# ISO TC184/SC4/WG3 N720 **Date:** 1998-01-16

**Supersedes ISO TC 184/SC4/WG3_ N 606**

**ISO/CD 10303-208**
**Product data representation and exchange - Application protocol: Life-cycle management - change process**


## COPYRIGHT NOTICE:

This ISO document is a working draft or committee draft and is copyright protected by ISO. While the reproduction of working drafts or committee drafts in any form for use by Participants in the ISO standards development process is permitted without prior permission from ISO, neither this document nor any extract from it may be reproduced, stored or transmitted in any form for any other purpose without prior written permission from ISO.

Requests for permission to reproduce this document for the purposes of selling it should be addressed as shown below (via the ISO TC 184/SC4 Secretariat's member body) or to ISO's member body in the country of the requester.

Copyright Manager
ANSI
11 West 42nd Street, New York, New York 10036, USA
phone: +1-212-642-4900, fax: +1-212-398-0023

Reproduction for sales purposes may be subject to royalty payments or a licensing agreement.

Violators may be prosecuted.



## ABSTRACT:

This document specifies the Application Protocol for the exchange of data required to support the management of change of configuration-worthy items. This includes the identification of a product anomaly, and its causes, the approval and performance of the resulting change (repairs) to the anomalized product of process definition, and the authorization of corrective actions to present anomaly reoccurrence.



## KEYWORDS:

application protocol, anomaly, change management, life-cycle management, authorization, fielded history, support resource


## COMMENTS TO READER:

"This document has been reviewed and noted by the ISO TC 184/SC4 Quality Committee and SC4 Secretariat and has been determined to be ready for this ballot cycle."


**Project Leader:** Chuck Amaral
**Address:** Boeing North American, Inc.
2600 Westminster Blvd. M/C SK49
Seal Beach CA 90740-7644 USA
**Telephone:** +1 562-797-4804
**Telefacsimile:** +1 562-797-4216
**Electronic mail:** charles.v.amaral@boeing.com

**Project Editor:** William C. Burkett
**Address:** P.D.I.T., Inc
100 W. Broadway, Suite 540
Long Beach CA 90802 USA
**Telephone:** 1-562-495-6500
**Telefacsimile:** 1-562-495-6509
**Electronic mail:** wburkett@pdit.com

# Contents

Page

1 Scope . . . . . 1

2 Normative references . . . . . 2

3 Definitions and abbreviations . . . . . 3
3.1 Terms defined in ISO 10303-1 . . . . . 3
3.2 Terms defined in ISO 10303-31 . . . . . 4
3.3 Terms defined in ISO 10303-44 . . . . . 4
3.4 Other definitions . . . . . 4
3.4.1 anomaly . . . . . 4
3.4.2 authorization . . . . . 4
3.4.3 change management . . . . . 4
3.4.4 change requirements . . . . . 4
3.4.5 corrective action . . . . . 5
3.4.6 fielded history . . . . . 5
3.4.7 item . . . . . 5
3.4.8 item version . . . . . 5
3.4.9 product functionality . . . . . 5
3.4.10 related change . . . . . 5
3.4.11 support resource . . . . . 5
3.4.12 task . . . . . 5
3.5 Abbreviations . . . . . 5

4 Information requirements . . . . . 6
4.1 Units of functionality . . . . . 6
4.1.1 change_definition UoF . . . . . 7
4.1.2 item_definition UoF . . . . . 7
4.1.3 item_properties UoF . . . . . 8
4.1.4 supporting_resources UoF . . . . . 9
4.1.5 task_definition UoF . . . . . 10
4.2 Application objects . . . . . 11
4.3 Application assertions . . . . . 45

5 Application interpreted model . . . . . 52
5.1 Mapping table . . . . . 52
5.2 AIM EXPRESS short listing . . . . . 125
5.2.1 AIM EXPRESS short listing types . . . . . 128
5.2.2 AIM EXPRESS short listing entities . . . . . 132
5.2.3 EXPRESS short listing rules . . . . . 144

6 Conformance requirements . . . . . 150

**Annexes**

A AIM EXPRESS expanded listing . . . . . 152

B AIM short names of entities . . . . . 191

C Implementation method-specific requirements . . . . . 198

D Protocol Implementation Conformance Statement proforma . . . . . 199

E Information object registration . . . . . 200

F Application activity model . . . . . 201
F.1 Application activity model definitions and abbreviations . . . . . 201
F.2 Application activity model diagrams . . . . . 205

G Application reference model . . . . . 212

H AIM EXPRESS-G . . . . . 225

J AIM EXPRESS listing . . . . . 254

K Bibliography . . . . . 255

Index . . . . . 256

**Figures**

Figure 1 - Life cycle management change process data planning model . . . . . xiii
Figure F.1 - IDEF0 basic notation . . . . . 205
Figure F.2 - A-0: Perform product change management (context) . . . . . 207
Figure F.3 - A0: Perform product change management . . . . . 208
Figure F.4 - A3: Identify and analyze change criteria . . . . . 209
Figure F.5 - A32: Develop solution . . . . . 210
Figure F.6 - A4: Implement change . . . . . 211
Figure G.1 - ARM diagram 1 of 12 . . . . . 213
Figure G.2 - ARM diagram 2 of 12 . . . . . 214
Figure G.3 - ARM diagram 3 of 12 . . . . . 215
Figure G.4 - ARM diagram 4 of 12 . . . . . 216
Figure G.5 - ARM diagram 5 of 12 . . . . . 217
Figure G.6 - ARM diagram 6 of 12 . . . . . 218
Figure G.7 - ARM diagram 7 of 12 . . . . . 219
Figure G.8 - ARM diagram 8 of 12 . . . . . 220
Figure G.9 - ARM diagram 9 of 12 . . . . . 221
Figure G.10 - ARM diagram 10 of 12 . . . . . 222

Figure G.11 - ARM diagram 11 of 12 . . . . . 223
Figure G.12 - ARM diagram 12 of 12 . . . . . 224
Figure H.1 - AIM EXPRESS-G diagram 1 of 28 . . . . . 226
Figure H.2 - AIM EXPRESS-G diagram 2 of 28 . . . . . 227
Figure H.3 - AIM EXPRESS-G diagram 3 of 28 . . . . . 228
Figure H.4 - AIM EXPRESS-G diagram 4 of 28 . . . . . 229
Figure H.5 - AIM EXPRESS-G diagram 5 of 28 . . . . . 230
Figure H.6 - AIM EXPRESS-G diagram 6 of 28 . . . . . 231
Figure H.7 - AIM EXPRESS-G diagram 7 of 28 . . . . . 232
Figure H.8 - AIM EXPRESS-G diagram 8 of 28 . . . . . 233
Figure H.9 - AIM EXPRESS-G diagram 9 of 28 . . . . . 234
Figure H.10 - AIM EXPRESS-G diagram 10 of 28 . . . . . 235
Figure H.11 - AIM EXPRESS-G diagram 11 of 28 . . . . . 236
Figure H.12 - AIM EXPRESS-G diagram 12 of 28 . . . . . 237
Figure H.13 - AIM EXPRESS-G diagram 13 of 28 . . . . . 238
Figure H.14 - AIM EXPRESS-G diagram 14 of 28 . . . . . 239
Figure H.15 - AIM EXPRESS-G diagram 15 of 28 . . . . . 240
Figure H.16 - AIM EXPRESS-G diagram 16 of 28 . . . . . 241
Figure H.17 - AIM EXPRESS-G diagram 17 of 28 . . . . . 242
Figure H.18 - AIM EXPRESS-G diagram 18 of 28 . . . . . 243
Figure H.19 - AIM EXPRESS-G diagram 19 of 28 . . . . . 244
Figure H.20 - AIM EXPRESS-G diagram 20 of 28 . . . . . 245
Figure H.21 - AIM EXPRESS-G diagram 21 of 28 . . . . . 246
Figure H.22 - AIM EXPRESS-G diagram 22 of 28 . . . . . 247
Figure H.23 - AIM EXPRESS-G diagram 23 of 28 . . . . . 248
Figure H.24 - AIM EXPRESS-G diagram 24 of 28 . . . . . 249
Figure H.25 - AIM EXPRESS-G diagram 25 of 28 . . . . . 250
Figure H.26 - AIM EXPRESS-G diagram 26 of 28 . . . . . 251
Figure H.27 - AIM EXPRESS-G diagram 27 of 28 . . . . . 252
Figure H.28 - AIM EXPRESS-G diagram 28 of 28 . . . . . 253

**Tables**

Table 1 - Mapping table for change_definition UoF . . . . . 54
Table 2 - Mapping table for item_definition UoF . . . . . 74
Table 3 - Mapping table for item_properties UoF . . . . . 94
Table 4 - Mapping table for supporting_resources UoF . . . . . 102
Table 5 - Mapping table for task_definition UoF . . . . . 107
Table B.1 - Short names of entities . . . . . 191

# Foreword

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

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

International Standard ISO 10303-208 was prepared by Technical Committee ISO/TC 184, *Industrial automation systems and integration*, Subcommittee SC4, *Industrial data*.

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

— Part 1, Overview and fundamental principles;

— Part 11, Description methods: The EXPRESS language reference manual;

— Part 12, Description method: The EXPRESS-I language reference manual;

— Part 21, Implementation methods: Clear text encoding of the exchange structure;

— Part 22, Implementation method: Standard data access interface specification;

— Part 23, Implementation method: C++ language binding to the standard data access interface;

— Part 24, Implementation method: C language binding to the standard data access interface;

— Part 26, Implementation method: Interface definition language binding to the standard data access;

— Part 31, Conformance testing methodology and framework: General concepts;

— Part 32, Conformance testing methodology and framework: Requirements on testing laboratories and clients;

— Part 33, Conformance testing methodology and framework: Structure and use of abstract test suites;

— Part 34, Conformance testing methodology and framework: Abstract test methods;

— Part 35, Conformance testing methodology and framework: Abstract test methods for SDAI implementations;

— Part 41, Integrated generic resources: Fundamentals of product description and support;

— Part 42, Integrated generic resources: Geometric and topological representation;

— Part 43, Integrated generic resources: Representation structures;

— Part 44, Integrated generic resources: Product structure configuration;

— Part 45, Integrated generic resource: Materials;

— Part 46, Integrated generic resources: Visual presentation;

— Part 47, Integrated generic resource: Shape variation tolerances;

— Part 49, Integrated generic resource: Process structure and properties;

— Part 101, Integrated application resource: Draughting;

— Part 104, Integrated application resource: Finite element analysis;

— Part 105, Integrated application resource: Kinematics;

— Part 106, Integrated application resource: Building construction core model;

— Part 201, Application protocol: Explicit draughting;

— Part 202, Application protocol: Associative draughting;

— Part 203, Application protocol: Configuration controlled design;

— Part 204, Application protocol: Mechanical design using boundary representation;

— Part 205, Application protocol: Mechanical design using surface representation;

— Part 207, Application protocol: Sheet metal die planning and design;

— Part 208, Application protocol: Life cycle management - Change process;

— Part 209, Application protocol: Composite and metallic structural analysis and related design;

— Part 210, Application protocol: Electronic assembly, interconnect, and packaging design;

— Part 212, Application protocol: Electrotechnical design and installation;

— Part 213, Application protocol: Numerical control process plans for machined parts;

— Part 214, Application protocol: Core data for automotive design;

— Part 215, Application protocol: Ship arrangement;

— Part 216, Application protocol: Ship moulded forms;

— Part 217, Application protocol: Ship piping;

— Part 218, Application protocol: Ship structures;

— Part 220, Application protocol: Process planning, manufacture, and assembly of layered electronic products;

— Part 221, Application protocol: Functional data and their schematic representation for process plant;

— Part 222, Application protocol: Exchange of product data for composite structures;

— Part 223, Application protocol: Exchange of design and manufacturing product information for casting parts;

— Part 224, Application protocol: Mechanical product definition for process plans using machining features;

— Part 225, Application protocol: Building elements using explicit shape representation;

— Part 226, Application protocol: Ship mechanical systems;

— Part 227, Application protocol: Plant spatial configuration;

— Part 228, Application protocol: Building services: Heating, ventilation, and air conditioning;

— Part 229, Application protocol: Exchange of design and manufacturing product information for forged parts;

— Part 230, Application protocol: Building structural frame: Steelwork;

— Part 231, Application protocol: Process engineering data: Process design and process specification of major equipment;

— Part 232, Application protocol: Technical data packaging core information and exchange;

— Part 301, Abstract test suite: Explicit draughting;

— Part 302, Abstract test suite: Associative draughting;

— Part 303, Abstract test suite: Configuration controlled design;

— Part 304, Abstract test suite: Mechanical design using boundary representation;

— Part 305, Abstract test suite: Mechanical design using surface representation;

— Part 307, Abstract test suite: Sheet metal die planning and design;

— Part 308, Abstract test suite: Life cycle management - Change process;

— Part 309, Abstract test suite: Composite and metallic structural analysis and related design;

— Part 310, Abstract test suite: Electronic assembly, interconnect, and packaging design;

— Part 312, Abstract test suite: Electrotechnical design and installation;

— Part 313, Abstract test suite: Numerical control process plans for machined parts;

— Part 314, Abstract test suite: Core data for automotive mechanical design;

— Part 315, Abstract test suite: Ship arrangement;

— Part 316, Abstract test suite: Ship moulded forms;

— Part 317, Abstract test suite: Ship piping;

— Part 318, Abstract test suite: Ship structures;

— Part 320, Abstract test suite: Process planning, manufacture, and assembly of layered electronic products;

— Part 321, Abstract test suite: Functional data and their schematic representation for process plant;

— Part 322, Abstract test suite: Exchange of product data for composite structures;

— Part 323, Abstract test suite: Exchange of design and manufacturing product information for casting parts;

— Part 324, Abstract test suite: Mechanical product definition for process plans using machining features;

— Part 325, Abstract test suite: Building elements using explicit shape representation;

— Part 326, Abstract test suite: Ship mechanical systems;

— Part 327, Abstract test suite: Plant spatial configuration;

— Part 328, Abstract test suite: Building services: Heating, ventilation, and air conditioning;

— Part 329, Abstract test suite: Exchange of design and manufacturing product information for forged parts;

— Part 330, Abstract test suite: Building structural frame: Steelwork;

— Part 331, Abstract test suite: Process engineering data: Process design and process specification of major equipment;

— Part 332, Abstract test suite: Technical data packaging core information and exchange;

— Part 501, Application interpreted construct: Edge-based wireframe;

— Part 502, Application interpreted construct: Shell-based wireframe;

— Part 503, Application interpreted construct: Geometrically bounded 2D wireframe;

— Part 504, Application interpreted construct: Draughting annotation;

— Part 505, Application interpreted construct: Drawing structure and administration;

— Part 506, Application interpreted construct: Draughting elements;

— Part 507, Application interpreted construct: Geometrically bounded surface;

— Part 508, Application interpreted construct: Non-manifold surface;

— Part 509, Application interpreted construct: Manifold surface;

— Part 510, Application interpreted construct: Geometrically bounded wireframe;

— Part 511, Application interpreted construct: Topologically bounded surface;

— Part 512, Application interpreted construct: Faceted boundary representation;

— Part 513, Application interpreted construct: Elementary boundary representation;

— Part 514, Application interpreted construct: Advanced boundary representation;

— Part 515, Application interpreted construct: Constructive solid geometry;

— Part 517, Application interpreted construct: Mechanical design geometric presentation;

— Part 518, Application interpreted construct: Mechanical design shaded representation.

