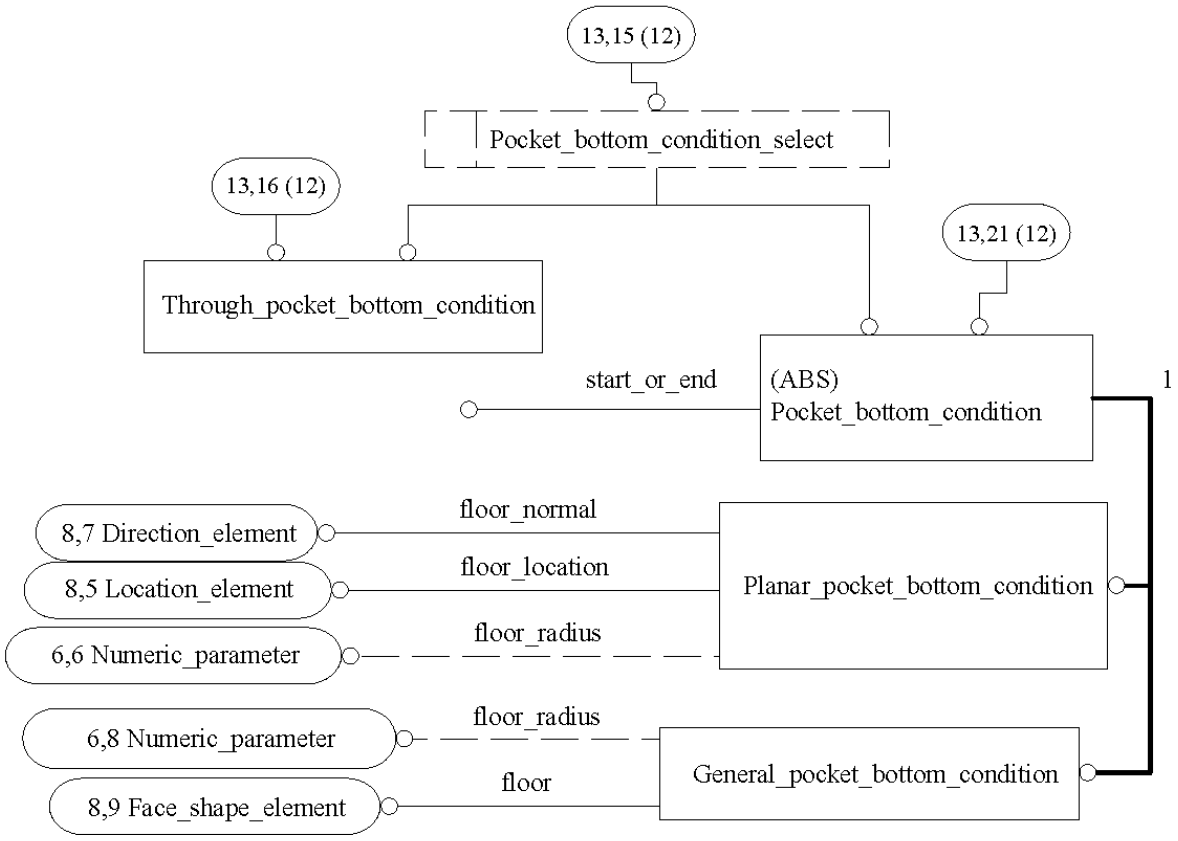


Figure G.13 - ARM EXPRESS-G diagram 13 of 28

©ISO 2000 - All rights reserved

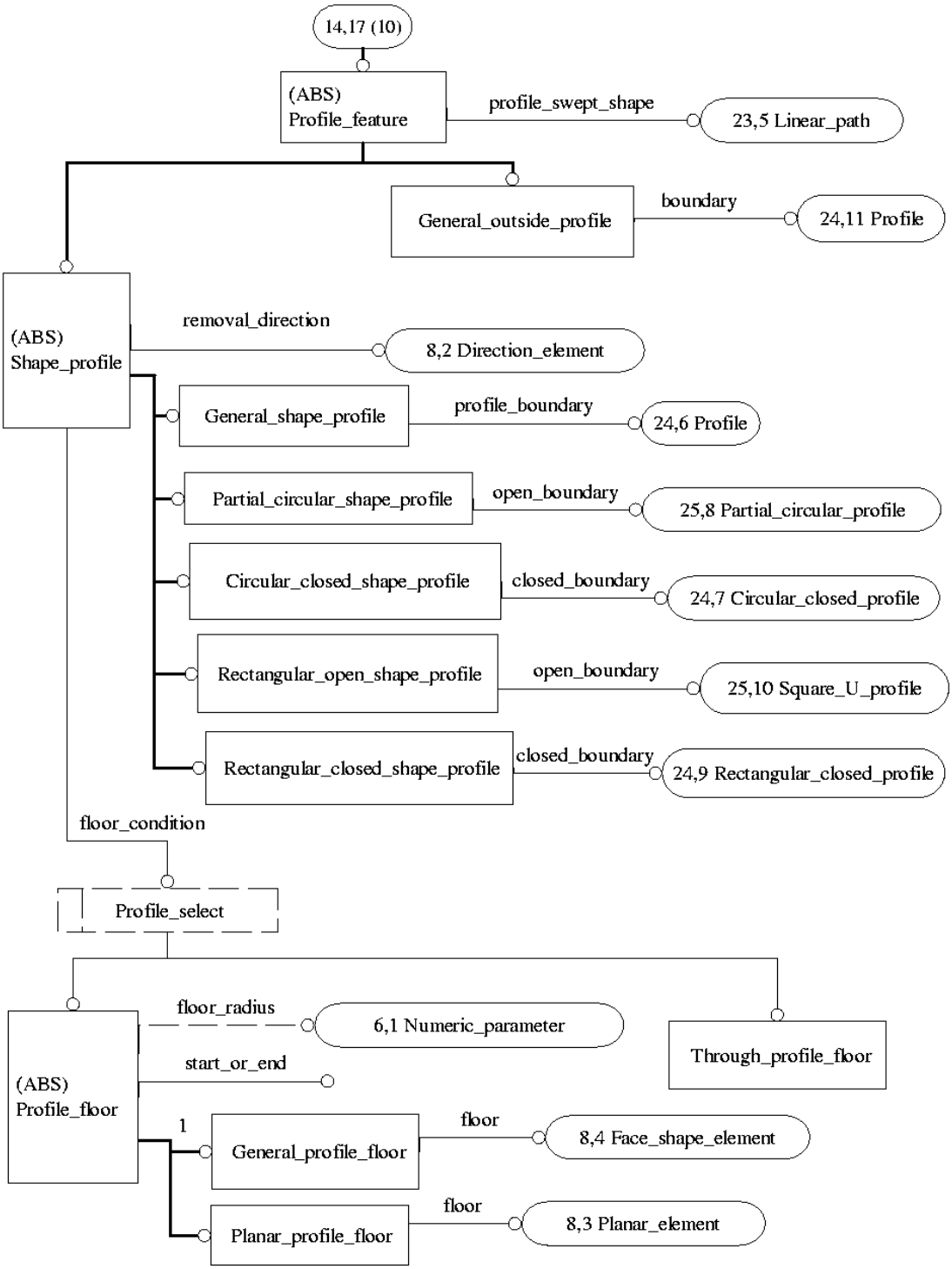


Figure G.14 - ARM EXPRESS-G diagram 14 of 28

 ©ISO 2000 - All rights reserved

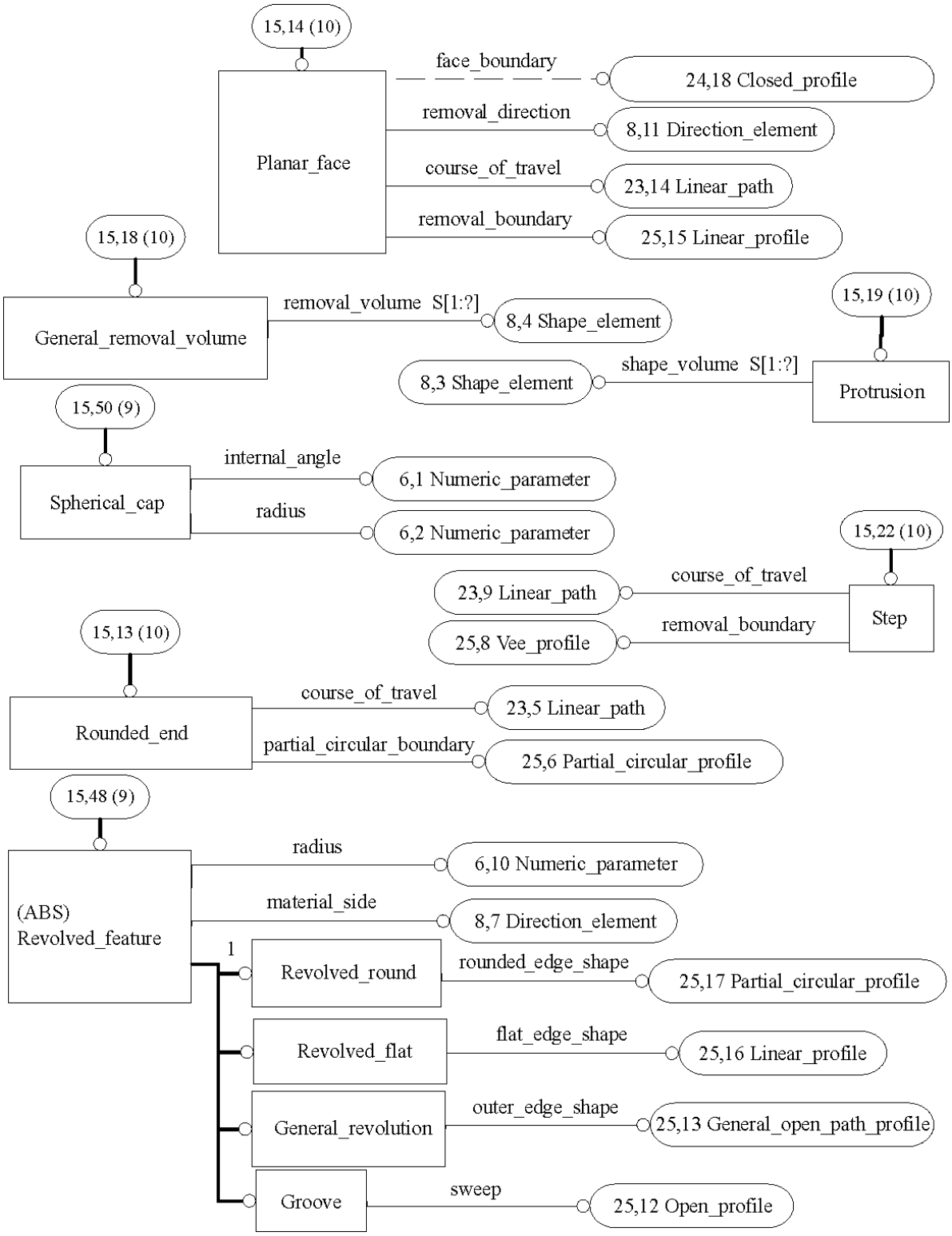


Figure G.15 - ARM EXPRESS-G diagram 15 of 28

©ISO 2000 - All rights reserved

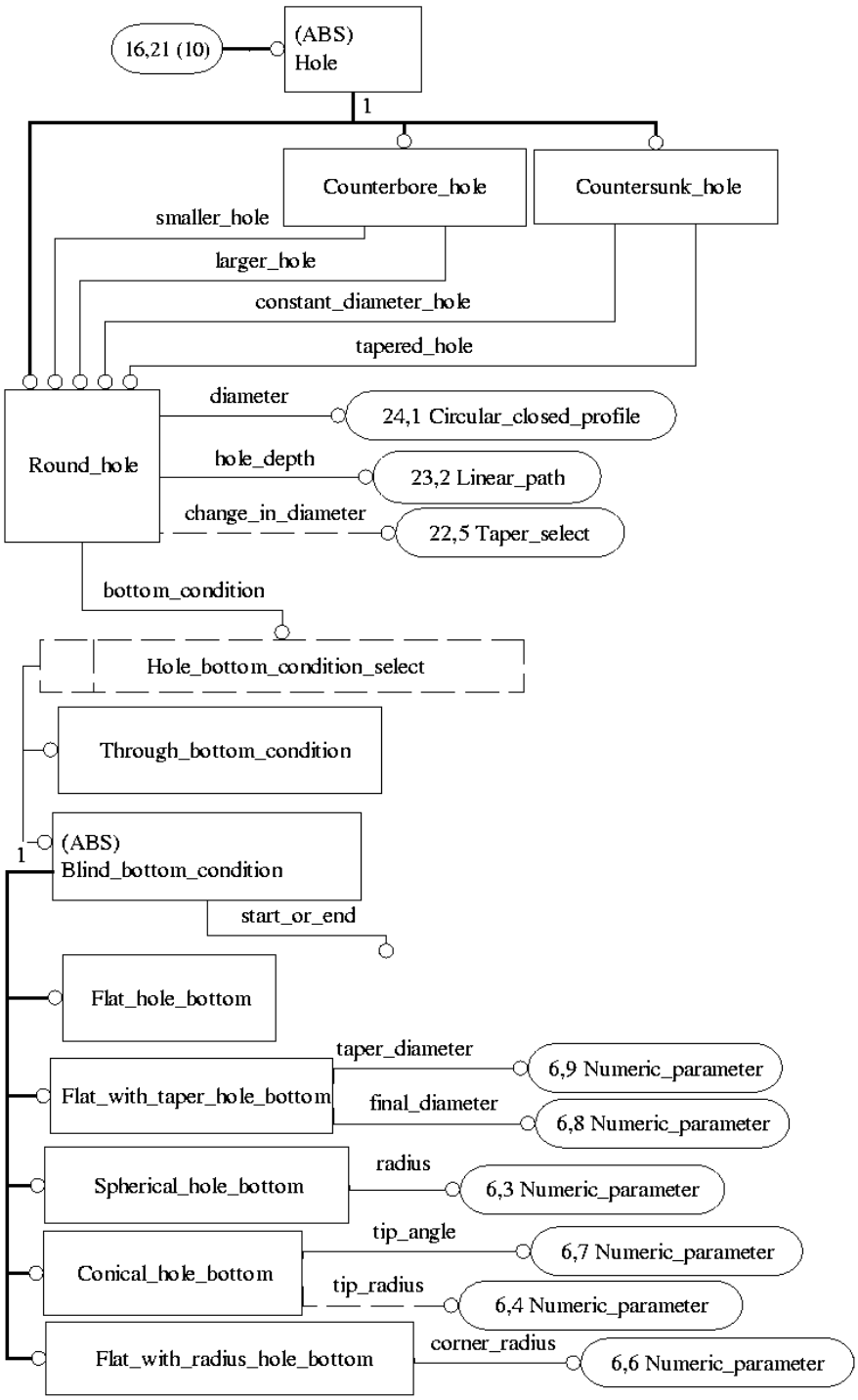


Figure G.16 - ARM EXPRESS-G diagram 16 of 28

 ©ISO 2000 - All rights reserved

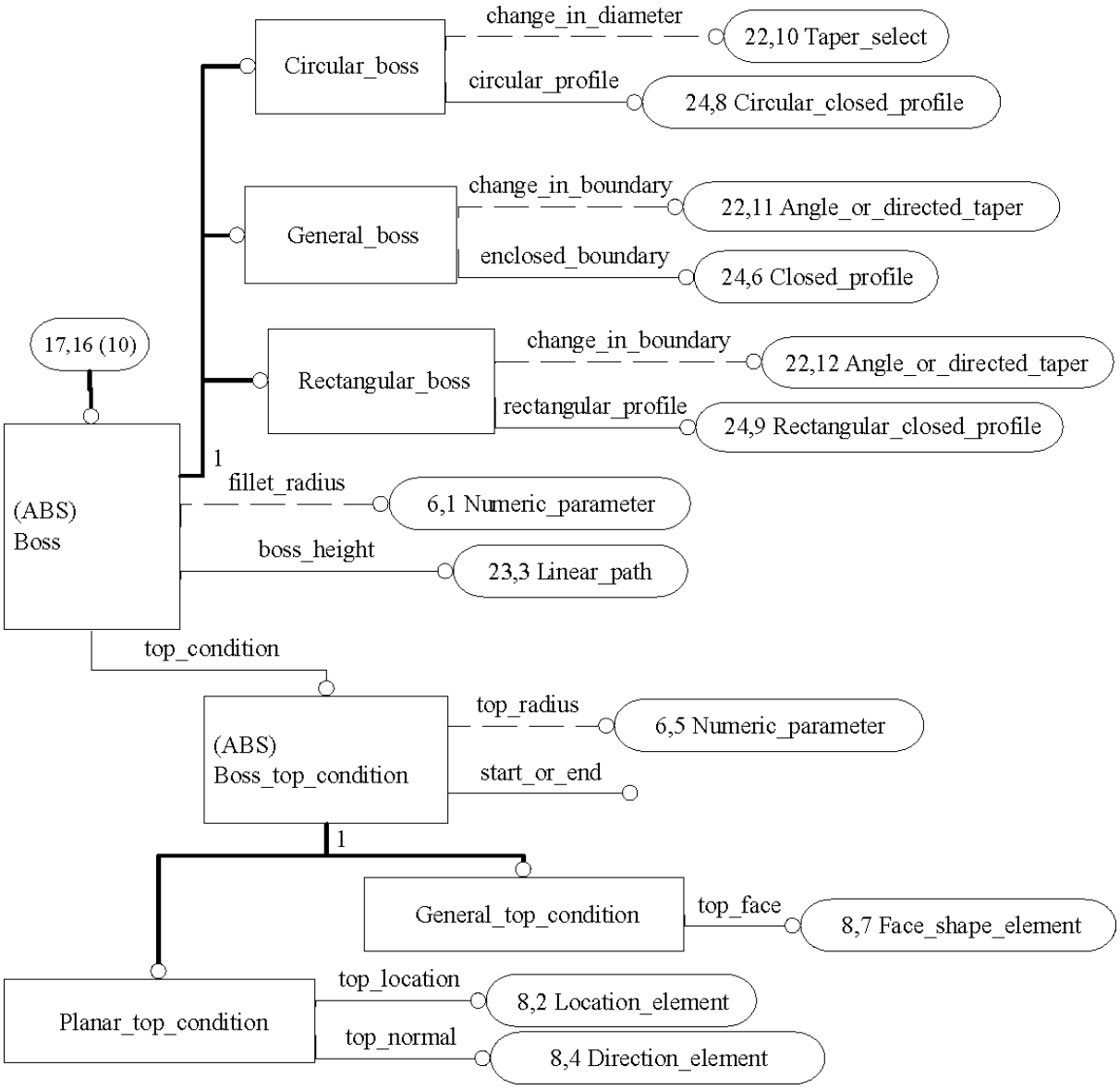


Figure G.17 - ARM EXPRESS-G diagram 17 of 28

©ISO 2000 - All rights reserved

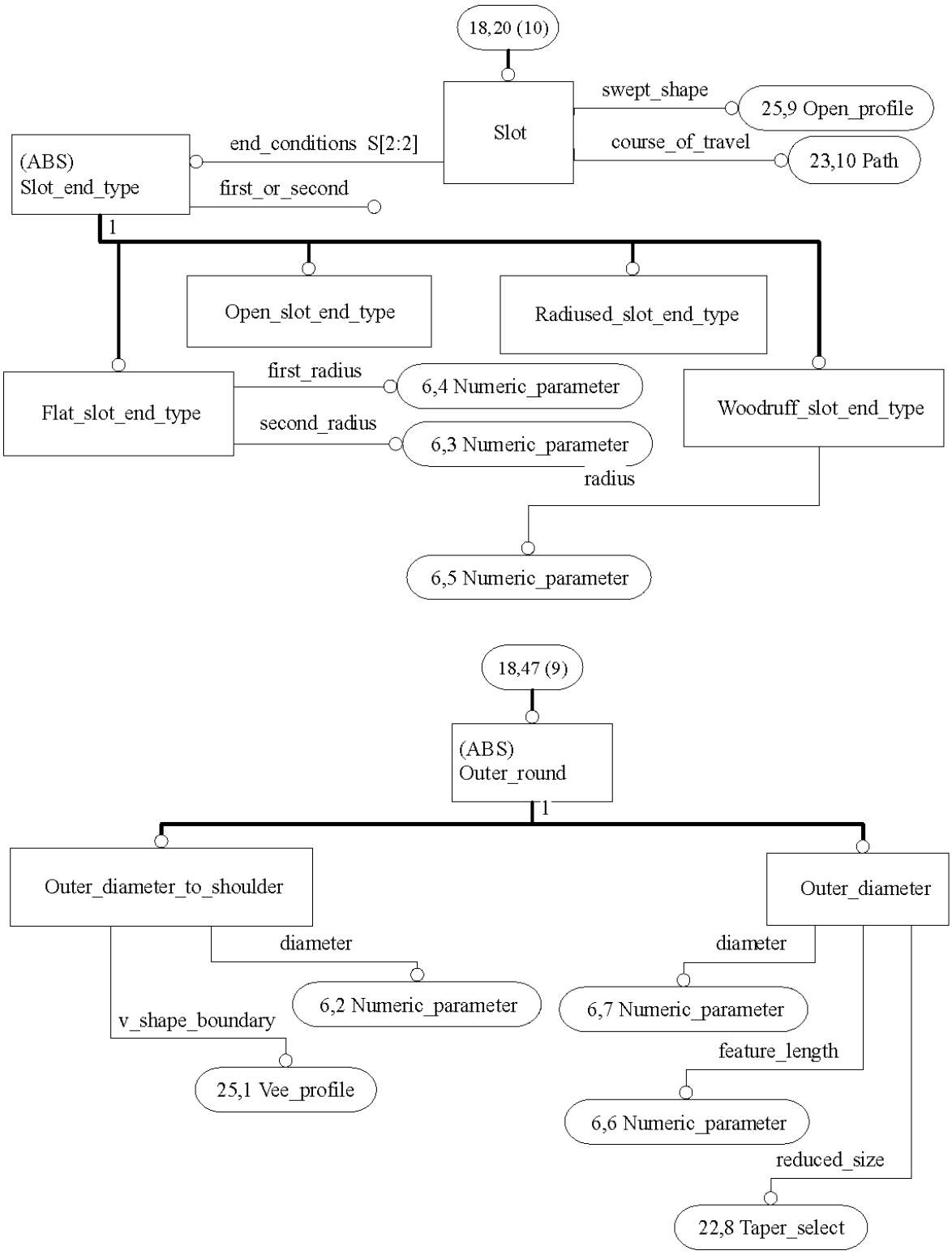


Figure G.18 - ARM EXPRESS-G diagram 18 of 28

 ©ISO 2000 - All rights reserved

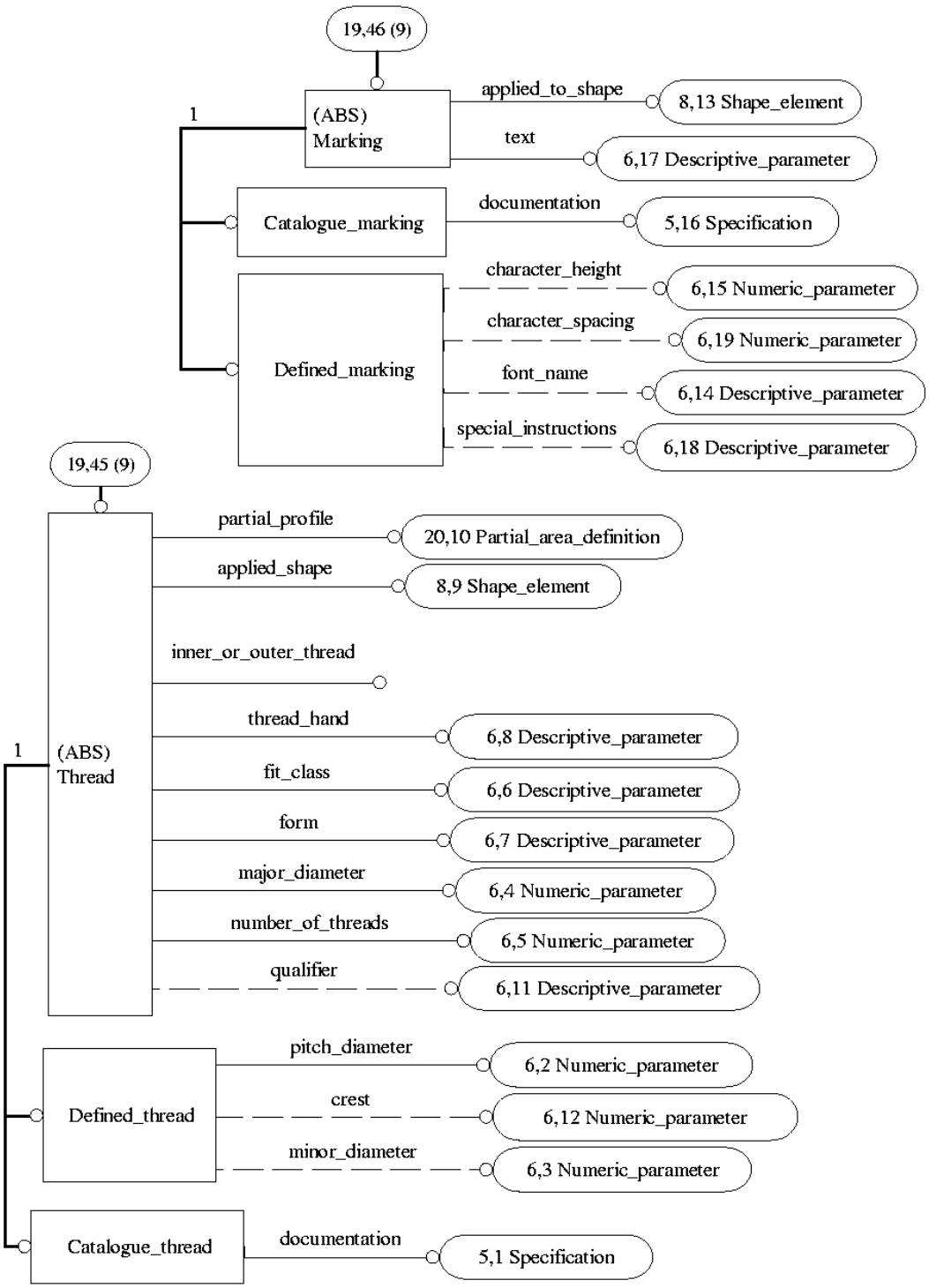


Figure G.19 - ARM EXPRESS-G diagram 19 of 28

©ISO 2000 - All rights reserved

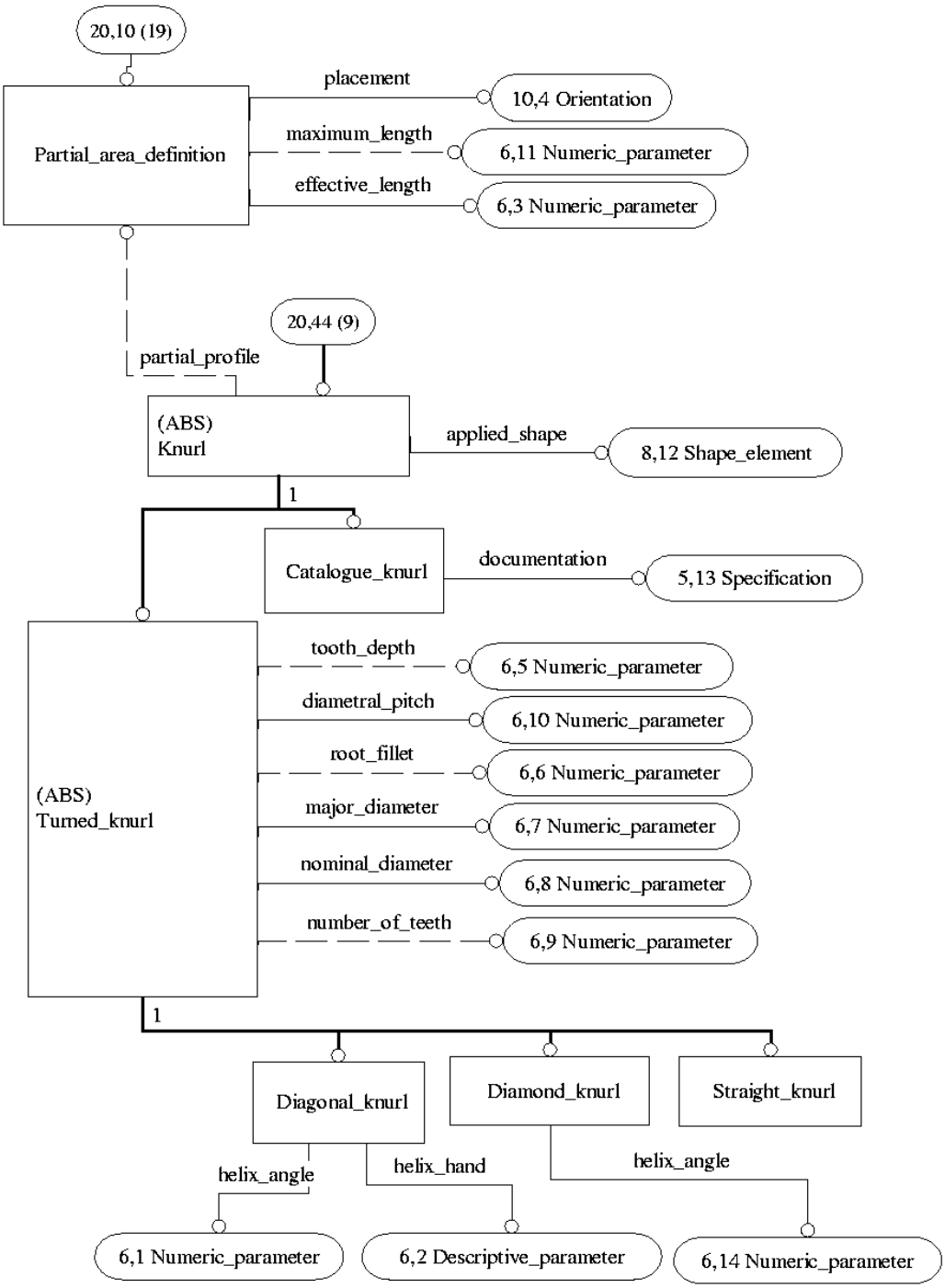


Figure G.20 - ARM EXPRESS-G diagram 20 of 28

 ©ISO 2000 - All rights reserved

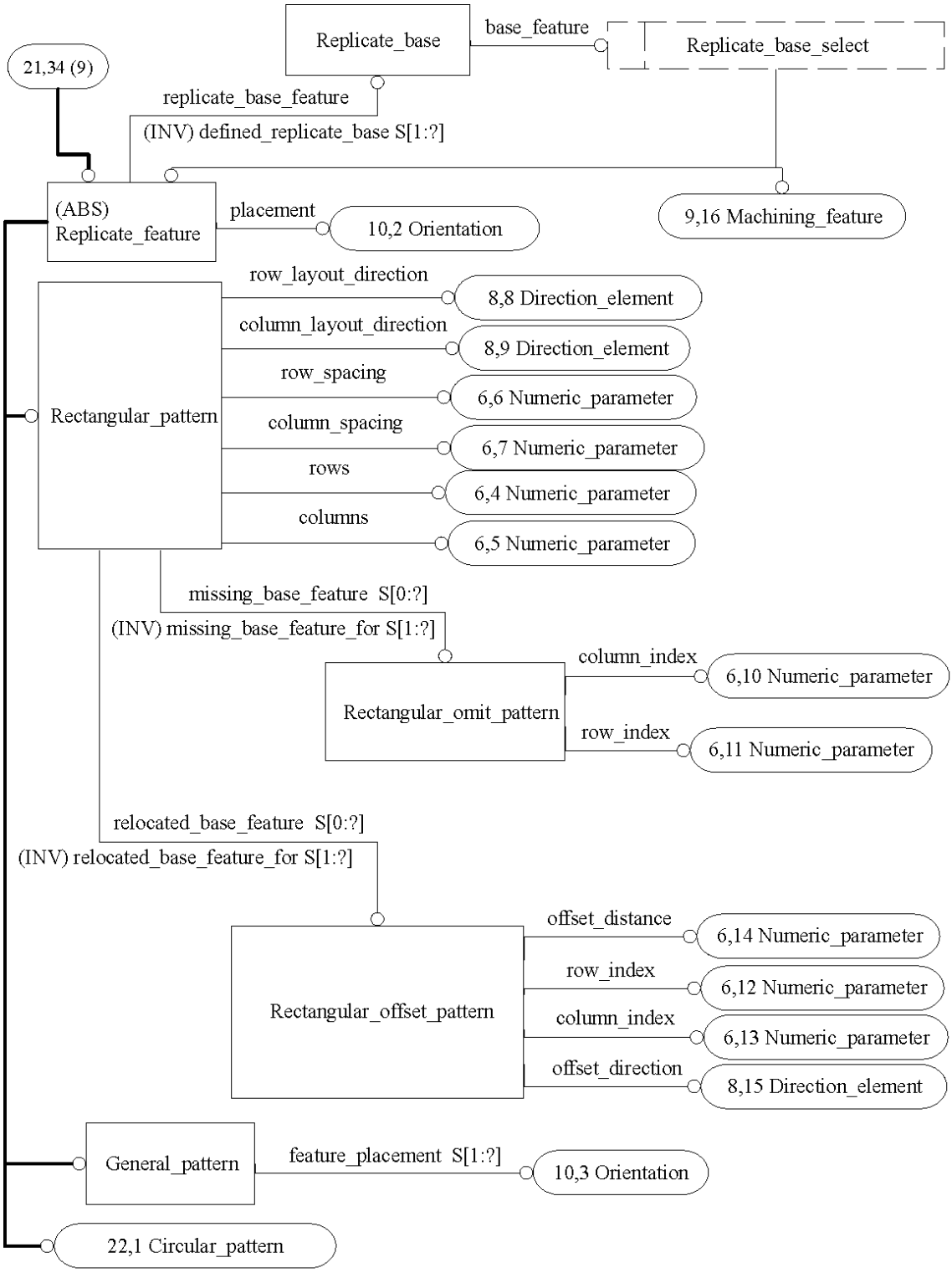


Figure G.21 - ARM EXPRESS-G diagram 21 of 28

©ISO 2000 - All rights reserved

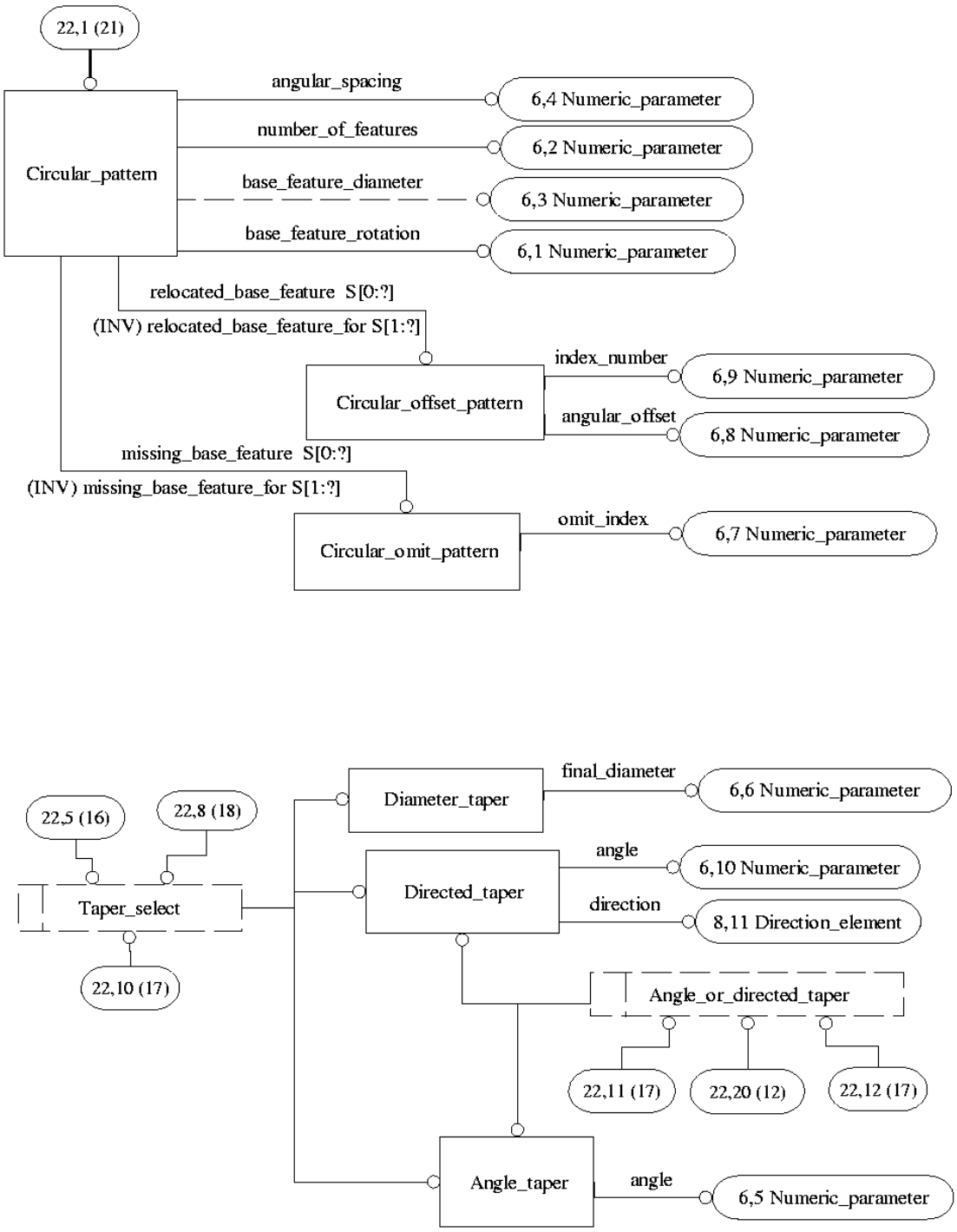


Figure G.22 - ARM EXPRESS-G diagram 22 of 28

 ©ISO 2000 - All rights reserved

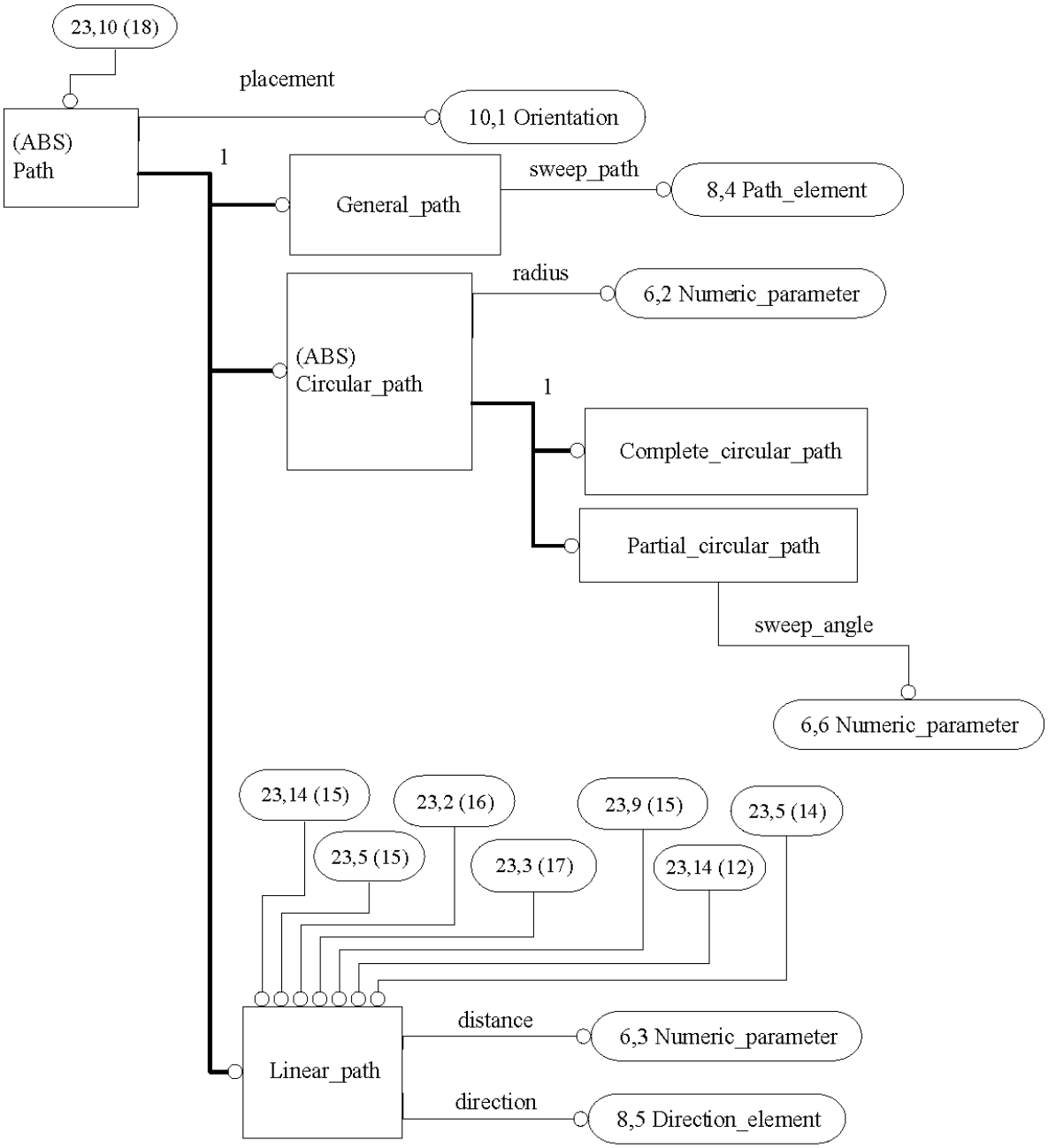


Figure G.23 - ARM EXPRESS-G diagram 23 of 28

©ISO 2000 - All rights reserved

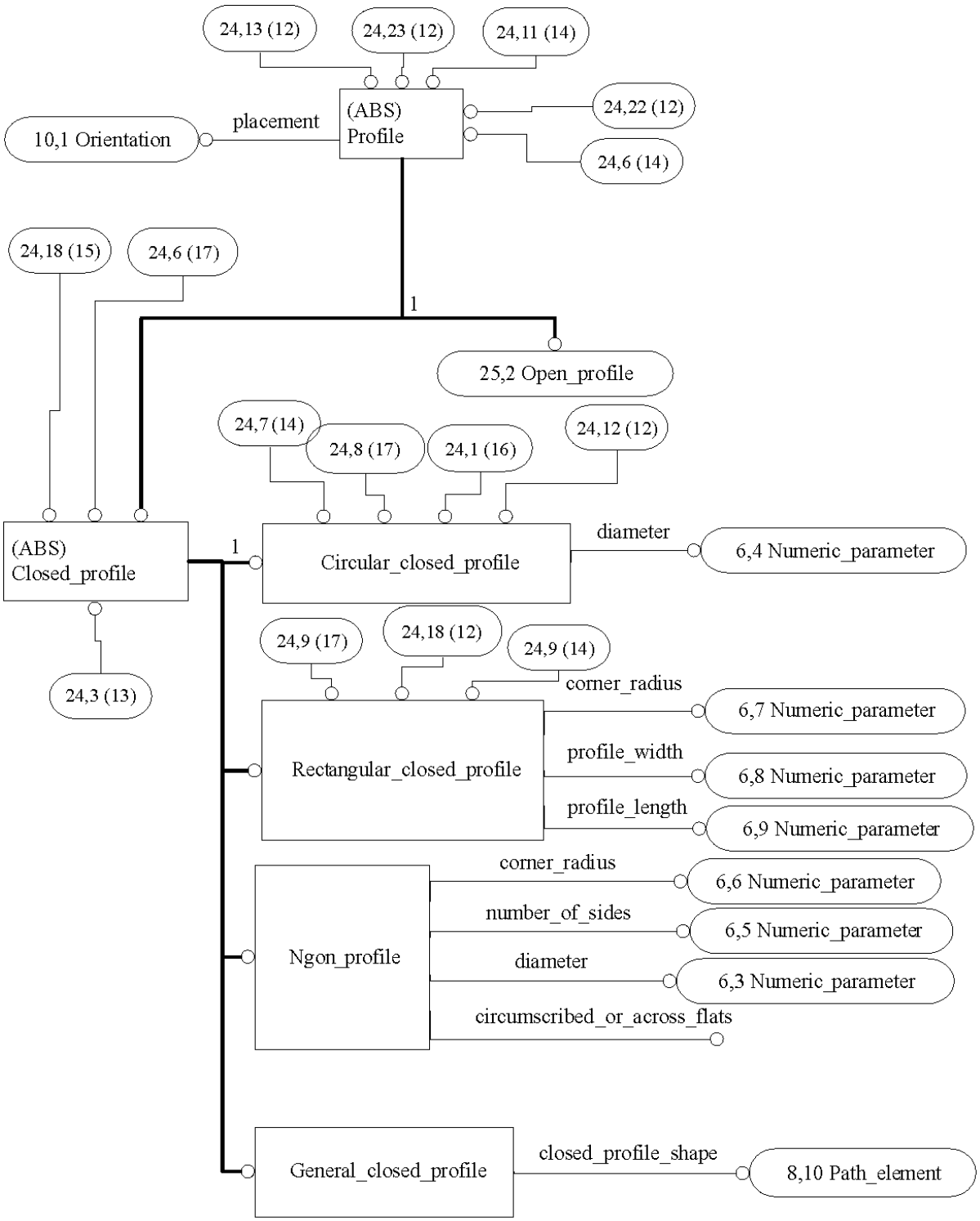


Figure G.24 - ARM EXPRESS-G diagram 24 of 28

 ©ISO 2000 - All rights reserved

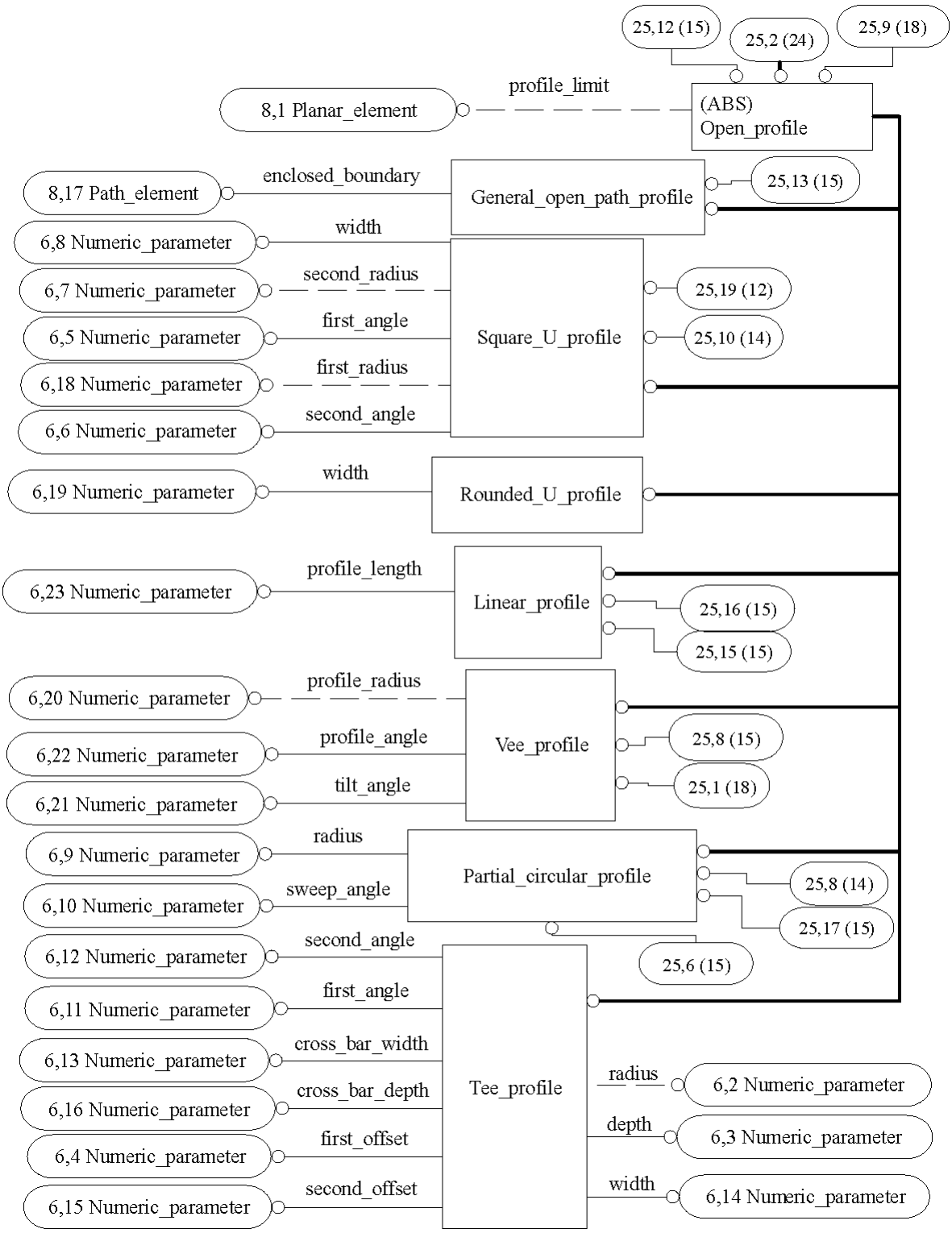


Figure G.25 - ARM EXPRESS-G diagram 25 of 28

©ISO 2000 - All rights reserved

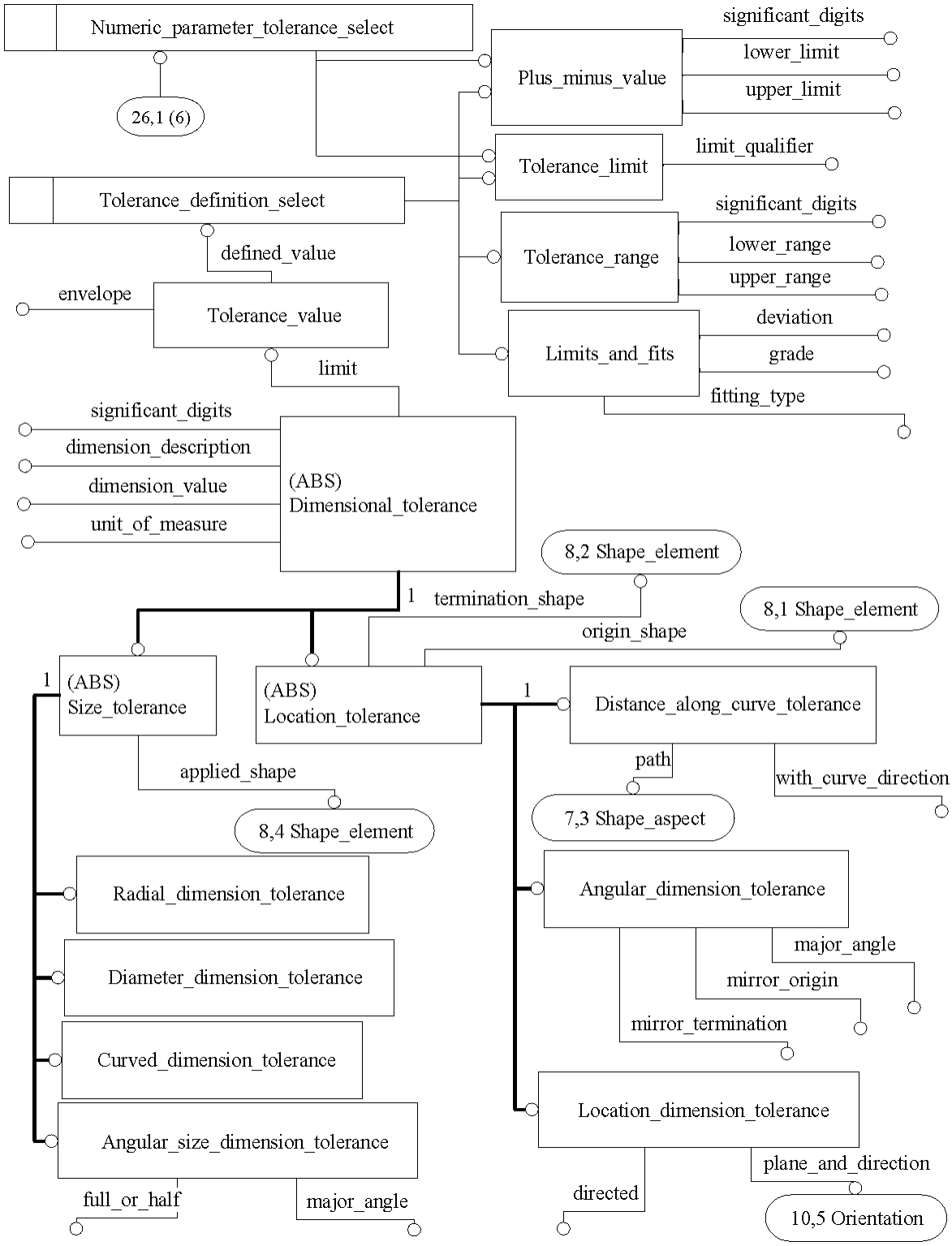


Figure G.26 - ARM EXPRESS-G diagram 26 of 28

 ©ISO 2000 - All rights reserved

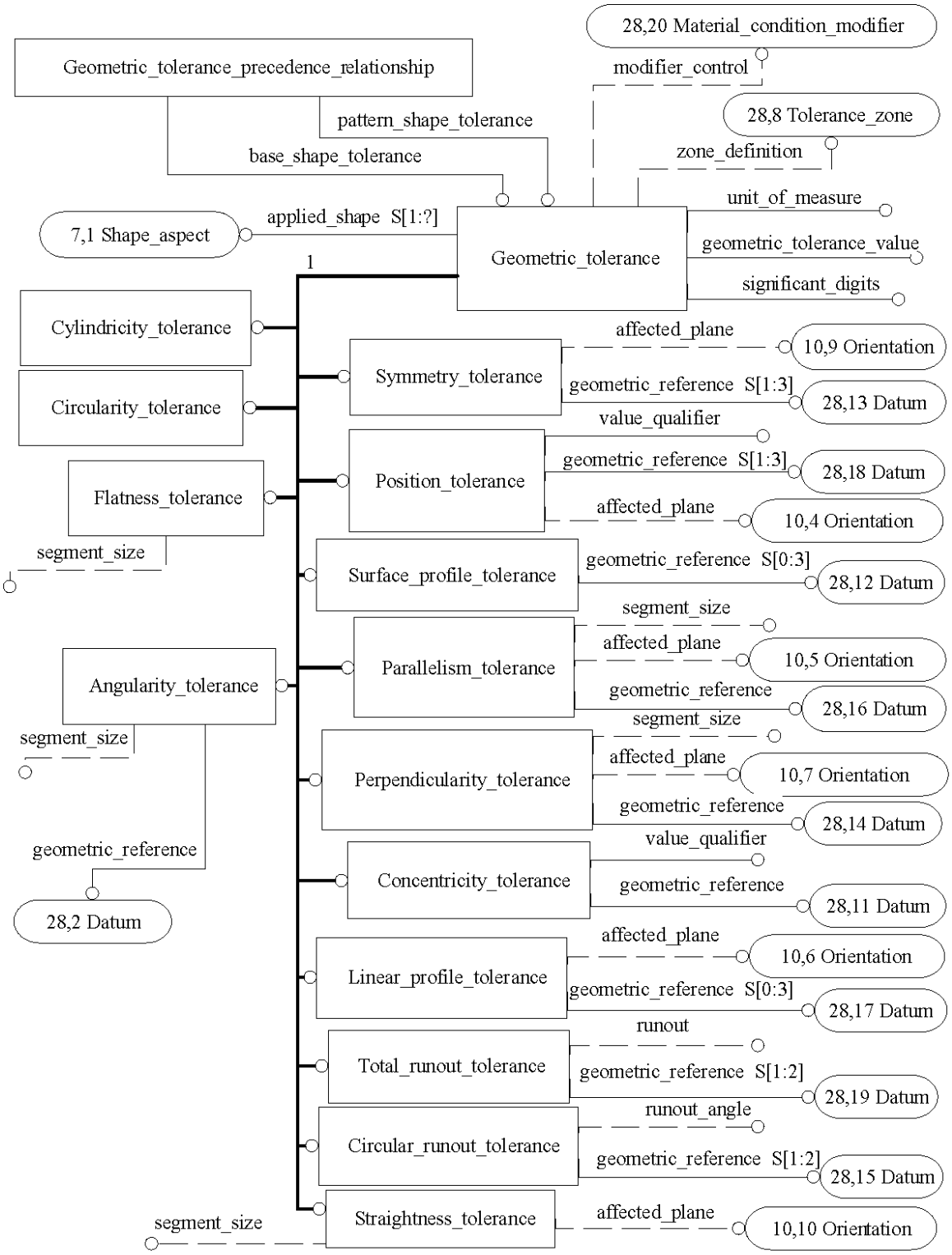


Figure G.27 - ARM EXPRESS-G diagram 27 of 28

©ISO 2000 - All rights reserved

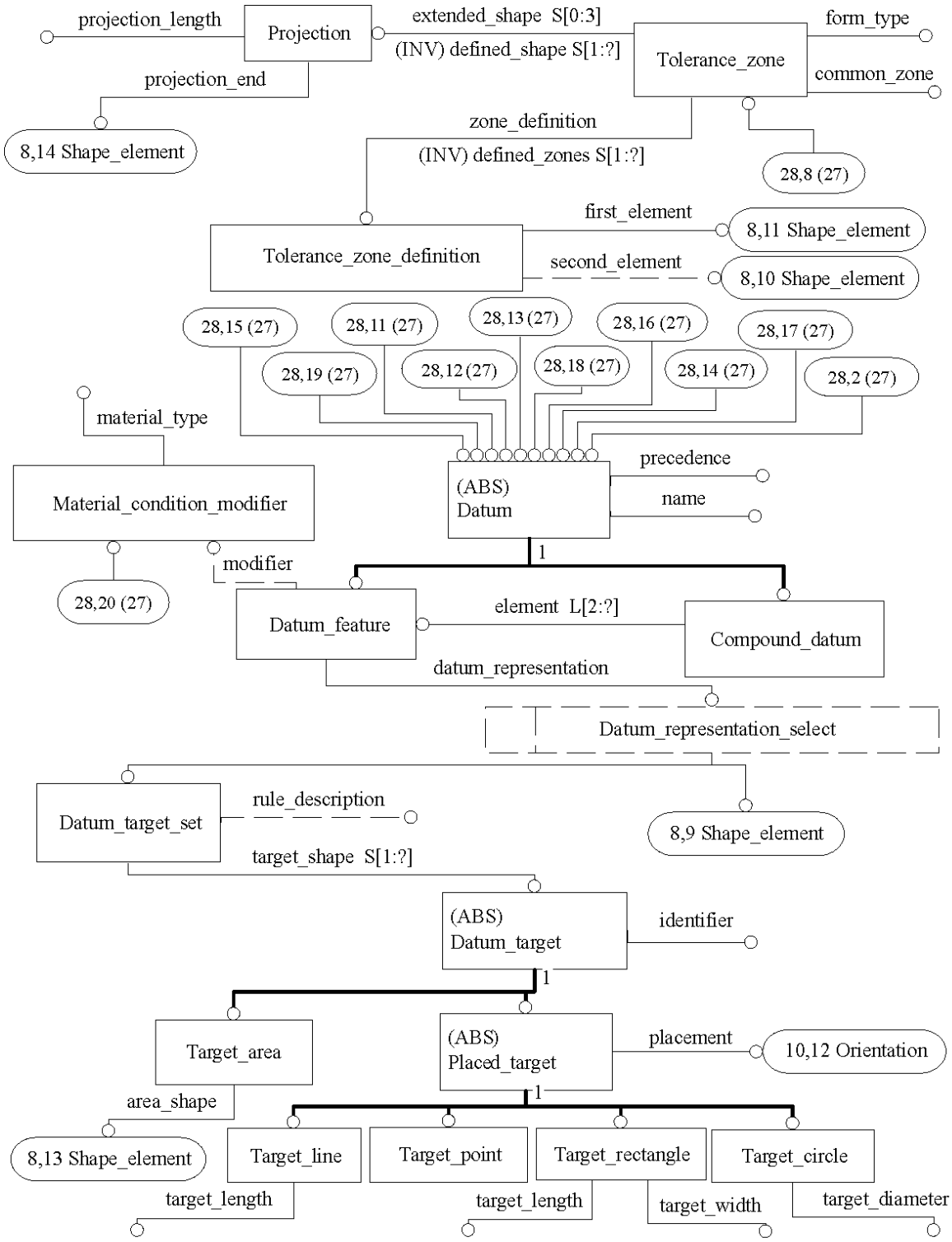


Figure G.28 - ARM EXPRESS-G diagram 28 of 28

 ©ISO 2000 - All rights reserved

Annex H
(informative)

AIM EXPRESS-G

Figure H.1 through H.37 correspond to the AIM EXPRESS annotated listing given in annex A. The figures use the EXPRESS-G graphical notation for the EXPRESS language. EXPRESS-G is defined an annex A of ISO 10303-11.