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

— Parts 11 to 12 specify the description methods,

— Parts 21 to 26 specify the implementation methods,

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

— Parts 41 to 49 specify the integrated generic resources,

— Parts 101 to 106 specify the integrated application resources,

— Parts 201 to 232 specify the application protocols,

— Parts 301 to 332 specify the abstract test suites, and

— Parts 501 to 518 specify the application interpreted constructs.

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

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

# Introduction

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

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

This part of ISO 10303 specifies an application protocol (AP) for the exchange of the life cycle product change process information. Industries have a need to communicate to their suppliers, customers, clients, and contractors any product problems or anomalies, the corrections for these problems and any resulting corrective actions or changes.

Products supported by the product life cycle change process which an enterprise wishes to maintain a change history such as: discrete parts or components, assemblies, systems, facilities, maintenance and modification kits, support equipment, training materials and personnel requirements. The product life cycle process change definition also addresses acquisition, quality assessment, and the tasks required to support product definition.

Figure 1 shows the high-level data planning model for this application protocol. The lines depict the associations that exist between the different types of information. At this level, an item requiring change can be described as that configuration managed product which is defined from the point of view of some context. Life-cycle contexts include stages such as preliminary design, detailed design, and other points of view such as the product usage. The need for change is discovered during the performance of some task against the item. The task can include inspection procedures and test activities as well as normal product operation activities. A support resource is also a configuration managed product that is used to either support the identification of an item requiring change or actually supporting the changes to another product.

This application protocol defines the context, scope, and information requirements for the communication of information necessary to identify, report, and correct post-delivery product anomalies and specifies the integrated resources necessary to satisfy these requirements.

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

Clause 1 defines the scope of the application protocol and summarizes the functionality and data covered by the AP. 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

File Contains Data for PostScript Printers Only

Contains Data for Postscript Only

**Figure - Life cycle management change process data planning model**

annex F. The information requirements of the application are specified in clause 4 using terminology appropriate to the application. A graphical representation of the information requirements, referred to as the application reference model, is given in annex G.

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

# Industrial automation systems and integration — Product data representation and exchange — Part 208: Application protocol: Life cycle management - change process

## 1 Scope

This part of ISO 10303 specifies the use of the integrated resources necessary for the scope and information requirements for the exchange of life cycle product change process information. Change management for a product includes the identification of the reason for change, its cause, the approval and performance of the resulting changes to the product, and the authorization of corrective actions to prevent reoccurrence of the anomaly. The identified information provides change management support for activities across the life cycle of a product such as technical publication generation, retrofit planning, change proposal coordination, acquisition management, quality assement and material requirements planning. This is information is not unique to a particular kind of product or discipline, but is applicable across all products and disciplines.

NOTE - The application activity model in annex F provides a graphical representation of the processes and information flows that 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 identification of the product requiring change;

— the classification of the change as either a deviation from the expected characteristics from the authorized design or needing enhancement;

— the identification of an anomaly due to a flaw or other cause that results in corrective, perfecting, adaptive, or preventative needs. An identified anomaly may require a change to more than one version of a product that requires a change;

— the specification of the tasks requiring implementation of a change and inspection of that changed product to verify that the change requirements have been properly implemented;

— the required corrective actions to prevent reoccurrence of the change.

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

— the usage of the change information in project management planning, project management, and administration functions;

— product design data and product manufacturing data;

— design, manufacturing, and other life-cycle phases prior to delivery of the product.

## 2 Normative references

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

ISO/IEC 8824-1:1995, *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 10303-41:TC184/SC4/WG12 N33[1)], *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:TC184/SC4/WG12 N43[1)], *Industrial automation systems and integration — Product data representation and exchange — Part 43: Integrated generic resources: Representation structures*.

ISO 10303-44:TC184/SC4/WG12 N30[1)], *Industrial automation systems and integration — Product data representation and exchange — Part 44: Integrated generic resources: Product structure configuration*.

---

[1)] This document may be obtained at http://www.nist.gov/sc4/wg_qc/wg12/.

ISO 10303-45:1996, *Industrial automation systems and integration — Product data representation and exchange — Part 45: Integrated generic resource: Materials*.

ISO 10303-49:[1)] *Industrial automation systems and integration — Product data representation and exchange — Part 49: Integrated generic resource: Process plan structure*.

# 3 Definitions and abbreviations

## 3.1 Terms defined in ISO 10303-1

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

— application;

— application activity model (AAM);

— application interpreted model (AIM);

— application object;

— application protocol (AP);

— application reference model (ARM);

— assembly;

— component;

— conformance testing;

— implementation method;

— integrated resource;

— PICS proforma;

— product;

— product data;

— unit of functionality (UoF).

---

[1)] To be published.

## 3.2 Terms defined in ISO 10303-31

This part of ISO 10303 makes use of the following terms defined in ISO 10303-31:

— abstract test suite (ATS);

— conformance class;

— implementation under test (IUT);

— protocol information and conformance statement (PICS).

## 3.3 Terms defined in ISO 10303-44

This part of ISO 10303 makes use of the following terms defined in ISO 10303-44:

— effectivity;

— form, fit, function;

— lot.

## 3.4 Other definitions

For the purposes of this part of ISO 10303, the following definitions apply:

**3.4.1 anomaly:** a description of either a product problem or enhancement that may result in a change requirement. The product problems are deviations from the expected product characteristics or failure to perform. A product enhancement identifies the need for new and or improved product characteristics. Product characteristics are the form, fit, and function properties of a product or specific descriptive traits.

**3.4.2 authorization:** the decision making mechanism through which the appropriate level of permission is granted to proceed with the execution of planned actions or resource allocations. This includes a commitment or acknowledgment to perform a particular task or series of tasks.

**3.4.3 change management:** a procedure used by the functional organizations within an enterprise. The purpose is to determine which functions are impacted by a change activity and coordinate the tasks that will be involved throughout the entire change procedure. Change management includes the conceptual design, final design process, testing procedures and final delivery.

**3.4.4 change requirements:** the reason for actions that include changing, altering or modifying, replacing an item version with a substitute or creating a new item version, condition, or phase to another of that which is required.

**3.4.5 corrective action:** an action taken to prevent a product anomaly from reoccurring. Corrective action may include any or all of the following steps: localization, isolation, disassembly, re-assembly, alignment, and checkout.

**3.4.6 fielded history:** the performance information about an operational product. An operational product is one that has completed the manufacturing cycle and is available for test and use to assess performance of a product in the field. Fielded history may ascertain user satisfaction, obtain lessons learned and provide a mechanism for reporting findings and appropriate recommendations.

**3.4.7 item:** a single article or unit within a collection, that defines a product but is identified separately from the product.

**3.4.8 item version:** the definition of an item at a given point in time. It may be one of several time-sequenced definitions of a given item.

**3.4.9 product functionality:** the behavior of a product that meets or satisfies a particular performance requirement.

**3.4.10 related change:** a required change to a product due to a problem, enhancement , or corrective action that is associated with an anomaly.

**3.4.11 support resource:** a product required to design, build, operate, and maintain another product. A resource may be a facility, tool, person, or documentation.

**3.4.12 task:** a unit of specific work behavior with a clear beginning and ending, and has a meaningful function.

## 3.5 Abbreviations

For the purposes 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 |
| ATS | abstract test suite |
| id | identifier |
| ICOM | input, control, output, or mechanism |

PICS protocol information and conformance statement

UoF unit of functionality

# 4 Information requirements

This clause specifies the information required for the exchange of information between application systems in the life cycle product change process.

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

NOTES

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

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

3 - The mapping table specified in 5.1 shows how the integrated resources are used to meet the information requirements of this application protocol. The use of the integrated resources introduces additional requirements that are common to application protocols.

## 4.1 Units of functionality

This subclause specifies the UoFs for the life cycle product change process. This part of ISO 10303 specifies the following units of functionality:

— change_definition;

— item_definition;

— item_properties;

— supporting_resources;

— task.

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.

### 4.1.1 change_definition UoF

The change_definition UoF defines the data required to capture the change history of an Item_version. The change_definition UoF identifies the Item_version that is changing, why it is changing, how it is changing, and provides the change identification for the Item_version to be tracked.

The following application objects are used by the change_definition UoF:

— Discrepant_item;

— Enhancement_item;

— Item_anomaly;

— Item_anomaly_disposition;

— Item_change;

— Item_flaw;

— Item_flaw_classification;

— Item_issue_or_concern;

— Item_requiring_change;

— Related_change.

### 4.1.2 item_definition UoF

The item_definition UoF contains the data that identifies an Item_version and classifies an Item_version.

The following application objects are used by the item_definition UoF:

— Assembly_or_component_usage;

— Authorization;

— Conversion_factor;

— Data_template;

— End_item_effectivity;

— Enterprise;

— Functional_definition;

— Functional_definition_usage;

— Item_documentation_relationship;

— Item_program;

— Item_responsibility;

— Item_version;

— Item_version_relationship;

— Make_from_usage;

— Organization;

— Part;

— Part_system_relationship;

— Physical_unit;

— Product_property;

— Program;

— Source_maintenance;

— Supplier;

— System.

### 4.1.3 item_properties UoF

The item_properties UoF defines the data that is used to describe the physical properties of an Item_version.

The following application objects are used by the item_properties UoF:

— Availability;

— Capacity_level;

— Facility_capacity_level;

— Item_age;

— Item_capacity_level;

— Item_product_property;

— Item_repairability_level;

— Task_requiring_capacity_level.

### 4.1.4 supporting_resources UoF

The supporting_resources UoF identifies the resource objects that are required to support the performance of a task.

The following application objects are used by the supporting_resources UoF:

— Approval_level;

— Documentation;

— Facility;

— Package;

— Personnel;

— Personnel_skill_level;

— Skill;

— Skill_level;

— Support_equipment;

— Support_resource;

— Support_resource_approval_authority.

### 4.1.5 task_definition UoF

The task_definition UoF defines the data required to track the activities performed in relation to an Item_version.

The following application objects are used by the task_definition UoF:

— Field_maintenance;

— Item_coordinate_location;

— Item_task;

— Item_task_authorization;

— Item_task_frequency;

— Item_task_time;

— Item_work_unit_code;

— Recommended_support_resource;

— Reference_activity;

— Resource_activity_location;

— Task;

— Task_condition;

— Task_execution;

— Task_execution_context_definition;

— Task_execution_status;

— Task_execution_support_resource;

— Task_skill_level.

## 4.2 Application objects

This subclause specifies the application objects for the life cycle management change process 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 Approval_level

An Approval_level is the degree of approval given in the acknowledgment of the performed task.

The data associated with Approval_level are the following:

— approval_description;

— approval_identifier.

#### 4.2.1.1 approval_description

The approval_description specifies the comments or elaborations associated with an approval in an acknowledgment of an Item_task (see 4.2.33).

#### 4.2.1.2 approval_identifier

The approval_identifier specifies the unique identification label given to the Approval_level for a given Item_task (see 4.2.33).

### 4.2.2 Assembly_or_component_usage

The Assembly_or_component_usage is a type of Functional_definition_usage (see 4.2.17) that is the context Item_version (see 4.2.37) with respect to a higher level (parent or ancestor) node within an assembly hierarchy. An instance of Assembly_or_component_usage in the make-up of an Item_version is in the context of having, or being subordinate to, or to uniquely identify instances of an Assembly_or_component_usage.

The data associated with Assembly_or_component_usage are the following:

— assembly_usage_type;

— sequence_number.

#### 4.2.2.1 assembly_usage_type

The assembly_usage_type specifies the assembly as being either a higher assembly usage or the next assembly usage.

#### 4.2.2.2 sequence_number

The sequence_number specifies a sequentially assigned value for Item_version (see 4.2.37) contained in the assembly breakdown.

### 4.2.3 Authorization

The Authorization is a commitment to perform specific actions.

The data associated with Authorization are the following:

— authorization_function_type;

— authorization_identifier;

— authorization_perspective;

— authorization_source_type;

— authorizing_enterprise.

#### 4.2.3.1 authorization_function_type

The authorization_function_type specifies a type of Authorization that designates a function or activity.

#### 4.2.3.2 authorization_identifier

The authorization_identifier specifies the unique identification label associated with a commitment to perform specific actions.