©ISO 2000 - All rights reserved

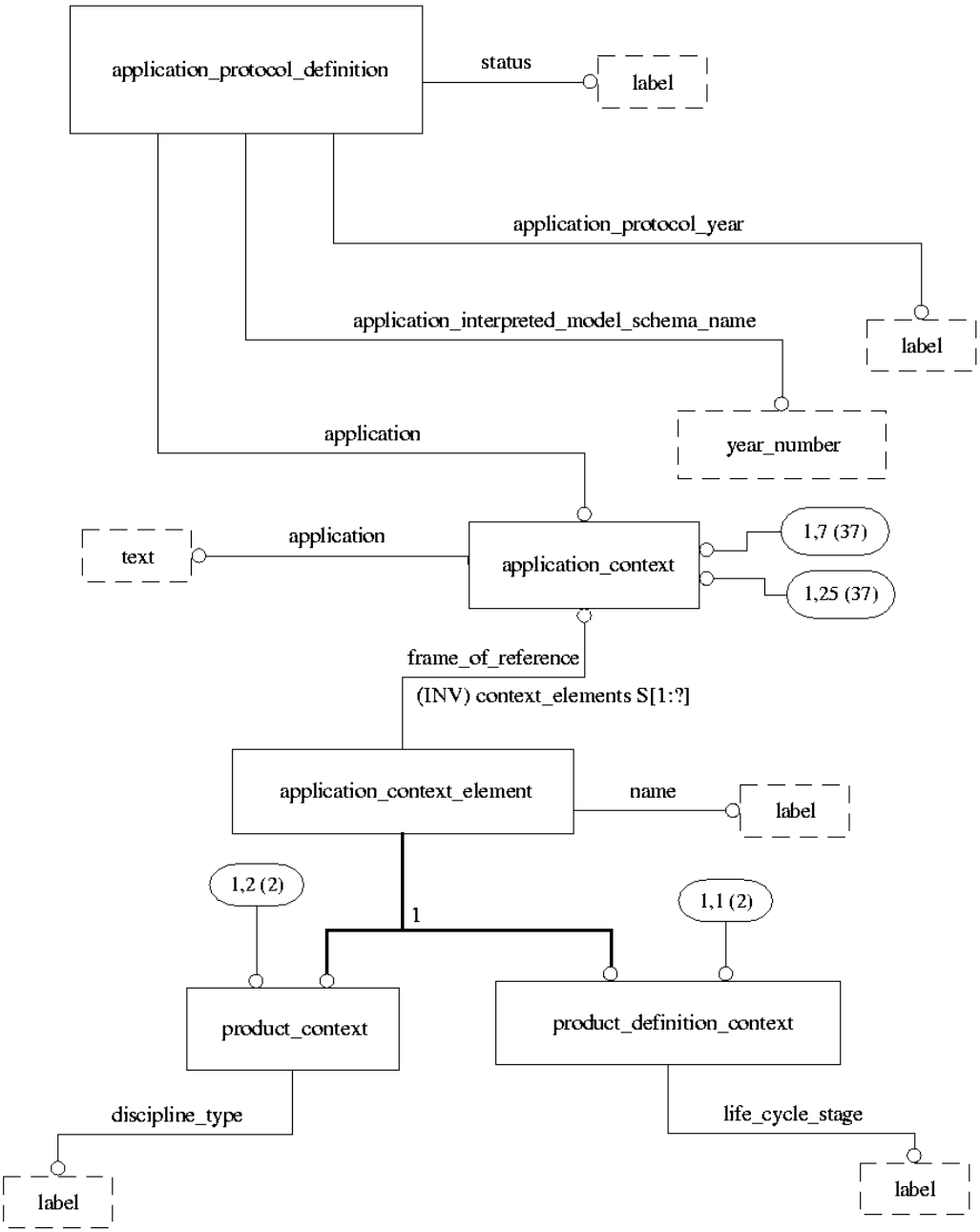


Figure H.1 - application context - AIM EXPRESS-G diagram 1 of 37

 ©ISO 2000 - All rights reserved

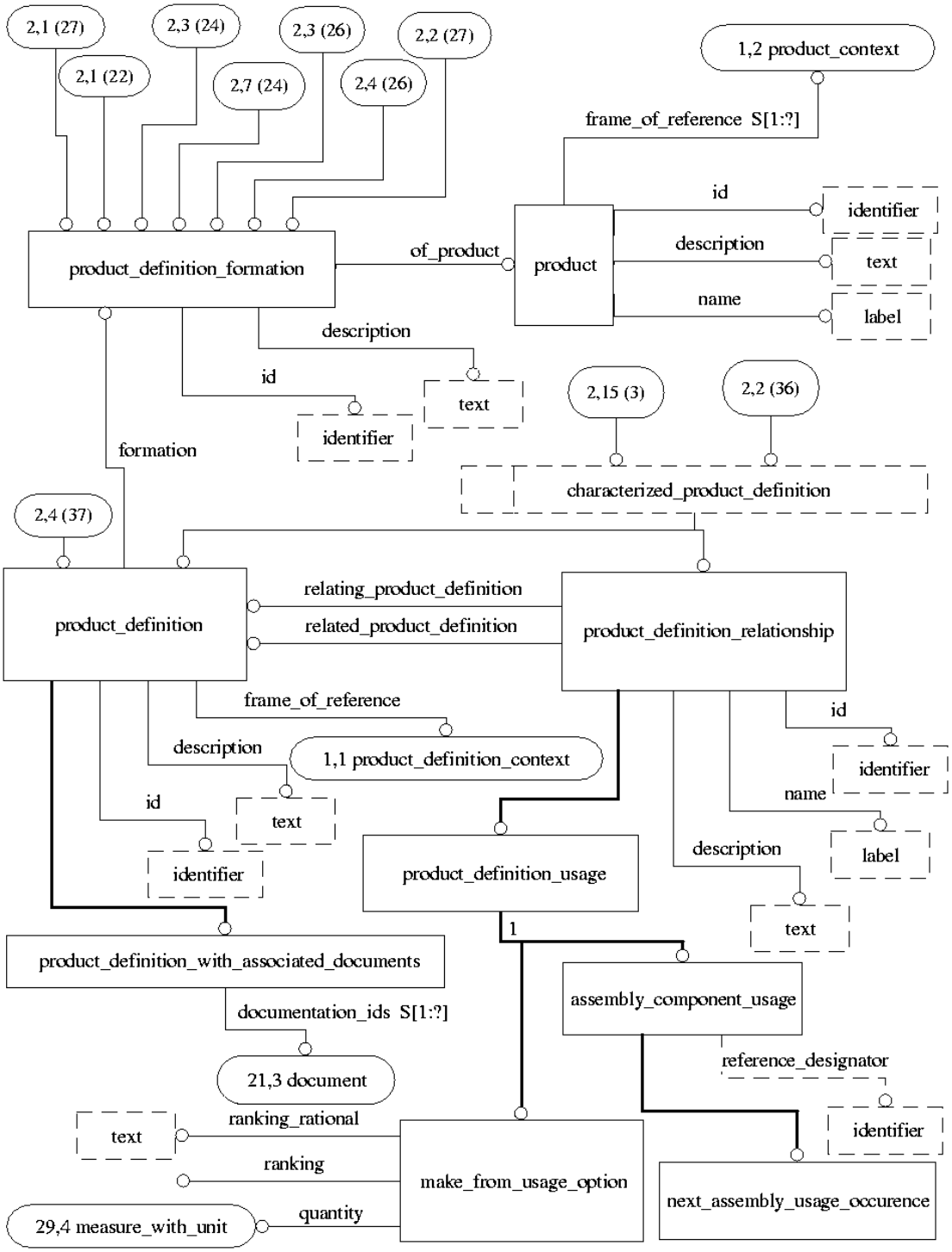


Figure H.2 - product definition - AIM EXPRESS-G diagram 2 of 37

©ISO 2000 - All rights reserved

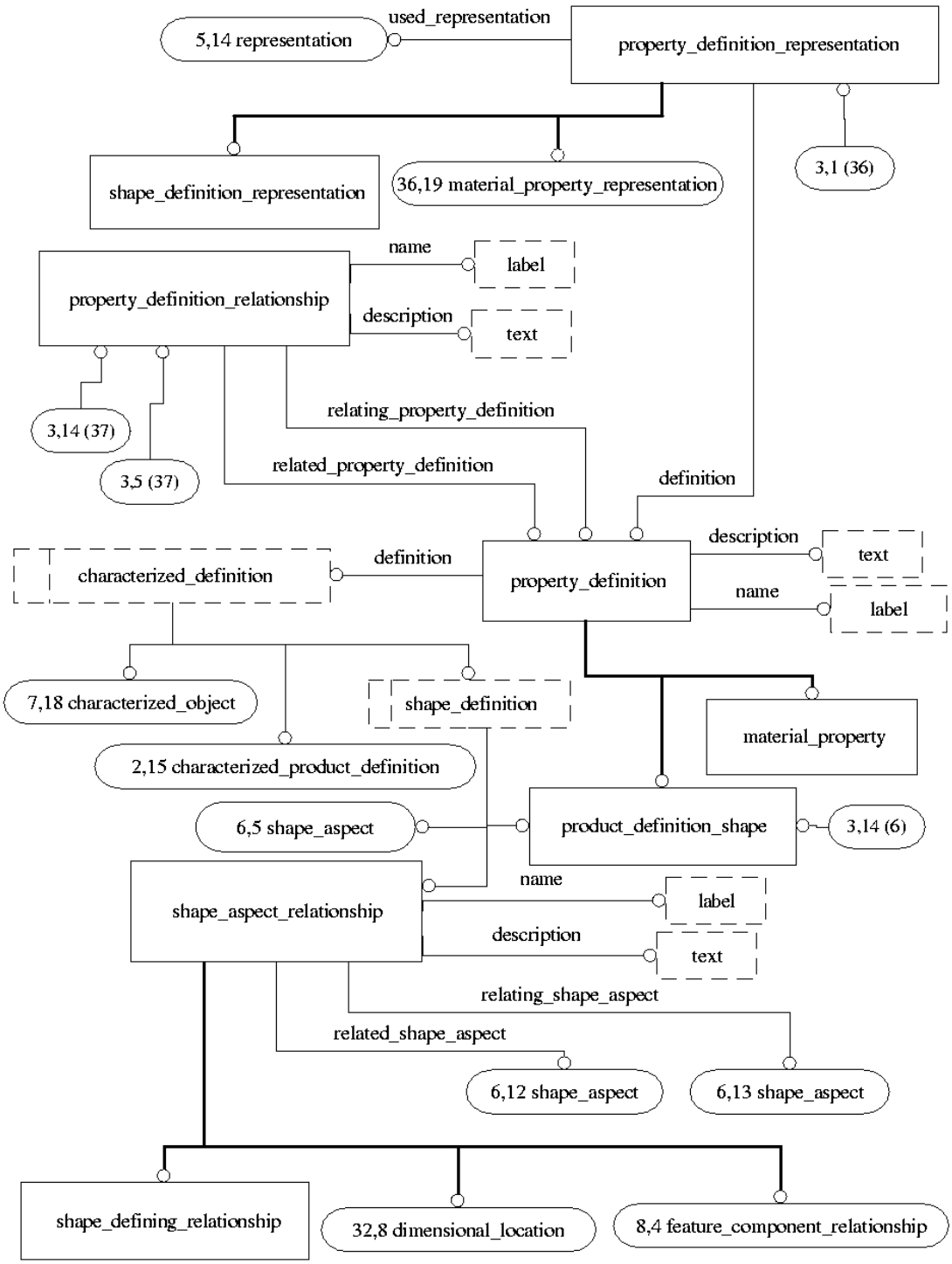


Figure H.3 - property definition - AIM EXPRESS-G diagram 3 of 37

 ©ISO 2000 - All rights reserved

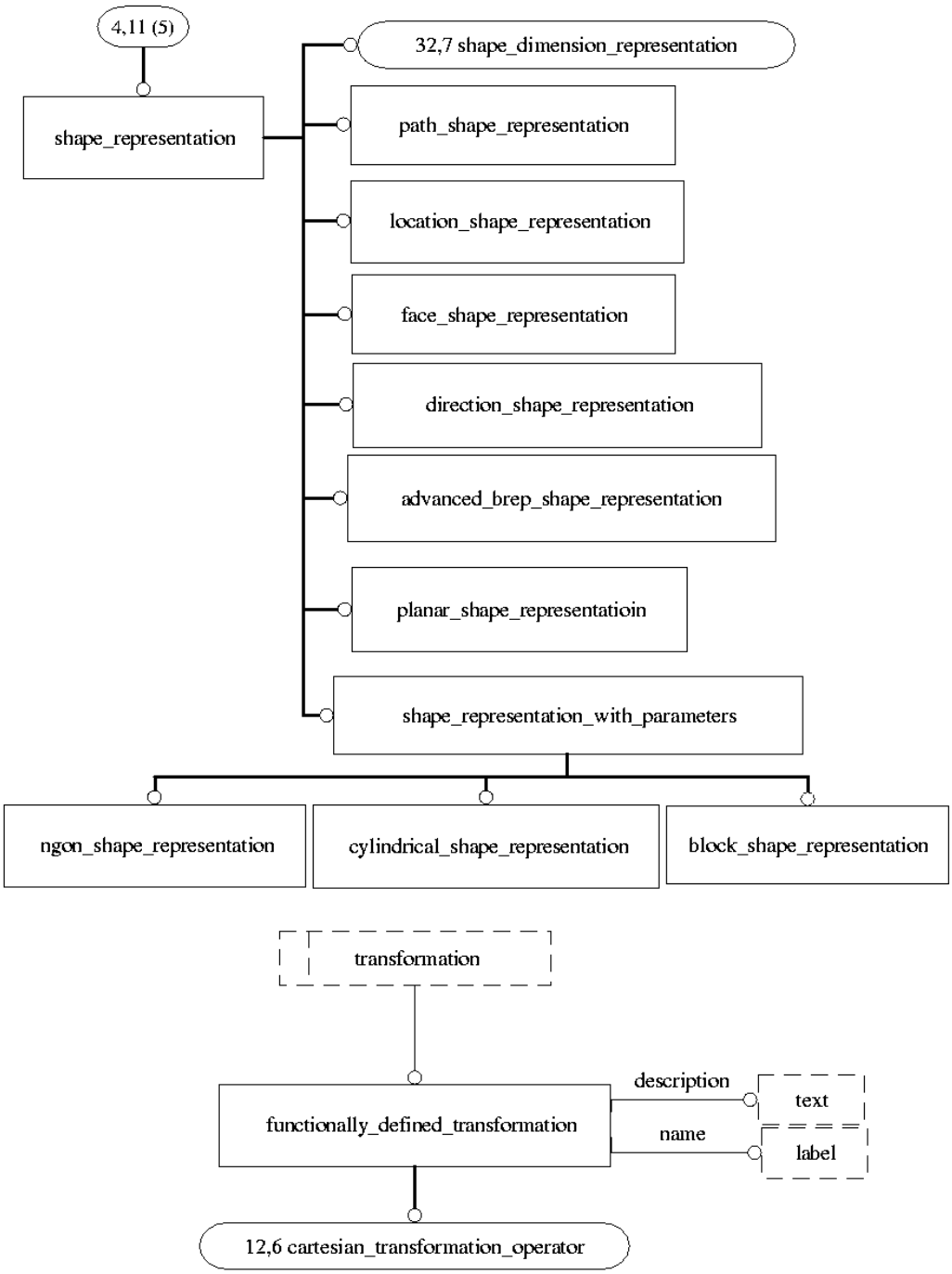


Figure H.4 - shape representation - AIM EXPRESS-G diagram 4 of 37

©ISO 2000 - All rights reserved

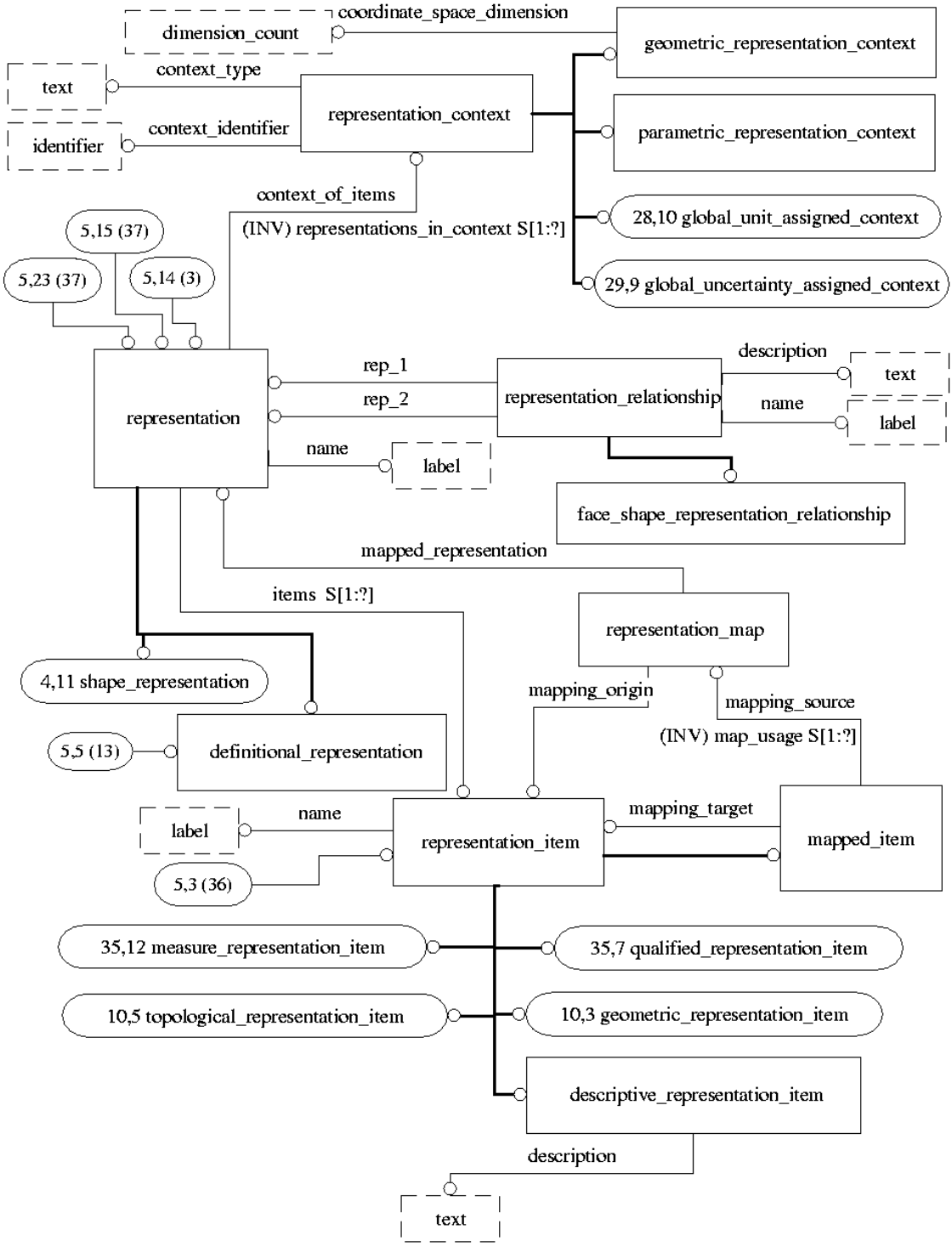


Figure H.5 - representation - AIM EXPRESS-G diagram 5 of 37

 ©ISO 2000 - All rights reserved

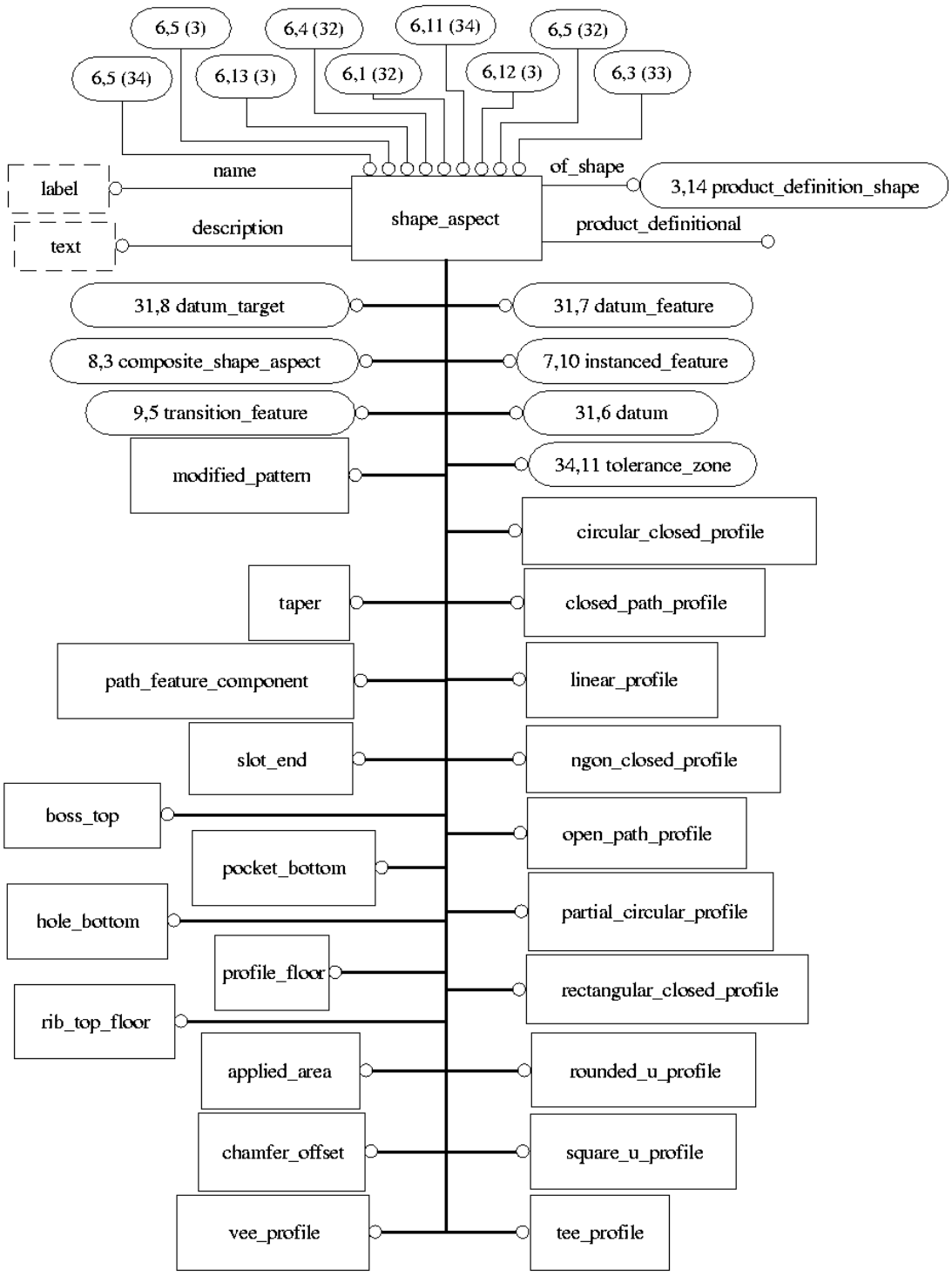


Figure H.6 - shape aspect - AIM EXPRESS-G diagram 6 of 37

©ISO 2000 - All rights reserved

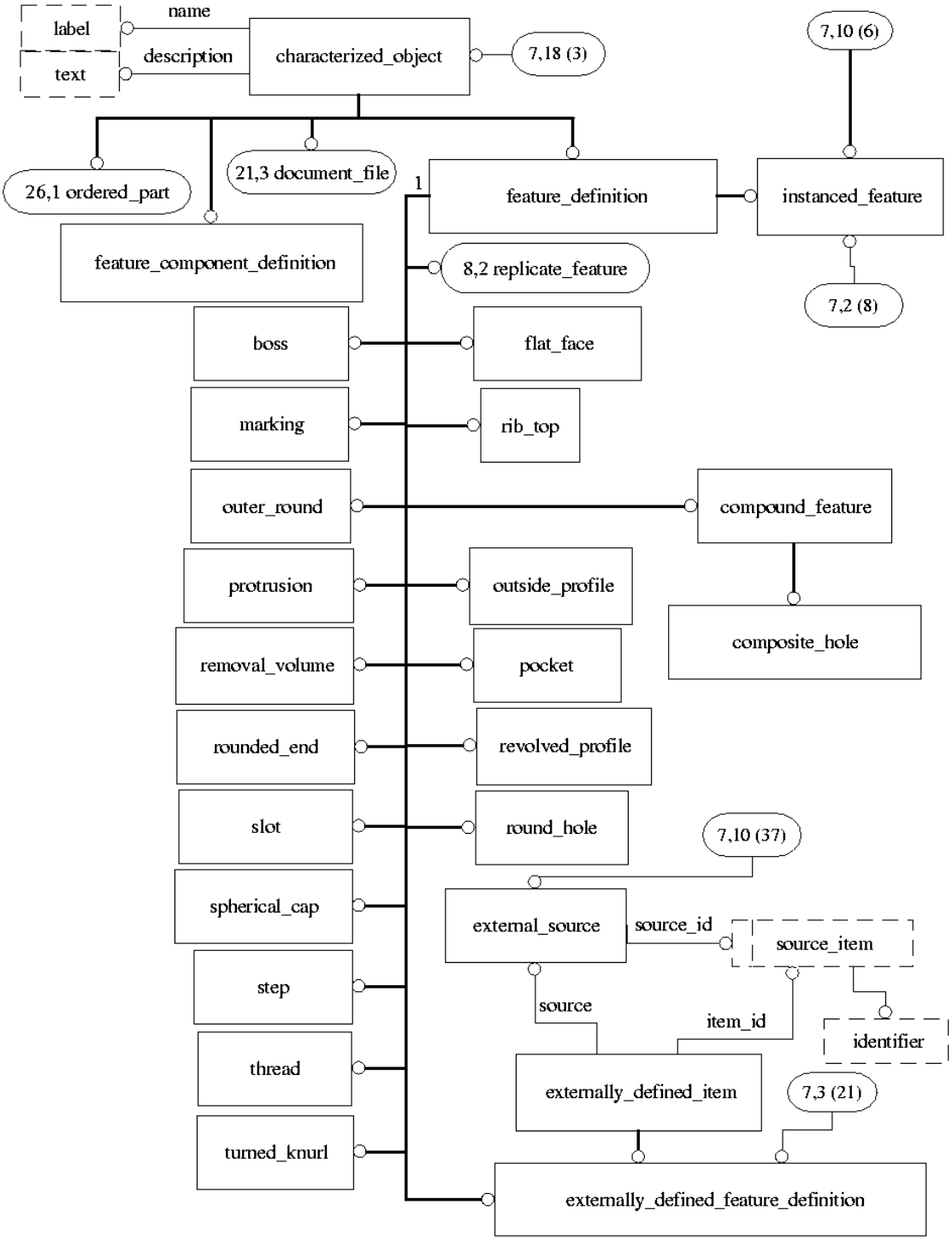


Figure H.7 - instanced feature - AIM EXPRESS-G diagram 7 of 37

 ©ISO 2000 - All rights reserved

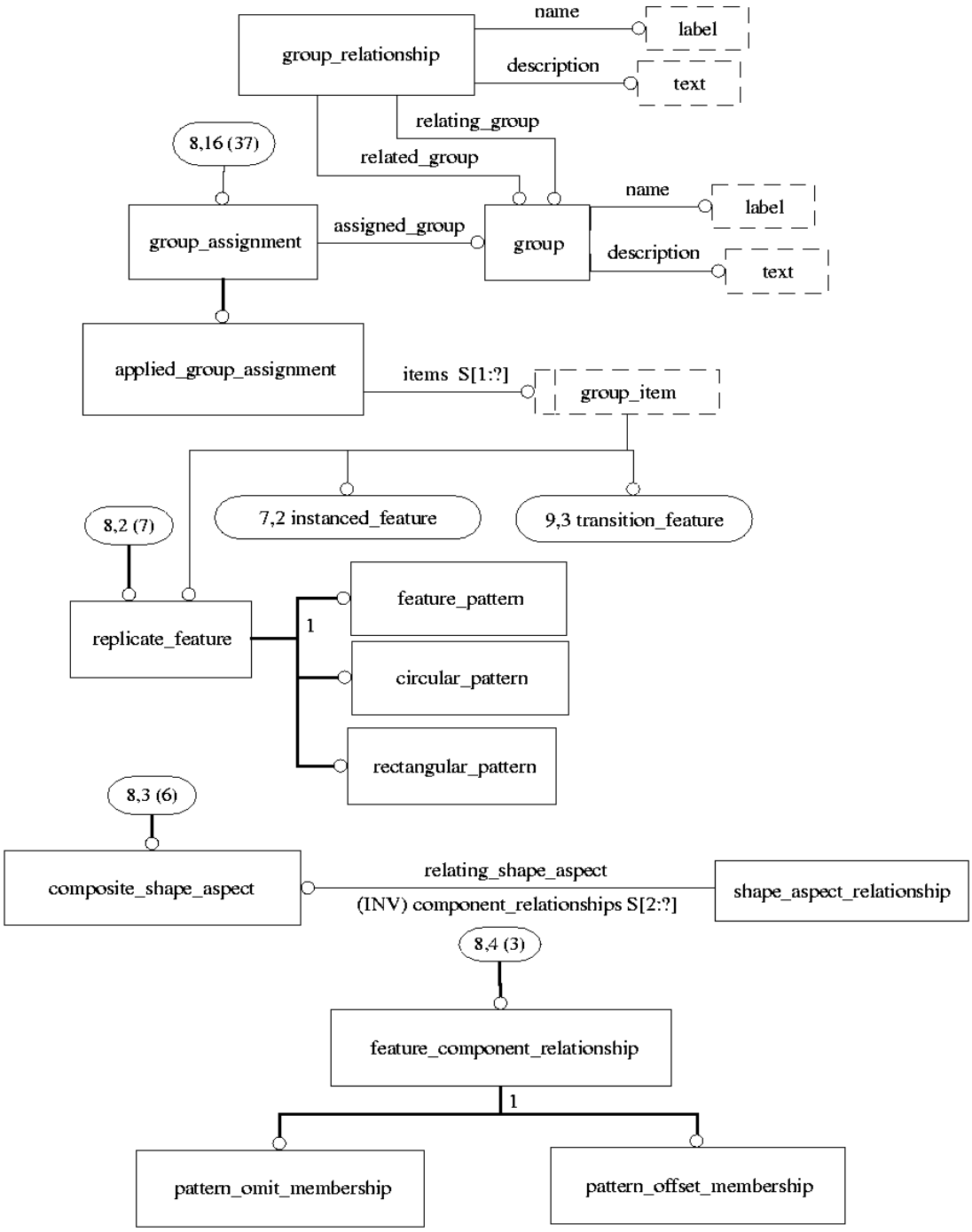


Figure H.8 - replicate feature and group - AIM EXPRESS-G diagram

8 of 37

©ISO 2000 - All rights reserved

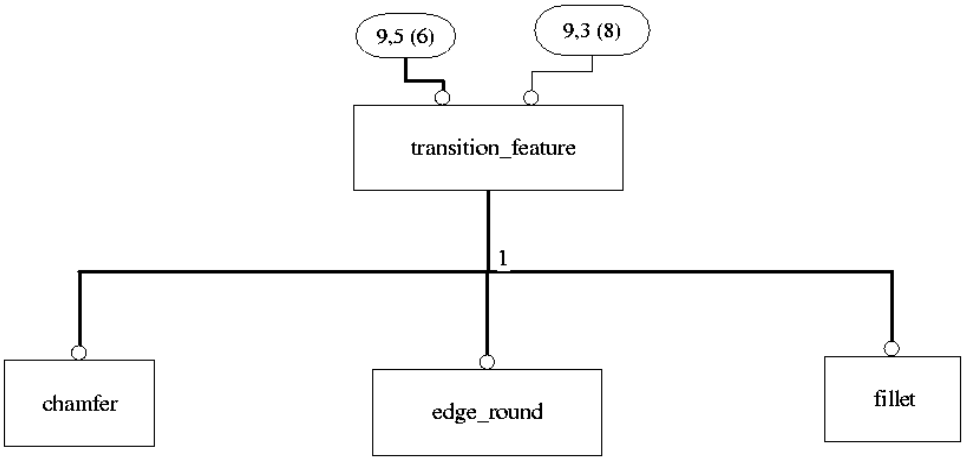


Figure H.9 - transition feature - AIM EXPRESS-G diagram 9 of 37

 ©ISO 2000 - All rights reserved

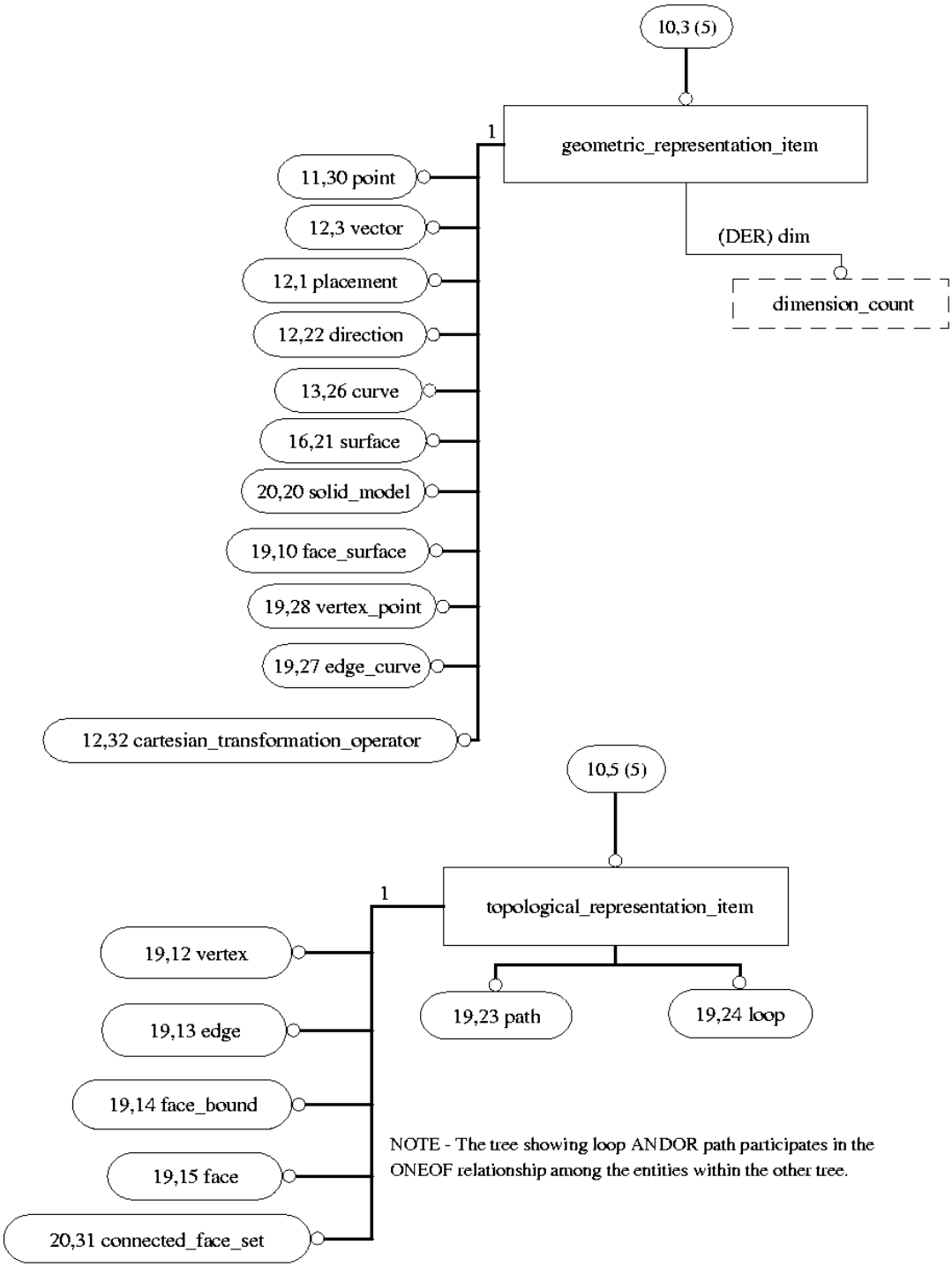


Figure H.10 - geometry and topology - AIM EXPRESS-G diagram 10 of 37

©ISO 2000 - All rights reserved

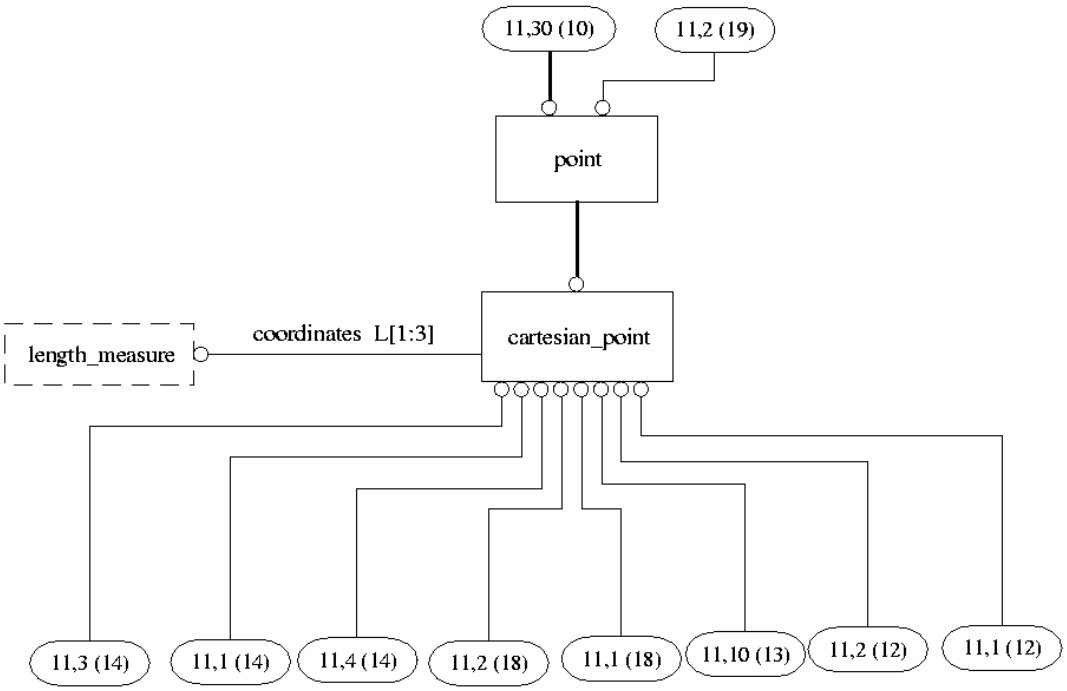


Figure H.11 - point - AIM EXPRESS-G diagram 11 of 37

 ©ISO 2000 - All rights reserved

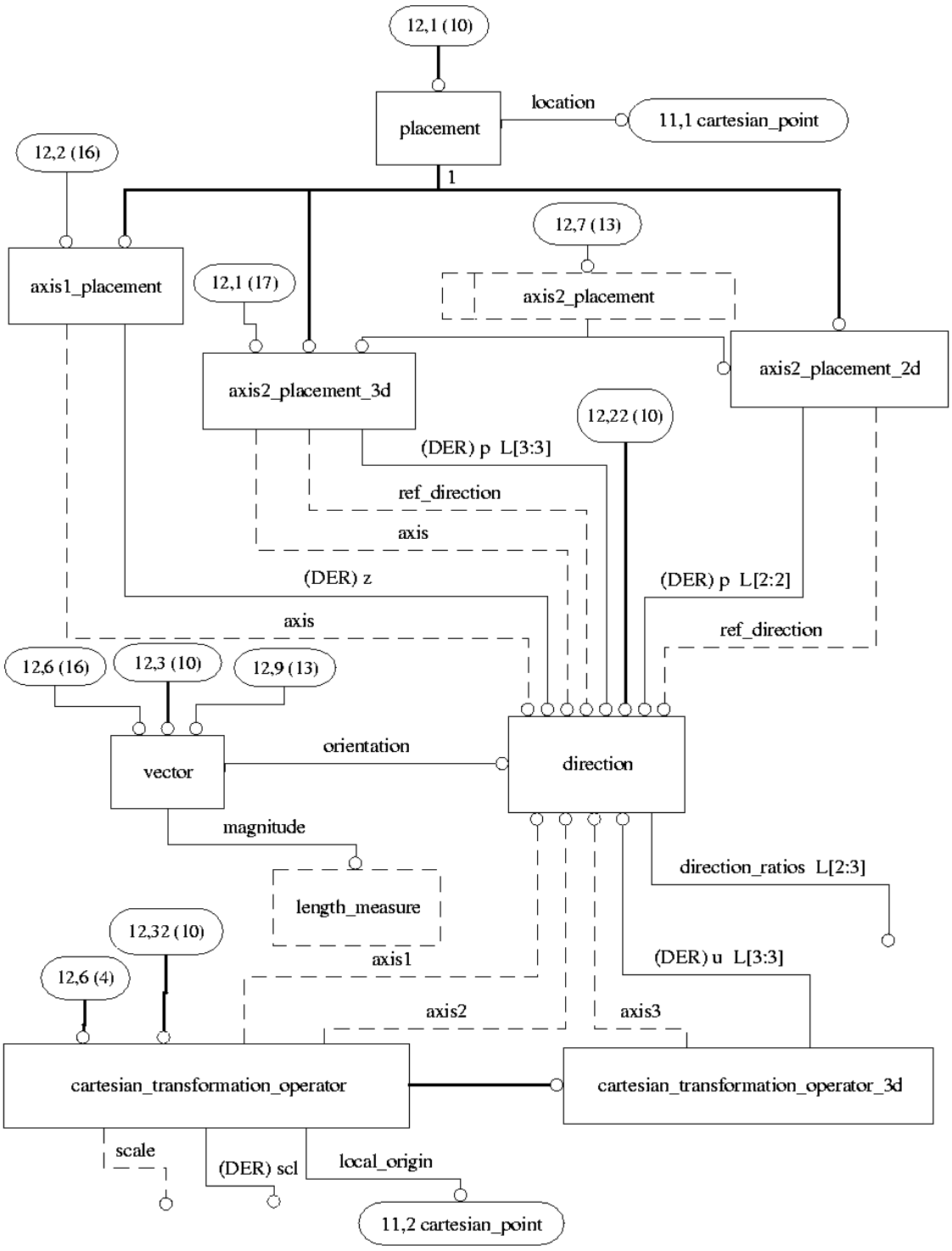


Figure H.12 - geometric orientation - AIM EXPRESS-G diagram 12 of 37

©ISO 2000 - All rights reserved

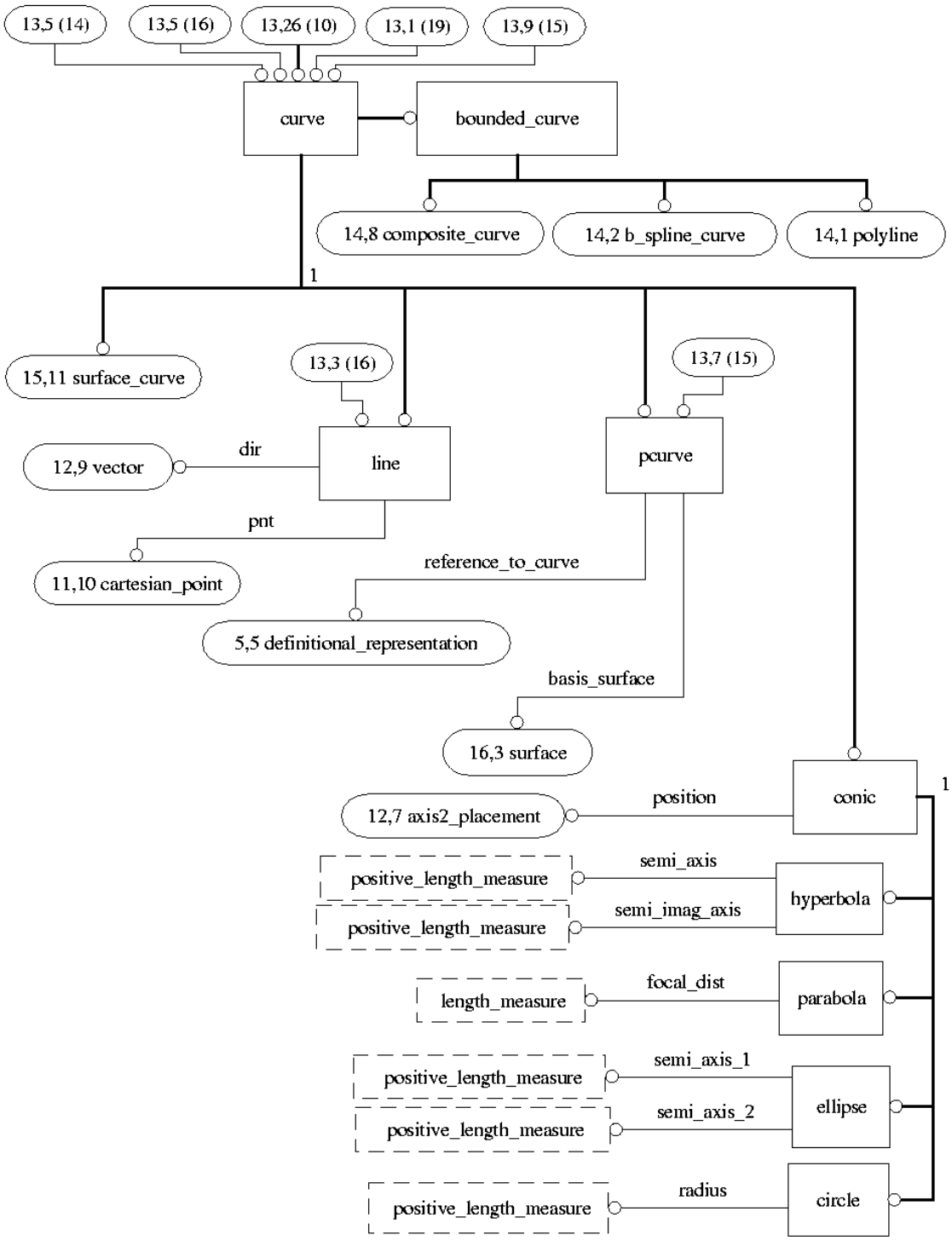


Figure H.13 - curve - AIM EXPRESS-G diagram 13 of 37

 ©ISO 2000 - All rights reserved

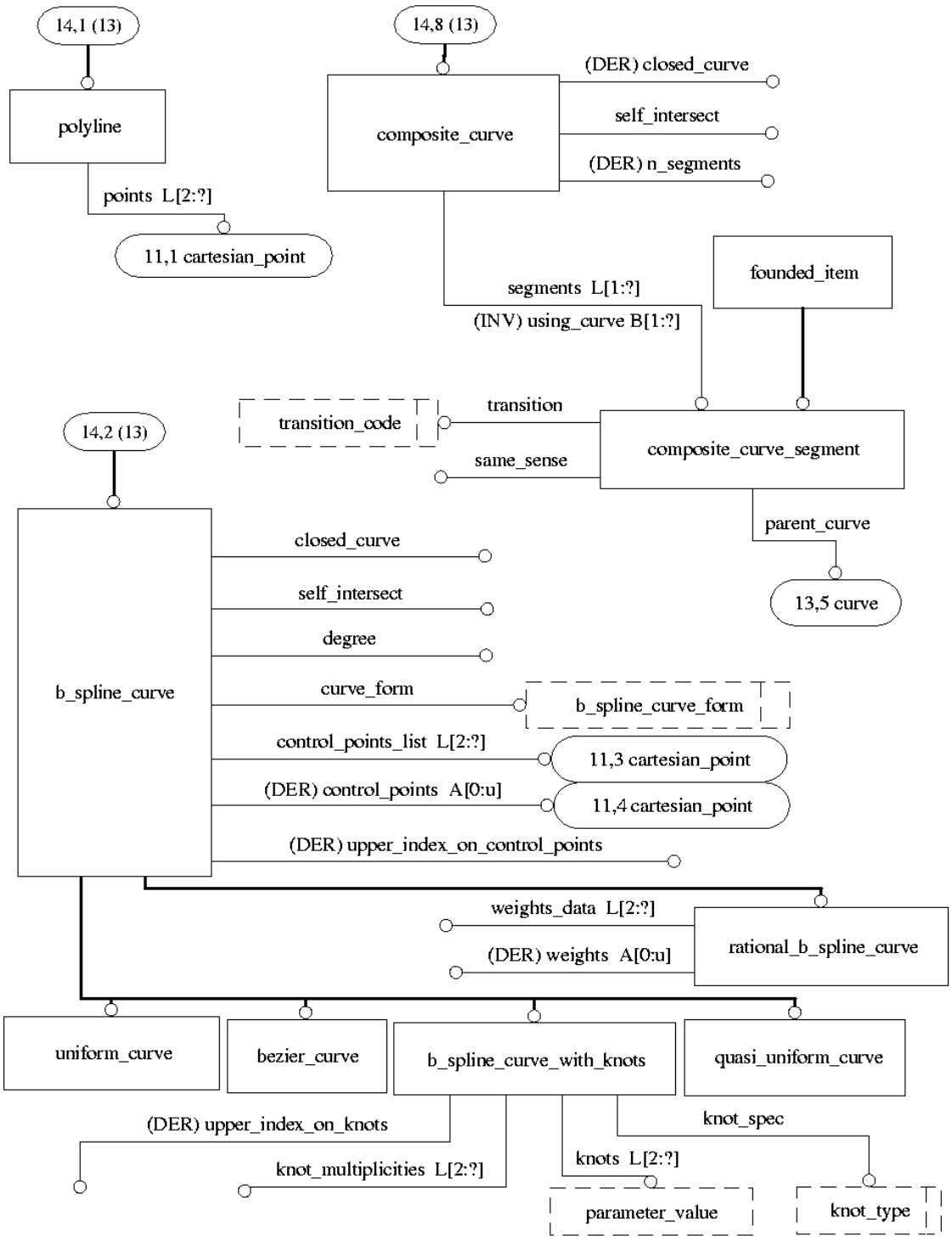


Figure H.14 - bounded curve - AIM EXPRESS-G diagram 14 of 37

©ISO 2000 - All rights reserved

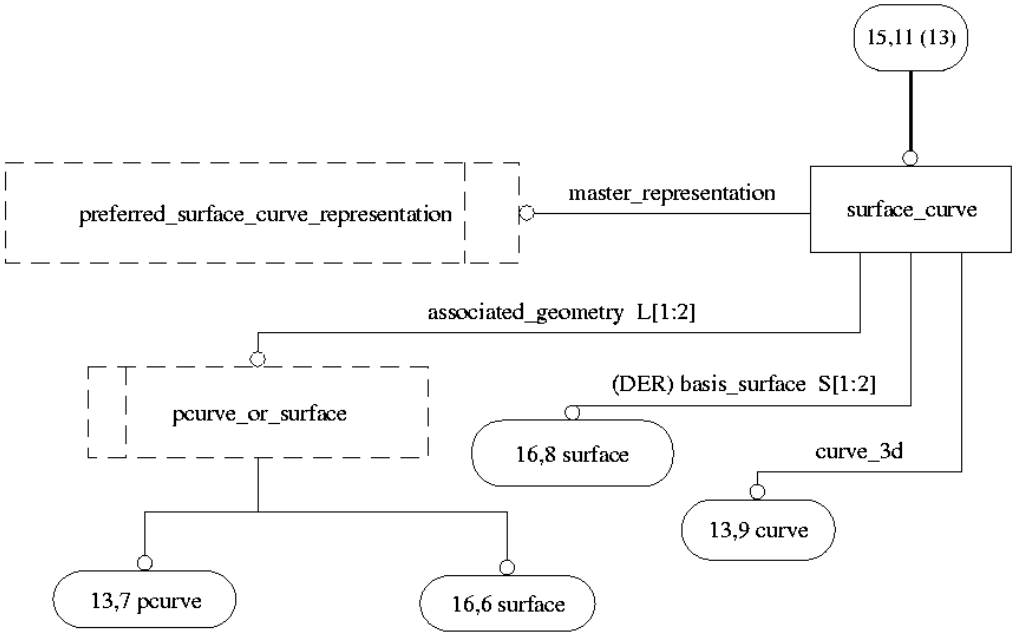


Figure H.15 - surface curve - AIM EXPRESS-G diagram 15 of 37

 ©ISO 2000 - All rights reserved

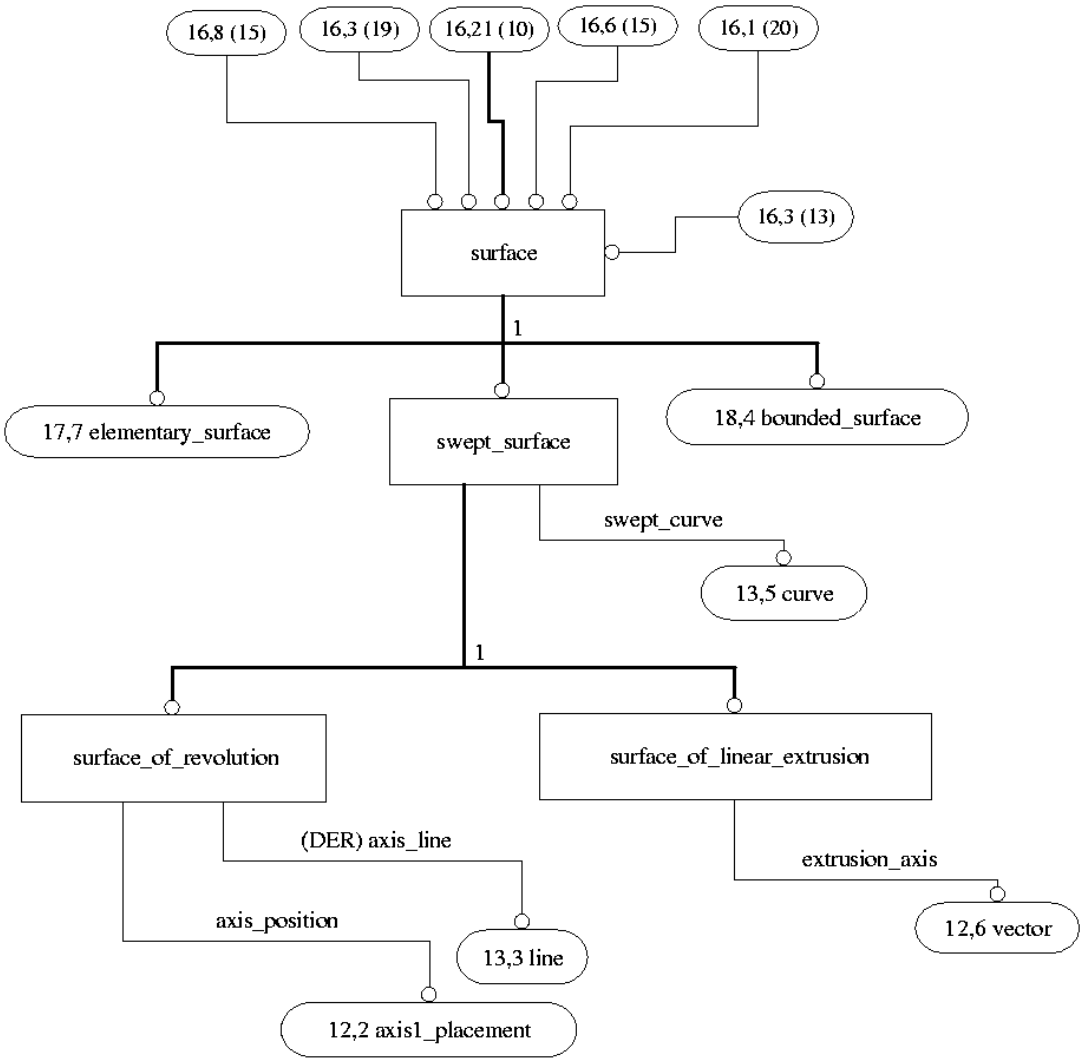


Figure H.16 - surface - AIM EXPRESS-G diagram 16 of 37

©ISO 2000 - All rights reserved

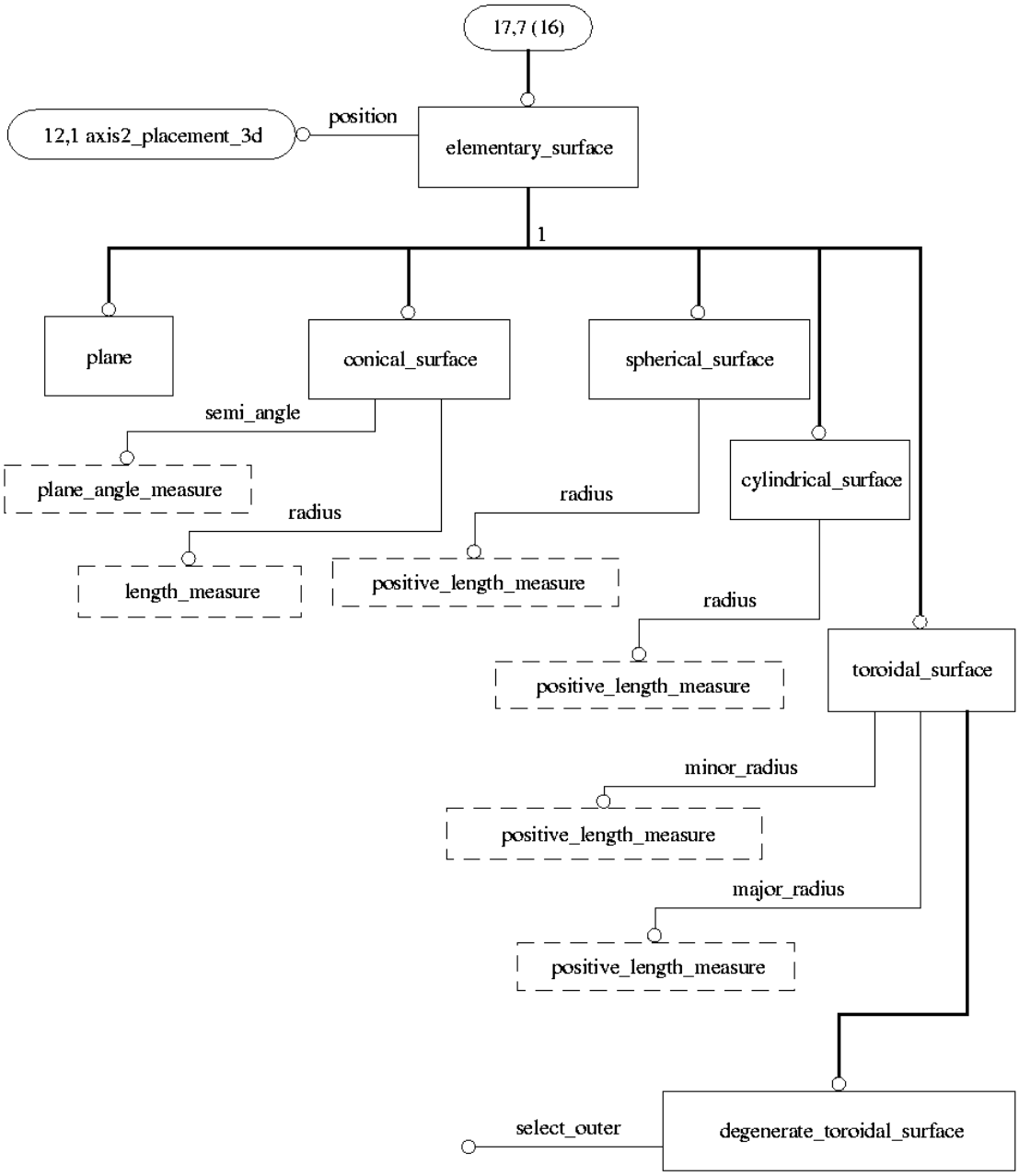


Figure H.17 - elementary surface - AIM EXPRESS-G diagram 17 of 37

 ©ISO 2000 - All rights reserved

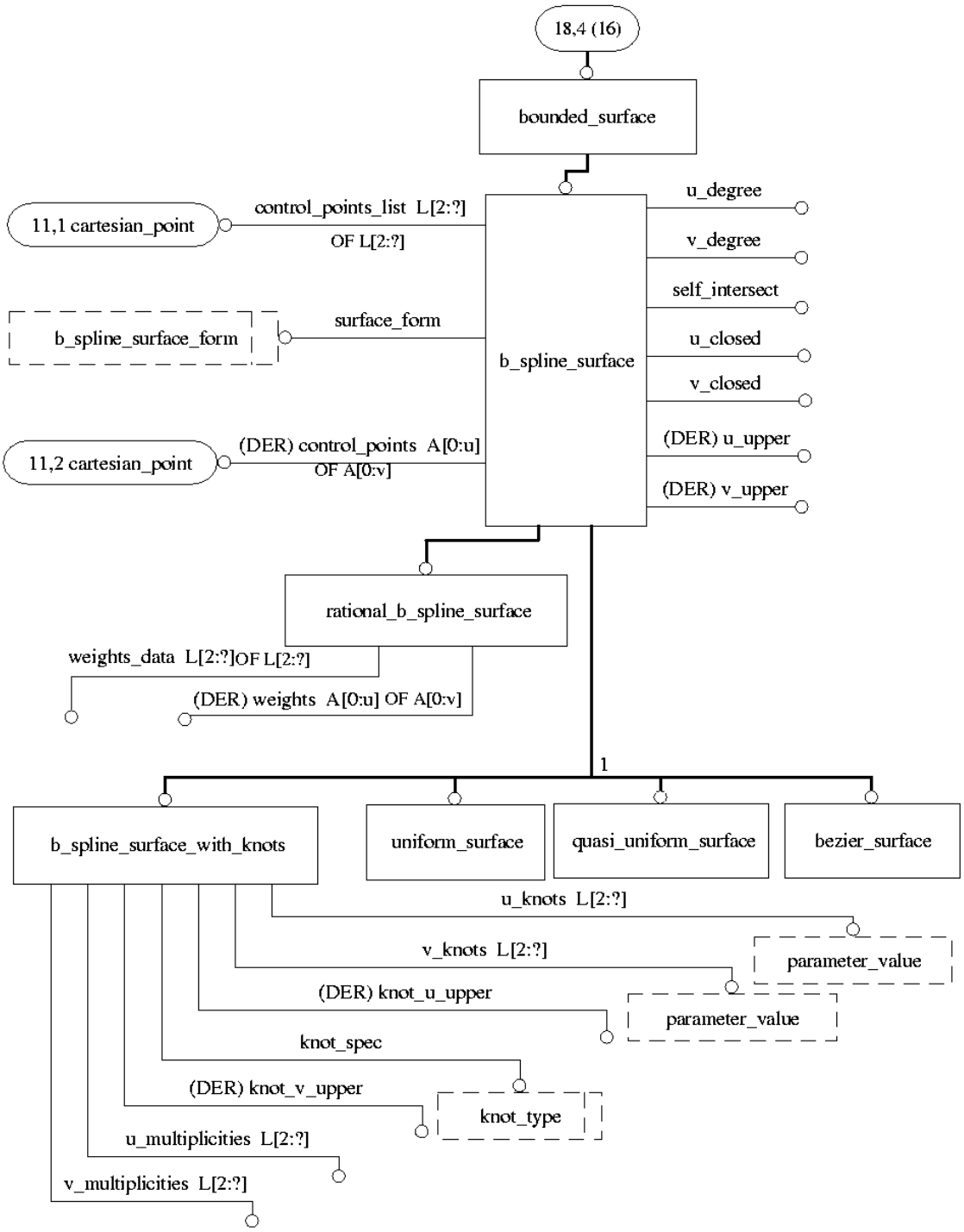


Figure H.18 - bounded surface - AIM EXPRESS-G diagram 18 of 37

©ISO 2000 - All rights reserved

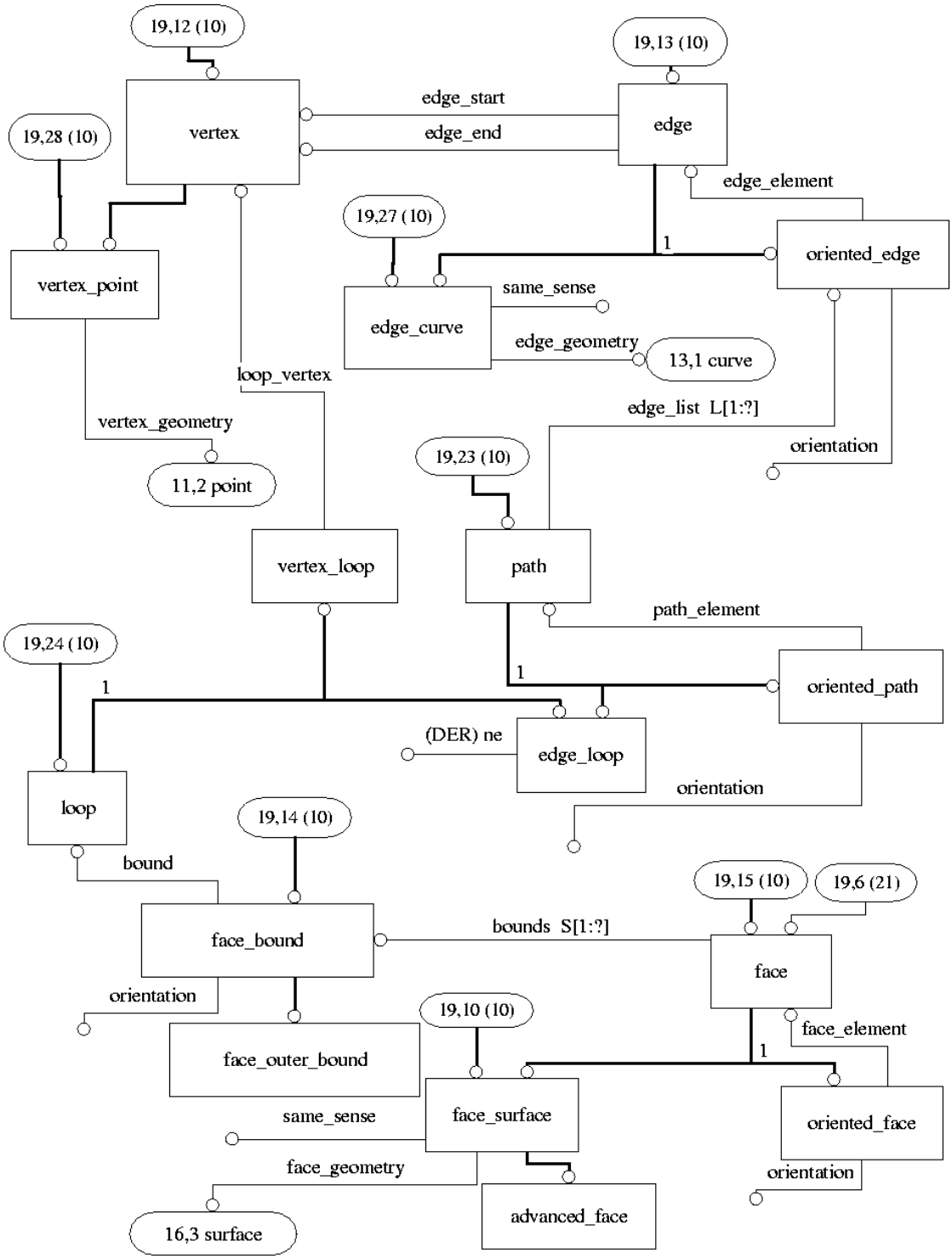


Figure H.19 - topology - AIM EXPRESS-G diagram 19 of 37

 ©ISO 2000 - All rights reserved

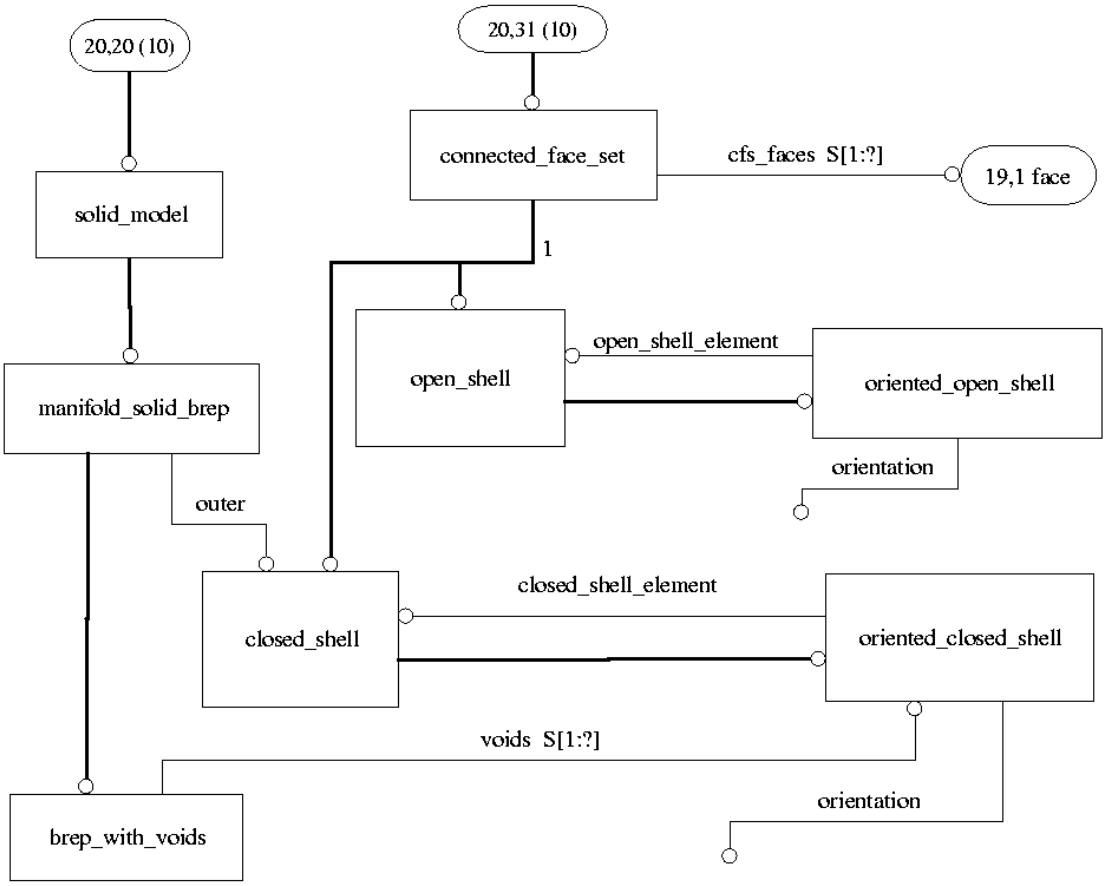


Figure H.20 - shell - AIM EXPRESS-G diagram 20 of 37

©ISO 2000 - All rights reserved

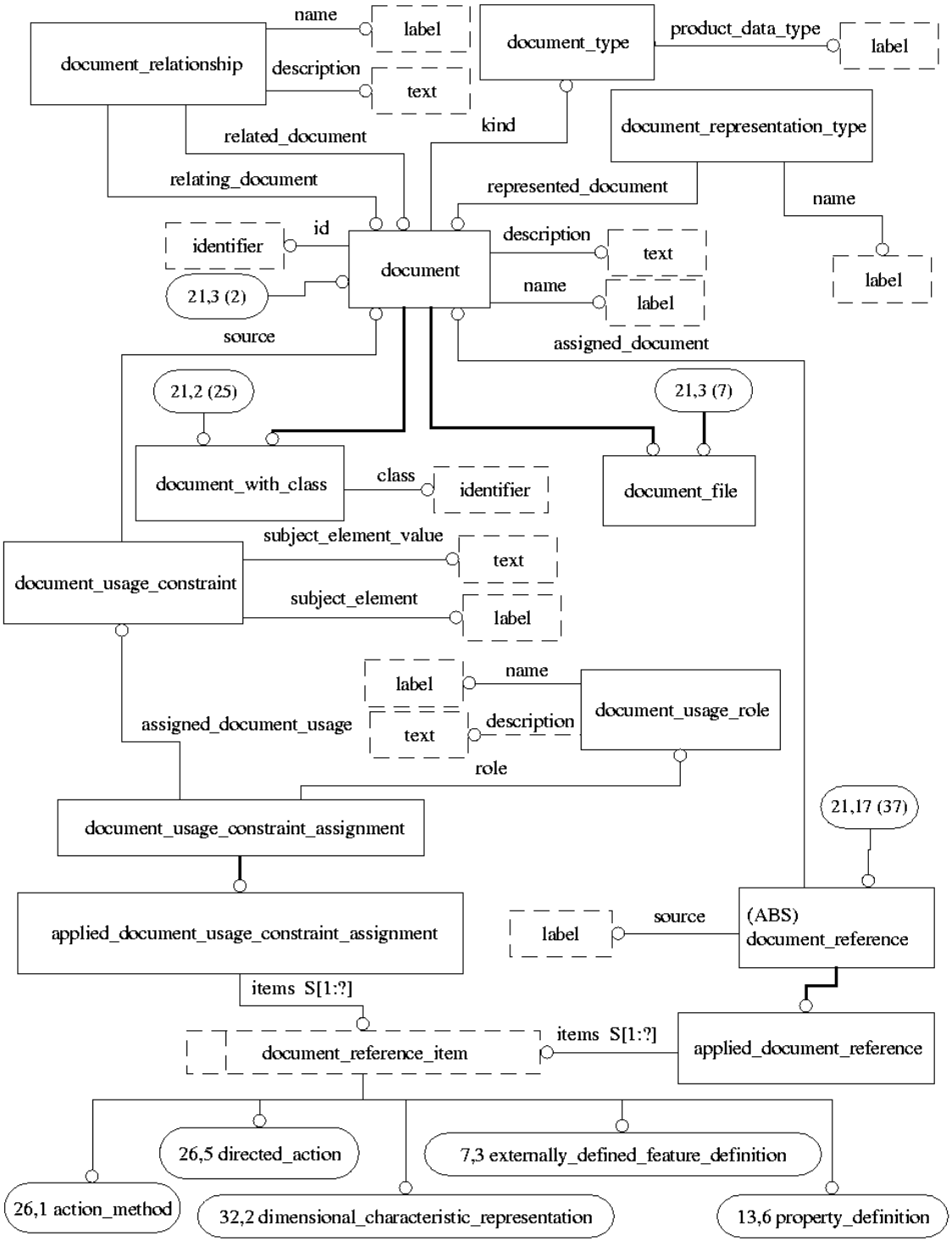


Figure H.21 - document - AIM EXPRESS-G diagram 21 of 37

 ©ISO 2000 - All rights reserved

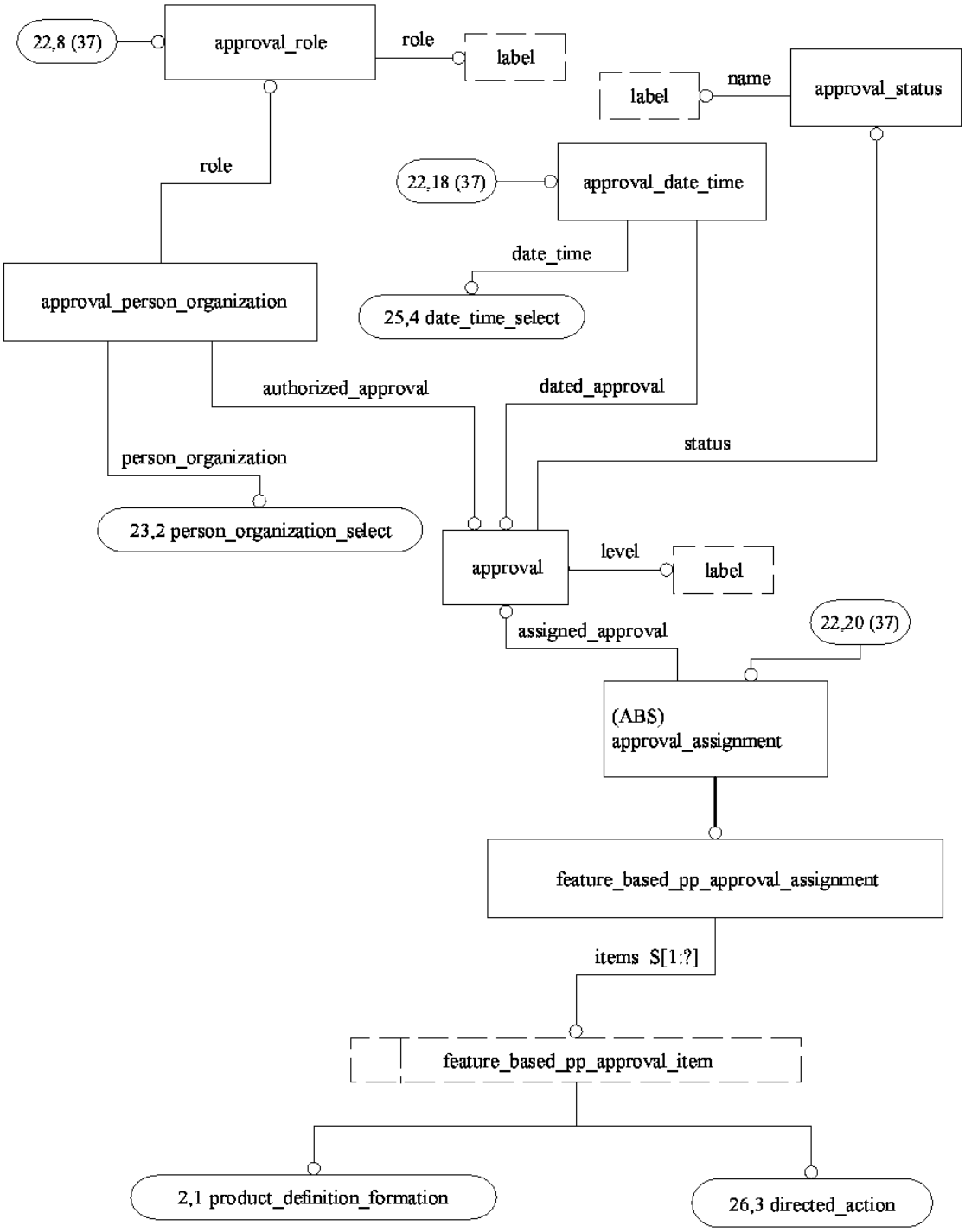


Figure H.22 - approval - AIM EXPRESS-G diagram 22 of 37

©ISO 2000 - All rights reserved

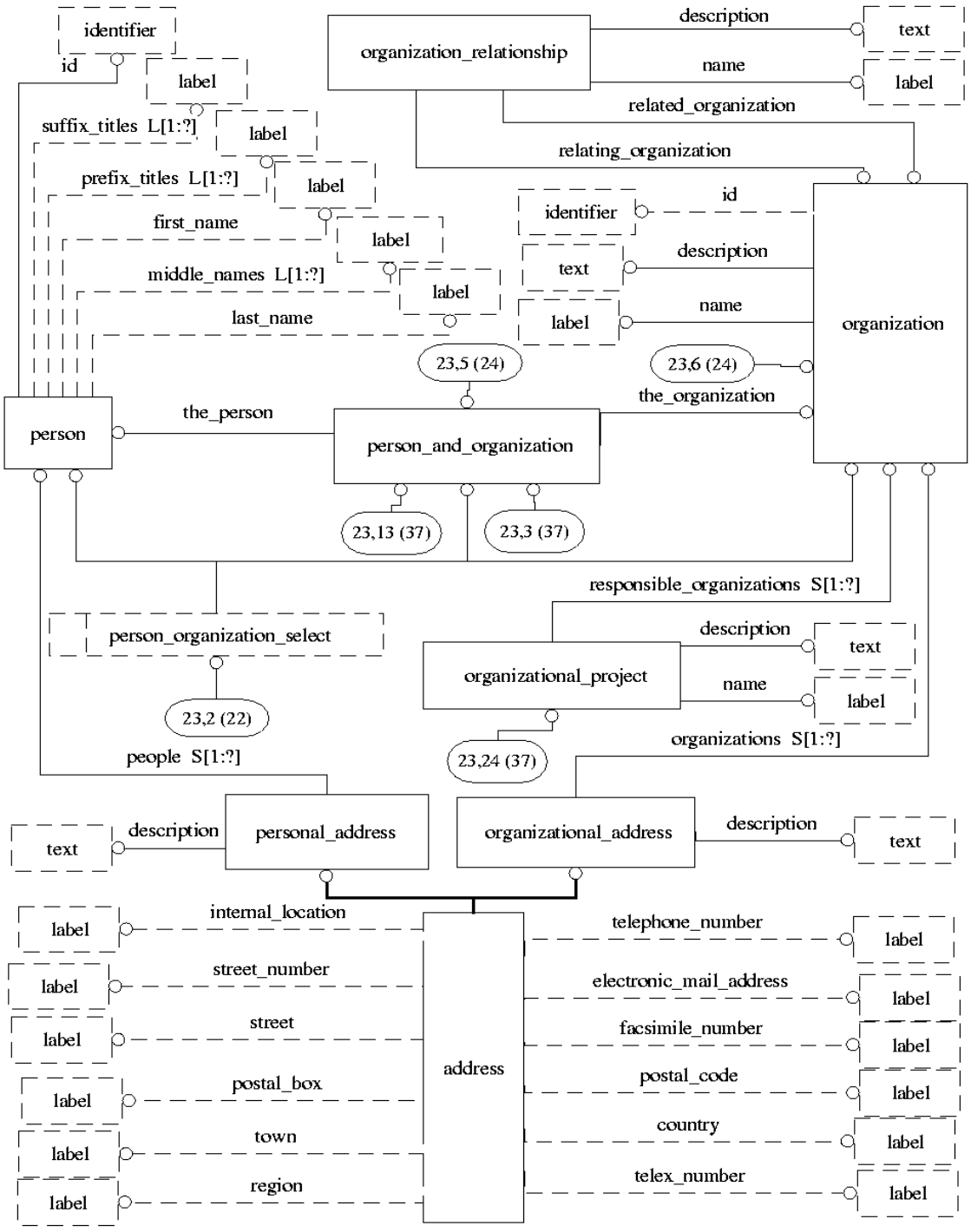


Figure H.23 - person and organization - AIM EXPRESS-G diagram

23 of 37

 ©ISO 2000 - All rights reserved

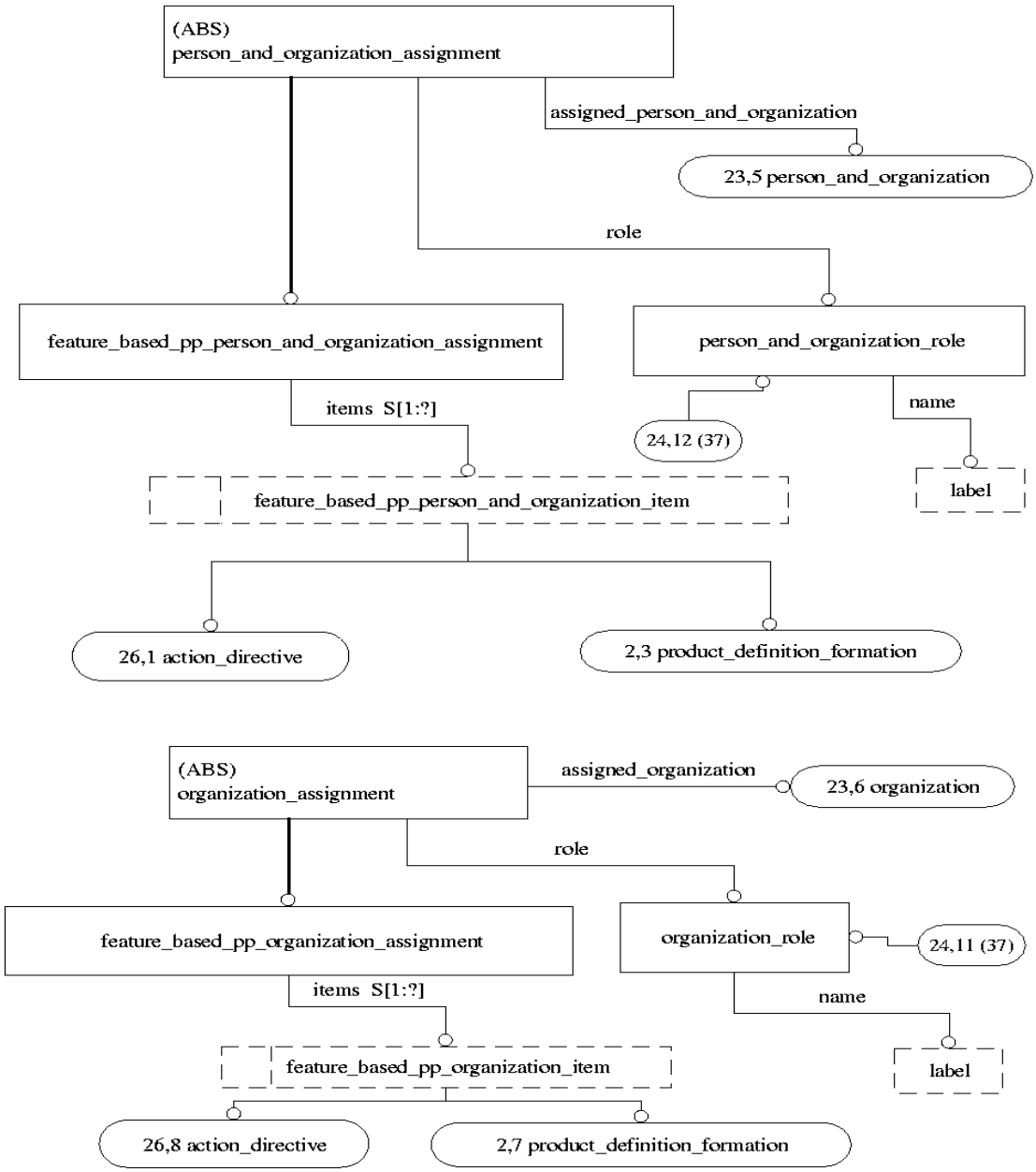


Figure H.24 - person and organization assignment - AIM EXPRESS-G diagram 24 of 37