#### 4.2.3.3 authorization_perspective

The authorization_perspective specifies an individuals ability to perform an evaluation of relative significance, a point of view.

#### 4.2.3.4 authorization_source_type

The authorization_source_type specifies the role of the granting enterprise as either an internal or external authorizing entity.

#### 4.2.3.5 authorizing_enterprise

The enterprise specifies the organization that is granting the authorization. See 4.3.1 for the application assertion.

### 4.2.4 Availability

An Availability is a type of Item_product_property (see 4.2.28) that defines the percentage of time that a component within a given context will be able to perform the assigned function. The component may have multiple availability values that are classified as inherent, achieved or operational.

The data associated with Availability are the following:

— availability_value;

— component_usage;

— indicator.

#### 4.2.4.1 availability_value

The availability_value specifies the value as a percentage that an Item_version (see 4.2.37) is available to perform its Item_task (see 4.2.33).

#### 4.2.4.2 component_usage

The component_usage specifies the context assembly specific component Item_version (see 4.2.37) about which the availability_value applies. See 4.3.2 for the application assertion.

#### 4.2.4.3 indicator

The indicator specifies if the Availability is inherent, achieved, or operational.

NOTE - Achieved availability is calculated by dividing the mean time between maintenance plus the mean active maintenance downtime into the mean time between maintenance. Inherent availability is calculated by adding the mean time between failure plus the meantime to repair divided into the mean time between failure. Operational availability is calculated by the summation of operating time per calendar year plus the stand-by time plus total preventive maintenance time plus corrective action time plus administrative and logistics delay time.

### 4.2.5 Capacity_level

A Capacity_level is the definition of the levels of organizational capability breakdown that are to be used to classify the maintenance tasks applied to an Item_version (see 4.2.37).

The data associated with Capacity_level are the following:

— capacity_level_function.

The capacity_level_function specifies the organizational Capacity_level where a particular task is authorized to be accomplished against an Item_version (see 4.2.37).

### 4.2.6 Conversion_factor

A Conversion_factor is the ratio of the annual operating time of a component item to the annual operation time of the highest end product.

The data associated with Conversion_factor are the following:

— component_usage;

— conversion_percent.

#### 4.2.6.1 component_usage

The component_usage specifies the component item and the highest level assembly about which the Conversion_-factor is determined. See 4.3.3 for the application assertion.

#### 4.2.6.2 conversion_percent

The conversion_percent specifies the value of the ratio of the annual operating time of a component compared to the annual operating time of the assembly in which it is used. This value is calculated by dividing the identifying end context item operating time into the component item operating time.

### 4.2.7 Data_template

A Data_template is a type of Item_version (see 4.2.37) that defines the makeup and format of a user view of data pertaining to an Item_version. A Data_template may be used to create a technical report, a document, or a computer user interface screen.

The data associated with Data_template are the following:

— template_type.

The template_type specifies the classification of a Data_template.

### 4.2.8 Discrepant_item

A Discrepant_item is a type of Item_requiring_change (see 4.2.31) that fails to satisfy one or more design nominal criteria requirements.

The data associated with Discrepant_item are the following:

— failure_rate.

The failure_rate specifies the value of the total number of failures within an item divided by the total functional life during a specific measurement interval.

### 4.2.9 Documentation

A Documentation is a type of Support_resource (see 4.2.59) that is a resource representation of one or more Item_version (see 4.2.37) objects and their characteristics in a format that is used to facilitate user communications.

The data associated with Documentation are the following:

— format;

— name;

— type.

#### 4.2.9.1 format

The format specifies a Data_template (see 4.2.7) which defines the makeup of a user Documentation. See 4.3.4 for the application assertion.

#### 4.2.9.2 name

The name specifies the nomenclature for Documentation.

NOTE - The title usually indicates the functionality of that document and the functionality it performs or serves.

#### 4.2.9.3 type

The type specifies a class of user communication formats having similar characteristics.

NOTE - The documentation type can be one of the following: journal, article, technical manual, engineering drawing, associated list, illustration.

### 4.2.10 End_item_effectivity

An End_item_effectivity is the identification of the Physical_unit (see 4.2.47) end items that the released Item_-version (see 4.2.37) is used on. The Physical_unit may be an individual serial number or a range of serial numbers.

The data associated with End_item_effectivity are the following:

— effective_usage;

— end_item.

#### 4.2.10.1 effective_usage

The effective_usage specifies the component and the context for which the effectivity is applicable. See 4.3.5 for the application assertion.

#### 4.2.10.2 end_item

The end_item specifies the instance of an item version that is the Physical_unit (see 4.2.47) within an End_item_-effectivity. See 4.3.6 for the application assertion.

### 4.2.11 Enhancement_item

An Enhancement_item is a type of Item_requiring_change (see 4.2.31) that is the identification of a need for new or improved item functionality.

### 4.2.12 Enterprise

An Enterprise is a type of Organization (see 4.2.41) that identifies a supplier, manufacturer, or consumer of an Item_version (see 4.2.37). Each Enterprise may be a Supplier (see 4.2.57).

### 4.2.13 Facility

A Facility is a type of Support_resource (see 4.2.59) that may be a building, area, fixture, or integrated equipment recommended or required to support design, production, operation, training, maintenance, or retirement of an Item_version (see 4.2.37).

The data associated with Facility are the following:

— geographic_location;

— type.

#### 4.2.13.1 geographic_location

The geographic_location specifies the address identifying a facility locale.

#### 4.2.13.2 type

The type specifies the classification of a Facility being used.

### 4.2.14 Facility_capacity_level

A Facility_capacity_level is a type of Item_capacity_level (see 4.2.21) that is the organizational capability breakdown of maintenance support that a support facility is capable of providing.

The data associated with Facility_capacity_level are the following:

— subject_facility.

The subject_facility specifies the support resource products (building, area, or fixture) where maintenance of a specific Facility_capacity_level can be performed, recommended or required to support the design, production, operation, training, or maintenance of an Item_version (see 4.2.37) throughout its life cycle. See 4.3.7 for the application assertion.

### 4.2.15 Field_maintenance

A Field_maintenance is a type of Task_execution (see 4.2.64) that identifies a task being performed against an Item_version (see 4.2.37) that has been deployed within the user community. It identifies the test, inspection, operation, and repair activities performed against an Item_work_unit_code (see 4.2.39).

The data associated with Field_maintenance are the following:

— maintenance_type;

— work_unit_code.

#### 4.2.15.1 maintenance_type

The maintenance_type specifies the class of Field_maintenance as being either operational, inspection, or repair.

NOTE - A project can have three levels of maintenance, as follows: organizational, intermediate, or depot.

#### 4.2.15.2 work_unit_code

The work_unit_code specifies the system, subsystem and component on which field maintenance is required or was performed. See 4.3.8 for the application assertion.

NOTE - The definitions are created to describe the different organizational views or perspectives of the Item_-version, such as the as-designed, as-planned, as-built, as-maintained, view or perspective.

## 4.2.16 Functional_definition

A Functional_definition is a type of Item_version (see 4.2.37) that is a unique description of the underlying relationships and information defining the organizational view of an Item_version from the point of view of functional or assembly use.

## 4.2.17 Functional_definition_usage

A Functional_definition_usage is uniquely identifiable association between a component and an assembly between an input and an output Item_version (see 4.2.37). The purpose of identifying an association is to define information unique to that association. Each Functional_definition_usage may be an Assembly_or_component_-usage (see 4.2.2) or a Make_from_usage (see 4.2.40).

EXAMPLE 1 - Information unique to the association between a component and an assembly would include location information, effectivity or functional requirements.

The data associated with Functional_definition_usage are the following:

— component;

— component_function;

— component_quantity;

— context;

— occurrence_identifier;

— usage_type.

### 4.2.17.1 component

The component specifies an Item_version (see 4.2.37) that is configured within another Item_version (see 4.2.37). See 4.3.9 for the application assertion.

#### 4.2.17.2 component_function

The component_function specifies the role that a component plays in the association with the context Item_-version (see 4.2.37).

#### 4.2.17.3 component_quantity

The component_quantity specifies the number of components utilized in a given context Item_version (see 4.2.37).

#### 4.2.17.4 context

The context specifies an Item_version (see 4.2.37) which is composed of another Item_version. See 4.3.9 for the application assertion.

#### 4.2.17.5 occurrence_identifier

The occurrence_identifier specifies the quantity of the component relative to the context.

NOTE - This becomes apparent when multiple components are used within a given context.

#### 4.2.17.6 usage_type

The usage_type specifies if the usage association is a Make_from_usage (see 4.2.40) or an assembly usage.

### 4.2.18 Item_age

An Item_age is a type of Item_product_property (see 4.2.28) that is the amount of time either in the form of hours, minutes, cycles, or date that has elapsed during the operation of an Item_version (see 4.2.37).

The data associated with Item_age are the following:

— age_date_time;

— age_defined_state;

— age_value;

— related_item.

#### 4.2.18.1 age_date_time

The age_date_time specifies the point in time when the age_value was taken.

#### 4.2.18.2 age_defined_state

The age_defined_state specifies a description of the conditions under which the item existed for the duration equal to its age value.

EXAMPLE 2 - "In operation" or "in storage" might be values for the age_defined_state.

#### 4.2.18.3 age_value

The age_value specifies the measured operation duration.

#### 4.2.18.4 related_item

The related_item specifies the Physical_unit (see 4.2.47) that an age value is being specified for. See 4.3.10 for the application assertion.

### 4.2.19 Item_anomaly

An Item_anomaly is the identification of a nonconformance or a deviation from design nominal conditions for a product. The identification of a nonconformance or a deviation is reported by a Support_resource (see 4.2.59). Each Item_anomaly may be an Item_flaw (see 4.2.25) or an Item_issue_or_concern (see 4.2.27).

The data associated with Item_anomaly are the following:

— anomaly_cause;

— anomaly_type;

— detection_method;

— item;

— item_anomaly_description;

— item_anomaly_identifier.

#### 4.2.19.1 anomaly_cause

The anomaly_cause specifies a narrative identifying the reason why the nonconformance occurred.

#### 4.2.19.2 anomaly_type

The anomaly_type specifies a descriptive label for the Item_anomaly.

#### 4.2.19.3 detection_method

The detection_method specifies the procedure that a system, sub-system or assembly was evaluated and determined to be nonconforming.

#### 4.2.19.4 item

The item specifies the Item_version (see 4.2.37) that possesses or was found to have the anomaly. See 4.3.11 for the application assertion.

#### 4.2.19.5 item_anomaly_description

The item_anomaly_description specifies a narrative account describing the nonconformance.

#### 4.2.19.6 item_anomaly_identifier

The item_anomaly_identifier specifies the unique identification of an Item_issue_or_concern or an Item_flaw that is associated with an Item_version (see 4.2.37).

### 4.2.20 Item_anomaly_disposition

An Item_anomaly_disposition is the actual resolution applied to an Item_anomaly (see 4.2.19).

The data associated with Item_anomaly_disposition are the following:

— anomalized_item;

— approving_resource;

— disposition_task.

#### 4.2.20.1 anomalized_item

The anomalized_item specifies the identification of an item anomaly. See 4.3.12 for the application assertion.

#### 4.2.20.2 approving_resource

The approving_resource specifies the Support_resource (see 4.2.59) that has the responsibility for determining the acceptability of, authorization for, or action applied to Item_version (see 4.2.37). See 4.3.13 for the application assertion.

#### 4.2.20.3 disposition_task

The disposition_task specifies the performance of an Item_task (see 4.2.33) for answering the disposition of an Item_version (see 4.2.37) to the satisfaction of the controlling interest. See 4.3.14 for the application assertion.

### 4.2.21 Item_capacity_level

An Item_capacity_level is a type of Item_product_property (see 4.2.28) that is the definition of the levels of organizational maintenance capability breakdown for an Item_version (see 4.2.37). Each Item_capacity_level is either a Facility_capacity_level (see 4.2.14), an Item_repairability_level (see 4.2.30), or a Task_requiring_-capacity_level (see 4.2.68).

The data associated with Item_capacity_level are the following:

— level.

The level specifies the organizational maintenance capability associated with an Item_version (see 4.2.37). See 4.3.15 for the application assertion.

### 4.2.22 Item_change

An Item_change is the creation of a new Item_version (see 4.2.37) that results from an anomaly or concern about a baseline item.

NOTE - This entity identifies the new Item_version as well as the baseline Item_version that the new version was based upon, due to an anomaly or concern as well as the authorization that accounts for the Item_change.

The data associated with Item_change are the following:

— baseline_item;

— baseline_item_disposition;

— reason;

— resulting_item.

#### 4.2.22.1 baseline_item

The baseline_item specifies the Item_version (see 4.2.37) that undergoes a change process and results in a new Item_version. See 4.3.17 for the application assertion.

#### 4.2.22.2 baseline_item_disposition

The baseline_item_disposition specifies the resolution that is being applied to the baseline_item to satisfy an anomaly. See 4.3.16 for the application assertion.

#### 4.2.22.3 reason

The reason specifies the rationale of why an Item_change took place.

#### 4.2.22.4 resulting_item

The resulting_item specifies the Item_version (see 4.2.37) that results from a change in the process. See 4.3.18 for the application assertion.

### 4.2.23 Item_coordinate_location

An Item_coordinate_location is a specific position of a component Item_version (see 4.2.37) within an assembly Item_version.

The data associated with Item_coordinate_location are the following:

— component_location;

— component_usage;

— coordinate_system_identifier.

#### 4.2.23.1 component_location

The component_location specifies the physical location of a component in reference to the usage.

#### 4.2.23.2 component_usage

The component_usage specifies the Functional_definition_usage (see 4.2.17) for which the Item_coordinate_-location is based. See 4.3.19 for the application assertion.