©ISO 2000 - All rights reserved

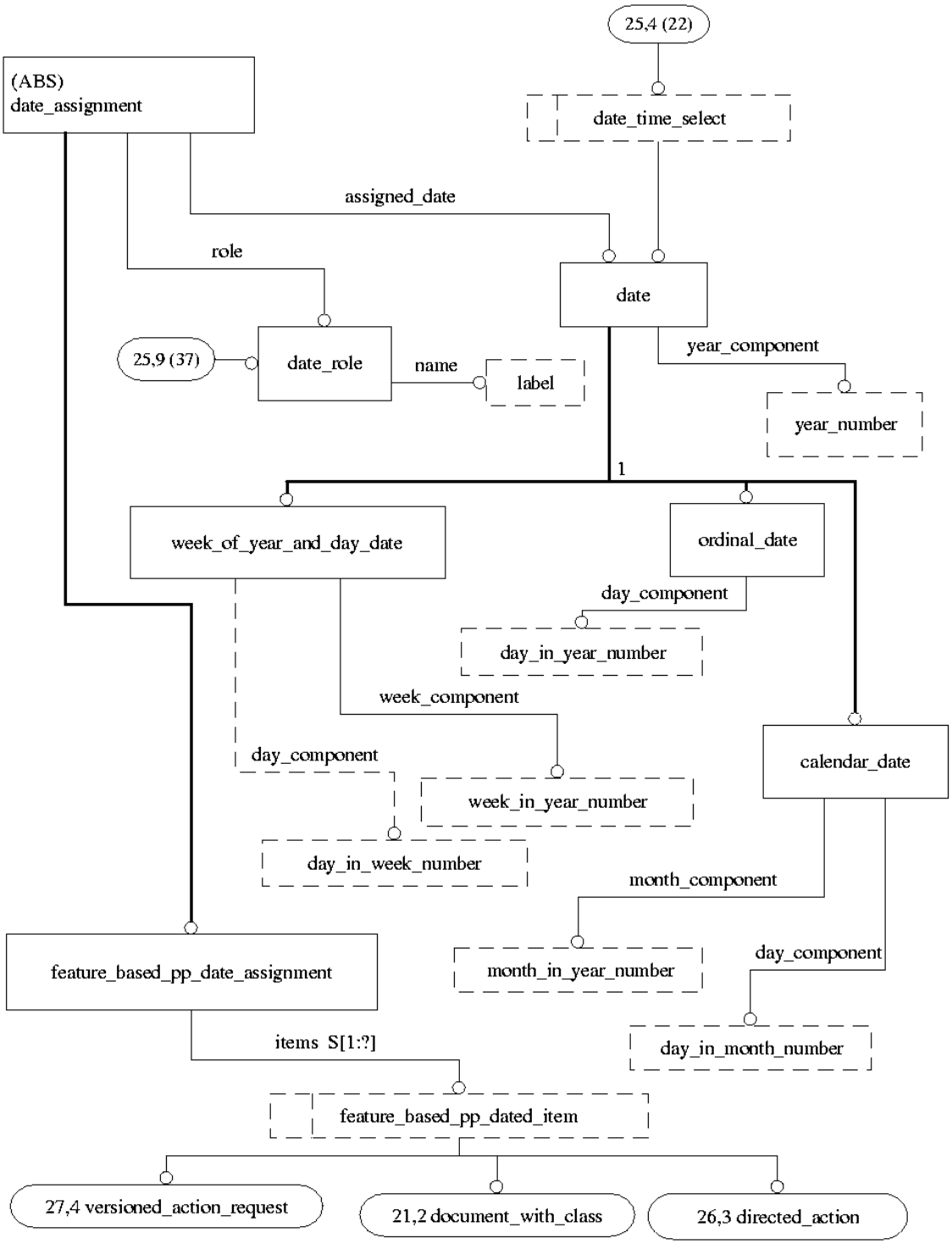


Figure H.25 - date - AIM EXPRESS-G diagram 25 of 37

 ©ISO 2000 - All rights reserved

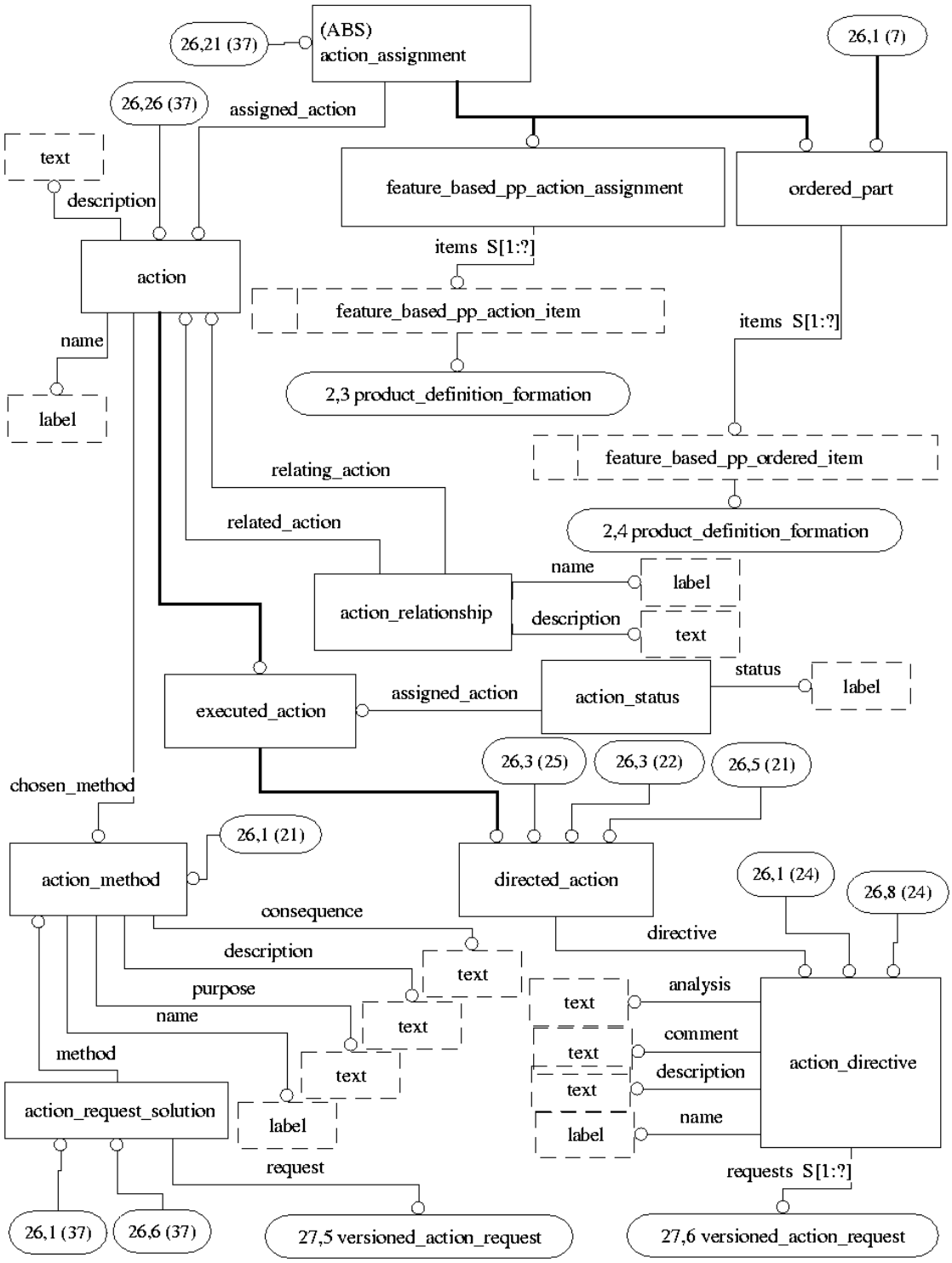


Figure H.26 - action - AIM EXPRESS-G diagram 26 of 37

©ISO 2000 - All rights reserved

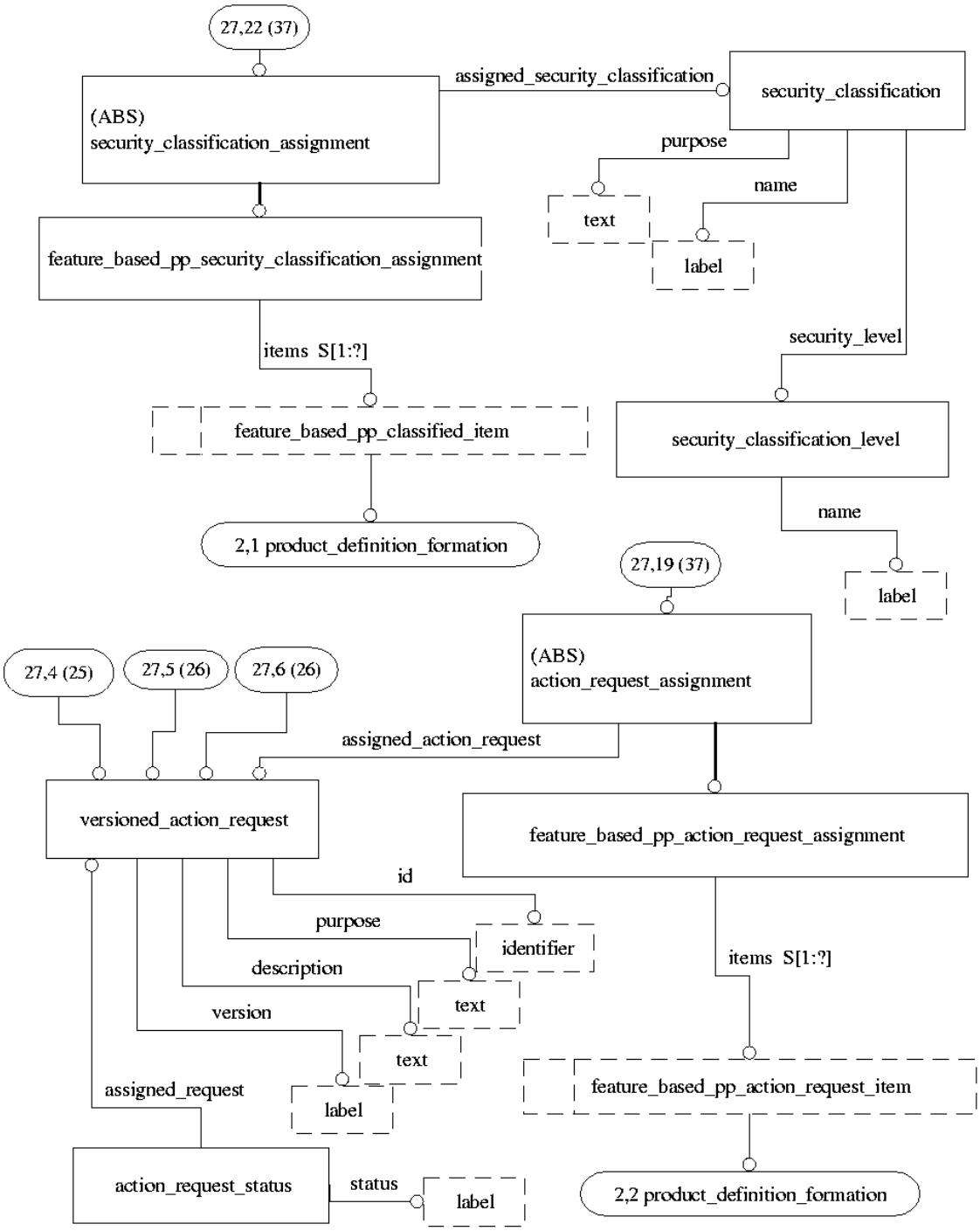


Figure H.27 - security classification - AIM EXPRESS-G diagram 27 of 37

 ©ISO 2000 - All rights reserved

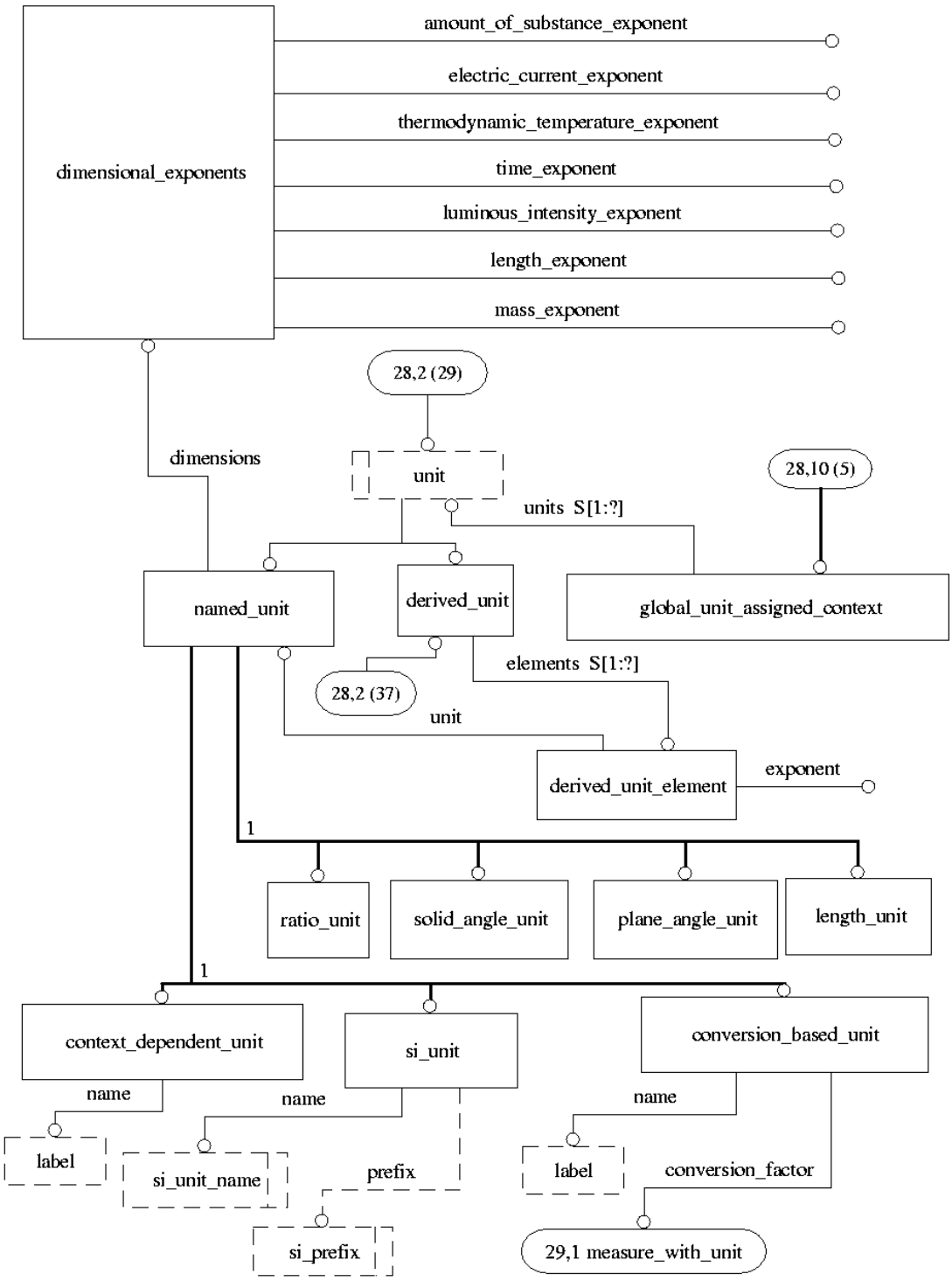


Figure H.28 - units - AIM EXPRESS-G diagram 28 of 37

©ISO 2000 - All rights reserved

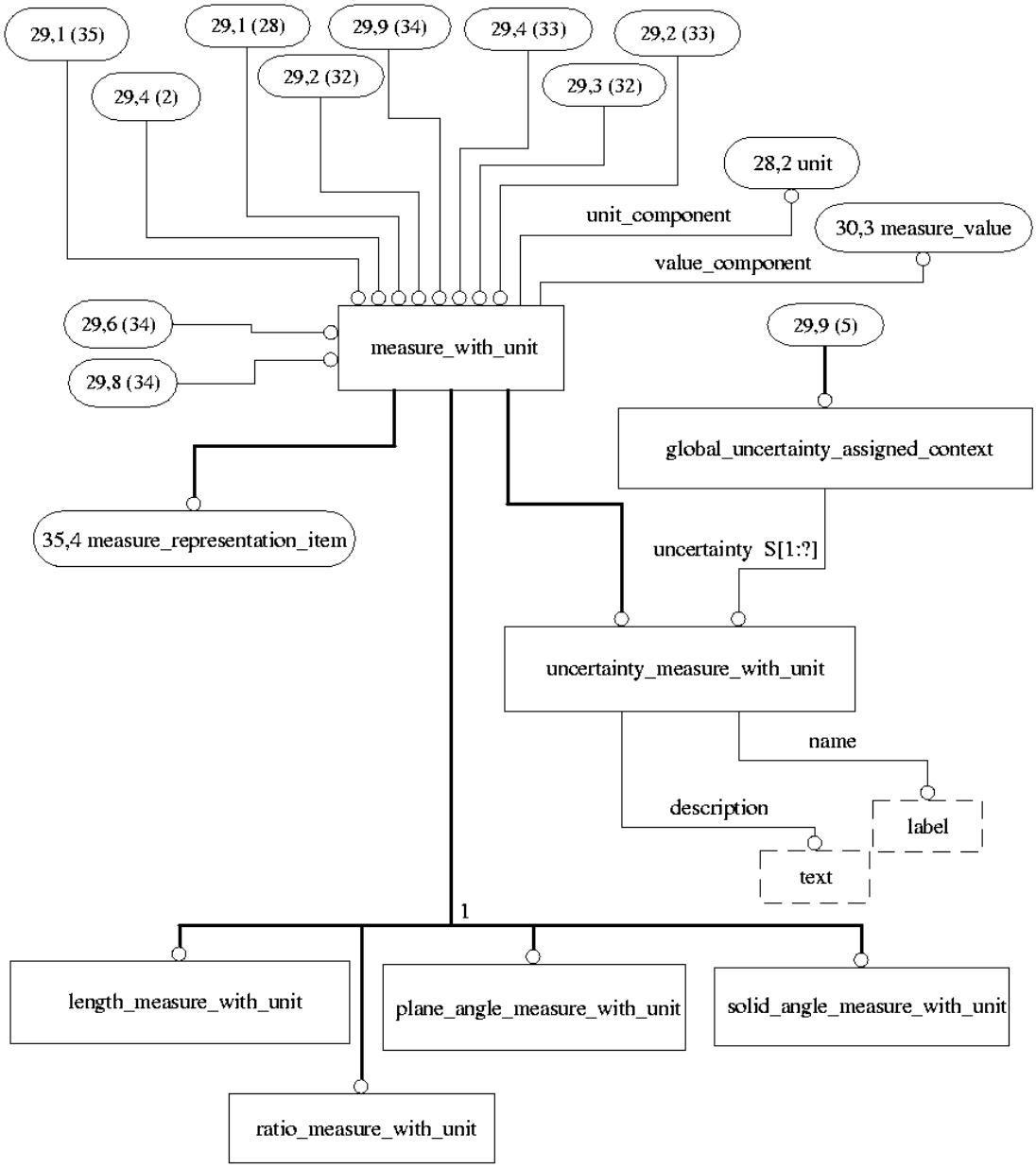


Figure H.29 - measure with units - AIM EXPRESS-G diagram 29 of 37

 ©ISO 2000 - All rights reserved

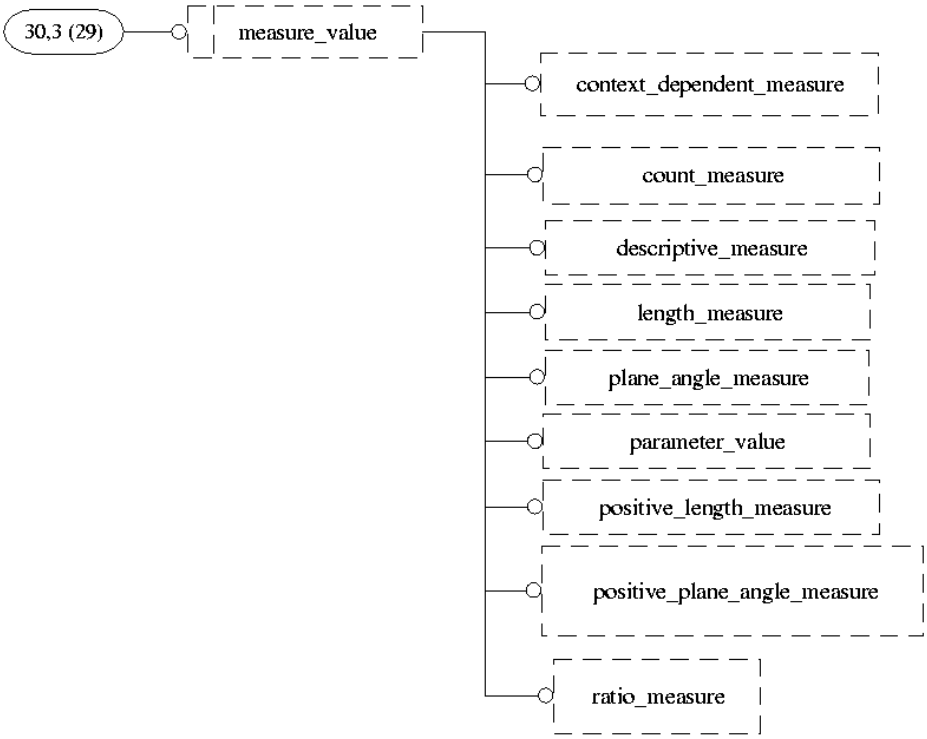


Figure H.30 - measures - AIM EXPRESS-G diagram 30 of 37

©ISO 2000 - All rights reserved

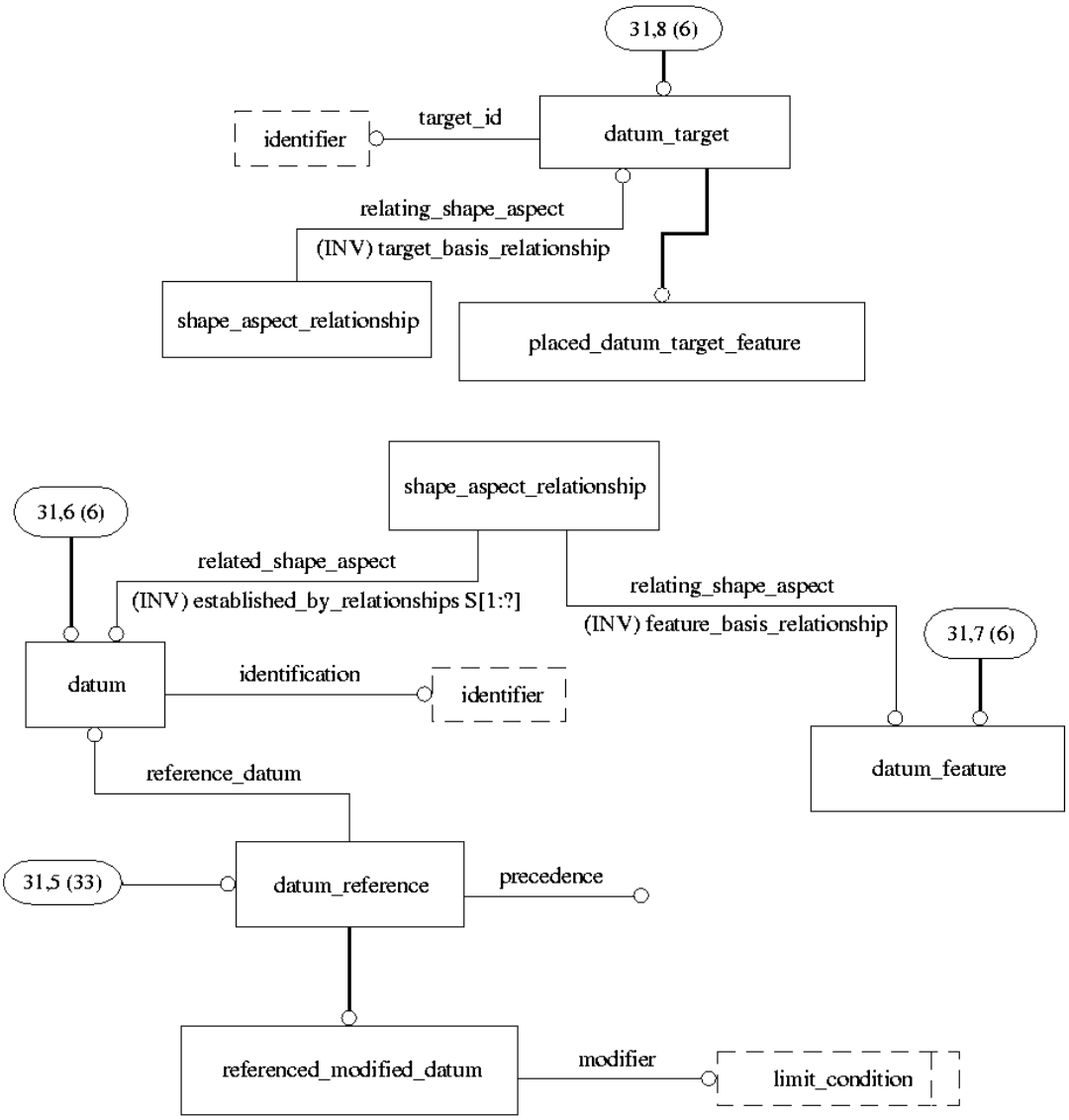


Figure H.31 - datums - AIM EXPRESS-G diagram 31 of 37

 ©ISO 2000 - All rights reserved

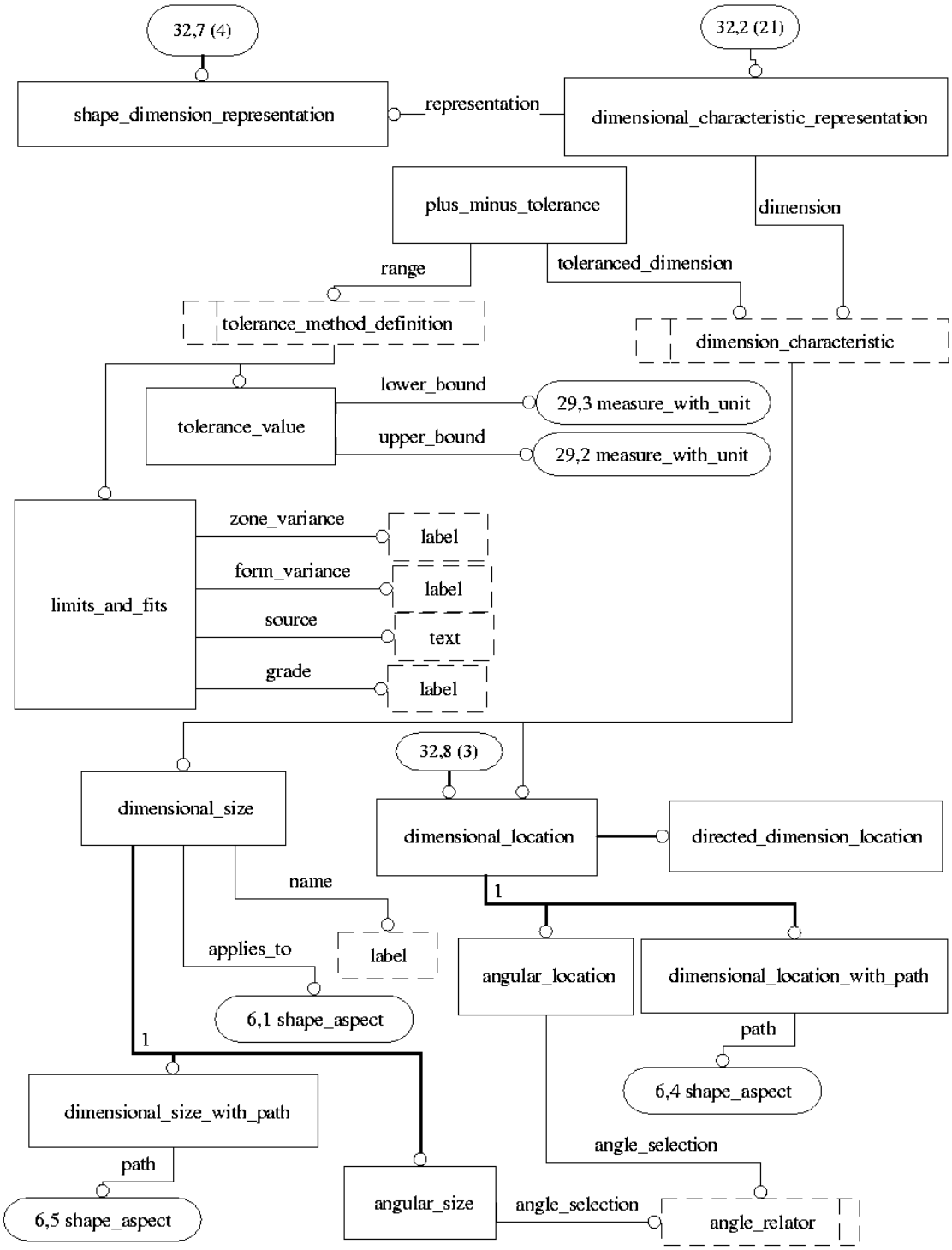


Figure H.32 - tolerances - AIM EXPRESS-G diagram 32 of 37

©ISO 2000 - All rights reserved

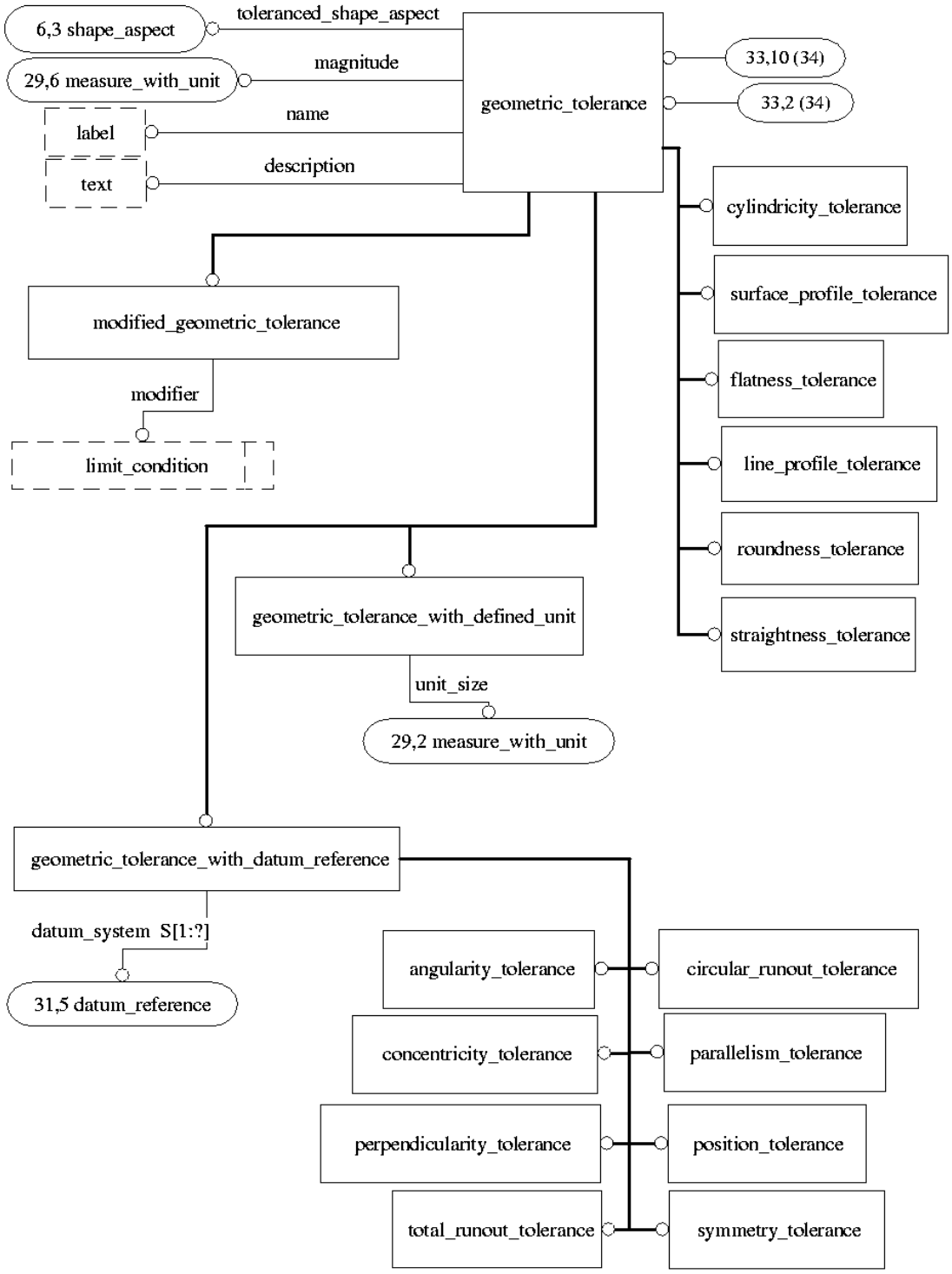
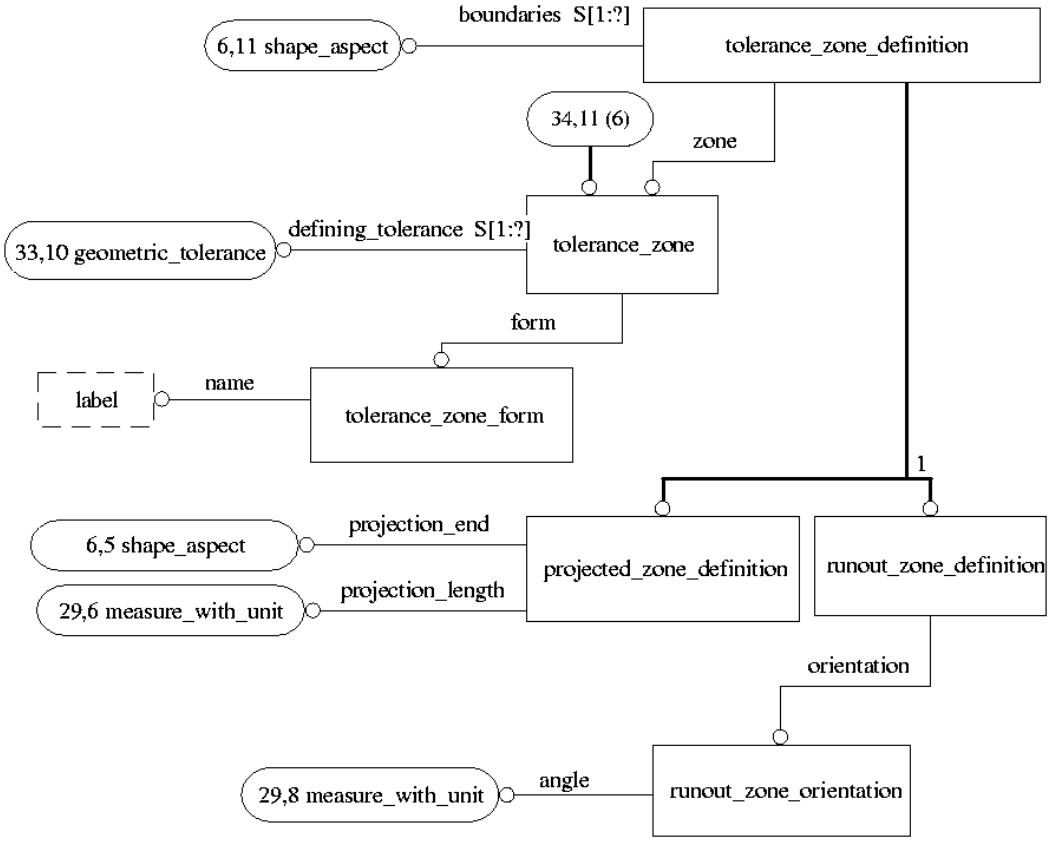