#### 4.2.23.3 coordinate_system_identifier

The coordinate_system_identifier specifies the identification of the coordinate system used in defining the location of a component.

### 4.2.24 Item_documentation_relationship

An Item_documentation_relationship is a type of Item_version_relationship (see 4.2.38) that is the association of an Item_version (see 4.2.37) with documentation used to describe the Item_version.

### 4.2.25 Item_flaw

An Item_flaw is a type of Item_anomaly (see 4.2.19) that is the description of a nonconformance or flaw in, on, or about an Item_version (see 4.2.37).

### 4.2.26 Item_flaw_classification

An Item_flaw_classification is the specification of one or more flaw categories that an Item_flaw (see 4.2.25) belongs to.

The data associated with Item_flaw_classification are the following:

— classified_item;

— flaw_class_identifier.

#### 4.2.26.1 classified_item

The classified_item specifies the Item_flaw being classified. See 4.3.20 for the application assertion.

#### 4.2.26.2 flaw_class_identifier

The flaw_class_identifies specifies the classification or type of an Item_flaw as inherent, natural failure, or caused by another Item_version (see 4.2.37) or Support_resource (see 4.2.59).

### 4.2.27 Item_issue_or_concern

An Item_issue_or_concern is a type of Item_anomaly (see 4.2.19) that is the identification of special issues or concerns associated with an Item_version (see 4.2.37) that are not flaws but require an Item_version change process to be initiated.

### 4.2.28 Item_product_property

An Item_product_property is a type of Item_version (see 4.2.37) and Item_version_relationship (see 4.2.38) that is a property or characteristic belonging to, associated with, or being used to define an Item_version that is undergoing product change management. Each Item_product_property is either an Availability (see 4.2.4), an Item_age (see 4.2.18), or an Item_capacity_level (see 4.2.21).

### 4.2.29 Item_program

An Item_program is a type of Item_version_relationship (see 4.2.38) that is the identification of the organizational authority to which an Item_version (see 4.2.37) being change managed is associated. This association is from the perspective of funding authorization, schedule development and general business management.

The data associated with Item_program are the following:

— program_authorization.

The program_authorization specifies the authority under which change management of an Item_version takes place. See 4.3.21 for the application assertion.

### 4.2.30 Item_repairability_level

An Item_repairability_level is a type of Item_capacity_level (see 4.2.21) that represents the amount of an Item_-version (see 4.2.37) undergoing change management that can be repaired at a particular Capacity_level (see 4.2.5).

The data associated with Item_repairability_level are the following:

— component_usage;

— condemnation_percentage;

— not_repairable_percentage;

— repairable_percentage.

#### 4.2.30.1 component_usage

The component_usage specifies the usage relationship for the Item_version (see 4.2.37) undergoing change management. See 4.3.22 for the application assertion.

### 4.2.30.2 condemnation_percentage

The condemnation_percentage specifies a computational rate that represents the predicted quantity of change managed Item_version (see 4.2.37) objects to be condemned for a specified time period of usage.

### 4.2.30.3 not_repairable_percentage

The not_repairable_percentage specifies the predicted ratio of the number of non-repairable generations of the change managed Item_version (see 4.2.37) objects that the Capacity_level (see 4.2.5) will be unable to repair to the total quantity of changed managed Item_version objects for a particular Capacity_level.

### 4.2.30.4 repairable_percentage

The repairable_percentage specifies a computational rate that defines the number of change manage Item_version (see 4.2.37) objects predicted to be repaired at a specified level of repair.

## 4.2.31 Item_requiring_change

An Item_requiring_change is a type of Item_version (see 4.2.37) that is changed because of the identification of a flaw or need for capability enhancement. Each Item_requiring_change may be a Discrepant_item (see 4.2.8), an Enhancement_item (see 4.2.11), or a Related_change (see 4.2.52).

The data associated with Item_requiring_change are the following:

— anomalies;

— item_change_requirement_type;

— item_need;

— requiring_change_task.

### 4.2.31.1 anomalies

The anomalies specifies the anomalies in the item which prompted the need for a change. See 4.3.23 for the application assertion.

### 4.2.31.2 item_change_requirement_type

The item_change_requirement_type specifies whether the reason for an Item_version (see 4.2.37) change is either a discrepancy or enhancement.

#### 4.2.31.3 item_need

The item_need specifies the set of discrepancies or set of enhancements causing the change requirement.

#### 4.2.31.4 requiring_change_task

The requiring_change_task specifies the task that caused, initiated, or identified the need for an Item_version (see 4.2.37) to be changed. See 4.3.24 for the application assertion.

### 4.2.32 Item_responsibility

An Item_responsibility is the association of an Item_version (see 4.2.37) with the organization that controls the item.

The data associated with Item_responsibility are the following:

— controlling_unit;

— responsible_item.

#### 4.2.32.1 controlling_unit

The controlling_unit specifies the organization that has control and responsibility over the Item_version (see 4.2.37). See 4.3.26 for the application assertion.

#### 4.2.32.2 responsible_item

The responsible_item specifies the Item_version (see 4.2.37) that is controlled by a particular organization. See 4.3.25 for the application assertion.

### 4.2.33 Item_task

An Item_task is a type of Item_version_relationship (see 4.2.38) that is the association of an Item_version (see 4.2.37) with the task to be performed on or by the Item_version. Each Item_task may be an Item_task_-authorization (see 4.2.34).

The data associated with Item_task are the following:

— item_operability.

The item_operability indicates the operational status or mission readiness of an Item_version during an Item_-task.

### 4.2.34 Item_task_authorization

An Item_task_authorization is a type of Item_task (see 4.2.33) that is the association of an Item_task with an authorization to perform that Item_task.

The data associated with Item_task_authorization are the following:

— authority.

The authority specifies the acknowledgment of a commitment to perform specified actions against an Item_-version (see 4.2.37). See 4.3.27 for the application assertion.

### 4.2.35 Item_task_frequency

An Item_task_frequency is an indication of how often an Item_task (see 4.2.33) is performed over a specified time period.

The data associated with Item_task_frequency are the following:

— subject_item_task;

— task_frequency;

— task_frequency_indicator.

#### 4.2.35.1 subject_item_task

The subject_item_task specifies the particular association of an Item_version (see 4.2.37) with a Task (see 4.2.62) for which the frequency is being captured. See 4.3.28 for the application assertion.

#### 4.2.35.2 task_frequency

The task_frequency specifies the numeric value for how often a given Item_task (see 4.2.33) will be or has been performed over a certain time period.

#### 4.2.35.3 task_frequency_indicator

The task_frequency_indicator specifies if the task_frequency is allocated, measured, or predicted.

### 4.2.36 Item_task_time

An Item_task_time is the average duration of time that a task to be performed on or by the Item_version (see 4.2.37) will take.

The data associated with Item_task_time are the following:

— subject_item_task;

— task_time_indicator;

— time_duration.

### 4.2.36.1 subject_item_task

The subject_item_task specifies the particular association of an Item_version (see 4.2.37) with a Task (see 4.2.62) for which the time duration is being captured. See 4.3.29 for the application assertion.

### 4.2.36.2 task_time_indicator

The indicator specifies if the time_duration is allocated, predicted, or measured.

### 4.2.36.3 time_duration

The time_duration specifies the total duration of time to execute the performance of an Item_task (see 4.2.33).

## 4.2.37 Item_version

An Item_version is anything about which one wishes to maintain change history. An Item_version is created as the result of an initial release of an item or from a change in one or more characteristics of an Item_version. Each Item_version is either a Data_template (see 4.2.7), a Functional_definition (see 4.2.16), an Item_product_-property (see 4.2.28), an Organization (see 4.2.41), a Part (see 4.2.43), a Product_property (see 4.2.48), a Support_resource (see 4.2.59), a System (see 4.2.61), or a Task (see 4.2.62). Each Item_version may be an Item_requiring_change (see 4.2.31).

The data associated with Item_version are the following:

— creation_date_time;

— description;

— family_group;

— item_type;

— item_version_identifier;

— nomenclature.

#### 4.2.37.1 creation_date_time

The creation_date_time specifies the point in time when an Item_version was created.

#### 4.2.37.2 description

The description specifies a narrative for documenting in a textual fashion the purpose and scope of an Item_-version.

#### 4.2.37.3 family_group

The family_group specifies the textual string or name for classifying an Item_version as belonging to a set of products having similar functionality and definitional characteristics.

#### 4.2.37.4 item_type

The item_type specifies the category or classification of an Item_version.

#### 4.2.37.5 item_version_identifier

The item_version_identifier specifies a unique instance of an item within the item_type. It may be a control number issued by a functional organization and will allow the Item_version to be tracked while in the domain of that function.

NOTE - Functional organizations can be Engineering, Manufacturing, Material, Logistics, etc.

#### 4.2.37.6 nomenclature

The nomenclature specifies a naming convention that identifies the Item_version.

### 4.2.38 Item_version_relationship

An Item_version_relationship is the pairing of one Item_version (see 4.2.37) with another Item_version. Each Item_version_relationship may be one of the following: an Item_documentation_relationship (see 4.2.24), an Item_product_property (see 4.2.28), an Item_program (see 4.2.29), an Item_task (see 4.2.33), or a Part_system_-relationship (see 4.2.44).

NOTE - When the associated grouping of items is to be configuration managed as a group, a new item version is created and the associated items are related to the new item through the functional definition usage entity.

The data associated with Item_version_relationship are the following:

— related_item;

— relating_item.

### 4.2.38.1 related_item

The related_item specifies an Item_version (see 4.2.37) that is associated with another Item_version. See 4.3.30 for the application assertion.

### 4.2.38.2 relating_item

The relating_item specifies the Item_version (see 4.2.37) that has a relationship to another Item_version. See 4.3.30 for the application assertion.

## 4.2.39 Item_work_unit_code

An Item_work_unit_code is the identification of the system and the component assembly Item_version (see 4.2.37) pairing on which maintenance is accomplished or is required.

The data associated with Item_work_unit_code are the following:

— system_or_part;

— work_unit_code_identifier.

### 4.2.39.1 system_or_part

The system_or_part specifies an Item_version (see 4.2.37) of type system or part on which maintenance is accomplished or is required. See 4.3.31 for the application assertion.

### 4.2.39.2 work_unit_code_identifier

The work_unit_code_identifier specifies a distinguishing identifier associated with the system or part on which maintenance is required as specified on a manufacturing or operational analysis document.

## 4.2.40 Make_from_usage

A Make_from_usage is a type of Functional_definition_usage (see 4.2.17) that establishes an Item_version (see 4.2.37). The input or context Item_version is one which is physically transformed into another Item_version, the output or components.

NOTE - Make_from_usage is distinguished from assembly usage by the physical transformation of the input item.

### 4.2.41 Organization

An Organization is a type of Item_version (see 4.2.37) that identifies a number of persons or groups having specific responsibilities and united for some purpose of work. Each Organization may be an Enterprise or a Program (see 4.2.49).

The data associated with Organization are the following:

— organization_type.

The organization_type specifies a descriptive label for the Organization.

### 4.2.42 Package

A Package is a type of Support_resource (see 4.2.59) that is a set of objects consisting of one or more parts and one or more documents used to produce, replace, modify, or repair an Item_version (see 4.2.37) or specific portions of an Item_version.

The data associated with Package are the following:

— package_type.

The package_type specifies the style of kit used as being either for maintenance, repair, modification, standard use, production, or procurement.

### 4.2.43 Part

A Part is a type of Item_version (see 4.2.37) that is a discrete product of the organization. A Part may be a Physical_unit (see 4.2.47).

The data associated with Part are the following:

— function_type;

— name;

— type.

#### 4.2.43.1 function_type

The function_type specifies the part as either a mechanical part, electrical part, or combination part.

#### 4.2.43.2 name

The name specifies the title given to a product that is descriptive of the functionality of the part.

#### 4.2.43.3 type

The type specifies the part is either a unitary part or aggregate part from the perspective of its component materials.

### 4.2.44 Part_system_relationship

A Part_system_relationship is a type of Item_version_relationship (see 4.2.38) when a system is associated with another Item_version (see 4.2.37) or type part.

### 4.2.45 Personnel

A Personnel is a type of Support_resource (see 4.2.59) that is a human.

NOTE - A personnel may be a person who is using or supporting the Item_version or related items.

### 4.2.46 Personnel_skill_level

A Personnel_skill_level is the association of an instance of personnel with an instance of Skill_level (see 4.2.55), detailing the capabilities of the person referenced.

The data associated with Personnel_skill_level are the following:

— person;

— subject_skill_level.

#### 4.2.46.1 person

The person specifies that the skilled Support_resource (see 4.2.59) is an individual with specific skill. See 4.3.32 for the application assertion.

#### 4.2.46.2 subject_skill_level

The subject_skill_level specifies the incremental degree of a refined proficiency required to execute a Task (see 4.2.62). See 4.3.33 for the application assertion.

## 4.2.47 Physical_unit

A Physical_unit is a type of Part (see 4.2.43) that uniquely identifies a physical manifestation of a part design. It is a tracked instance of an Item_version (see 4.2.37), a serialized unit, or lot.

The data associated with Physical_unit are the following:

— baseline_part;

— physical_unit_type.

### 4.2.47.1 baseline_part

The baseline_part specifies the original part that undergoes a physical manifestation to become one or more physical units. See 4.3.34 for the application assertion.

### 4.2.47.2 physical_unit_type

The physical_unit_type specifies the physical unit as a discrete physical unit or a lot physical unit.

## 4.2.48 Product_property

A Product_property is a type of Item_version (see 4.2.37) that identifies the inherent qualities of a physical or functional characteristic.

## 4.2.49 Program

A Program is a type of Organization (see 4.2.41) denoting a particular organized thrust or development effort.

## 4.2.50 Recommended_support_resource

A Recommended_support_resource is a Support_resource (see 4.2.59) that is specified as needed to assist or facilitate the performance of an Item_task (see 4.2.33). A Recommended_support_resource is an item that performs, aids, is used by, or is consumed in the execution of an Item_task.

The data associated with Recommended_support_resource are the following:

— recommended_quantity;

— requiring_item_task;

— resource;

— technical_evaluation_priority.

### 4.2.50.1 recommended_quantity

The recommended_quantity is the number of Support_resource (see 4.2.59) objects recommended to perform the requiring Item_task (see 4.2.33).

### 4.2.50.2 requiring_item_task

The requiring_item_task specifies the Item_task (see 4.2.33) that requires or needs a Support_resource (see 4.2.59). See 4.3.35 for the application assertion.

### 4.2.50.3 resource

The resource specifies the Support_resource (see 4.2.59) that is required to facilitate design, production, training, operation, or maintenance of an Item_version (see 4.2.37). See 4.3.36 for the application assertion.

NOTE - A Support_resource may be personnel, support equipment, facility, or kit, among others.

### 4.2.50.4 technical_evaluation_priority

The technical_evaluation_priority specifies the detailed evaluation category, preference, and criteria for a Support_resource (see 4.2.59) used in an Item_task (see 4.2.33).

## 4.2.51 Reference_activity

A Reference_activity identifies a Task (see 4.2.62) that is associated to another Task.

The data associated with Reference_activity are the following:

— referenced_task;

— referencing_task.

### 4.2.51.1 referenced_task

The referenced_task specifies a task that is being associated to another task. See 4.3.37 for the application assertion.

### 4.2.51.2 referencing_task

The referencing_task specifies the task associating another task. See 4.3.37 for the application assertion.

### 4.2.52 Related_change

A Related_change is a type of Item_requiring_change (see 4.2.31) that identifies an Item_requiring_change due to an anomaly with another Item_requiring_change.

The data associated with Related_change are the following:

— related_item.

The related_item specifies the identification of a product anomaly that has identified an additional Item_-requiring_change. See 4.3.38 for the application assertion.

### 4.2.53 Resource_activity_location

A Resource_activity_location is the location on a given Item_version (see 4.2.37) where the Recommended_-support_resource (see 4.2.50) is used to perform the Item_task (see 4.2.33).

The data associated with Resource_activity_location are the following:

— location;

— recommended_resource.

#### 4.2.53.1 location

The location specifies the coordinate location where the Recommended_support_resource (see 4.2.50) is used to perform the Item_task (see 4.2.33). See 4.3.39 for the application assertion.

#### 4.2.53.2 recommended_resource

The recommended_resource specifies the Support_resource (see 4.2.59) that is being located. See 4.3.40 for the application assertion.

### 4.2.54 Skill

A Skill is a type of Support_resource (see 4.2.59) that is the ability or proficiency that comes from training, practice, or experience.

The data associated with Skill are the following:

— skill_description;

— skill_identifier.

### 4.2.54.1 skill_description

The skill_description specifies the portrayal of the skill needed to perform a given task.

EXAMPLE 3 - A pilot needs to be trained in the operation of a particular aircraft, or a technician needs training in repairing an electronic device.

### 4.2.54.2 skill_identifier

The skill_identifier specifies the unique label of a proficiency, either by training or practice.

## 4.2.55 Skill_level

A Skill_level is the incremental degree of a refined Skill (see 4.2.54).

The data associated with Skill_level are the following:

— description;

— subject_skill.

### 4.2.55.1 description

The description specifies the textual portrayal of the skill level needed in order to perform a given task.

NOTE - This task could be to fly an airplane, repair an engine, modify a fixture, build an assembly, or plot a navigational course. These tasks require a certain level of expertise.

### 4.2.55.2 subject_skill

The subject_skill specifies the ability or proficiency that the incremental degree is associated with. See 4.3.41 for the application assertion.

## 4.2.56 Source_maintenance

A Source_maintenance indicates the source, maintenance, and recoverability of a Functional_definition_usage (see 4.2.17). Source_maintenance is assigned to maintainable products early in the acquisition cycle to convey maintenance and supply instructions to the various support levels and using functions.

The data associated with Source_maintenance are the following:

— component_usage;

— source_maintenance_repair_code.

### 4.2.56.1 component_usage

The component_usage specifies the Assembly_or_component_usage (see 4.2.2) for which the Source_-maintenance is based. See 4.3.42 for the application assertion.

### 4.2.56.2 source_maintenance_repair_code

The source_maintenance_repair_code specifies a description of the Source_maintenance for a product.

## 4.2.57 Supplier

A Supplier is a type of Enterprise (see 4.2.12) that is a commercial or government entity which provides products to another enterprise in the acquisition process.

— code;

— type.

### 4.2.57.1 code

The code specifies a system of letters, characters or other symbols that arbitrarily are used to uniquely represent an enterprise at a location. It is designed to control an activity or actual enterprise for an Item_version (see 4.2.37).

### 4.2.57.2 type

The type specifies the supplier as either a manufacturer or non-manufacturer type of enterprise.

## 4.2.58 Support_equipment

A Support_equipment is a type of Support_resource (see 4.2.59) that is a device required to facilitate design, production, training, operation, or maintenance of an Item_version (see 4.2.37).

NOTE - All equipment required to support the operation and maintenance of a system. This includes associated multiuse end items, ground handling and maintenance equipment, tools, metrology and calibration equipment, communications resources, test equipment and automatic test equipment, with diagnostic software for equipment maintenance. It includes the acquisition of support for the support and test equipment itself.

The data associated with Support_equipment are the following:

— install_factor.

The install_factor specifies the considerations required for the installation of an Item_version such as vibration and shock mounting requirements, special environment factors.

## 4.2.59 Support_resource

A Support_resource is a type of Item_version (see 4.2.37) that plays a supporting role or serves as a resource within an Item_task (see 4.2.33). Each Support_resource may be one of the following: a Documentation (see 4.2.9), a Facility (see 4.2.13), a Package (see 4.2.42), a Personnel (see 4.2.45), a Skill (see 4.2.54), or a Support_-equipment (see 4.2.58).

## 4.2.60 Support_resource_approval_authority

A Support_resource_approval_authority is a Support_resource (see 4.2.59) which has the responsibility for acknowledging the completion of an action.

The data associated with Support_resource_approval_authority are the following:

— approval;

— granting_resource.

### 4.2.60.1 approval

The approval specifies the Approval_level (see 4.2.1) that a Support_resource (see 4.2.59) possesses. See 4.3.43 for the application assertion.

### 4.2.60.2 granting_resource

The granting_resource specifies the Support_resource (see 4.2.59) that is acknowledging the completion of an action. See 4.3.44 for the application assertion.

NOTE - This may be a computer software program that approves system operability or ground support equipment that shows that a system is operating properly for flight or a person granting approval.

## 4.2.61 System

A System is a type of Item_version (see 4.2.37) that is a regularly interacting or an interdependent group of items forming a unified whole under the influence of related forces.

NOTE - An organized collection of interdependent and interactive personnel, machines, and methods combined to accomplish a set of specific functions as a larger unit having the capabilities of all the separate units.

The data associated with System are the following:

— system_type.

The system_type specifies a category designated to identify the application unique platform for the functional group configuration.

NOTE - A category may be armament, electronic, avionics, electrical, mechanical, propulsion, or structural.

## 4.2.62 Task

A Task is a type of Item_version (see 4.2.37) that defines the step by step instructions on how to perform a particular activity or action.

The data associated with Task are the following:

— acceptance_criteria;

— consideration;

— name;

— procedure;

— purpose.

### 4.2.62.1 acceptance_criteria

The acceptance_criteria specifies the textual description on how to determine the acceptance of a task performance.

### 4.2.62.2 consideration

The consideration specifies any critical issues associated with the Task.

NOTE - The potential for exposing personnel to hazardous conditions or whether failure to accomplish a task in accordance with system requirements would result in adverse system reliability, efficiency, effectiveness, safety or degraded Item_task performance.

### 4.2.62.3 name

The name specifies the title given to the Task.

#### 4.2.62.4 procedure

The procedure specifies the step by step narrative instruction set outlining the requirements for performing the Task.

#### 4.2.62.5 purpose

The purpose specifies the logic established to perform the Task and the expected results from performing the Task.

### 4.2.63 Task_condition

A Task_condition is the identification of the special considerations that must be taken into account during the performance of a task.

— condition_description;

— condition_identifier;

— requiring_item_task.

#### 4.2.63.1 condition_description

The condition_description specifies the considerations that must be met in the performance of task and against an Item_version (see 4.2.37).

#### 4.2.63.2 condition_identifier

The condition_identifier specifies a value for uniquely identifying the Task_condition.

#### 4.2.63.3 requiring_item_task

The requiring_item_task specifies the task and associated Item_version (see 4.2.37) which require special considerations. See 4.3.45 for the application assertion.

### 4.2.64 Task_execution

A Task_execution is an actual performance of a task against an Item_version (see 4.2.37). Each Task_execution may be a Field_maintenance (see 4.2.15) or a Task_execution_context_definition (see 4.2.65).

The data associated with Task_execution are the following:

— requiring_item_task;

— run_identifier.

### 4.2.64.1 requiring_item_task

The requiring_item_task specifies the association of an Item_version (see 4.2.37) with a task that is being performed. See 4.3.46 for the application assertion.

### 4.2.64.2 run_identifier

The run_identifier specifies a unique identification for a particular performance of an Item_task (see 4.2.33). This attribute enables differentiation between multiple performances of an Item_task instance.

## 4.2.65 Task_execution_context_definition

A Task_execution_context_definition is a type of Task_execution (see 4.2.64) where the association of the context end item, as well as the supertask for a component Task_execution is identified.

EXAMPLE 4 - This allows the recording of the complete Task_execution structure or for the recording of a direct reference to a higher level supertask in which the Task_execution context was performed. In other words, the next higher task or a higher task association can be supported.

The data associated with Task_execution_context_definition are the following:

— context_relationship;

— context_task.

### 4.2.65.1 context_relationship

The context_relationship specifies the end item associated with the Task_execution (see 4.2.64). See 4.3.47 for the application assertion.

### 4.2.65.2 context_task

The context_task specifies the supertask or mission in which the use of the task within a task execution belongs. See 4.3.48 for the application assertion.

## 4.2.66 Task_execution_status

A Task_execution_status is the state of a Task_execution (see 4.2.64), that delineates the degree of execution.

The data associated with Task_execution_status are the following:

— executed_task;

— execution_status_value;

— status;

— status_date.

### 4.2.66.1 executed_task

The executed_task specifies the reference to the Item_version (see 4.2.37) being subject to the performance of a task. See 4.3.49 for the application assertion.

### 4.2.66.2 execution_status_value

The execution_status_value specifies the unique identifier of a status for a task execution.

### 4.2.66.3 status

The status specifies the text information that describes the state of the task execution.

### 4.2.66.4 status_date

The status_date specifies the time and date stamp that quantifies when a status is applied to a task execution.

## 4.2.67 Task_execution_support_resource

A Task_execution_support_resource is the identification of a Support_resource (see 4.2.59) that is actually assisting or facilitating the performance of an Item_task (see 4.2.33). A Task_execution_support_resource is an instance of an item that performs, aids, is used by, or is consumed in the execution of an Item_task.

The data associated with Task_execution_support_resource are the following:

— actual_quantity;

— executed_task;

— used_resource.

### 4.2.67.1 actual_quantity

The actual_quantity specifies the number of each type of Support_resource (see 4.2.59) instance that is actually assisting or facilitating the performance of an Item_task (see 4.2.33).

### 4.2.67.2 executed_task

The executed_task specifies the Task_execution (see 4.2.64) for which a Support_resource (see 4.2.59) is associated. See 4.3.51 for the application assertion.

### 4.2.67.3 used_resource

The used_resource specifies the Support_resource (see 4.2.59) that is actually assisting or facilitating the performance of an Item_task (see 4.2.33). See 4.3.50 for the application assertion.

## 4.2.68 Task_requiring_capacity_level

A Task_requiring_capacity_level is a type of Item_capacity_level (see 4.2.21) that defines the level of organizational maintenance capability that an Item_version (see 4.2.37) needs in the performance of an Item_task (see 4.2.33).

The data associated with Task_requiring_capacity_level are the following:

— requiring_item_task.

The requiring_item_task specifies the Item_version and task to which the maintenance capability is defined. See 4.3.52 for the application assertion.

## 4.2.69 Task_skill_level

A Task_skill_level is the assigned Skill_level (see 4.2.55) that is needed for a certain Item_task (see 4.2.33) to be performed.

The data associated with Task_skill_level are the following:

— condition;

— subject_skill_level.

### 4.2.69.1 condition

The condition specifies the special considerations that must be taken into account during the assignment of a Skill_level (see 4.2.55) to a task associated with an Item_version (see 4.2.37). See 4.3.54 for the application assertion.

### 4.2.69.2 subject_skill_level

The subject_skill_level specifies the incremental degree of a refined skill that is required in order to perform a specific task associated with an Item_version (see 4.2.37). See 4.3.53 for the application assertion.

## 4.3 Application assertions

This subclause specifies the application assertions for the life cycle product change process application protocol. Application assertions specify the relationships among 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 Authorization to Enterprise

Each Authorization is authorized by exactly one Enterprise. Each Enterprise is the authorizing enterprise of zero, one, or many Authorization objects.

### 4.3.2 Availability to Assembly_or_component_usage

Each Availability has as the component usage exactly one Assembly_or_component_usage. Each Assembly_or_-component_usage is the component usage for zero, one, or many Availability objects.

### 4.3.3 Conversion_factor to Assembly_or_component_usage

Each Conversion_factor has as the component usage exactly one Assembly_or_component_usage. Each Assembly_or_component_usage is the usage ratio for zero or one Conversion_factor objects.

### 4.3.4 Documentation to Data_template

Each Documentation is formatted using one or more Data_template objects. Each Data_template is the format for zero, one, or many Documentation objects.