Figure H.33 - geometric tolerances - AIM EXPRESS-G diagram 33 of 37

 ©ISO 2000 - All rights reserved



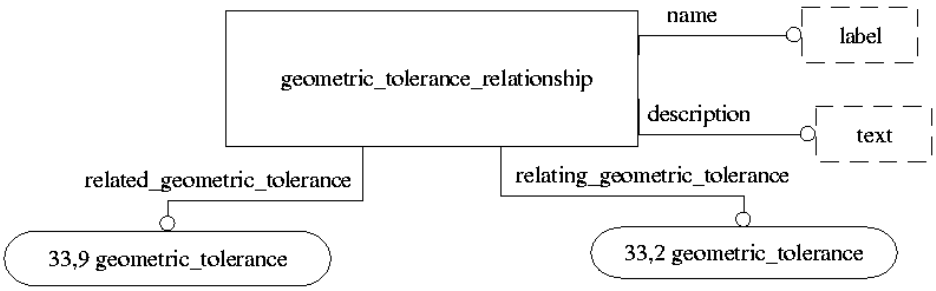


Figure H.34 - tolerance zone - AIM EXPRESS-G diagram 34 of 37

©ISO 2000 - All rights reserved

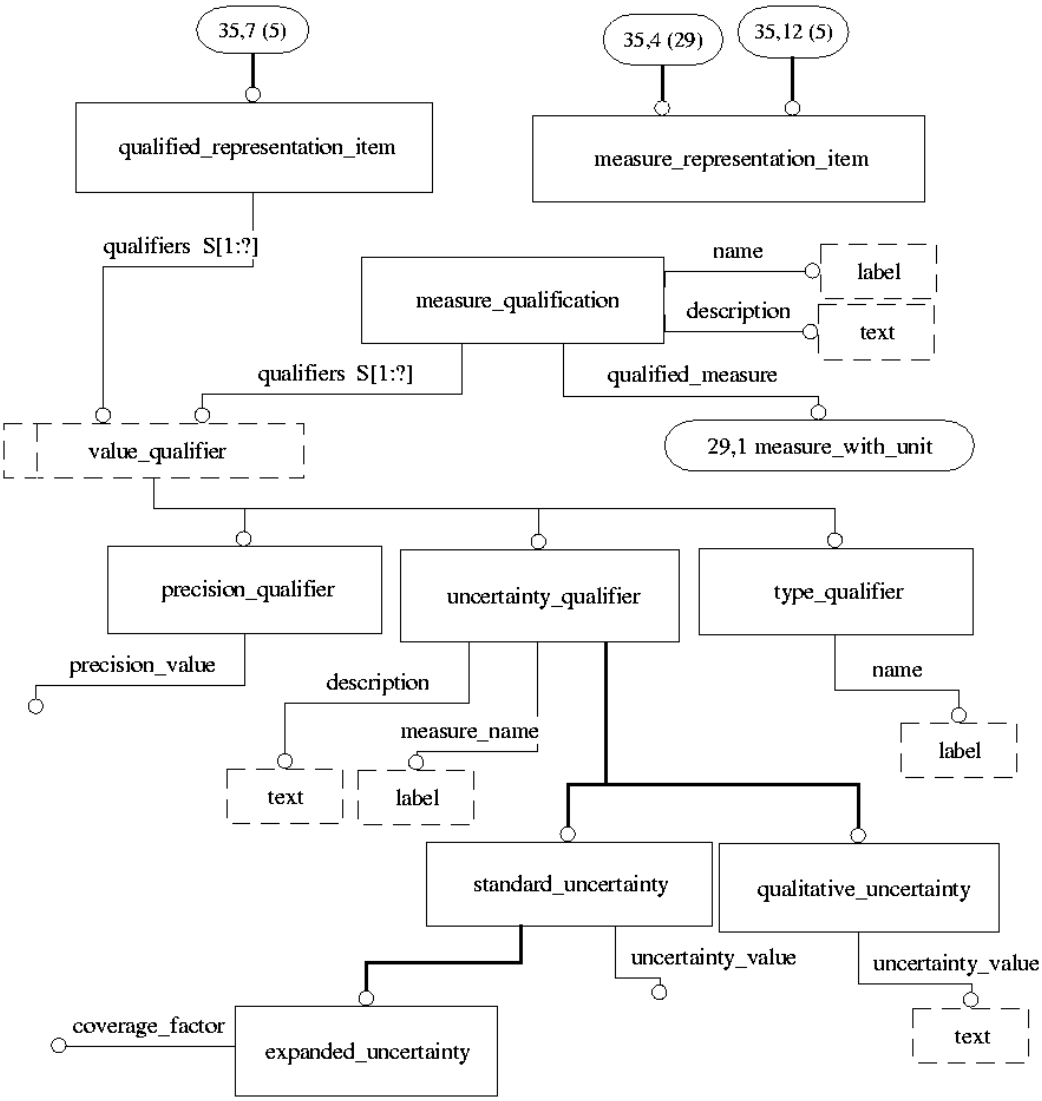


Figure H.35 - materials - AIM EXPRESS-G diagram 35 of 37

 ©ISO 2000 - All rights reserved

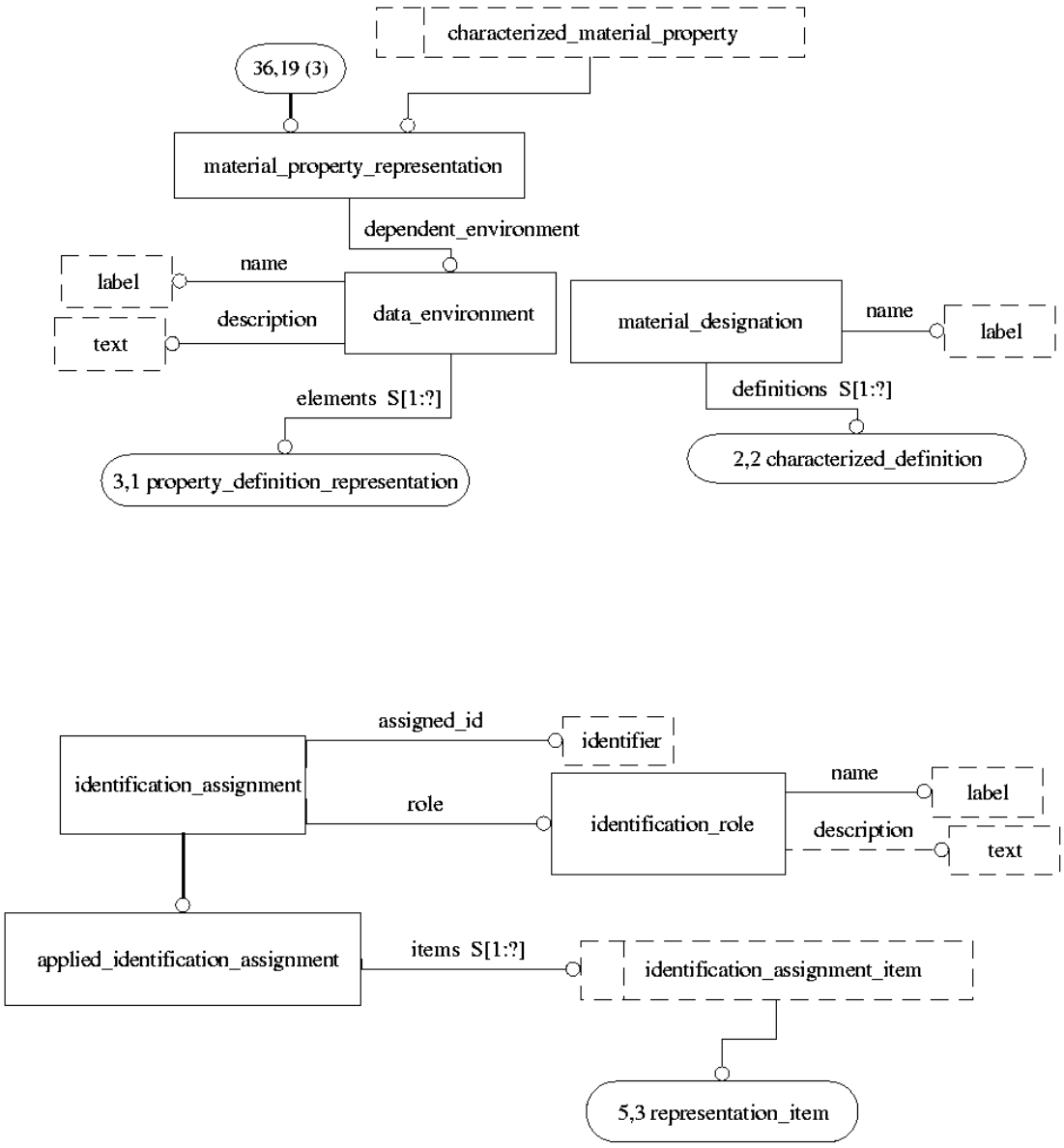


Figure H.36 - material property and identification - AIM EXPRESS-G diagram 36 of 37

©ISO 2000 - All rights reserved

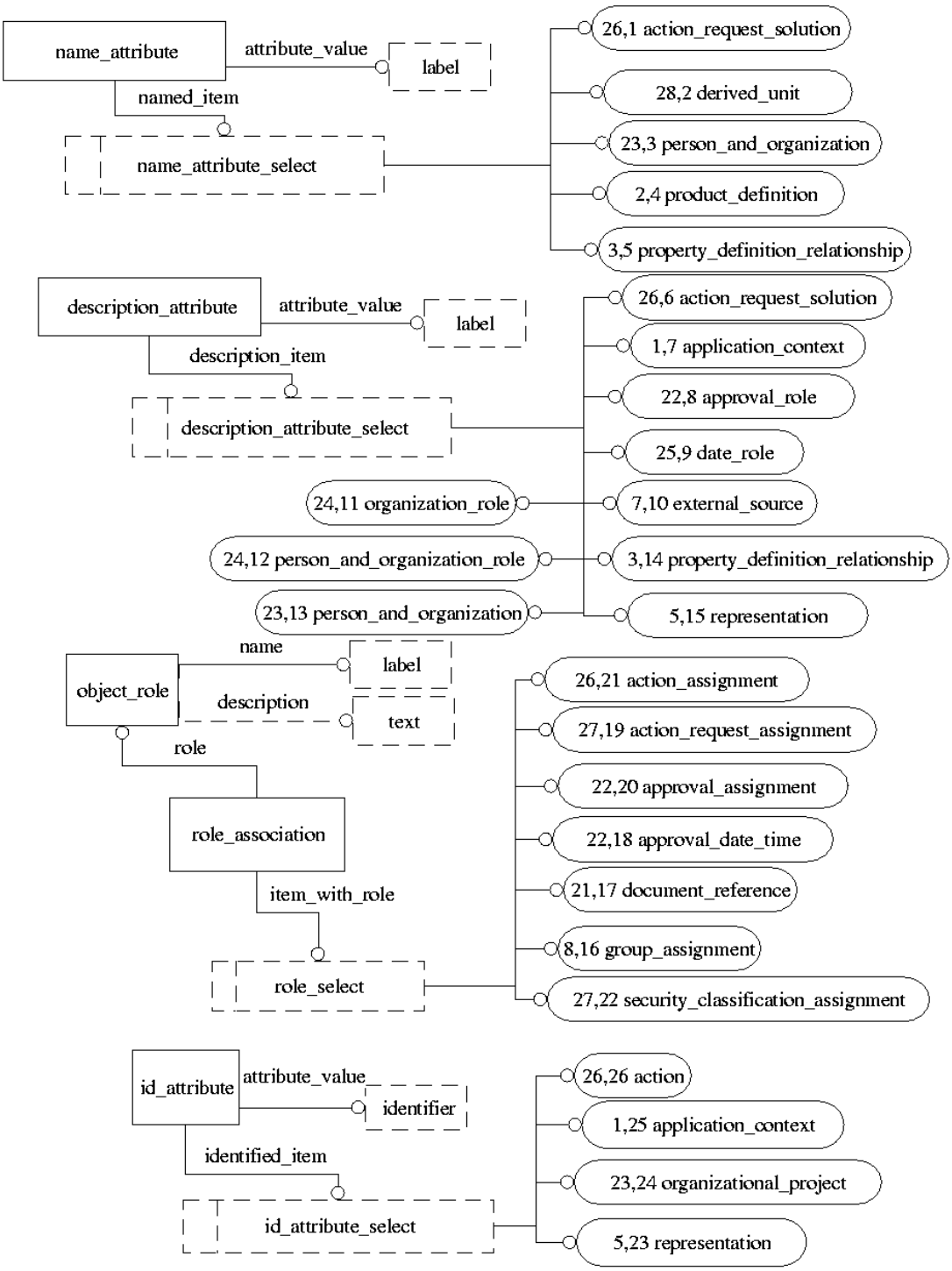


Figure H.37 - attributes - AIM EXPRESS-G diagram 37 of 37

 ©ISO 2000 - All rights reserved

Annex J
(informative)

Computer interpretable listings

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

EXPRESS: http://www.mel.nist.gov/step/parts/part224/is/
Short names: http://www.mel.nist.gov/div826/subject/apde/snr/

If there is difficulty accessing these sites contact ISO Central Secretariat or contact the ISO TC 184/SC4 Secretariat directly at: sc4sec@cme.nist.gov.

NOTE - The information provided in computer-interpretable form at the above URLs is informative. The information that is contained in the body of this part of ISO 10303 is normative.

©ISO 2000 - All rights reserved

Annex K
(informative)

Application protocol usage guide

The following clauses provide the usage/functional test purposes. The objective of these test purposes is to test the functionality of an implementation, that is the ability of an implementation to be able to provide the correct results of complex queries combining a number of entities and attributes in order to obtain an answer to a real world question.

The test purposes are located in ISO 10303-324, the Abstract Test Suite for ISO 10303-224. They are available from the ISO Central Secretariat.

 ©ISO 2000 - All rights reserved

Annex L
(informative)

Technical discussion

This annex contains the summary of technical changes for the second edition of this part. Table L.1 summarizes the changes of the ARM EXPRESS-G model (see annex G), Application objects (see 4.2), and the Application assertions (see 4.3). Table L.2 summarizes the changes of the AIM EXPRESS short listing (see 5.2).

Table L.1 is organized into 6 columns.

Column 1) Application element: Name of an application element as it appears in the application object definition in 4.2., and the ARM EXPRESS-G model in annex G.

Column 2) Attribute: Name of an application element attribute, if applicable, as it appears in the application object definition in 4.2., and the ARM EXPRESS-G model in annex G.

Column 3) Status: Description of the change made in this second edition part. Types of changes include: new, modified, or removed application elements or attributes.

Column 4) Upward Compatible: Indicates if the change is or is not upward compatible. 'Yes' indicates the changes is upward compatible, 'no' indicates the changes in not upward compatible.

Column 5) Annex G diagram: The reference to the Annex G diagram that shows the application element or attribute.

Column 6) Comments: General summary of the type of change made to the application element or attribute for the 2nd edition of this part.

Table L.2 is organized into 6 columns.

Column 1) AIM element: Name of an element as it appears in the application object definition in 5.2., AIM EXPRESS expanded listing in annex A, and the AIM EXPRESS-G model in annex H.

Column 2) Status: Description of the change made in this second edition part. Types of changes include: new, modified, or removed AIM elements, types, rules, or functions.

Column 3) 1st edition clause: Clause number where the AIM element can be found in the ISO 10303-224:1999 edition.

Column 4) 2nd edition clause: Clause number where the AIM element can be found in this second edition part.

Column 5) Upward Compatible: Indicates if the change is or is not upward compatible. 'Yes' indicates the changes is upward compatible, 'no' indicates the changes in not upward compatible.

Column 6) Comments: General summary of the type of change made to the application element or attribute for the 2nd edition of this part.

©ISO 2000 - All rights reserved

Table L.1 Summary of AP224 2nd edition ARM changes

Application Entity	Attribute	Status	Upward Compatible	Annex G diagram	Comments
Engineering_change_order	incorporated_proposal	Moved attribute	Yes, AIM maps the same	G.1	attribute moved to Engineering change proposal
Engineering_change_proposal	incorporated_proposal	New attribute	Yes, AIM maps the same	G.1	
Part		New sub-types	Yes	G.1	1st edition parts are upward compatible to 2nd edition
Manufactured_assembly		New entity	Yes	G.2	
Manufactured_assembly_-relationship		New entity	Yes	G.2	
Single_piece_part		New entity	Yes	G.2	Part is now an ABSTRACT SUPERTYPE. Single_piece_part equates to Part in 1st edition
Mating_definition		New entity	Yes	G.2	
Mating_definition_relationship		New entity	Yes	G.2	
Mating_relationship		New entity	Yes	G.2	
Part Placement		New entity	Yes	G.2	
Specification	specification_-description	Modified	No (Note: AIM remapped)	G.5	attribute changed to OPTIONAL to harmonize with other Aps. AIM remapped to harmonize with other AP.
	specification_class	Modified	Yes	G.5	attribute changed to OPTIONAL to harmonize with other Aps. AIM remapped to harmonize with other AP.

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table L.1 Summary of AP224 2nd edition ARM changes (continued)

Application Entity	Attribute	Status	Upward Compatible	Annex G diagram	Comments
Property_parameter	parameter_name	Modified	No (Note: AIM remapped)	G.6	attribute changed to OPTIONAL to harmonize with other Aps. AIM remapped to harmonize with other AP.
Ngon_base_shape	circumscribed_diameter	Modified attribute	Yes, AIM mapping not effected.	G.7	Attribute changed to diameter
	circumscribed_or_-across_flats	New attribute	Yes	G.7	Attribute added to support two different ways to define the diameter value
Shape	base_shape_definition	Modified	Yes	G.7	attribute changed to OPTIONAL to harmonize with other Aps. AIM remapped to harmonize with other AP.
Planar_element		New entity	Yes	G.8	
Face_shape_element_relationship		New entity	Yes	G.8	
Manufacturing_feature_group		New entity	Yes	G.9	
Multi_axis_feature		New entity	Yes	G.10	entity created to add a unique attribute used by multi axis processes
	maximum_feature_-limit	New attribute	Yes	G.10	attribute changed to OPTIONAL to harmonize with other Aps. AIM remapped to harmonize with other AP.
Constant_radius_edge_round	first_face_offset	Modified	Yes	G.11	attribute changed to OPTIONAL to harmonize with other Aps. AIM remapped to harmonize with other AP.

Table L.1 Summary of AP224 2nd edition ARM changes (continued)

Application Entity	Attribute	Status	Upward Compatible	Annex G diagram	Comments
	second_face_offset	Modified	Yes	G.11	attribute changed to OPTIONAL to harmonize with other Aps. AIM remapped to harmonize with other AP.
Constant_radius_fillet	first_face_offset	Modified	Yes	G.11	attribute changed to OPTIONAL to harmonize with other Aps. AIM remapped to harmonize with other AP.
	second_face_offset	Modified	Yes	G.11	attribute changed to OPTIONAL to harmonize with other Aps. AIM remapped to harmonize with other AP.
Pocket		Modified entity	Yes	G.12	New sub-types added.
	base_radius	New attribute	Yes	G.12	
	change_in_boundary	Modified attribute	Yes	G.12	additional taper type added
Recess		New entity	Yes	G.12	
Cutout		New entity	Yes	G.12	
Circular_cutout		New entity	Yes	G.12	
General_cutout		New entity	Yes	G.12	
Rib_top		New entity	Yes	G.13	
Rib_top_floor		New entity	Yes	G.13	
Planar_rib_top_floor		New entity	Yes	G.13	
General_rib_top_floor		New entity	Yes	G.13	

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table L.1 Summary of AP224 2nd edition ARM changes (continued)

Application Entity	Attribute	Status	Upward Compatible	Annex G diagram	Comments
Planar_pocket_bottom_condition	floor_radius	Modified	Yes	G.13	attribute changed to OPTIONAL to harmonize with other Aps. AIM remapped to harmonize with other AP.
General_pocket_bottom_condition	floor_radius	Modified	Yes	G.13	attribute changed to OPTIONAL to harmonize with other Aps. AIM remapped to harmonize with other AP.
Profile_feature		New entity	Yes	G.14	
General_outside_profile		Modified entity	Yes	G.14	Now a sub-type of Profile_feature
Shape_profile		New entity	Yes	G.14	
General_shape_profile		New entity	Yes	G.14	
Partial_circular_shape_profile		New entity	Yes	G.14	
Circular_closed_shape_profile		New entity	Yes	G.14	
Rectangular_open_shape_profile		New entity	Yes	G.14	
Rectangular_closed_shape_-profile		New entity	Yes	G.14	
Profile_floor		New entity	Yes	G.14	
General_profile_floor		New entity	Yes	G.14	
Planar_profile_floor		New entity	Yes	G.14	
Through_profile_floor		New entity	Yes	G.14	
Planar_face	face_boundary	New attribute	Yes	G.15	attribute OPTIONAL
Blind_bottom_condition		Modified entity	Yes	G.16	New sub-type added.

Table L.1 Summary of AP224 2nd edition ARM changes (continued)

Application Entity	Attribute	Status	Upward Compatible	Annex G diagram	Comments
Flat_with_taper_hole_bottom		New entity	Yes	G.16	
Conical_hole_bottom	tip_radius	Modified	Yes		attribute changed to OPTIONAL to harmonize with other Aps. AIM remapped to harmonize with other AP.
Boss		Modified entity	Yes	G.17	New sub-type added.
	fillet_radius	Modified	Yes		attribute changed to OPTIONAL to harmonize with other Aps. AIM remapped to harmonize with other AP.
	top_radius	New attribute	Yes	G.17	attribute OPTIONAL
Rectangular_boss		New entity	Yes	G.17	
Slot	slot_depth	Removed attribute	No	G.18	attribute deletede to harmonize with other APs.
Thread	english_or_metric	Removed attribute	No	G.19	attribute deletede to harmonize with other APs.
	applied_area	Modified attribute	Yes	G.19	Changed attribute from Shape_element to Shape, less constraint.
	qualifier	Modified attribute	Yes	G.19	attribute changed to OPTIONAL to harmonize with other APs.
	fit_class	Modified attribute	No	G.19	attribute changed to MANDITORY to harmonize with other APs.
	partial_profile	Modified attribute	No	G.19	attribute changed to MANDITORY to harmonize with other APs.
Defined_thread	crest	New attribute	Yes	G.19	attribute OPTIONAL

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table L.1 Summary of AP224 2nd edition ARM changes (continued)

Application Entity	Attribute	Status	Upward Compatible	Annex G diagram	Comments
	pitch_diameter	Modified attribute	No	G.19	attribute changed to MANDITORY to harmonize with other APs.
Marking	applied_to_shape	Modified attribute	Yes	G.19	Changed attribute from Shape_element to Shape, less constraint.
Partial_area_definition	applied_length	Modified attribute	No	G.20	attribute changed from applied_length to effective_length
	effective_length	New attribute	Yes	G.20	attribute OPTIONAL
Knurl	applied_shape	Modified attribute	Yes	G.20	Changed attribute from Shape_element to Shape, less constraint.
Circular_pattern	base_feature_diameter	Modified_attribute	Yes	G.22	attribute changed to OPTIONAL to harmonize with other APs.
Directed_taper		New entity	Yes	G.22	
Linear_path	direction	Missing attribute	Yes	G.23	Attribute defined in AP224 1st edition AIM, but was missing in the ARM.
Rectangular_closed_profile	corner_radius	Modified attribute	Yes	G.24	attribute changed to OPTIONAL to harmonize with other APs.
Ngon_profile	corner_radius	Modified attribute	Yes	G.24	attribute changed to OPTIONAL to harmonize with other APs.
	circumscribed_diameter	Modified attribute	Yes, AIM mapping not effected.	G.24	Attribute changed to diameter
	circumscribed_or_-across_flats	New attribute	Yes	G.24	Attribute added to support two different ways to define the diameter value
Open_profile	profile_limit	New attribute	Yes	G.25	attribute OPTIONAL
Square_U_profile	first_radius	Modified attribute	Yes	G.25	attribute changed to OPTIONAL to harmonize with other APs.

Table L.1 Summary of AP224 2nd edition ARM changes (concluded)

Application Entity	Attribute	Status	Upward Compatible	Annex G diagram	Comments
	second_radius	Modified attribute	Yes	G.25	attribute changed to OPTIONAL to harmonize with other APs.
Vee_profile	profile_radius	Modified attribute	Yes	G.25	attribute changed to OPTIONAL to harmonize with other APs.
Tee_profile	radius	Modified attribute	Yes	G.25	attribute changed to OPTIONAL to harmonize with other APs.
Externally_defined_tolerance_-value		Renamed entity	No	G.26	Entity changed to limits_and_fits to harmonize with other APs.

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table L.2 Summary of AP224 2nd edition AIM changes

AIM element	status	1st edition clause	2nd edition clause	Upward compatible	comments
document_reference_item	new type		5.1.2.1	No	Replaces AP224 1st edition entity feature_based_pp_-document_item. Harmonization with other APs.
group_item	new type		5.1.2.10	Yes	
feature_based_pp_-document_item	renamed type	5.2.2.6	5.1.2.1	No	Renamed to document_reference_item
advanced_brep_shape_-representation	removed entity	5.2.3.1.1		Yes	AP224 now references AIC 514
advanced_face	removed entity	5.2.3.1.2		Yes	AP224 now references AIC 511
angularity_tolerance	removed entity	5.2.3.1.3		Yes	AP224 now references AIC 519
applied_area	modified entity	5.2.3.1.5	5.2.3.1.1	No	New rules to support ARM changes. 1st edition had 5 rules, 2nd edition has 7.
applied_document_reference	new entity	5.2.3.1.27	5.2.3.1.2	No	Replaces AP224 1st edition entity feature_based_pp_-document_reference. Harmonization with other APs.
applied_document_usage_-constraint_assignment	new entity	5.2.3.1.27	5.2.3.1.3	No	created in the re-mapping of 1st edition entity feature_-based_pp_document_reference. Harmonization with other APs.
applied_group_assignment	new entity		5.2.3.1.4	Yes	
block_shape_representation	no change	5.2.3.1.5	5.2.3.1.6	Yes	
boss	modified entity	5.2.3.1.6	5.2.3.1.7	No	New rules to support ARM changes. 1st edition had 8 rules, 2nd edition has 12. Moved 5 informal propositions from 1st edition entity machining_feature.

Table L.2 Summary of AP224 2nd edition AIM changes (continued)

boss_top	modified entity	5.2.3.1.7	5.2.3.1.8	No	No new rules, existing rules modified to support AP harmonization.
chamfer	no change	5.2.3.1.8	5.2.3.1.9	Yes	
chamfer_offset	no change	5.2.3.1.9	5.2.3.1.10	Yes	
circular_closed_profile	no change	5.2.3.1.10	5.2.3.1.11	Yes	
circular_pattern	modified entity	5.2.3.1.11	5.2.3.1.12	Yes	WR2 removed, existing rules modified to support AP harmonization. Base_feature_rotation made optional.
circular_runout_tolerance	removed entity	5.2.3.1.12		Yes	AP224 now references AIC 519
closed_path_profile	no change	5.2.3.1.13	5.2.3.1.13	Yes	
composite_hole	modified entity	5.2.3.1.14	5.2.3.1.14	Yes	WR3 removed and changed to in informal proposition - IP1.
composite_machining_feature	modified entity	5.2.3.1.15	5.2.3.1.15	No	Renamed entity to compound_feature New rules to support ARM changes. 1st edition had 3 rules, 2nd edition has 6.
compound_feature	modified entity	5.2.3.1.15	5.2.3.1.15	No	
concentricity_tolerance	removed entity	5.2.3.1.16		Yes	AP224 now references AIC 519
cylindrical_shape_-representation	no change	5.2.3.1.17	5.2.3.1.16	Yes	
directed_dimensional_-location	no change	5.2.3.1.18	5.2.3.1.17	Yes	
direction_shape_-representation	no change	5.2.3.1.19	5.2.3.1.18	Yes	
document_file	new entity		5.2.3.1.19	No	New mapping for Thread to Specification to support AP harmonization.

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table L.2 Summary of AP224 2nd edition AIM changes (continued)

edge_round	modified entity	5.2.3.1.20	5.2.3.1.20	Yes	Existing rules modified to support AP harmonization.
externally_defined_feature_-definition	modified entity	5.2.3.1.21	5.2.3.1.21	No	New rules to support ARM changes. 1st edition had 12 rules, 2nd edition has 15.
face_shape_representation	modified entity	5.2.3.1.22	5.2.3.1.22	Yes	Changed WR1.
face_shape_representation_-relationship	new entity		5.2.3.1.23	Yes	
feature_based_pp_action_-assignment	no change	5.2.3.1.23	5.2.3.1.24	Yes	
feature_based_pp_action_-request_assignment	no change	5.2.3.1.24	5.2.3.1.25	Yes	
feature_based_pp_approval_-assignment	no change	5.2.3.1.25	5.2.3.1.26	Yes	
feature_based_pp_date_-assignment	no change	5.2.3.1.26	5.2.3.1.27	Yes	
feature_based_pp_-document_reference	modified entity	5.2.3.1.27	5.2.3.1.2	No	Renamed to applied_document_reference
feature_based_pp_-organization_assignment	no change	5.2.3.1.28	5.2.3.1.28	Yes	
feature_based_pp_person_-and_organization_assignment	no change	5.2.3.1.29	5.2.3.1.29	Yes	
feature_based_pp_security_-classification_assignment	no change	5.2.3.1.30	5.2.3.1.30	Yes	
feature_component_definition	no change	5.2.3.1.31	5.2.3.1.31	Yes	
feature_component_-relationship	modified entity	5.2.3.1.32	5.2.3.1.32	Yes	Existing rules modified to support AP harmonization.

Table L.2 Summary of AP224 2nd edition AIM changes (continued)

feature_definition	modified entity	5.2.3.1.33	5.2.3.1.33	Yes	Existing rules modified to support AP harmonization, rules from 1st edition entity machining_feature moved to this entity. 1st edition had 2 rules, 2nd edition has 5.
feature_pattern	modified entity	5.2.3.1.34	5.2.3.1.34	Yes	Existing rules modified to support AP harmonization. 1st edition had 1 rule, 2nd edition has 2.
fillet	modified entity	5.2.3.1.35	5.2.3.1.35	Yes	Existing rules modified to support AP harmonization. 1st edition had 9 rules, 2nd edition has 8.
flat_face	modified entity	5.2.3.1.36	5.2.3.1.36	Yes	rule added to support new ARM attribute
hole_bottom	modified entity	5.2.3.1.37	5.2.3.1.37	yes	New rules to support ARM changes. Existing rules modified to support AP harmonization. 1st edition had 13 rules, 2nd edition has 16.
instanced_feature	modified entity	5.2.3.1.38	5.2.3.1.38	yes	Moved 1 informal propositions from 1st edition entity machining_feature.
linear_profile	no change	5.2.3.1.39	5.2.3.1.39	yes	
linear_profile_tolerance	removed entity	5.2.3.1.40		yes	AP224 now references AIC 519
location_shape_-representation	no change	5.2.3.1.41	5.2.3.1.40	yes	
machining_feature	removed entity	5.2.3.1.41		no	support AP harmonization.
marking	modified entity	5.2.3.1.43	5.2.3.1.41	yes	Existing rules modified, 1st edition had 5 rule, 2nd edition has 7.
modified_pattern	modified entity	5.2.3.1.44	5.2.3.1.42	no	removed as sub-type of replicate_feature. New rules to support AP harmonization. Existing rules modified to support AP harmonization. 1st edition had 2 rules, 2nd edition has 3.
ngon_closed_profile	modified entity	5.2.3.1.45	5.2.3.1.43	yes	New rules to support ARM changes. 1st edition had 7 rules, 2nd edition has 8.

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table L.2 Summary of AP224 2nd edition AIM changes (continued)

ngon_shape_representation	modified entity	5.2.3.1.46	5.2.3.1.44	yes	New rules to support ARM changes. 1st edition had 6 rules, 2nd edition has 6.
open_path_profile	modified entity	5.2.3.1.47	5.2.3.1.45	yes	New rules to support ARM changes. 1st edition had 5 rules, 2nd edition has 6.
ordered_part	no change	5.2.3.1.48	5.2.3.1.46	yes	
outer_round	modified entity	5.2.3.1.49	5.2.3.1.47	yes	1st edition entity machining_feature moved to this entity. 1st edition had 1 rules, 2nd edition has 11. Entity machining_feature informal propositions moved to outer_round.
outside_profile	modified entity	5.2.3.1.50	5.2.3.1.48	yes	New rules to support ARM changes. 1st edition entity machining_feature moved to this entity. 1st edition had 1 rules, 2nd edition has 11. Entity machining_feature informal propositions moved to outer_round.
parallelism_tolerance	removed entity	5.2.3.1.51		yes	AP224 now references AIC 519
partial_circular_profile	no change	5.2.3.1.52	5.2.3.1.49	yes	
path_feature_component	modified entity	5.2.3.1.53	5.2.3.1.50	yes	Existing rules modified to support AP harmonization. 1st edition had 14 rules, 2nd edition has 13.
path_shape_representation	modified entity	5.2.3.1.54	5.2.3.1.51	yes	Existing rules modified to support AP harmonization. 1st edition had 3 rules, 2nd edition has 2.
pattern_offset_membership	modified entity	5.2.3.1.55	5.2.3.1.52	no	Existing rules modified to support AP harmonization, rules from 1st edition entity machining_feature moved to this entity. 1st edition had 12 rules, 2nd edition has 13.
pattern_omit_membership	modified entity	5.2.3.1.56	5.2.3.1.53	Yes	Existing rules modified to support AP harmonization, rules from 1st edition entity machining_feature moved to this entity. 1st edition had 9 rules, 2nd edition has 9.
perpendicularity_tolerance	removed entity	5.2.3.1.57		yes	AP224 now references AIC 519
placed_datum_target_feature	no change	5.2.3.1.58	5.2.3.1.54	yes	

Table L.2 Summary of AP224 2nd edition AIM changes (continued)

planar_shape_representation	new entity		5.2.3.1.55	yes	
pocket	modified entity	5.2.3.1.59	5.2.3.1.56	no	Existing rules modified to support AP harmonization, rules from 1st edition entity machining_feature moved to this entity. 1st edition had 7 rules, 2nd edition has 14. New rules to support ARM changes. Entity machining_-feature informal propositions moved .
pocket_bottom	modified entity	5.2.3.1.60	5.2.3.1.57	no	Existing rules modified to support AP harmonization, rules. 1st edition had 9 rules, 2nd edition has 11.
position_tolerance	removed entity	5.2.3.1.61		yes	AP224 now references AIC 519
profile_floor	new entity		5.2.3.1.58	yes	
protrusion	no change	5.2.3.1.62	5.2.3.1.59	yes	
rectangular_closed_profile	modified entity	5.2.3.1.63	5.2.3.1.60	yes	Existing rules modified to support AP harmonization, rules. 1st edition had 7 rules, 2nd edition has 8.
rectangular_pattern	modified entity	5.2.3.1.64	5.2.3.1.61	yes	Existing rules modified to support AP harmonization, rules. 1st edition had 11 rules, 2nd edition has 10.
removal_volume	modified entity	5.2.3.1.65	5.2.3.1.62	yes	Moved informal propositions from Machining_feature
replicate_feature	modified entity	5.2.3.1.66	5.2.3.1.63	yes	Existing rules modified to support AP harmonization, rules. 1st edition had 1 rules, 2nd edition has 2.
revolved_profile	modified entity	5.2.3.1.67	5.2.3.1.64	yes	Entity machining_feature informal propositions moved.
rib_top	new entity		5.2.3.1.65	yes	
rib_top_floor	new entity		5.2.3.1.66	yes	
round_hole	modified entity	5.2.3.1.68	5.2.3.1.67	yes	1st edition entity machining_feature moved to this entity. Entity machining_feature informal propositions moved.

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table L.2 Summary of AP224 2nd edition AIM changes (continued)

rounded_end	modified entity	5.2.3.1.69	5.2.3.1.68	yes	1st edition entity machining_feature moved to this entity. Entity machining_feature informal propositions moved.
rounded_u_profile	modified entity	5.2.3.1.70	5.2.3.1.69	yes	Existing rules modified to support AP harmonization, rules. 1st edition had 5 rules, 2nd edition has 6.
shape_defining_relationship	no change	5.2.3.1.71	5.2.3.1.70	yes	
shape_representation_with_-parameters	no change	5.2.3.1.72	5.2.3.1.71	yes	
slot	modified entity	5.2.3.1.73	5.2.3.1.72	no	Entity machining_feature informal propositions moved. Existing rules modified to support AP harmonization, rules. 1st edition had 5 rules, 2nd edition has 4. Removed an ARM attribute.
slot_end	no change	5.2.3.1.74	5.2.3.1.73	No	text strings change
spherical_cap	no change	5.2.3.1.75	5.2.3.1.74	yes	Informal proposition moved from Machining_feature
square_u_profile	modified entity	5.2.3.1.76	5.2.3.1.75	yes	Existing rules modified to support AP harmonization. 1st edition had 9 rules, 2nd edition has 11. New rules to support ARM changes.
step	no change	5.2.3.1.77	5.2.3.1.76	yes	Informal proposition moved from Machining_feature
surface_profile_tolerance	removed entity	5.2.3.1.78		yes	AP224 now references AIC 519
symmetry_tolerance	removed entity	5.2.3.1.79		yes	AP224 now references AIC 519
taper	modified entity	5.2.3.1.80	5.2.3.1.77	yes	Existing rules modified to support AP harmonization. 1st edition had 6 rules, 2nd edition has 8. New rules to support ARM changes.
tee_profile	modified entity	5.2.3.1.81	5.2.3.1.78	yes	Existing rules modified to support AP harmonization.