### 4.3.5 End_item_effectivity to Functional_definition_usage

Each End_item_effectivity has as the effectivity usage exactly one Functional_defintion_usage. Each Functional_definition_usage is the effectivity usage for zero, one, or many End_item_effectivity objects.

### 4.3.6 End_item_effectivity to Physical_unit

Each End_item_effectivity has as the end item exactly one Physical_unit. Each Physical_unit is the end item for zero, one, or many End_item_effectivity objects.

### 4.3.7 Facility_capacity_level to Facility

Each Facility_capacity_level is for exactly one subject Facility. Each Facility is the subject facility for zero, one, or many Facility_capacity_level objects.

### 4.3.8 Field_maintenance to Item_work_unit_code

Each Field_maintenance has exactly one Item_work_unit_code. Each Item_work_unit_code is for zero, one, or many Field_maintenance objects.

### 4.3.9 Functional_definition_usage to Functional_definition

Each Functional_definition_usage has as the component exactly one Functional_definition. Each Functional_-definition is the component for zero, one, or many Functional_defintional_usage objects.

Each Functional_definition_usage has as the context exactly one Functional_definition. Each Functional_-definition is the context for zero, one, or many Functional_defintional_usage objects.

### 4.3.10 Item_age to Physical_unit

Each Item_age is for exactly one Physical_unit. Each Physical_unit is the related item for zero, one, or many Item_age objects.

### 4.3.11 Item_anomaly to Item_version

Each Item_anomaly has as the item exactly one Item_version. Each Item_version is the item for zero, one, or many Item_anomaly objects.

### 4.3.12 Item_anomaly_disposition to Item_anomaly

Each Item_anomaly_disposition has as the anomalized item exactly one Item_anomaly. Each Item_anomaly is the anomalized item for zero, one, or many Item_anomaly_disposition objects.

### 4.3.13 Item_anomaly_disposition to Support_resource_approval_authority

Each Item_anomaly_disposition is approved by exactly one Support_resource_approval_authority. Each Support_resource_approval_authority is the approving resource for zero, one, or many Item_anomaly_disposition objects.

### 4.3.14 Item_anomaly_disposition to Task_execution

Each Item_anomaly_disposition has as the disposition task exactly one Task_execution. Each Task_execution is the disposition task for zero, one, or many Item_anomaly_disposition objects.

### 4.3.15 Item_capacity_level to Capacity_level

Each Item_capacity_level has as the level exactly one Capacity_level. Each Capacity_level is the level for zero, one, or many Item_capacity_level objects.

### 4.3.16 Item_change to Item_anomaly_disposition

Each Item_change has as the baseline disposition exactly one Item_anomaly_disposition. Each Item_anomaly_-disposition is the baseline disposition for zero, one, or many Item_change objects.

### 4.3.17 Item_change to Item_requiring_change

Each Item_change has as the baseline item exactly one Item_requiring_change. Each Item_requiring_change is the baseline item for zero, one, or many Item_change objects.

### 4.3.18 Item_change to Item_version

Each Item_change has as the resulting item exactly one Item_version. Each Item_version is the resulting item for zero, one, or many Item_change objects.

### 4.3.19 Item_coordinate_location to Functional_definition_usage

Each Item_coordinate_location has as the component usage exactly one Functional_definition_usage. Each Functional_definition_usage is the component usage for zero, one, or many Item_coordinate_location objects.

### 4.3.20 Item_flaw_classification to Item_flaw

Each Item_flaw_classification classifies exactly one Item_flaw. Each Item_flaw is classified by zero, one, or many Item_flaw_classification objects.

### 4.3.21 Item_program to Authorization

Each Item_program is authorized by exactly one Authorization. Each Authorization is the program authorization for zero, one, or many Item_program objects.

### 4.3.22 Item_repairability_level to Assembly_or_component_usage

Each Item_repairability_level has as the component usage exactly one Assembly_or_component_usage. Each Assembly_or_component_usage is the component usage for zero, one, or many Item_repairability_level objects.

### 4.3.23 Item_requiring_change to Item_anomaly

Each Item_requiring_change has as anomalies one or more Item_anomaly objects. Each Item_anomaly is an anomaly for zero, one, or many Item_requiring_change objects.

### 4.3.24 Item_requiring_change to Task_execution

Each Item_requiring_change is required by exactly one Task_execution. Each Task_execution is the requiring change task for zero, one, or many Item_requiring_change objects.

### 4.3.25 Item_responsibility to Item_version

Each Item_responsibility has as the responsible item exactly one Item_version. Each Item_version is the responsible item for zero, one, or many Item_responsibility objects.

### 4.3.26 Item_responsibility to Organization

Each Item_responsibility has as the controlling unit exactly one Organization. Each Organization is the controlling unit for zero, one, or many Item_responsibility objects.

### 4.3.27 Item_task_authorization to Authorization

Each Item_task_authorization is authorized by exactly one Authorization. Each Authorization is the authority for zero, one, or many Item_task_authorization objects.

### 4.3.28 Item_task_frequency to Item_task

Each Item_task_frequency is the frequency for exactly one Item_task. Each Item_task is the subject item task for one or more Item_task_frequency objects.

### 4.3.29 Item_task_time to Item_task

Each Item_task_time is the time for exactly one Item_task. Each Item_task is the subject item task for one or more Item_task_time objects.

### 4.3.30 Item_version_relationship to Item_version

Each Item_version_relationship has as the related item exactly one Item_version. Each Item_version is the related item for zero, one, or many Item_version_relationship objects.

Each Item_version_relationship has as the relating item exactly one Item_version. Each Item_version is the relating item for zero, one, or many Item_version_relationship objects.

### 4.3.31 Item_work_unit_code to Item_version

Each Item_work_unit_code has as the system or part exactly one Item_version. Each Item_version is the system or part for zero, one, or many Item_work_unit_code objects.

### 4.3.32 Personnel_skill_level to Personnel

Each Personnel_skill_level has as the person exactly one Personnel. Each Personnel is the person for zero, one, or many Personnel_skill_level objects.

### 4.3.33 Personnel_skill_level to Skill_level

Each Personnel_skill_level has exactly one subject Skill_level. Each Skill_level is the subject skill level for zero, one, or many Personnel_skill_level objects.

### 4.3.34 Physical_unit to Part

Each Physical_unit has as the baseline part exactly one Part. Each Part is the baseline part for zero, one, or many Physical_unit objects.

### 4.3.35 Recommended_support_resource to Item_task

Each Recommended_support_resource is required by exactly one Item_task. Each Item_task is the requiring task for zero, one, or many Recommended_support_resource objects.

### 4.3.36 Recommended_support_resource to Support_resource

Each Recommended_support_resource has as the resource exactly one Support_resource. Each Support_resource is the resource for zero, one, or many Recommended_support_resource objects.

### 4.3.37 Reference_activity to Task

Each Reference_activity has exactly one Task as the referencing task. Each Task is the referencing task for zero, one, or many Reference_activity objects.

Each Reference_activity has exactly one Task as the referenced task. Each Task is the referenced task for zero, one, or many Reference_activity objects.

### 4.3.38 Related_change to Item_requiring_change

Each Related_change is for exactly one Item_requiring_change. Each Item_requiring_change is the related item for zero, one, or many Related_change objects.

### 4.3.39 Resource_activity_location to Item_coordinate_location

Each Resource_activity_location has as the location exactly one Item_coordinate_location. Each Item_-coordinate_location is the location for zero, one, or many Resource_activity_location objects.

### 4.3.40 Resource_activity_location to Recommended_support_resource

Each Resource_activity_location has as the recommended resources exactly one Recommended_support_-resource. Each Recommended_support_resource is the resource for zero, one, or many Resource_activity_location objects.

### 4.3.41 Skill_level to Skill

Each Skill_level has exactly one subject Skill. Each Skill is the subject skill for zero, one, or many Skill_level objects.

### 4.3.42 Source_maintenance to Assembly_or_component_usage

Each Source_maintenance has as the component usage exactly one Assembly_or_component_usage. Each Assembly_or_component_usage is the component usage for zero or one Source_maintenance objects.

### 4.3.43 Support_resource_approval_authority to Approval_level

Each Support_resource_approval_authority has as an approval exactly one Approval_level. Each Approval_level is the approval for one or more Support_resource_approval_authority objects.

### 4.3.44 Support_resource_approval_authority to Support_resource

Each Support_resource_approval_authority is granted by exactly one Support_resource. Each Support_resource is the granting resource for zero, one, or many Support_resource_approval_authority objects.

### 4.3.45 Task_condition to Item_task

Each Task_condition is required by exactly one Item_task. Each Item_task is the requiring item task for zero, one, or many Task_condition objects.

### 4.3.46 Task_execution to Item_task

Each Task_execution is required by exactly one Item_task. Each Item_task is the requiring task for zero, one, or many Task_execution objects.

### 4.3.47 Task_execution_context_definition to End_item_effectivity

Each Task_execution_context_definition has as the context relationship exactly one End_item_effectivity. Each End_item_effectivity is the context relationship for zero, one, or many Task_execution_context_definition objects.

### 4.3.48 Task_execution_context_definition to Functional_definition_usage

Each Task_execution_context_definition has as the context task exactly one Functional_definition_usage. Each Function_definition_usage is the context task for zero, one, or many Task_execution_context_definition objects.

### 4.3.49 Task_execution_status to Task_execution

Each Task_execution_status is the status of exactly one Task_execution. Each Task_execution is the executed task for one or more Task_execution_status objects.

### 4.3.50 Task_execution_support_resource to Support_resource

Each Task_execution_support_resource uses as the resource exactly one Support_resource. Each Support_-resource is the used resource for zero, one, or many Task_execution_support_resource objects.

### 4.3.51 Task_execution_support_resource to Task_execution

Each Task_execution_support_resource has as the executed task exactly one Task_execution. Each Task_-execution is the executed task for zero, one, or many Task_execution_support_resource objects.

### 4.3.52 Task_requiring_capacity_level to Item_task

Each Task_requiring_capacity_level is required by exactly one Item_task. Each Item_task is the requiring task for zero, one, or many Task_requiring_capacity_level objects.

### 4.3.53 Task_skill_level to Skill_level

Each Task_skill_level has exactly one subject Skill_level. Each Skill_level is the subject skill level for zero, one, or many Task_skill_level objects.

### 4.3.54 Task_skill_level to Task_condition

Each Task_skill_level has exactly one Task_condition. Each Task_condition is the condition for zero, one, or many Task_skill_level 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 an 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. For those AIM elements that are directly incorporated from an application interpreted protocol (AIC), this is the AIC reference.

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.

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) \: line continuation for strings that wrap.

# Table 1 - Mapping table for change_definition UoF

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| DISCREPANT_ITEM | applied_action_assignment | 208 | 6,7,9 | {applied_action_assignme<br>grouped_item = applied_action_a<br>grouped_item <-<br>applied_group_assignment.it<br>applied_group_assignment<br>group_assignment<br>group_assignment.assigned_g<br>group<br>group.name = `discrepant ch |
| failure_rate | [measure_with_unit.value_component]<br>[measure_with_unit.unit_component] | 41<br>41 | 6,8,10 | applied_action_assignment<br>action_assignment<br>action_assignment.assigned_a<br>action<br>supported_item = actio<br>supported_item <-<br>action_resource.usage[i<br>action_resource<br>characterized_resource_definition = a<br>characterized_resource_defini<br>resource_property.resour<br>resource_property <-<br>resource_property_representatio<br>resource_property_represent<br>resource_property_representation.rep<br>representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `fail<br>representation_item =><br>measure_representation_ite<br>{measure_with_unit =><br>ratio_measure_with_uni<br>measure_with_unit<br>[measure_with_unit.value_com<br>[measure_with_unit.unit_com |

## Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| ENHANCEMENT_ITEM | applied_action_assignment | 208 | 6,7,9 | {applied_action_assignme<br>grouped_item = applied_action_a<br>grouped_item <-<br>applied_group_assignment.ite<br>applied_group_assignment<br>group_assignment<br>group_assignment.assigned_gr<br>group<br>group.name = `enhancement c |
| ITEM_ANOMALY | (property_definition)<br>(resource_property)<br>(action_resource) | 41<br>49<br>41 | 1,8,10 | {(property_definition<br>[property_definition.name = `item<br>[property_definition.definiti<br>characterized_definition<br>characterized_definition = characterized_<br>characterized_product_defin<br>characterized_product_definition = pro<br>product_definition<br>product_definition.frame_of_refe<br>product_definition_contex<br>application_context_elem<br>application_context_element.name = `cha<br>(resource_property<br>resource_property.name = `item<br>(action_resource<br>action_resource.name = `item ar |



55

# Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| anomaly_cause | descriptive_representation_-<br>item.description | 45 | 8,10 | (property_definition<br>represented_definition = property<br>represented_definition <<br>property_definition_representatior<br>property_definition_represen<br>property_definition_representation.used_<br>(resource_property <-<br>resource_property_representatior<br>resource_property_represent<br>resource_property_representation.rep<br>(action_resource<br>characterized_resource_definition = a<br>characterized_resource_defini<br>resource_property.resour<br>resource_property <-<br>resource_property_representatior<br>resource_property_represent<br>resource_property_representation.rep<br>representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `anon<br>representation_item =><br>descriptive_representation_<br>descriptive_representation_item.c |

## Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| anomaly_type | group.name | 41 | 7,8,9, 10 | (property_definition<br>grouped_item = property_defi<br>(resource_property<br>grouped_item = resource_prc<br>(action_resource<br>grouped_item = action_reso<br>grouped_item <-<br>applied_group_assignment.it<br>applied_group_assignment<br>{group_assignment<br>group_assignment.role -<br>group_role<br>group_role.name = `anomaly<br>group_assignment<br>group_assignment.assigned_g<br>group<br>group.name |