Table L.2 Summary of AP224 2nd edition AIM changes (continued)

thread	modified entity	5.2.3.1.82	5.2.3.1.79	no	Existing rules modified to support AP harmonization, rules from 1st edition entity machining_feature moved to this entity. 1st edition had 10 rules, 2nd edition has 13. New rules to support ARM changes. Entity machining_-feature informal propositions moved.
total_runout_tolerance	removed entity	5.2.3.1.83		yes	AP224 now references AIC 519
transition_feature	modified entity	5.2.3.1.84	5.2.3.1.80	yes	Existing rules modified to support AP harmonization. 1st edition had 1 rule, 2nd edition has 2.
turned_knurl	modified entity	5.2.3.1.85	5.2.3.1.81	yes	Existing rules modified to support AP harmonization, rules from 1st edition entity machining_feature moved to this entity. 1st edition had 10 rules, 2nd edition has 12. New rules to support ARM changes. Entity machining_-feature informal propositions moved.
vee_profile	modified entity	5.2.3.1.86	5.2.3.1.82	yes	Existing rules modified to support AP harmonization. 1st edition had 7 rules, 2nd edition has 9. New rules to support ARM changes.
application_context_-requires_ap_definition	no change	5.2.4.1	5.2.4.1	yes	
approval_requires_approval_-date_time	no change	5.2.4.2	5.2.4.2	yes	
approval_requires_approval_-person_organization	no change	5.2.4.3	5.2.4.3	yes	
dependent_instantiable_-action_request_status	no change	5.2.4.4	5.2.4.4	yes	
dependent_instantiable_-action_status	rule removed	5.2.4.5		yes	rule not required for 2nd edition
dependent_instantiable_-approval_status	no change	5.2.4.6	5.2.4.5	yes	

©ISO 2000 - All rights reserve

©ISO 2000 - All rights reserved

Table L.2 Summary of AP224 2nd edition AIM changes (continued)

dependent_instantiable_date	no change	5.2.4.7	5.2.4.6	yes	
dependent_instantiable_-named_unit	no change	5.2.4.8	5.2.4.7	yes	
dependent_instantiable_-precision_qualifier	no change	5.2.4.9	5.2.4.8	yes	
dependent_instantiable_-type_qualifier	no change	5.2.4.10	5.2.4.9	yes	
dependent_instantiable_-uncertainty_qualifier	no change	5.2.4.11	5.2.4.10	yes	
dependent_instantiable_-security_classification_level	no change	5.2.4.12	5.2.4.11	yes	
dependent_instantiable_-shape_representation	no change	5.2.4.13	5.2.4.12	yes	
geometric_tolerance_-subtype_exclusiveness	modified rule	5.2.4.14	5.2.4.13	yes	new types added to rule
machining_feature_life_cycle	yes	5.2.4.15	5.2.4.14	yes	Machining_feature changed to instanced_feature
material_is_specified_for_-part	no change	5.2.4.16	5.2.4.15	yes	
part_requires_project_order	no change	5.2.4.17	5.2.4.16	yes	
part_to_approval	no change	5.2.4.18	5.2.4.17	yes	
product_requires_version	no change	5.2.4.19	5.2.4.18	yes	
product_definition_-formation_requires_-security_classification	no change	5.2.4.20	5.2.4.19	yes	
project_order_requires_-approval	no change	5.2.4.21	5.2.4.20	yes	

Table L.2 Summary of AP224 2nd edition AIM changes (concluded)

project_order_tracking_-relationships	no change	5.2.4.22	5.2.4.21	yes	
representation_subtype_-exclusiveness	modified rule	5.2.4.23	5.2.4.22	yes	new types added to rule, planar_shape_representation
restrict_approval_status	no change	5.2.4.24	5.2.4.23	yes	
restrict_security_-classification_level	no change	5.2.4.25	5.2.4.24	yes	
shape_aspect_relationship_-subtype_exclusiveness	no change	5.2.4.26	5.2.4.25	yes	
shape_aspect_subtype_-exclusiveness	modified rule	5.2.4.27	5.2.4.26	yes	new types added to rule
shape_representation_-subtype_exclusiveness	no change	5.2.4.28	5.2.4.27	yes	
subtype_mandatory_-characterized_object	modified rule	5.2.4.29	5.2.4.28	yes	modified because of material_property
transition_feature_life_cycle	no change	5.2.4.30	5.2.4.29	yes	
transition_feature_on_part_-boundary	no change	5.2.4.31	5.2.4.30	yes	
msb_shells	removed function	5.1.5.1		yes	rule supported in AIC 514
number_of_representation_-items	removed function	5.1.5.2		yes	rules moved to specific entities

©ISO 2000 - All rights reserve

Bibliography

[1] Guidelines for the Development and Approval of STEP Application Protocols, Version 1.2, ISO TC184/SC4/WG4 N433, April 1996.

[2] *IDEF0 Federal Information Processing Standards Publication 183, Integration Definition for Function Modeling (IDEF0)*, FIPS PUB 183, National Institute of Standards and Technology, December 1993.

[3] ANSI Y14.5M:1982 American National Standard, Dimensioning and Tolerancing

©ISO 2000 - All rights reserved

Index

AAM 4
action
AIM diagrams 981
AIM EXPRESS long listing entity 737
action_assignment
AIM diagrams 981
AIM EXPRESS long listing entity 737
action_directive
AIM diagrams 981
AIM EXPRESS long listing entity 737
mapping table 227
action_method
AIM diagrams 981
AIM EXPRESS long listing entity 738
action_relationship
AIM diagrams 981
AIM EXPRESS long listing entity 738
AIM EXPRESS short listing imported entity modifications 701
action_request_assignment
AIM diagrams 982
AIM EXPRESS long listing entity 738
action_request_solution
AIM diagrams 981
AIM EXPRESS long listing entity 738
action_request_status
AIM diagrams 982
AIM EXPRESS long listing entity 738
AIM EXPRESS short listing imported entity modifications 701
action_status
AIM diagrams 981
AIM EXPRESS long listing entity 738
AIM EXPRESS short listing imported entity modifications 701
acyclic_mapped_representation
AIM EXPRESS long listing function 862
acyclic_product_definition_relationship
AIM EXPRESS long listing function 862
address
AIM diagrams 978
AIM EXPRESS long listing entity 738
advanced_brep_shape_representation
AIM diagrams 959
AIM EXPRESS long listing entity 739
advanced_face
AIM diagrams 974
AIM EXPRESS long listing entity 739
AIM 4
Alternate_material

 ©ISO 2000 - All rights reserved

application assertion 186, 218
application object 19
ARM diagams 931
mapping table 463
angle_relator
AIM EXPRESS long listing type 730
Angle_taper
application assertion 186, 188, 196, 204, 209, 212, 216
application object 20
ARM diagams 948
mapping table 229
Angular_dimension_tolerance
application object 21
ARM diagams 952
mapping table 479
angular_location
AIM diagrams 987
AIM EXPRESS long listing entity 741
angular_size
AIM diagrams 987
AIM EXPRESS long listing entity 741
Angular_size_dimension_tolerance
application object 23
ARM diagams 952
mapping table 479
angularity_tolerance
AIM diagrams 988
AIM EXPRESS long listing entity 741
application assertion 186
application object 24
ARM diagams 953
mapping table 480
AP 4
Application 4, 5
Application activity model 4
Application interpreted model 4
Application protocol 4
Application reference model 4, 5
application_context
AIM diagrams 956
AIM EXPRESS long listing entity 741
AIM EXPRESS short listing imported entity modifications 701
application_context_element
AIM diagrams 956
AIM EXPRESS long listing entity 741
application_context_requires_ap_definition
AIM EXPRESS long listing rule 855
AIM EXPRESS short listing rules 708
application_protocol_definition
AIM diagrams 956

©ISO 2000 - All rights reserved

AIM EXPRESS long listing entity 741
AIM EXPRESS short listing imported entity modifications 702
applied_area
AIM diagrams 961
AIM EXPRESS long listing entity 741
AIM EXPRESS short listing entities 545
mapping table 253-256, 373, 455
applied_document_reference
AIM diagrams 976
AIM EXPRESS long listing entity 743
AIM EXPRESS short listing enities 547
mapping table 309, 311, 313, 469
applied_document_usage_constraint_assignment
AIM diagrams 976
AIM EXPRESS long listing entity 743
applied_document_usage_constraint_assignment 547
applied_group_assignment
AIM diagrams 963
AIM EXPRESS long listing entity 743
AIM EXPRESS short listing enities 547
approval
AIM diagrams 977
AIM EXPRESS long listing entity 743
AIM EXPRESS short listing imported entity modifications 702
application assertion 186, 205, 210
application object 25
ARM diagams 930
mapping table 509
approval_assignment
AIM diagrams 977
AIM EXPRESS long listing entity 743
approval_date_time
AIM diagrams 977
AIM EXPRESS long listing entity 743
AIM EXPRESS short listing imported entity modifications 702
approval_person_organization
AIM diagrams 977
AIM EXPRESS long listing entity 743
AIM EXPRESS short listing imported entity modifications 702
approval_requires_approval_date_time
AIM EXPRESS long listing rule 855
AIM EXPRESS short listing rules 709
approval_requires_approval_person_organization
AIM EXPRESS long listing rule 855
AIM EXPRESS short listing rules 709
approval_role
AIM diagrams 977
AIM EXPRESS long listing entity 743
approval_status
AIM diagrams 977

 ©ISO 2000 - All rights reserved

AIM EXPRESS long listing entity 744
AIM EXPRESS short listing imported entity modifications 703
ARM 4, 5
assembly_component_usage
AIM diagrams 957
AIM EXPRESS long listing entity 744
associated_surface
AIM EXPRESS long listing function 863
attribute_type
AIM EXPRESS long listing type 730
axis1_placement
AIM diagrams 967
AIM EXPRESS long listing entity 744
axis2_placement
AIM EXPRESS long listing type 730
axis2_placement_2d
AIM diagrams 967
AIM EXPRESS long listing entity 744
axis2_placement_3d
AIM diagrams 967
AIM EXPRESS long listing entity 744
B-rep_model
application assertion 187
application object 28
ARM diagams 933
mapping table 523
B-rep_model_element
application assertion 187, 218
application object 29
ARM diagams 933
mapping table 524
B-rep_shape_aspect_representation
application assertion 187, 218
application object 29
ARM diagams 933
mapping table 524
B-rep_shape_representation
application assertion 187, 195, 217
application object 29
ARM diagams 933
mapping table 524
b_spline_curve
AIM diagrams 969
AIM EXPRESS long listing entity 744
b_spline_curve_form
AIM EXPRESS long listing type 730
b_spline_curve_with_knots
AIM diagrams 969
AIM EXPRESS long listing entity 745
b_spline_surface

©ISO 2000 - All rights reserved

AIM diagrams 973
AIM EXPRESS long listing entity 745
b_spline_surface_form
AIM EXPRESS long listing type 730
b_spline_surface_with_knots
AIM diagrams 973
AIM EXPRESS long listing entity 745
bag_to_set
AIM EXPRESS long listing function 863
base_axis
AIM EXPRESS long listing function 863
Base_shape
application assertion 217
application object 26
ARM diagams 933
mapping table 523
bezier_curve
AIM diagrams 969
AIM EXPRESS long listing entity 746
bezier_surface
AIM diagrams 973
AIM EXPRESS long listing entity 746
Blind_bottom_condition
application assertion 216
application object 26
ARM diagams 942
mapping table 231
Block_base_shape
application assertion 186
application object 26
ARM diagams 933
mapping table 523
block_shape_representation
AIM diagrams 959
AIM EXPRESS long listing entity 746
AIM EXPRESS short listing entities 548
mapping table 523
boolean_choose
AIM EXPRESS long listing function 864
boolean_operand
AIM EXPRESS long listing type 730
boss
AIM diagrams 962
AIM EXPRESS long listing entity 746
AIM EXPRESS short listing entities 549
application assertion 186, 187
application object 27
ARM diagams 943
mapping table 305-307, 315-319, 359-361, 400-403
boss_top

 ©ISO 2000 - All rights reserved

AIM diagrams 961
AIM EXPRESS long listing entity 749
AIM EXPRESS short listing entities 553
mapping table 232, 251, 262, 263, 305
Boss_top_condition
application assertion 186
application object 28
ARM diagams 943
mapping table 232
bounded_curve
AIM diagrams 968
AIM EXPRESS long listing entity 750
bounded_surface
AIM diagrams 973
AIM EXPRESS long listing entity 750
brep_with_voids
AIM diagrams 975
AIM EXPRESS long listing entity 750
build_2axes
AIM EXPRESS long listing function 864
build_axes
AIM EXPRESS long listing function 864
calendar_date
AIM diagrams 980
AIM EXPRESS long listing entity 750
cartesian_point
AIM diagrams 966
AIM EXPRESS long listing entity 750
cartesian_transformation_operator
AIM diagrams 967
AIM EXPRESS long listing entity 751
cartesian_transformation_operator_3d
AIM diagrams 967
AIM EXPRESS long listing entity 751
Catalogue_knurl
application assertion 187
application object 29
ARM diagams 946
mapping table 308
Catalogue_marking
application assertion 187
application object 30
ARM diagams 945
mapping table 310
Catalogue_thread
application assertion 187
application object 30
ARM diagams 945
mapping table 312
chamfer

©ISO 2000 - All rights reserved

AIM diagrams 964
AIM EXPRESS long listing entity 751
AIM EXPRESS short listing entities 556
application assertion 187, 188
application object 32
ARM diagams 937
mapping table 313-315
Chamfer_angle
application assertion 188
application object 33
ARM diagams 937
mapping table 233
chamfer_offset
AIM diagrams 961
AIM EXPRESS long listing entity 751
AIM EXPRESS short listing entities 557
mapping table 233, 234, 240-242, 267, 268, 314, 315
characterized_definition
AIM EXPRESS long listing type 731
characterized_material_property
AIM EXPRESS long listing type 731
characterized_object
AIM diagrams 962
AIM EXPRESS long listing entity 753
AIM EXPRESS short listing imported entity modifications 703
characterized_product_definition
AIM EXPRESS long listing type 731
circle
AIM diagrams 968
AIM EXPRESS long listing entity 753
Circular_boss
application assertion 188
application object 33
ARM diagams 943
mapping table 315
circular_closed_profile
AIM diagrams 961
AIM EXPRESS long listing entity 753
AIM EXPRESS short listing entities 559
application assertion 188, 189, 216
application object 35
ARM diagams 950
mapping table 261, 272, 282, 319, 321, 362, 364, 367, 370, 399, 432
Circular_closed_shape_profile
application assertion 189
application object 35
ARM diagams 940
mapping table 319
Circular_cutout
application assertion 189

 ©ISO 2000 - All rights reserved

application object 36
ARM diagams 938
mapping table 320
Circular_offset_pattern
application assertion 189
application object 36
ARM diagams 948
mapping table 321
Circular_omit_pattern
application assertion 189
application object 37
ARM diagams 948
mapping table 324
Circular_path
application assertion 189
application object 38
ARM diagams 949
mapping table 234
circular_pattern
AIM diagrams 963
AIM EXPRESS long listing entity 754
AIM EXPRESS short listing entities 561
application assertion 189
application object 38
ARM diagams 948
mapping table 325, 326, 328-331
circular_runout_tolerance
AIM diagrams 988
AIM EXPRESS long listing entity 755
application assertion 190
application object 40
ARM diagams 953
mapping table 480, 481
Circularity_tolerance
application object 41
ARM diagams 953
mapping table 481
closed_path_profile
AIM diagrams 961
AIM EXPRESS long listing entity 755
AIM EXPRESS short listing entities 563
mapping table 261, 272, 273, 282, 361, 362, 364, 367, 370, 399
Closed_profile
application assertion 196, 207, 208
application object 41
ARM diagams 950
mapping table 272
closed_shell
AIM diagrams 975
AIM EXPRESS long listing entity 756

©ISO 2000 - All rights reserved

closed_shell_reversed
AIM EXPRESS long listing function 865
compatible_dimension
AIM EXPRESS long listing rule 855
Complete_circular_path
application object 42
ARM diagams 949
mapping table 235
composite_curve
AIM diagrams 969
AIM EXPRESS long listing entity 756
composite_curve_on_surface
AIM diagrams 969
composite_curve_segment
AIM diagrams 969
AIM EXPRESS long listing entity 756
composite_hole
AIM diagrams 962
AIM EXPRESS long listing entity 757
AIM EXPRESS short listing entities 565
mapping table 340, 342, 372
composite_shape_aspect
AIM diagrams 963
AIM EXPRESS long listing entity 757
Compound_datum
application assertion 190
application object 43
ARM diagams 954
mapping table 481
compound_feature
AIM diagrams 962
AIM EXPRESS long listing entity 758
AIM EXPRESS short listing enities 566
application assertion 190
application object 43
ARM diagams 935
mapping table 332
Compound_feature_element
application assertion 190
application object 44
ARM diagams 935
mapping table 333
Compound_feature_relationship
application assertion 190
application object 44
ARM diagams 935
mapping table 334
concentricity_tolerance
AIM diagrams 988
AIM EXPRESS long listing entity 759

 ©ISO 2000 - All rights reserved

application assertion 191
application object 45
ARM diagams 953
mapping table 481, 482, 496
conditional_reverse
AIM EXPRESS long listing function 865
Conformance testing 5
conic
AIM diagrams 968
AIM EXPRESS long listing entity 759
Conical_hole_bottom
application object 46
ARM diagams 942
mapping table 235
conical_surface
AIM diagrams 972
AIM EXPRESS long listing entity 759
connected_face_set
AIM diagrams 975
AIM EXPRESS long listing entity 759
Constant_radius_edge_round
application assertion 191
application object 47
ARM diagams 937
mapping table 334
Constant_radius_fillet
application assertion 191
application object 48
ARM diagams 937
mapping table 337
constraints_param_b_spline
AIM EXPRESS long listing function 865
context_dependent_measure
AIM EXPRESS long listing type 731
context_dependent_unit
AIM diagrams 983
AIM EXPRESS long listing entity 759
conversion_based_unit
AIM diagrams 983
AIM EXPRESS long listing entity 759
count_measure
AIM EXPRESS long listing type 731
Counterbore_hole
application assertion 191
application object 49
ARM diagams 942
mapping table 340
Countersunk_hole
application assertion 192
application object 50

©ISO 2000 - All rights reserved

ARM diagams 942
mapping table 342
cross_product
AIM EXPRESS long listing function 866
curve
AIM diagrams 968
AIM EXPRESS long listing entity 759
curve_on_surface
AIM EXPRESS long listing type 731
curve_weights_positive
AIM EXPRESS long listing function 867
Curved_dimension_tolerance
application object 51
ARM diagams 952
mapping table 482
Customer_order
application assertion 192
application object 52
ARM diagams 930
mapping table 471
Cutout
application assertion 192
application object 53
ARM diagams 938
mapping table 345
Cutting_tool_requisition
application object 54
ARM diagams 929
mapping table 521
Cylindrical_base_shape
application assertion 192
application object 54
ARM diagams 933
mapping table 525
cylindrical_shape_representation
AIM diagrams 959
AIM EXPRESS long listing entity 759
AIM EXPRESS short listing entities 568
mapping table 525
cylindrical_surface
AIM diagrams 972
AIM EXPRESS long listing entity 760
cylindricity_tolerance
AIM diagrams 988
AIM EXPRESS long listing entity 760
application object 55
ARM diagams 953
mapping table 482
data_environment
AIM diagrams 991

 ©ISO 2000 - All rights reserved

AIM EXPRESS long listing entity 760
date
AIM diagrams 980
AIM EXPRESS long listing entity 760
AIM EXPRESS short listing imported entity modifications 703
mapping table 226
date_assignment
AIM diagrams 980
AIM EXPRESS long listing entity 760
date_role
AIM diagrams 980
AIM EXPRESS long listing entity 760
date_time_or_event_occurrence
AIM EXPRESS long listing type 731
date_time_select
AIM EXPRESS long listing type 731
datum
AIM diagrams 986
AIM EXPRESS long listing entity 760
application assertion 186, 190, 191, 200, 204, 207, 209, 220, 223
application object 55
ARM diagams 954
mapping table 482
datum_feature
AIM diagrams 986
AIM EXPRESS long listing entity 761
application assertion 190, 192, 193
application object 55
ARM diagams 954
mapping table 482
datum_reference
AIM diagrams 986
AIM EXPRESS long listing entity 761
datum_target
AIM diagrams 986
AIM EXPRESS long listing entity 761
application assertion 193
application object 56
ARM diagams 954
mapping table 483
Datum_target_set
application assertion 192, 193
application object 56
ARM diagams 954
mapping table 483
day_in_month_number
AIM EXPRESS long listing type 731
day_in_week_number
AIM EXPRESS long listing type 731
day_in_year_number

©ISO 2000 - All rights reserved

AIM EXPRESS long listing type 731
Dedicated_fixture_requisition
application object 57
ARM diagams 929
mapping table 521
Defined_marking
application assertion 193
application object 57
ARM diagams 945
mapping table 345
Defined_thread
application assertion 193
application object 58
ARM diagams 945
mapping table 349
definitional_representation
AIM diagrams 960
AIM EXPRESS long listing entity 761
degenerate_toroidal_surface
AIM diagrams 972
AIM EXPRESS long listing entity 761
dependent_instantiable_action_request_status
AIM EXPRESS long listing rule 856
AIM EXPRESS short listing rules 710
dependent_instantiable_approval_status
AIM EXPRESS long listing rule 856
AIM EXPRESS short listing rules 710
dependent_instantiable_date
AIM EXPRESS long listing rule 856
AIM EXPRESS short listing rules 711
dependent_instantiable_named_unit
AIM EXPRESS long listing rule 856
AIM EXPRESS short listing rules 711
dependent_instantiable_precision_qualifier
AIM EXPRESS long listing rule 856
AIM EXPRESS short listing rules 712
dependent_instantiable_security_classification_level
AIM EXPRESS long listing rule 856
AIM EXPRESS short listing rules 713
dependent_instantiable_shape_representation
AIM EXPRESS long listing rule 856
AIM EXPRESS short listing rules 714
dependent_instantiable_type_qualifier
AIM EXPRESS long listing rule 857
AIM EXPRESS short listing rules 712
dependent_instantiable_uncertainty_qualifier
AIM EXPRESS long listing rule 857
AIM EXPRESS short listing rules 713
derive_dimensional_exponents
AIM EXPRESS long listing function 867

 ©ISO 2000 - All rights reserved

derived_property_select
AIM EXPRESS long listing type 731
derived_unit
AIM diagrams 983
AIM EXPRESS long listing entity 761
derived_unit_element
AIM diagrams 983
AIM EXPRESS long listing entity 762
description_attribute
AIM diagrams 992
AIM EXPRESS long listing entity 762
AIM EXPRESS short listing imported entity modifications 703
description_attribute_select
AIM EXPRESS long listing type 731
descriptive_measure
AIM EXPRESS long listing type 732
Descriptive_parameter
application assertion 193, 194, 201, 221
application object 59
ARM diagams 932
mapping table 463
descriptive_representation_item
AIM diagrams 960
AIM EXPRESS long listing entity 762
Design_exception_notice
application assertion 193
application object 60
ARM diagams 927
mapping table 226
Diagonal_knurl
application assertion 194
application object 61
ARM diagams 946
mapping table 352
Diameter_dimension_tolerance
application object 62
ARM diagams 952
mapping table 483
Diameter_taper
application assertion 188, 194, 204, 216
application object 62
ARM diagams 948
mapping table 237
Diamond_knurl
application assertion 194
application object 63
ARM diagams 946
mapping table 354
Digital_technical_data_package_work_order
application assertion 210

©ISO 2000 - All rights reserved

application object 64
ARM diagams 929
mapping table 472
dimension_count
AIM EXPRESS long listing type 732
dimension_of
AIM EXPRESS long listing function 867
dimensional_characteristic
AIM EXPRESS long listing type 732
dimensional_characteristic_representation
AIM diagrams 987
AIM EXPRESS long listing entity 762
dimensional_exponents
AIM diagrams 983
AIM EXPRESS long listing entity 762
dimensional_location
AIM diagrams 987
AIM EXPRESS long listing entity 762
AIM EXPRESS short listing imported entity modifications 704
dimensional_location_with_path
AIM diagrams 987
AIM EXPRESS long listing entity 762
dimensional_size
AIM diagrams 987
AIM EXPRESS long listing entity 762
dimensional_size_with_path
AIM diagrams 987
AIM EXPRESS long listing entity 762
Dimensional_tolerance
application assertion 194
application object 64
ARM diagams 952
mapping table 483
dimensions_for_si_unit
AIM EXPRESS long listing function 868
directed_action
AIM diagrams 981
AIM EXPRESS long listing entity 762
AIM EXPRESS short listing imported entity modifications 704
directed_dimensional_location
AIM diagrams 987
AIM EXPRESS long listing entity 763
AIM EXPRESS short listing entities 569
mapping table 488, 489
Directed_taper
application assertion 188, 194, 196, 209, 212, 217
application object 65
ARM diagams 948
direction
AIM diagrams 967

 ©ISO 2000 - All rights reserved

AIM EXPRESS long listing entity 763
Direction_element
application assertion 194, 199, 207, 208, 213-216, 218
application object 66
ARM diagams 934
mapping table 525
direction_shape_representation
AIM diagrams 959
AIM EXPRESS long listing entity 763
AIM EXPRESS short listing enities 570
mapping table 239, 252, 259, 262, 386, 406, 415, 424, 441, 525
Distance_along_curve_tolerance
application assertion 194
application object 66
ARM diagams 952
mapping table 485
document
AIM diagrams 976
AIM EXPRESS long listing entity 763
mapping table 228
document_file
AIM diagrams 976
AIM EXPRESS long listing entity 763
AIM EXPRESS short listing enities 570
document_reference
AIM diagrams 976
AIM EXPRESS long listing entity 763
document_reference_item
AIM EXPRESS long listing type 732
AIM EXPRESS short listing types 542
mapping table 309, 311, 313, 469
document_relationship
AIM diagrams 976
AIM EXPRESS long listing entity 764
document_representation_type
AIM diagrams 976
AIM EXPRESS long listing entity 764
document_type
AIM diagrams 976
AIM EXPRESS long listing entity 764
document_usage_constraint
AIM diagrams 976
AIM EXPRESS long listing entity 764
document_usage_constraint_assignment
AIM diagrams 976
AIM EXPRESS long listing entity 764
document_usage_role
AIM diagrams 976
AIM EXPRESS long listing entity 764
document_with_class

©ISO 2000 - All rights reserved

AIM diagrams 976
AIM EXPRESS long listing entity 764
dot_product
AIM EXPRESS long listing function 869
edge
AIM diagrams 974
AIM EXPRESS long listing entity 764
edge_curve
AIM diagrams 974
AIM EXPRESS long listing entity 764
edge_loop
AIM diagrams 974
AIM EXPRESS long listing entity 764
edge_reversed
AIM EXPRESS long listing function 869
edge_round
AIM diagrams 964
AIM EXPRESS long listing entity 765
AIM EXPRESS short listing entities 571
application assertion 194
application object 67
ARM diagams 937
mapping table 334-337, 355-357
elementary_surface
AIM diagrams 972
AIM EXPRESS long listing entity 766
ellipse
AIM diagrams 968
AIM EXPRESS long listing entity 766
Engineering_change_order
application assertion 195
application object 68
ARM diagams 927
mapping table 227
Engineering_change_proposal
application assertion 193, 195
application object 68
ARM diagams 927
mapping table 228
executed_action
AIM diagrams 981
AIM EXPRESS long listing entity 766
expanded_uncertainty
AIM diagrams 990
AIM EXPRESS long listing entity 766
Explicit_base_shape_representation
application assertion 195
application object 69
ARM diagams 933
mapping table 525

 ©ISO 2000 - All rights reserved

external_source
AIM diagrams 962
AIM EXPRESS long listing entity 767
externally_defined_feature_definition
AIM diagrams 962
AIM EXPRESS long listing entity 767
AIM EXPRESS short listing entities 573
mapping table 308-313, 372, 373, 375, 376, 449-456
externally_defined_item
AIM diagrams 962
AIM EXPRESS long listing entity 770
face
AIM diagrams 974
AIM EXPRESS long listing entity 770
face_bound
AIM diagrams 974
AIM EXPRESS long listing entity 770
face_bound_reversed
AIM EXPRESS long listing function 869
face_outer_bound
AIM diagrams 974
AIM EXPRESS long listing entity 770
face_reversed
AIM EXPRESS long listing function 870
Face_shape_element
application assertion 187, 194, 195, 197, 198, 217
application object 69
ARM diagams 934
mapping table 526
Face_shape_element_relationship
application assertion 195
application object 69
ARM diagams 934
mapping table 526
face_shape_representation
AIM diagrams 959
AIM EXPRESS long listing entity 770
AIM EXPRESS short listing entities 579
mapping table 241, 248, 249, 251, 267, 356-359, 526
face_shape_representation_relationship
AIM diagrams 960
AIM EXPRESS long listing entity 771
AIM EXPRESS short listing enities 579
mapping table 526
face_surface
AIM diagrams 974
AIM EXPRESS long listing entity 771
feature_based_pp_action_assignment
AIM diagrams 981
AIM EXPRESS long listing entity 771

©ISO 2000 - All rights reserved

AIM EXPRESS short listing entities 580
mapping table 228, 476
feature_based_pp_action_item
AIM EXPRESS long listing type 732
AIM EXPRESS short listing types 542
mapping table 228, 476
feature_based_pp_action_request_assignment
AIM diagrams 982
AIM EXPRESS long listing entity 771
AIM EXPRESS short listing entities 580
mapping table 227
feature_based_pp_action_request_item
AIM EXPRESS long listing type 732
AIM EXPRESS short listing types 542
mapping table 227
feature_based_pp_approval_assignment
AIM diagrams 977
AIM EXPRESS long listing entity 771
AIM EXPRESS short listing entities 581
mapping table 475, 516
feature_based_pp_approved_item
AIM EXPRESS long listing type 732
AIM EXPRESS short listing types 543
mapping table 475, 516
feature_based_pp_classified_item
AIM EXPRESS long listing type 732
AIM EXPRESS short listing types 543
mapping table 515
feature_based_pp_date_assignment
AIM diagrams 980
AIM EXPRESS long listing entity 771
AIM EXPRESS short listing entities 581
mapping table 226, 471, 522
feature_based_pp_dated_item
AIM EXPRESS long listing type 732
AIM EXPRESS short listing types 543
mapping table 226, 471, 522
Feature_based_pp_document_item
mapping table 227, 228, 309, 311, 313, 476
Feature_based_pp_document_reference
mapping table 227, 228, 309, 311, 313, 476
feature_based_pp_ordered_item
AIM EXPRESS long listing type 732
AIM EXPRESS short listing types 544
mapping table 516
feature_based_pp_organization_assignment
AIM diagrams 979
AIM EXPRESS long listing entity 771
AIM EXPRESS short listing entities 581
mapping table 516, 517

 ©ISO 2000 - All rights reserved

feature_based_pp_organization_item
AIM EXPRESS long listing type 732
AIM EXPRESS short listing types 544
mapping table 516, 517
feature_based_pp_person_and_organization_assignment
AIM diagrams 979
AIM EXPRESS long listing entity 771
AIM EXPRESS short listing entities 582
mapping table 472, 517
feature_based_pp_person_and_organization_item
AIM EXPRESS long listing type 733
AIM EXPRESS short listing types 544
mapping table 472, 517
feature_based_pp_security_classification_assignment
AIM diagrams 982
AIM EXPRESS long listing entity 771
AIM EXPRESS short listing entities 582
mapping table 515
feature_component_definition
AIM diagrams 962
AIM EXPRESS long listing entity 771
AIM EXPRESS short listing entities 582
feature_component_relationship
AIM diagrams 963
AIM EXPRESS long listing entity 772
AIM EXPRESS short listing entities 583
mapping table . . 231, 261, 264, 305, 314-318, 321, 324, 326, 327, 341-345, 360, 378-380, 391, 392, 394, 395, 398, 400-402, 405, 410, 420, 421, 423, 429-431, 433, 434, 436, 439, 440, 444
feature_definition
AIM diagrams 962
AIM EXPRESS long listing entity 772
AIM EXPRESS short listing entities 583
mapping table . . 231, 305-308, 310, 312, 315-321, 332, 333, 340-355, 359-364, 366-398, 400-405, 412-414, 424-462
feature_pattern
AIM diagrams 963
AIM EXPRESS long listing entity 773
AIM EXPRESS short listing entities 585
mapping table 365
fillet
AIM diagrams 964
AIM EXPRESS long listing entity 774
AIM EXPRESS short listing entities 586
application assertion 195
application object 70
ARM diagams 937
mapping table 337-340, 357-359
First_offset
application assertion 188, 195, 196
application object 71

©ISO 2000 - All rights reserved

ARM diagams 937
mapping table 240
first_proj_axis
AIM EXPRESS long listing function 870
flat_face
AIM diagrams 962
AIM EXPRESS long listing entity 775
AIM EXPRESS short listing entities 589
mapping table 386-389
Flat_hole_bottom
application object 71
ARM diagams 942
mapping table 242
Flat_slot_end_type
application assertion 196
application object 72
ARM diagams 944
mapping table 242
Flat_with_radius_hole_bottom
application object 73
ARM diagams 942
mapping table 244
Flat_with_taper_hole_bottom
application assertion 196
application object 73
ARM diagams 942
mapping table 245
flatness_tolerance
AIM diagrams 988
AIM EXPRESS long listing entity 776
application object 74
ARM diagams 953
mapping table 485
founded_item
AIM diagrams 969
AIM EXPRESS long listing entity 776
founded_item_select
AIM EXPRESS long listing type 733
functionally_defined_transformation
AIM diagrams 959
AIM EXPRESS long listing entity 776
General_boss
application assertion 196
application object 74
ARM diagams 943
mapping table 359
General_closed_profile
application assertion 197
application object 75
ARM diagams 950

 ©ISO 2000 - All rights reserved

mapping table 273
General_cutout
application assertion 197
application object 76
ARM diagams 938
mapping table 361
General_open_profile
application assertion 197, 198
application object 76
ARM diagams 951
mapping table 273
General_outside_profile
application assertion 197
application object 77
ARM diagams 940
mapping table 363
General_path
application assertion 197
application object 78
ARM diagams 949
mapping table 247
General_pattern
application assertion 197
application object 79
ARM diagams 947
mapping table 365
General_pocket
application assertion 197
application object 80
ARM diagams 938
mapping table 366
General_pocket_bottom_condition
application assertion 197
application object 81
ARM diagams 939
mapping table 248
General_profile_floor
application assertion 198
application object 81
ARM diagams 940
mapping table 248
General_removal_volume
application assertion 198
application object 81
ARM diagams 941
mapping table 367
General_revolution
application assertion 198
application object 82
ARM diagams 941

©ISO 2000 - All rights reserved

mapping table 368
General_rib_top_floor
application assertion 198
application object 83
ARM diagams 939
mapping table 250
General_shape_profile
application assertion 198
application object 84
ARM diagams 940
mapping table 369
General_top_condition
application assertion 198
application object 84
ARM diagams 943
mapping table 251
geometric_representation_context
AIM diagrams 960
AIM EXPRESS long listing entity 776
geometric_representation_item
AIM diagrams 965
geometric_set_select
AIM EXPRESS long listing type 733
geometric_tolerance
AIM diagrams 988
AIM EXPRESS long listing entity 777
AIM EXPRESS short listing imported entity modifications 704
application assertion 198, 199
application object 84
ARM diagams 953
mapping table 485
Geometric_tolerance_precedence_relationship
application assertion 199
application object 86
ARM diagams 953
mapping table 486, 487
geometric_tolerance_relationship
AIM diagrams 989
AIM EXPRESS long listing entity 777
geometric_tolerance_subtype_exclusiveness
AIM EXPRESS long listing rule 857
AIM EXPRESS short listing rules 714
geometric_tolerance_with_datum_reference
AIM diagrams 988
AIM EXPRESS long listing entity 777
geometric_tolerance_with_defined_unit
AIM diagrams 988
AIM EXPRESS long listing entity 777
get_basis_surface
AIM EXPRESS long listing function 871

 ©ISO 2000 - All rights reserved

get_description_value
AIM EXPRESS long listing function 871
get_id_value
AIM EXPRESS long listing function 871
get_name_value
AIM EXPRESS long listing function 872
get_role
AIM EXPRESS long listing function 872
global_uncertainty_assigned_context
AIM diagrams 984
AIM EXPRESS long listing entity 777
global_unit_assigned_context
AIM diagrams 983
AIM EXPRESS long listing entity 777
Groove
application assertion 199
application object 87
ARM diagams 941
mapping table 371
group
AIM diagrams 963
AIM EXPRESS long listing entity 777
group_assignment
AIM diagrams 963
AIM EXPRESS long listing entity 777
group_item
AIM EXPRESS long listing type 733
AIM EXPRESS short listing types 544
group_relationship
AIM diagrams 963
AIM EXPRESS long listing entity 778
Hardness
application assertion 201
application object 88
ARM diagams 931
mapping table 463
Hole
application object 88
ARM diagams 942
mapping table 372
hole_bottom
AIM diagrams 961
AIM EXPRESS long listing entity 778
AIM EXPRESS short listing entities 591
mapping table 231, 235-237, 242, 244-247, 269, 270, 431, 436
hyperbola
AIM diagrams 968
AIM EXPRESS long listing entity 781
id_attribute
AIM diagrams 992

©ISO 2000 - All rights reserved

AIM EXPRESS long listing entity 781
id_attribute_select
AIM EXPRESS long listing type 733
identifier
AIM EXPRESS long listing type 733
Implementation method 4
Implicit_base_shape_representation
application assertion 199
application object 88
ARM diagams 933
mapping table 526
Indirect_stock_requisition
application object 90
ARM diagams 929
mapping table 521
instanced_feature
AIM diagrams 962
AIM EXPRESS long listing entity 781
AIM EXPRESS short listing entities 595
mapping table . 305, 308, 310, 312, 315, 319, 320, 333, 334, 340-345, 349, 352, 354, 359, 361, 363, 366-369, 371-377, 382, 384-386, 389, 395-397, 400, 403, 404, 412, 413, 424-427, 429, 436, 438, 441, 444, 446, 448, 449, 455, 456
Integrated resource 4
item_in_context
AIM EXPRESS long listing function 872
knot_type
AIM EXPRESS long listing type 733
Knurl
application assertion 199
application object 90
ARM diagams 946
mapping table 372
label
AIM EXPRESS long listing type 733
leap_year
AIM EXPRESS long listing function 873
length_measure
AIM EXPRESS long listing type 733
length_measure_with_unit
AIM diagrams 984
AIM EXPRESS long listing entity 781
length_unit
AIM diagrams 983
AIM EXPRESS long listing entity 781
limit_condition
AIM EXPRESS long listing type 733
limits_and_fits
AIM diagrams 987
AIM EXPRESS long listing entity 781
application assertion 222