57

# Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| detection_method | action_method.name | 41 | 1,6,8, 10 | (property_definition<br>property_definition.definitic<br>characterized_definitior<br>characterized_definition = characterized_<br>characterized_product_defir<br>characterized_product_definition = prc<br>product_definition<br>product_definition.formatic<br>product_definition_format<br>action_item = product_definition_<br>action_item <-<br>applied_action_assignment.it<br>applied_action_assignmen<br>{action_assignment<br>action_assignment.role -<br>action_role<br>action_role.name = `detectior<br>action_assignment<br>action_assignment.assigned_ac<br>(resource_property<br>resource_property.resourc<br>characterized_resource_defi<br>characterized_resource_definition = a<br>action_resource<br>action_resource.usage[i]<br>supported_item<br>supported_item = actior<br>(action_resource<br>action_resource.usage[i]<br>supported_item<br>supported_item = actior<br>action<br>action.chosen_method -<br>action_method<br>action_method.name |

## Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_anomaly_description | descriptive_representation_-item.description | 45 | 8,10 | (property_definition<br>represented_definition = property_<br>represented_definition <<br>property_definition_representatior<br>property_definition_represen<br>property_definition_representation.used_<br>(resource_property <-<br>resource_property_representatior<br>resource_property_represent<br>resource_property_representation.rep<br>(action_resource<br>characterized_resource_definition = a<br>characterized_resource_defini<br>resource_property.resour<br>resource_property <-<br>resource_property_representatior<br>resource_property_represent<br>resource_property_representation.rep<br>representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `anomaly<br>representation_item =><br>descriptive_representation_<br>descriptive_representation_item.d |



59

# Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_anomaly_identifier | representation.name | 43 | 8,10 | (property_definition<br>represented_definition = property<br>represented_definition <<br>property_definition_representatior<br>property_definition_represen<br>property_definition_representation.used_<br>(resource_property <-<br>resource_property_representatior<br>resource_property_represent<br>resource_property_representation.rep<br>(action_resource<br>characterized_resource_definition = a<br>characterized_resource_defini<br>resource_property.resour<br>resource_property <-<br>resource_property_representatior<br>resource_property_represent<br>resource_property_representation.rep<br>representation<br>representation.name |

## Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_anomaly to<br>item_version<br>(as item) | PATH | | 8,10 | ((property_definition<br>represented_definition = property_<br>represented_definition <<br>property_definition_representatior<br>property_definition_represen<br>property_definition_representation.used_<br>representation <-<br>resource_property_representation.re<br>resource_property_represent<br>resource_property_representation.<br>resource_property<br>resource_property.resourc<br>characterized_resource_defi<br>characterized_resource_definition = a<br>(action_resource =><br>document_action_resource<br>document)<br>(action_resource)<br>(action_resource =><br>organization_action_resourc<br>organization))<br>(property_definition <-<br>property_definition_relationship.related_p<br>property_definition_relatior<br>property_definition_relationship.relating_pr<br>property_definition)<br>(property_definition<br>property_definition.definitic<br>characterized_definitior<br>characterized_definition = characterized_<br>characterized_product_defir<br>characterized_product_definition = prc<br>product_definition<br>product_definition.formatic<br>product_definition_format<br>product_definition_formation.of_<br>product) |



61

# Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_anomaly to<br>item_version<br>(as item)<br><br>(continued) | | | 1,8,10 | represented_definition = property_<br>represented_definition <<br>property_definition_representatior<br>property_definition_represen<br>property_definition_representation.used_<br>representation <-<br>(action_property_representation.rep<br>action_property_representa<br>action_property_representation.p<br>action_property<br>action_property.definition<br>characterized_action_defini<br>characterized_action_definition<br>action)<br>(action_property_representation.rep<br>action_property_representa<br>action_property_representation.p<br>action_property)<br>(resource_property_representation.r<br>resource_property_represent<br>resource_property_representation.<br>resource_property)))<br>((resource_property<br>resource_property.resourc<br>characterized_resource_defi<br>characterized_resource_definition = a<br>(action_resource =><br>document_action_resource<br>document)<br>(action_resource)<br>(action_resource =><br>organization_action_resourc<br>organization))<br>(resource_property <-<br>resource_property_representatior<br>resource_property_represent<br>resource_property_representation.rep |

## Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_anomaly to<br>item_version<br>(as item)<br><br>(continued) | | | 1,8,10 | property_definition_representation.used<br>property_definition_represen<br>property_definition_representation.<br>represented_definition<br>represented_definition = property_<br>(property_definition)<br>(property_definition<br>property_definition.definitic<br>characterized_definition<br>characterized_definition = characterized_<br>characterized_product_defir<br>characterized_product_definition = pro<br>product_definition<br>product_definition.formatic<br>product_definition_format<br>product_definition_formation.of_<br>product))<br>(resource_property<br>resource_property.resourc<br>characterized_resource_defi<br>characterized_resource_definition = a<br>action_resource<br>action_resource.usage[i]<br>supported_item<br>supported_item = actio<br>action)<br>(resource_property <-<br>resource_property_representatio<br>resource_property_represent<br>resource_property_representation.rep<br>representation <-<br>action_property_representation.rep<br>action_property_representa<br>action_property_representation.p<br>action_property)<br>(resource_property <-<br>resource_property_relationship.related_r |



63

# Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_anomaly to<br>item_version<br>(as item)<br><br>(continued) | | | 1,8,10 | resource_property_relationship.relating_re<br>resource_property))<br>((action_resource =><br>document_action_resource<br>document)<br>(action_resource <-<br>action_resource_relationship.relate<br>action_resource_relationsl<br>action_resource_relationship.relating<br>action_resource)<br>(action_resource =><br>organization_action_resourc<br>organization)<br>(action_resource<br>characterized_resource_definition = a<br>characterized_resource_defini<br>resource_property.resour<br>resource_property <-<br>resource_property_representatior<br>resource_property_represent<br>resource_property_representation.rep<br>representation <-<br>property_definition_representation.usec<br>property_definition_represen<br>property_definition_representation.<br>represented_definition<br>represented_definition = property_<br>(property_definition)<br>(property_definition<br>property_definition.definitic<br>characterized_definitior<br>characterized_definition = characterized_<br>characterized_product_defir<br>characterized_product_definition = prc<br>product_definition<br>product_definition.formatic<br>product_definition_format |

## Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_anomaly to<br>item_version<br>(as item)<br><br>(concluded) | | | | product))<br>(action_resource<br>action_resource.usage[i]<br>supported_item<br>supported_item = actio<br>action)<br>(action_resource<br>characterized_resource_definition = a<br>characterized_resource_defini<br>resource_property.resour<br>(resource_property <-<br>resource_property_representatio<br>resource_property_represen<br>resource_property_representation.rep<br>representation <-<br>action_property_representation.rep<br>action_property_representa<br>action_property_representation.p<br>action_property)<br>(resource_property))) |
| ITEM_ANOMALY_-<br>DISPOSITION | versioned_action_request | 41 | 1 | |



65

# Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_anomaly_disposition<br>to item_anomaly<br>(as anomalized_item) | PATH | | 1 | versioned_action_request<br>(action_request_assignment.assigned_<br>action_request_assignmen<br>applied_action_request_assig<br>applied_action_request_assignmen<br>action_request_item<br>action_request_item = product_<br>product_definition<br>characterized_product_definition = pro<br>characterized_product_defin<br>characterized_definition = characterized_<br>characterized_definition<br>property_definition.definit<br>property_definition<br>{property_definition<br>property_definition.name = `item<br>(action_request_solution.re<br>action_request_solutio<br>action_request_solution.met<br>action_method<br>supported_item = action_m<br>supported_item <-<br>action_resource.usage[i<br>(action_resource<br>characterized_resource_definition = a<br>characterized_resource_defini<br>resource_property.resour<br>resource_property<br>{resource_property<br>resource_property.name = `item a<br>(action_resource<br>{action_resource<br>action_resource.name = `item an |

## Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_anomaly_disposition to support_resource_-approval_authority (as approving_resource) | PATH | | 2,3,8, 10 | versioned_action_request<br>action_directive.requests<br>action_directive<br>supported_item = action_dir<br>supported_item <-<br>action_resource.usage[i<br>action_resource<br>approved_item = action_res<br>approved_item <-<br>applied_approval_assignment.<br>{applied_approval_assignme<br>approval_assignment<br>approval_assignment.role<br>approval_role<br>approval_role.name = `support resource a<br>applied_approval_assignm |
| item_anomaly_disposition to task_execution (as disposition_task) | PATH | | 6 | versioned_action_request<br>action_request_solution.rec<br>action_request_solutior<br>action_request_solution.metl<br>action_method <-<br>action.chosen_method<br>action =><br>executed_action |
| ITEM_CHANGE | [applied_action_assignment]<br>[applied_action_assignment] | 208<br>208 | 6 | {[applied_action_assignmer<br>action_assignment<br>action_assignment.role -<br>action_role<br>action_role.name = `change<br>[applied_action_assignmen<br>action_assignment<br>action_assignment.role -<br>action_role<br>action_role.name = `change |



67

# Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| reason | versioned_action_request.purpose | 41 | 6 | applied_action_assignment<br>{action_assignment<br>action_assignment.role -<br>action_role<br>action_role.name = `change<br>action_assignment<br>action_assignment.assigned_a<br>action<br>action.chosen_method -<br>action_method <-<br>action_request_solution.me<br>action_request_solution<br>action_request_solution.requ<br>versioned_action_reque<br>versioned_action_request.pu |
| item_change to item_-<br>anomaly_disposition<br>(as baseline_item_-<br>disposition) | PATH | | 6 | applied_action_assignment<br>{action_assignment<br>action_assignment.role -<br>action_role<br>action_role.name = `change<br>action_assignment<br>action_assignment.assigned_a<br>action<br>action.chosen_method -<br>action_method <-<br>action_request_solution.me<br>action_request_solution<br>action_request_solution.requ<br>versioned_action_reque |
| item_change to item_-<br>requiring_change<br>(as baseline_item) | PATH | | 6 | {applied_action_assignmen<br>action_assignment<br>action_assignment.role -<br>action_role<br>action_role.name = `change<br>applied_action_assignme<br>applied_action_assignment.it |

## Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_change to item_-<br>version<br>(as resulting_item) | PATH | | 1,6,8,<br>10 | {applied_action_assignmen<br>action_assignment<br>action_assignment.role -<br>action_role<br>action_role.name = `change<br>(applied_action_assignme<br>applied_action_assignment.iten<br>action_item<br>(action_item = documen<br>document)<br>(action_item = action_reso<br>action_resource)<br>(action_item = organizati<br>organization)<br>(action_item = property_defi<br>property_definition)<br>(action_item = product<br>product)<br>(action_item = resource_pro<br>resource_property))<br>(applied_action_assignmen<br>action_assignment<br>action_assignment.assigned_a<br>(action)<br>(action<br>characterized_action_definition<br>characterized_action_definiti<br>action_property.definitio<br>action_property)) |



69

# Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| ITEM_FLAW | property_definition | 41 | 7,9 | {property_definition<br>grouped_item = property_def<br>grouped_item <-<br>applied_group_assignment.it<br>applied_group_assignment<br>group_assignment<br>group_assignment.assigned_g<br>group <-<br>group_relationship.related_g<br>group_relationship<br>group_relationship.relating_gr<br>group<br>group.name = `item flaw |
| ITEM_FLAW_-<br>CLASSIFICATION | group | 41 | 7 | {group <-<br>[group_assignment.assigned_<br>group_assignment<br>group_assignment.role -<br>group_role<br>group_role.name = `item flaw clas<br>[group_relationship.related_<br>group_relationship<br>group_relationship.relating_gr<br>group<br>group.name = `item flaw |
| flaw_class_identifier | group.name | 41 | 7 | |

## Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_flaw_classification to item_flaw (as classified_item) | PATH | | 7,9 | group <- group_assignment.assigned_ group_assignment => applied_group_assignme applied_group_assignment.iter grouped_item grouped_item = property_def property_definition {property_definition grouped_item = property_def grouped_item <- applied_group_assignment.it applied_group_assignment group_assignment group_assignment.assigned_g group <- group_relationship.related_g group_relationship group_relationship.relating_gr group group.name = `item flaw |
| ITEM_ISSUE_OR_-CONCERN | property_definition | 41 | 7,9 | {property_definition grouped_item = property_def grouped_item <- applied_group_assignment.it applied_group_assignment group_assignment group_assignment.assigned_g group group.name = `issue or conc |
| ITEM_REQUIRING_-CHANGE | applied_action_assignment.items[i] | 208 | 6 | {applied_action_assignment.i applied_action_assignment action_assignment action_assignment.role - action_role action_role.name = `change |



71

# Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_change_-<br>requirement_type | action_role.name | 41 | 6 | applied_action_assignment.it<br>applied_action_assignmen<br>action_assignment<br>action_assignment.role -<br>action_role<br>action_role.name<br>{(action_role.name = `chang<br>(action_role.name = `change |
| item_need | action_method.name | 41 | 6 | applied_action_assignment.it<br>applied_action_assignmen<br>action_assignment<br>action_assignment.assigned_a<br>{action =><br>executed_action}<br>action<br>action.chosen_method -<br>action_method<br>action_method.name |
| item_requiring_change to<br>item_anomaly<br>(as anomalies) | PATH | | 1,6 | applied_action_assignment.it<br>applied_action_assignme<br>applied_action_assignment.ite<br>action_item<br>action_item = product_defi<br>product_definition <-<br>product_definition_relationship.relating_<br>product_definition_relation<br>product_definition_relationship.related_pr<br>product_definition<br>characterized_product_definition = pr<br>characterized_product_defi<br>characterized_definition = characterized_<br>characterized_definition<br>property_definition.definit<br>property_definition<br>{property_definition<br>property_definition.name = `item |