 ©ISO 2000 - All rights reserved

application object 91
ARM diagams 952
mapping table 487
line
AIM diagrams 968
AIM EXPRESS long listing entity 781
Line_profile_tolerance 487, 488
AIM diagrams 988
AIM EXPRESS long listing entity 781
Linear_path
application assertion 186, 199, 208-210, 216, 217, 220
application object 91
ARM diagams 949
mapping table 252
linear_profile
AIM diagrams 961
AIM EXPRESS long listing entity 782
AIM EXPRESS short listing entities 596
application assertion 200, 208, 215
application object 92
ARM diagams 951
mapping table 273, 274, 278, 282, 362, 364, 388, 396, 399, 426
Linear_profile_tolerance
application assertion 200
application object 93
ARM diagams 953
mapping table 487
list_face_loops
AIM EXPRESS long listing function 873
list_of_reversible_topology_item
AIM EXPRESS long listing type 733
list_of_topology_reversed
AIM EXPRESS long listing function 873
list_to_array
AIM EXPRESS long listing function 873
list_to_set
AIM EXPRESS long listing function 874
Location_dimension_tolerance
application assertion 200
application object 94
ARM diagams 952
mapping table 488
Location_element
application assertion 207-209
application object 95
ARM diagams 934
mapping table 527
location_shape_representation
AIM diagrams 959
AIM EXPRESS long listing entity 782

©ISO 2000 - All rights reserved

AIM EXPRESS short listing entities 598
mapping table 259, 263, 527
Location_tolerance
application assertion 200
application object 95
ARM diagams 952
mapping table 489
loop
AIM diagrams 974
AIM EXPRESS long listing entity 783
Machine_requisition
application object 96
ARM diagams 929
mapping table 521
Machining_feature
application assertion 190, 200, 215
application object 96
ARM diagams 935
mapping table 332, 333, 374
machining_feature_life_cycle
AIM EXPRESS long listing rule 857
AIM EXPRESS short listing rules 715
make_array_of_array
AIM EXPRESS long listing function 874
make_from_usage_option
AIM diagrams 957
AIM EXPRESS long listing entity 783
AIM EXPRESS short listing imported listing modifications 704
manifold_solid_brep
AIM diagrams 975
AIM EXPRESS long listing entity 783
Manufactured_assembly
application assertion 202
application object 96
ARM diagams 928
mapping table 511
Manufactured_assembly_relationship
application object 96
ARM diagams 928
Manufacturing_feature
application assertion 201
application object 98
ARM diagams 935
mapping table 374
Manufacturing_feature_group
application assertion 201
application object 98
ARM diagams 935
mapped_item
AIM diagrams 960

 ©ISO 2000 - All rights reserved

AIM EXPRESS long listing entity 783
mapping table 529
marking
AIM diagrams 962
AIM EXPRESS long listing entity 783
AIM EXPRESS short listing entities 598
application assertion 201
application object 98
ARM diagams 945
mapping table 345-349, 375, 376
Material
application assertion 186, 201, 218
application object 99
ARM diagams 931
mapping table 465
Material_condition_modifier
application assertion 192, 198
application object 100
ARM diagams 954
mapping table 489
material_designation
AIM diagrams 991
AIM EXPRESS long listing entity 785
material_is_specified_for_part
AIM EXPRESS long listing rule 857
AIM EXPRESS short listing rules 716
material_property
AIM diagrams 958
AIM EXPRESS long listing entity 785
application assertion 201, 202, 211
application object 100
ARM diagams 931
mapping table 466
material_property_representation
AIM diagrams 991
AIM EXPRESS long listing entity 785
Material_requisition
application object 101
ARM diagams 929
mapping table 521
Mating_definition
application assertion 202
application object 101
Mating_definition_relationship
application assertion 202
application object 102
ARM diagams 928
Mating_definiton
ARM diagams 928
Mating_relationship

©ISO 2000 - All rights reserved

application assertion 202
application object 102
ARM diagams 928
measure_qualification
AIM diagrams 990
AIM EXPRESS long listing entity 785
measure_representation_item
AIM diagrams 990
AIM EXPRESS long listing entity 785
measure_value
AIM EXPRESS long listing type 733
measure_with_unit
AIM diagrams 984
AIM EXPRESS long listing entity 785
mixed_loop_type_set
AIM EXPRESS long listing function 874
Model 4
modified_geometric_tolerance
AIM diagrams 988
AIM EXPRESS long listing entity 786
modified_pattern
AIM diagrams 961
AIM EXPRESS long listing entity 786
AIM EXPRESS short listing entities 601
Modular_fixture_requisition
application object 103
ARM diagams 929
mapping table 521
month_in_year_number
AIM EXPRESS long listing type 734
msb_shells
AIM EXPRESS long listing function 875
Multi_axis_feature
application object 103
ARM diagams 936
mapping table 376
name_attribute
AIM diagrams 992
AIM EXPRESS long listing entity 786
name_attribute_select
AIM EXPRESS long listing type 734
named_unit
AIM diagrams 983
AIM EXPRESS long listing entity 786
AIM EXPRESS short listing imported entity modifications 705
next_assembly_usage_occurrence
AIM diagrams 957
AIM EXPRESS long listing entity 786
Ngon_base_shape
application assertion 203

 ©ISO 2000 - All rights reserved

application object 103
ARM diagams 933
mapping table 527
ngon_closed_profile
AIM diagrams 961
AIM EXPRESS long listing entity 786
AIM EXPRESS short listing entities 602
mapping table 261, 272, 274, 276-278, 282, 361, 362, 364, 367, 370, 399
Ngon_profile
application assertion 203
application object 104
ARM diagams 950
mapping table 274
ngon_shape_representation
AIM diagrams 959
AIM EXPRESS long listing entity 788
AIM EXPRESS short listing entities 604
mapping table 527-529
non_instantiable_description_attribute
AIM EXPRESS long listing rule 857
AIM EXPRESS short listing rules 716
non_instantiable_object_role
AIM EXPRESS long listing rule 858
AIM EXPRESS short listing rules 717
non_instantiable_role_association
AIM EXPRESS long listing rule 858
AIM EXPRESS short listing rules 717
normalise
AIM EXPRESS long listing function 875
Numeric_parameter
application assertion . 186-189, 191-194, 196, 197, 199, 200, 203, 204, 206, 208-210, 213-215, 217, 219, 221-224
application object 106
ARM diagams 932
mapping table 466
Numeric_parameter_with_tolerance
application assertion 203
application object 106
ARM diagams 932
mapping table 467
object_role
AIM diagrams 992
AIM EXPRESS long listing entity 788
AIM EXPRESS short listing imported entity modifications 705, 706
open_path_profile
AIM diagrams 961
AIM EXPRESS long listing entity 788
AIM EXPRESS short listing entities 606
mapping table 273, 278, 279, 282, 362, 364, 367, 369-371, 399, 442
Open_profile

©ISO 2000 - All rights reserved

application assertion 199, 203
application object 107
ARM diagams 951
mapping table 278
open_shell
AIM diagrams 975
AIM EXPRESS long listing entity 789
open_shell_reversed
AIM EXPRESS long listing function 876
Open_slot_end_type
application object 107
ARM diagams 944
mapping table 253
ordered_part
AIM diagrams 981
AIM EXPRESS long listing entity 789
AIM EXPRESS short listing entities 608
application assertion 192, 205
application object 107
ARM diagams 930
mapping table 472-474, 516
ordinal_date
AIM diagrams 980
AIM EXPRESS long listing entity 790
organization
AIM diagrams 978
AIM EXPRESS long listing entity 790
application assertion 205, 207
application object 108
ARM diagams 930
mapping table 509
organization_assignment
AIM diagrams 979
AIM EXPRESS long listing entity 790
organization_role
AIM diagrams 979
AIM EXPRESS long listing entity 790
organizational_address
AIM diagrams 978
Orientation
application assertion 197, 199, 200, 204-207, 210, 215, 220
application object 108
ARM diagams 936
mapping table 529
oriented_closed_shell
AIM diagrams 975
AIM EXPRESS long listing entity 790
oriented_edge
AIM diagrams 974
AIM EXPRESS long listing entity 791

 ©ISO 2000 - All rights reserved

oriented_face
AIM diagrams 974
AIM EXPRESS long listing entity 791
oriented_open_shell
AIM diagrams 975
AIM EXPRESS long listing entity 791
oriented_path
AIM diagrams 974
AIM EXPRESS long listing entity 791
orthogonal_complement
AIM EXPRESS long listing function 876
Outer_diameter
application assertion 204
application object 109
ARM diagams 944
mapping table 377
Outer_diameter_to_shoulder
application assertion 204
application object 110
ARM diagams 944
mapping table 382
outer_round
AIM diagrams 962
AIM EXPRESS long listing entity 792
AIM EXPRESS short listing entities 609
application object 111
ARM diagams 944
mapping table 377-384
outside_profile
AIM diagrams 962
AIM EXPRESS long listing entity 793
AIM EXPRESS short listing entities 611
mapping table 363, 364, 441
parabola
AIM diagrams 968
AIM EXPRESS long listing entity 797
parallelism_tolerance
AIM diagrams 988
AIM EXPRESS long listing entity 798
application assertion 204
application object 112
ARM diagams 953
mapping table 489-491
parameter_value
AIM EXPRESS long listing type 734
parametric_representation_context
AIM diagrams 960
AIM EXPRESS long listing entity 798
Part
application assertion 193, 195, 205, 210

©ISO 2000 - All rights reserved

application object 113
ARM diagams 927
mapping table 515
Part_placement
application assertion 202, 205, 206
application object 116
ARM diagams 928
mapping table 529
Part_property
application assertion 206, 211
application object 116
ARM diagams 931
mapping table 467
part_requires_project_order
AIM EXPRESS long listing rule 858
AIM EXPRESS short listing rules 717
part_to_approval
AIM EXPRESS long listing rule 858
AIM EXPRESS short listing rules 718
Partial_area_definition
application assertion 199, 206, 222
application object 116
ARM diagams 946
mapping table 253
Partial_circular_path
application assertion 206
application object 117
ARM diagams 949
mapping table 256
partial_circular_profile
AIM diagrams 961
AIM EXPRESS long listing entity 798
AIM EXPRESS short listing entities 617
application assertion 206, 215, 217
application object 118
ARM diagams 951
mapping table 278-282, 362, 364, 367, 370, 371, 399, 427, 438, 442
Partial_circular_shape_profile
application assertion 206
application object 119
ARM diagams 940
mapping table 385
path
AIM diagrams 974
AIM EXPRESS long listing entity 799
application assertion 206, 219
application object 120
ARM diagams 949
mapping table 257
Path_element

 ©ISO 2000 - All rights reserved

application assertion 197
application object 120
ARM diagams 934
mapping table 530
path_feature_component
AIM diagrams 961
AIM EXPRESS long listing entity 799
AIM EXPRESS short listing entities 620
mapping table 234, 235, 247, 248, 252, 253, 256-258, 306, 387, 393, 435, 437, 443, 447
path_head_to_tail
AIM EXPRESS long listing function 876
path_reversed
AIM EXPRESS long listing function 877
path_shape_representation
AIM diagrams 959
AIM EXPRESS long listing entity 801
AIM EXPRESS short listing entities 623
mapping table 248, 273, 530
pattern_offset_membership
AIM diagrams 963
AIM EXPRESS long listing entity 801
AIM EXPRESS short listing entities 624
mapping table 321, 326, 405, 420
pattern_omit_membership
AIM diagrams 963
AIM EXPRESS long listing entity 804
AIM EXPRESS short listing entities 629
mapping table 324, 327, 410, 421
pcurve
AIM diagrams 968
AIM EXPRESS long listing entity 806
pcurve_or_surface
AIM EXPRESS long listing type 734
Pedigree_creation_order
application assertion 210
application object 120
ARM diagams 929
mapping table 474
perpendicularity_tolerance
AIM diagrams 988
AIM EXPRESS long listing entity 806
application assertion 207
application object 120
ARM diagams 953
mapping table 491-493
person
AIM diagrams 978
AIM EXPRESS long listing entity 806
application assertion 207
application object 122

©ISO 2000 - All rights reserved

ARM diagams 930
mapping table 510
person_and_organization
AIM diagrams 978
AIM EXPRESS long listing entity 807
person_and_organization_assignment
AIM diagrams 979
AIM EXPRESS long listing entity 807
person_and_organization_role
AIM diagrams 979
AIM EXPRESS long listing entity 807
Person_in_organization
application assertion 186, 192, 205, 207
application object 122
ARM diagams 930
mapping table 510
person_organization_select
AIM EXPRESS long listing type 734
personal_address
AIM diagrams 978
AIM EXPRESS long listing entity 807
placed_datum_target_feature
AIM diagrams 986
AIM EXPRESS long listing entity 807
AIM EXPRESS short listing entities 633
mapping table 493, 494, 501-504
Placed_target
application assertion 207
application object 123
ARM diagams 954
mapping table 493
placement
AIM diagrams 967
AIM EXPRESS long listing entity 809
Planar_element
application assertion 203, 207, 208
application object 123
ARM diagams 934
mapping table 531
Planar_face
application assertion 207, 208
application object 124
ARM diagams 941
mapping table 386
Planar_pocket_bottom_condition
application assertion 208
application object 125
ARM diagams 939
mapping table 258
Planar_profile_floor

 ©ISO 2000 - All rights reserved

application assertion 208
application object 126
Planar_rib_top_floor
application assertion 208
application object 126
ARM diagams 939
mapping table 261
Planar_shape_representation 261
AIM diagrams 959
AIM EXPRESS long listing entity 809
AIM express short listing entities 636
Planar_top_condition
application assertion 208, 209
application object 126
ARM diagams 943
mapping table 262
plane
AIM diagrams 972
AIM EXPRESS long listing entity 809
mapping table 531
plane_angle_measure
AIM EXPRESS long listing type 734
plane_angle_measure_with_unit
AIM diagrams 984
AIM EXPRESS long listing entity 809
plane_angle_unit
AIM diagrams 983
AIM EXPRESS long listing entity 810
plus_minus_tolerance
AIM diagrams 987
AIM EXPRESS long listing entity 810
Plus_minus_value
application assertion 203, 222
application object 127
ARM diagams 952
mapping table 494
pocket
AIM diagrams 962
AIM EXPRESS long listing entity 810
AIM EXPRESS short listing entities 637
application assertion 209
application object 128
ARM diagams 938
mapping table 320, 321, 345, 361, 366, 367, 389, 391-399, 403, 404, 412, 413
pocket_bottom
AIM diagrams 961
AIM EXPRESS long listing entity 813
AIM EXPRESS short listing entities 642
mapping table 248-250, 258-260, 263, 264, 270, 345, 394, 395, 398
Pocket_bottom_condition

©ISO 2000 - All rights reserved

application assertion 209, 212
application object 129
ARM diagams 939
mapping table 263
point
AIM diagrams 966
AIM EXPRESS long listing entity 815
polyline
AIM diagrams 969
AIM EXPRESS long listing entity 815
position_tolerance
AIM diagrams 988
AIM EXPRESS long listing entity 815
application assertion 209, 210
application object 130
ARM diagams 953
mapping table 495-497
positive_length_measure
AIM EXPRESS long listing type 734
positive_plane_angle_measure
AIM EXPRESS long listing type 734
precision_qualifier
AIM diagrams 990
AIM EXPRESS long listing entity 815
AIM EXPRESS short listing imported entity modifications 705
preferred_surface_curve_representation
AIM EXPRESS long listing type 734
Process_property
application assertion 210, 211
application object 131
ARM diagams 931
mapping table 467
product
AIM diagrams 957
AIM EXPRESS long listing entity 815
AIM EXPRESS short listing imported entity modifications 705
definition 4
Product data 4
product_context
AIM diagrams 956
AIM EXPRESS long listing entity 816
product_definition
AIM diagrams 957
AIM EXPRESS long listing entity 816
AIM EXPRESS short listing imported entity modifications 705
product_definition_context
AIM diagrams 956
AIM EXPRESS long listing entity 816
product_definition_formation
AIM diagrams 957

 ©ISO 2000 - All rights reserved

AIM EXPRESS long listing entity 816
AIM EXPRESS short listing imported entity modifications 706
product_definition_formation_requires_security_classification
AIM EXPRESS long listing rule 858
AIM EXPRESS short listing rules 719
product_definition_relationship
AIM diagrams 957
AIM EXPRESS long listing entity 816
product_definition_shape
AIM diagrams 958
AIM EXPRESS long listing entity 816
product_definition_usage
AIM diagrams 957
AIM EXPRESS long listing entity 816
product_definition_with_associated_documents
AIM EXPRESS long listing entity 817
product_definition_with_associated_documents
AIM diagrams 957
product_or_formation_or_definition
AIM EXPRESS long listing type 734
product_requires_version
AIM EXPRESS long listing rule 858
AIM EXPRESS short listing rules 719
Profile
application assertion 197, 198, 210, 219
application object 131
ARM diagams 950
mapping table 282
Profile_feature
application assertion 210
application object 132
ARM diagams 940
mapping table 395
profile_floor
AIM diagrams 961
AIM EXPRESS long listing entity 817
AIM EXPRESS short listing enities 645
application assertion 210, 218
application object 132
ARM diagams 940
mapping table 264-266, 439, 440
Project_order
application assertion 192, 210, 211
application object 132
ARM diagams 929
mapping table 474
project_order_requires_approval
AIM EXPRESS long listing rule 859
AIM EXPRESS short listing rules 720
project_order_tracking_relationships

©ISO 2000 - All rights reserved

AIM EXPRESS long listing rule 859
AIM EXPRESS short listing rules 721
projected_zone_definition
AIM diagrams 989
AIM EXPRESS long listing entity 818
Projection
application assertion 223
application object 134
ARM diagams 954
mapping table 497
Property
application assertion 205, 211, 212
application object 134
ARM diagams 931
mapping table 468
property_definition
AIM diagrams 958
AIM EXPRESS long listing entity 818
property_definition_relationship
AIM diagrams 958
AIM EXPRESS long listing entity 818
property_definition_representation
AIM diagrams 958
AIM EXPRESS long listing entity 818
Property_parameter
application assertion 202, 206, 210, 220
application object 136
ARM diagams 932
mapping table 469
protrusion
AIM diagrams 962
AIM EXPRESS long listing entity 819
AIM EXPRESS short listing entities 648
application assertion 212
application object 136
ARM diagams 941
mapping table 396, 397
qualified_representation_item
AIM diagrams 990
AIM EXPRESS long listing entity 819
qualitative_uncertainty
AIM diagrams 990
AIM EXPRESS long listing entity 819
quasi_uniform_curve
AIM diagrams 969
AIM EXPRESS long listing entity 819
quasi_uniform_surface
AIM diagrams 973
AIM EXPRESS long listing entity 819
Radial_dimension_tolerance

 ©ISO 2000 - All rights reserved

application object 137
ARM diagams 952
mapping table 498
Radiused_slot_end_type
application object 137
ARM diagams 944
mapping table 266
ratio_measure
AIM EXPRESS long listing type 734
ratio_measure_with_unit
AIM diagrams 984
ratio_unit
AIM diagrams 983
AIM EXPRESS long listing entity 820
rational_b_spline_curve
AIM diagrams 969
AIM EXPRESS long listing entity 820
rational_b_spline_surface
AIM diagrams 973
AIM EXPRESS long listing entity 820
Recess
application assertion 212
application object 138
ARM diagams 938
mapping table 397
Rectangular_boss
application assertion 212
application object 139
ARM diagams 943
mapping table 400
Rectangular_closed_pocket
application assertion 212
application object 139
ARM diagams 938
mapping table 403
rectangular_closed_profile
AIM diagrams 961
AIM EXPRESS long listing entity 820
AIM EXPRESS short listing entities 649
application assertion 212, 213
application object 140
ARM diagams 950
mapping table 261, 272, 282-286, 362, 364, 367, 370, 399, 403-405
Rectangular_closed_shape_profile
application assertion 213
application object 141
ARM diagams 940
mapping table 404
Rectangular_offset_pattern
application assertion 213, 214

©ISO 2000 - All rights reserved

application object 142
ARM diagams 947
mapping table 405
Rectangular_omit_pattern
application assertion 213, 214
application object 143
ARM diagams 947
mapping table 410
Rectangular_open_pocket
application assertion 214
application object 144
ARM diagams 938
mapping table 412
Rectangular_open_shape_profile
application assertion 214
application object 145
ARM diagams 940
mapping table 413
rectangular_pattern
AIM diagrams 963
AIM EXPRESS long listing entity 822
AIM EXPRESS short listing entities 651
application assertion 214
application object 145
ARM diagams 947
mapping table 414-421
referenced_modified_datum
AIM diagrams 986
AIM EXPRESS long listing entity 824
removal_volume
AIM diagrams 962
AIM EXPRESS long listing entity 824
AIM EXPRESS short listing entities 655
mapping table 367, 368
Replicate_base
application assertion 215
application object 147
ARM diagams 947
mapping table 421
replicate_feature
AIM diagrams 963
AIM EXPRESS long listing entity 824
AIM EXPRESS short listing entities 656
application assertion 215
application object 147
ARM diagams 947
mapping table 326-331, 365, 415-423
representation
AIM diagrams 960
AIM EXPRESS long listing entity 825

 ©ISO 2000 - All rights reserved

AIM EXPRESS short listing imported entity modifications 706
representation_context
AIM diagrams 960
AIM EXPRESS long listing entity 825
representation_item
AIM diagrams 960
AIM EXPRESS long listing entity 825
representation_map
AIM diagrams 960
AIM EXPRESS long listing entity 825
representation_relationship
AIM diagrams 960
AIM EXPRESS long listing entity 825
representation_subtype_exclusiveness
AIM EXPRESS long listing rule 860
AIM EXPRESS short listing rules 723
represented_definition
AIM EXPRESS long listing type 734
Requisition
application assertion 211
application object 147
ARM diagams 929
mapping table 521
Resource_acquisition_order
application assertion 211
application object 148
ARM diagams 929
mapping table 477
restrict_approval_status
AIM EXPRESS long listing rule 860
AIM EXPRESS short listing rules 723
restrict_security_classification_level
AIM EXPRESS long listing rule 860
AIM EXPRESS short linsting rules 724
reversible_topology
AIM EXPRESS long listing type 735
reversible_topology_item
AIM EXPRESS long listing type 735
Revolved_feature
application assertion 215
application object 149
ARM diagams 941
mapping table 424
Revolved_flat
application assertion 215
application object 149
ARM diagams 941
mapping table 425
revolved_profile
AIM diagrams 962

©ISO 2000 - All rights reserved

AIM EXPRESS long listing entity 826
AIM EXPRESS short listing entities 657
mapping table 368, 369, 371, 424-427
Revolved_round
application assertion 215
application object 150
ARM diagams 941
mapping table 426
rib_top
AIM diagrams 962
AIM EXPRESS long listing entity 827
AIM EXPRESS short listing enities 661
application assertion 216
application object 151
ARM diagams 939
mapping table 427
rib_top_floor
AIM diagrams 961
AIM EXPRESS long listing entity 828
AIM EXPRESS short listing enities 661
application assertion 216
application object 152
ARM diagams 939
mapping table 267
ribtop_floor
mapping table 429
role_association
AIM diagrams 992
AIM EXPRESS long listing entity 829
AIM EXPRESS short listing imported entity modifications 706
role_select
AIM EXPRESS long listing type 735
round_hole
AIM diagrams 962
AIM EXPRESS long listing entity 829
AIM EXPRESS short listing entities 663
application assertion 191, 192, 216, 217
application object 152
ARM diagams 942
mapping table 341-344, 372, 429-436
rounded_end
AIM diagrams 962
AIM EXPRESS long listing entity 830
AIM EXPRESS short listing entities 666
application assertion 217
application object 154
ARM diagams 941
mapping table 436-438
rounded_u_profile
AIM diagrams 961

 ©ISO 2000 - All rights reserved

AIM EXPRESS long listing entity 831
AIM EXPRESS short listing entities 667
application assertion 217
application object 155
ARM diagams 951
mapping table 278, 282, 286, 287, 362, 364, 367, 370, 371, 399, 442
roundness_tolerance
AIM diagrams 988
AIM EXPRESS long listing entity 832
runout_zone_definition
AIM diagrams 989
runout_zone_orientation
AIM diagrams 989
AIM EXPRESS long listing entity 832
scalar_times_vector
AIM EXPRESS long listing function 877
Second_chamfer_offset
application assertion 188, 217
application object 155
ARM diagams 937
mapping table 267
Second_offset
application assertion 217
application object 156
ARM diagams 937
mapping table 267
second_proj_axis
AIM EXPRESS long listing function 878
security_classification
AIM diagrams 982
AIM EXPRESS long listing entity 832
security_classification_assignment
AIM diagrams 982
AIM EXPRESS long listing entity 832
security_classification_level
AIM diagrams 982
AIM EXPRESS long listing entity 833
AIM EXPRESS short listing imported entity modifications 707
set_of_reversible_topology_item
AIM EXPRESS long listing type 735
set_of_topology_reversed
AIM EXPRESS long listing function 878
Shape
application assertion 205, 206, 217
application object 156
ARM diagams 933
mapping table 531
shape_aspect
AIM diagrams 961
AIM EXPRESS long listing entity 833

©ISO 2000 - All rights reserved

AIM EXPRESS short listing imported entity modifications 707
application assertion 194, 198, 211, 217, 218
application object 157
ARM diagams 933
mapping table 533
shape_aspect_relationship
AIM diagrams 958
AIM EXPRESS long listing entity 833
AIM EXPRESS short listing imported entity modifications 707
shape_aspect_relationship_subtype_exclusiveness
AIM EXPRESS long listing rule 860
AIM EXPRESS short listing rules 725
shape_aspect_subtype_exclusiveness
AIM EXPRESS long listing rule 860
AIM EXPRESS short listing rules 726
shape_defining_relationship
AIM diagrams 958
AIM EXPRESS long listing entity 833
AIM EXPRESS short listing entities 669
mapping table . . 306, 319-321, 361, 362, 364, 367-371, 373, 384, 385, 387-389, 393, 396, 397, 399, 403-405, 413, 414, 426, 427, 432, 435, 437, 438, 442, 443, 447, 448, 455
shape_definition
AIM EXPRESS long listing type 735
shape_definition_representation
AIM diagrams 958
AIM EXPRESS long listing entity 833
shape_dimension_representation
AIM diagrams 987
AIM EXPRESS long listing entity 833
Shape_element
application assertion 193, 199-202, 212, 218, 219, 222, 223
application object 158
ARM diagams 934
mapping table 534
Shape_profile
application assertion 218
application object 158
ARM diagams 940
mapping table 438
shape_representation
AIM diagrams 959
AIM EXPRESS long listing entity 833
AIM EXPRESS short listing imported entity modifications 707
shape_representation_subtype_exclusiveness
AIM EXPRESS long listing rule 861
AIM EXPRESS short listing rules 727
shape_representation_with_parameters
AIM diagrams 959
AIM EXPRESS long listing entity 833
AIM EXPRESS short listing entities 669

 ©ISO 2000 - All rights reserved

mapping table . . 230, 233-238, 240, 242-247, 250, 253-258, 260, 265, 266, 268, 269, 271, 272, 274-278, 280, 281, 283-304, 307, 322, 323, 325, 328-331, 335-340, 346-355, 365, 374, 381-383, 390, 407-409, 411, 412, 416-419, 423, 425, 445, 446, 449-454, 456-462, 494, 501-504, 523, 525-529
shell
AIM EXPRESS long listing type 735
shell_reversed
AIM EXPRESS long listing function 878
Shop_work_order
application assertion 211
application object 159
ARM diagams 929
mapping table 477
si_prefix
AIM EXPRESS long listing type 735
si_unit
AIM diagrams 983
AIM EXPRESS long listing entity 834
si_unit_name
AIM EXPRESS long listing type 736
Single_piece_part
application assertion 202, 218
application object 159
ARM diagams 928
mapping table 518
Size_tolerance
application assertion 219
application object 160
ARM diagams 952
mapping table 498
slot
AIM diagrams 962
AIM EXPRESS long listing entity 834
AIM EXPRESS short listing entities 670
application assertion 219
application object 160
ARM diagams 944
mapping table 441-444
slot_end
AIM diagrams 961
AIM EXPRESS long listing entity 835
AIM EXPRESS short listing entities 672
mapping table 242-244, 253, 266, 268, 269, 271, 444
Slot_end_type
application assertion 219
application object 162
ARM diagams 944
mapping table 268
solid_angle_measure_with_unit
AIM diagrams 984

©ISO 2000 - All rights reserved

AIM EXPRESS long listing entity 836
solid_angle_unit
AIM diagrams 983
AIM EXPRESS long listing entity 837
solid_model
AIM diagrams 975
AIM EXPRESS long listing entity 837
source_item
AIM EXPRESS long listing type 736
Specification
application assertion 187, 201, 211, 219
application object 162
ARM diagams 931
mapping table 478
Specification_usage_constraint
application assertion 219
application object 163
mapping table 478
spherical_cap
AIM diagrams 962
AIM EXPRESS long listing entity 837
AIM EXPRESS short listing entities 675
application assertion 219
application object 163
ARM diagams 941
mapping table 444-446
Spherical_hole_bottom
application assertion 219
application object 164
ARM diagams 942
mapping table 269
spherical_surface
AIM diagrams 972
AIM EXPRESS long listing entity 838
square_u_profile
AIM diagrams 961
AIM EXPRESS long listing entity 838
AIM EXPRESS short listing entities 677
application assertion 214, 219
application object 165
ARM diagams 951
mapping table 278, 282, 287-292, 362, 364, 367, 370, 371, 399, 413, 414, 442
standard_uncertainty
AIM diagrams 990
AIM EXPRESS long listing entity 840
step
AIM diagrams 962
AIM EXPRESS long listing entity 840
AIM EXPRESS short listing entities 680
application assertion 220

 ©ISO 2000 - All rights reserved

application object 166
ARM diagams 941
mapping table 446-448
Straight_knurl
application object 167
ARM diagams 946
mapping table 448
straightness_tolerance
AIM diagrams 988
AIM EXPRESS long listing entity 841
application assertion 220
application object 167
ARM diagams 953
mapping table 498
subtype_mandatory_characterized_object
AIM EXPRESS long listing rule 861
AIM EXPRESS short listing rules 727
supported_item
AIM EXPRESS long listing type 736
surface
AIM diagrams 971
AIM EXPRESS long listing entity 841
surface_curve
AIM diagrams 970
AIM EXPRESS long listing entity 841
surface_of_linear_extrusion
AIM diagrams 971
AIM EXPRESS long listing entity 841
surface_of_revolution
AIM diagrams 971
AIM EXPRESS long listing entity 841
surface_profile_tolerance
AIM diagrams 988
AIM EXPRESS long listing entity 841
application assertion 220
application object 169
ARM diagams 953
mapping table 499
Surface_property
application assertion 212, 220
application object 170
ARM diagams 931
mapping table 469
surface_weights_positive
AIM EXPRESS long listing function 878
swept_surface
AIM diagrams 971
AIM EXPRESS long listing entity 841
symmetry_tolerance
AIM diagrams 988

©ISO 2000 - All rights reserved

AIM EXPRESS long listing entity 842
application assertion 220
application object 170
ARM diagams 953
mapping table 500
taper
AIM diagrams 961
AIM EXPRESS long listing entity 842
AIM EXPRESS short listing entities 682
mapping table . . 229, 230, 237-240, 316-318, 344, 360, 378-380, 391, 392, 400-402, 430, 433, 434
Target_area
application assertion 221
application object 171
ARM diagams 954
mapping table 500
Target_circle
application object 171
ARM diagams 954
mapping table 501
Target_line
application object 172
ARM diagams 954
mapping table 502
Target_point
application object 172
ARM diagams 954
mapping table 502
Target_rectangle
application object 172
ARM diagams 954
mapping table 503
tee_profile
AIM diagrams 961
AIM EXPRESS long listing entity 843
AIM EXPRESS short listing entities 684
application assertion 221
application object 172
ARM diagams 951
mapping table 278, 282, 292-301, 362, 364, 367, 370, 371, 399, 442
text
AIM EXPRESS long listing type 736
thread
AIM diagrams 962
AIM EXPRESS long listing entity 846
AIM EXPRESS short listing entities 689
application assertion 221, 222
application object 175
ARM diagams 945
mapping table 349-352, 449-456

 ©ISO 2000 - All rights reserved

Through_bottom_condition
application assertion 216
application object 177
ARM diagams 942
mapping table 270
Through_pocket_bottom_condition
application assertion 192, 209
application object 177
ARM diagams 939
mapping table 270
Through_profile_floor
application assertion 218
application object 177
ARM diagams 940
mapping table 270
Tolerance_limit
application assertion 222
application object 177
ARM diagams 952
mapping table 504
Tolerance_range
application assertion 222
application object 178
ARM diagams 952
mapping table 504
tolerance_select
AIM EXPRESS long listing type 736
tolerance_value
AIM diagrams 987
AIM EXPRESS long listing entity 849
application assertion 194, 222
application object 178
ARM diagams 952
mapping table 505
tolerance_zone
AIM diagrams 989
AIM EXPRESS long listing entity 849
application assertion 198, 223
application object 179
ARM diagams 954
mapping table 506
tolerance_zone_definition
AIM diagrams 989
AIM EXPRESS long listing entity 849
application assertion 223
application object 180
ARM diagams 954
mapping table 507
tolerance_zone_form
AIM diagrams 989

©ISO 2000 - All rights reserved

AIM EXPRESS long listing entity 849
topological_representation_item
AIM diagrams 965
AIM EXPRESS long listing entity 849
topology_reversed
AIM EXPRESS long listing function 879
toroidal_surface
AIM diagrams 972
AIM EXPRESS long listing entity 849
total_runout_tolerance
AIM diagrams 988
AIM EXPRESS long listing entity 849
application assertion 223
application object 180
ARM diagams 953
mapping table 507, 508
transformation
AIM EXPRESS long listing type 736
transition_code
AIM EXPRESS long listing type 737
transition_feature
AIM diagrams 964
AIM EXPRESS long listing entity 849
AIM EXPRESS short listing entities 693
application assertion 190
application object 181
ARM diagams 937
mapping table 313-315, 333-340, 355-359, 456
transition_feature_life_cycle
AIM EXPRESS long listing rule 861
AIM EXPRESS short listing rules 728
transition_feature_on_part_boundary
AIM EXPRESS long listing rule 862
AIM EXPRESS short listing rules 729
trimming_select
AIM EXPRESS long listing type 737
turned_knurl
AIM diagrams 962
AIM EXPRESS long listing entity 850
AIM EXPRESS short listing entities 694
application assertion 223
application object 182
ARM diagams 946
mapping table 352-355, 372, 373, 448, 456-462
type_qualifier
AIM diagrams 990
AIM EXPRESS long listing entity 852
AIM EXPRESS short listing imported entity modifications 708
uncertainty_measure_with_unit
AIM diagrams 984

 ©ISO 2000 - All rights reserved

AIM EXPRESS long listing entity 852
uncertainty_qualifier
AIM diagrams 990
AIM EXPRESS long listing entity 852
AIM EXPRESS short listing imported entity modifications 708
uniform_curve
AIM diagrams 969
AIM EXPRESS long listing entity 852
uniform_surface
AIM diagrams 973
AIM EXPRESS long listing entity 852
unit
AIM EXPRESS long listing type 737
Unit of functionality 4
UoF 4
using_items
AIM EXPRESS long listing function 879
using_representations
AIM EXPRESS long listing function 880
valid_calendar_date
AIM EXPRESS long listing function 880
valid_measure_value
AIM EXPRESS long listing function 881
valid_units
AIM EXPRESS long listing function 881
value_qualifier
AIM EXPRESS long listing type 737
vector
AIM diagrams 967
vector_difference
AIM EXPRESS long listing function 883
vector_or_direction
AIM EXPRESS long listing type 737
vee_profile
AIM diagrams 961
AIM EXPRESS long listing entity 853
AIM EXPRESS short listing entities 698
application assertion 204, 220, 223
application object 183
ARM diagams 951
mapping table 278, 282, 301-304, 362, 364, 367, 370, 371, 384, 399, 442, 448
versioned_action_request
AIM diagrams 982
AIM EXPRESS long listing entity 854
mapping table 226
vertex
AIM diagrams 974
AIM EXPRESS long listing entity 854
vertex_loop
AIM diagrams 974

©ISO 2000 - All rights reserved

vertex_point
AIM diagrams 974
AIM EXPRESS long listing entity 855
week_in_year_number
AIM EXPRESS long listing type 737
week_of_year_and_day_date
AIM diagrams 980
AIM EXPRESS long listing entity 855
Woodruff_slot_end_type
application assertion 224
application object 185
ARM diagams 944
mapping table 271
year_number
AIM EXPRESS long listing type 737

 ©ISO 2000 - All rights reserved