## Table 1 - Mapping table for change_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_requiring_change to task_execution (as requiring_change_-task) | PATH | | 6 | applied_action_assignment.it<br>applied_action_assignmen<br>action_assignment<br>action_assignment.assigned_a<br>action =><br>executed_action |
| RELATED_CHANGE | [applied_action_assignment]<br>[applied_action_assignment] | 208<br>208 | 6 | {[applied_action_assignme<br>action_assignment<br>action_assignment.role -<br>action_role<br>action_role.name = `change<br>[applied_action_assignmen<br>action_assignment<br>action_assignment.role -<br>action_role<br>action_role.name = `change |
| related_change to item_-requiring_change (as related_item) | PATH | | | applied_action_assignme<br>applied_action_assignment.it<br>{applied_action_assignment.i<br>applied_action_assignmen<br>action_assignment<br>action_assignment.role -<br>action_role<br>action_role.name = `change |



73

# Table 2 - Mapping table for item_definition UoF

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| ASSEMBLY_OR_-<br>COMPONENT_USAGE | assembly_component_usage | 44 | 9 | |
| assembly_usage_type | (next_assembly_usage_occurrence)<br>(specified_higher_usage_occurrence)<br>(promissory_usage_occurrence) | 44<br>44<br>44 | 9 | assembly_component_usage<br>(next_assembly_usage_occurr<br>(specified_higher_usage_occu<br>(promissory_usage_occurre |
| sequence_number | identification_assignment.assigned_id | 41 | 9 | assembly_component_usage<br>(next_assembly_usage_occur<br>identification_item = next_assembly_usa<br>(specified_higher_usage_occu<br>identification_item = specified_higher_us<br>(promissory_usage_occurre<br>identification_item = promissory_usag<br>identification_item <-<br>applied_identification_assignmen<br>applied_identification_assignm<br>{identification_assignmer<br>identification_assignment.rol<br>identification_role<br>identification_role.name = `sequenc<br>identification_assignmen<br>identification_assignment.assig |

## Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| AUTHORIZATION | applied_approval_assignment | 208 | 2,3 | {applied_approval_assignm<br>applied_approval_assignment.ite<br>approved_item<br>(approved_item = applied_action_a<br>applied_action_assignmen<br>(approved_item = applied_organizatio<br>applied_organization_assignm<br>(approved_item = action_relati<br>action_relationship)<br>(approved_item = organization_re<br>organization_relationship)<br>applied_approval_assignmen<br>approval_assignment |
| authorization_function_-type | approval_role.name | 41 | 2,3 | applied_approval_assignmen<br>approval_assignment<br>approval_assignment.role<br>approval_role<br>approval_role.name |
| authorization_identifier | identification_assignment.assigned_id | 41 | 2,3 | applied_approval_assignmen<br>approval_assignment<br>identification_item = approval_as<br>identification_item <-<br>applied_identification_assignmen<br>applied_identification_assignm<br>identification_assignmen<br>identification_assignment.assig |



75

# Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| authorization_perspective | approver_role.description | 41 | 2,3,4 | applied_approval_assignmen<br>approval_assignment<br>approval_assignment.assigned_ap<br>approval <-<br>approval_person_organization.authori<br>{approval_person_organiza<br>approval_person_organization.person_<br>person_organization_sele<br>person_organization_select = p<br>person}<br>approval_person_organizat<br>approval_person_organization.<br>approver_role<br>approver_role.descriptio |
| authorization_source_-type | organization.name | 41 | 2,3 | applied_approval_assignmen<br>approval_assignment<br>approval_assignment.assigned_ap<br>approval <-<br>approval_person_organization.authori<br>approval_person_organizat<br>approval_person_organization.person_<br>person_organization_sele<br>person_organization_select = org<br>organization<br>organization.name |
| authorization to enterprise (as authorizing_-enterprise) | PATH | | 2,3 | applied_approval_assignmen<br>approval_assignment<br>approval_assignment.assigned_ap<br>approval <-<br>approval_person_organization.authori<br>approval_person_organizat<br>approval_person_organization.person_<br>person_organization_sele<br>person_organization_select = org<br>organization |

## Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| CONVERSION_FACTOR | representation_item | 43 | | {representation_item<br>representation_item.name = `conver |
| conversion_percent | [measure_with_unit.value_component]<br>[measure_with_unit.unit_component] | 41<br>41 | | representation_item =><br>measure_representation_iter<br>{measure_with_unit =><br>ratio_measure_with_unit<br>measure_with_unit<br>[measure_with_unit.value_com<br>[measure_with_unit.unit_comp |
| conversion_factor to assembly_or_-component_usage (as component_usage) | PATH | | 9 | representation_item <-<br>representation.items[i]<br>representation <-<br>property_definition_representation.used<br>property_definition_represent<br>property_definition_representation.d<br>represented_definition<br>represented_definition = property_<br>property_definition<br>property_definition.definitio<br>characterized_definition<br>characterized_definition = characterized_p<br>characterized_product_defin<br>characterized_product_definition = product_d<br>product_definition_relationsh<br>product_definition_usage =<br>assembly_component_usa |
| DATA_TEMPLATE | document | 41 | | |
| template_type | document_type.product_data_type | 41 | | document<br>document.kind -><br>document_type<br>document_type.product_data |
| END_ITEM_-EFFECTIVITY | product_definition_effectivity | 41 | | |



77

# Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| end_item_effectivity to functional_definition_-usage (as effective_usage) | PATH | | | product_definition_effectiv<br>product_definition_effectivity.u<br>product_definition_relationsh<br>product_definition_usag |
| end_item_effectivity to physical_unit (as end_item) | PATH | | | product_definition_effectivit<br>configuration_effectivity<br>configuration_effectivity.configu<br>configuration_design<br>configuration_design.desig<br>configuration_design_ite<br>configuration_design_item = product_def<br>product_definition_formati |
| ENTERPRISE | organization | 41 | | |
| FUNCTIONAL_-DEFINITION | product_definition | 41 | 1 | |
| FUNCTIONAL_-DEFINITION_USAGE | product_definition_usage | 44 | | |
| component_function | product_definition_relationship.description | 41 | | product_definition_usage<br>product_definition_relation<br>product_definition_relationship.d |
| component_quantity | (quantified_assembly_component_-usage.quantity)<br>(make_from_usage_option.quantity) | 44<br>44 | 9 | product_definition_usage<br>(assembly_component_usag<br>quantified_assembly_componen<br>quantified_assembly_component_us<br>(make_from_usage_optio<br>make_from_usage_option.qua |
| occurrence_identifier | product_definition_relationship.id | 41 | 9 | product_definition_usage<br>product_definition_relation<br>product_definition_relationsh |
| usage_type | (assembly_component_usage)<br>(make_from_usage_option) | 44<br>44 | 9 | product_definition_usage<br>(assembly_component_usa<br>(make_from_usage_optio |

## Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| functional_definition_-usage to functional_-definition (as component) | PATH | | 1 | product_definition_usage · product_definition_relations product_definition_relationship.related_pro product_definition |
| functional_definition_-usage to functional_-definition (as context) | PATH | | 1 | product_definition_usage · product_definition_relations product_definition_relationship.relating_pro product_definition |
| ITEM_-DOCUMENTATION_-RELATIONSHIP | applied_document_reference | 208 | | applied_document_referenc document_reference |
| ITEM_PROGRAM | (applied_organization_assignment) (organization_relationship) | 208 41 | | applied_organization_assignm organization_assignment |
| item_program to authorization (as program_-authorization) | PATH | | 2,3 | (applied_organization_assign approved_item = applied_organizatior (organization_relationshi approved_item = organization_rel approved_item <- applied_approval_assignment.it applied_approval_assignm |
| ITEM_RESPONSIBILITY | applied_organization_assignment | 208 | | {applied_organization_assignm organization_assignment organization_assignment.rol organization_role organization_role.name = `item resp applied_organization_assignm organization_assignment |



79

# Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_responsibility to<br>item_version<br>(as responsible_item) | PATH | | 1,8,10 | (applied_organization_assign<br>applied_organization_assignment.i<br>organized_item<br>(organized_item = docume<br>document)<br>(organized_item = action_res<br>action_resource)<br>(organized_item = property_de<br>property_definition)<br>(organized_item = produ<br>product)<br>(organized_item = actior<br>action)<br>(organized_item = action_pro<br>action_property)<br>(organized_item = resource_pr<br>resource_property))<br>(applied_organization_assignm<br>organization_assignment<br>organization_assignment.assigned_or<br>organization) |
| item_responsibility to<br>organization<br>(as controlling_unit) | PATH | | | applied_organization_assignm<br>organization_assignment<br>organization_assignment.assigned_or<br>organization |
| ITEM_VERSION | (document)<br>(action_resource)<br>(organization)<br>(property_definition)<br>(product)<br>(action)<br>(action_property)<br>(resource_property) | 41<br>41<br>41<br>41<br>41<br>41<br>49<br>49 | 1,8,10 | |

## Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| creation_date_time | (date)<br>(date_and_time) | 41<br>41 | 1,8,10 | ((document<br>dated_item = document)<br>(action_resource<br>dated_item = action_resour<br>(organization<br>dated_item = organizatior<br>(property_definition<br>dated_item = property_defini<br>(product<br>dated_item = product)<br>(action<br>dated_item = action)<br>(action_property<br>dated_item = action_propei<br>(resource_property<br>dated_item = resource_prop<br>dated_item <-<br>applied_date_assignment.iter<br>applied_date_assignment<br>{date_assignment<br>date_assignment.role -><br>date_role<br>date_role.name = `creation d<br>date_assignment<br>date_assignment.assigned_da<br>date) |



81

# Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| creation_date_time<br><br>(concluded) | | | 1 | ((document<br>date_and_time_item = docur<br>(action_resource<br>date_and_time_item = action_re<br>(organization<br>date_and_time_item = organiz<br>(property_definition<br>date_and_time_item = property_d<br>(product<br>date_and_time_item = prod<br>(action<br>date_and_time_item = acti<br>(action_property<br>date_and_time_item = action_p<br>(resource_property<br>date_and_time_item = resource_<br>date_and_time_item <-<br>applied_date_and_time_assignmer<br>applied_date_and_time_assignn<br>{date_and_time_assignme<br>date_and_time_assignment.ro<br>date_time_role<br>date_time_role.name = `creation dat<br>date_and_time_assignmer<br>date_and_time_assignment.assigned_da<br>date_and_time) |
| description | (document.description)<br>(action_resource.description)<br>(organization.description)<br>(property_definition.description)<br>(product.description)<br>(action.description)<br>(action_property.description)<br>(resource_property.description) | 41<br>41<br>41<br>41<br>41<br>41<br>49<br>49 | 1,8,10 | |

## Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| family_group | group.name | 41 | 1,7,8, 9,10 | (document<br>grouped_item = documen<br>(action_resource<br>grouped_item = action_reso<br>(organization<br>grouped_item = organizati<br>(property_definition<br>grouped_item = property_defi<br>(product<br>grouped_item = product<br>(action<br>grouped_item = action)<br>(action_property<br>grouped_item = action_prop<br>(resource_property<br>grouped_item = resource_pro<br>grouped_item <-<br>applied_group_assignment.ite<br>applied_group_assignment<br>{group_assignment<br>group_assignment.role -><br>group_role<br>group_role.name = `family<br>group_assignment<br>group_assignment.assigned_gr<br>group<br>group.name |



83

# Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_type | group.name | 41 | 1,7,8, 9,10 | (document<br>grouped_item = documen<br>(action_resource<br>grouped_item = action_reso<br>(organization<br>grouped_item = organizati<br>(property_definition<br>grouped_item = property_defi<br>(product<br>grouped_item = product<br>(action<br>grouped_item = action)<br>(action_property<br>grouped_item = action_prop<br>(resource_property<br>grouped_item = resource_pro<br>grouped_item <-<br>applied_group_assignment.ite<br>applied_group_assignment<br>{group_assignment<br>group_assignment.role -><br>group_role<br>group_role.name = `classifica<br>group_assignment<br>group_assignment.assigned_gr<br>group<br>group.name |

## Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_version_identifier | (identification_assignment.assigned_id)<br>(product_definition_formation.id) | 41<br>41 | 1,8,10 | ((document<br>identification_item = docum<br>(action_resource<br>identification_item = action_re<br>(organization<br>identification_item = organiz<br>(property_definition<br>identification_item = property_d<br>(action<br>identification_item = actio<br>(action_property<br>identification_item = action_pr<br>(resource_property<br>identification_item = resource_p<br>identification_item <-<br>applied_identification_assignmen<br>applied_identification_assignm<br>{identification_assignmer<br>identification_assignment.ro<br>identification_role<br>identification_role.name = `vers<br>identification_assignmen<br>identification_assignment.assig<br>(product <-<br>product_definition_formation.of_<br>product_definition_formati<br>product_definition_formatio |
| nomenclature | (document.name)<br>(action_resource.name)<br>(organization.name)<br>(property_definition.name)<br>(product.name)<br>(action.name)<br>(action_property.name)<br>(resource_property.name) | 41<br>41<br>41<br>41<br>41<br>41<br>49<br>49 | 1,8,10 | |



85

# Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| ITEM_VERSION_-<br>RELATIONSHIP | (applied_document_reference)<br>(action_resource_relationship)<br>(organization_relationship)<br>(property_definition_relationship)<br>(product_definition_formation_-<br>relationship)<br>(product_definition_relationship)<br>(action_relationship)<br>(property_definition)<br>(action_property)<br>(resource_property) | 208<br>41<br>41<br>41<br><br>41<br>41<br>41<br>41<br>49<br>49 | 8 | (applied_document_referenc<br>document_reference) |

## Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_version_-<br>relationship to item_-<br>version<br>(as related_item) | PATH | | 1,8,10 | (applied_document_referenc<br>document_reference<br>document_reference.assigned_doc<br>document)<br>(action_resource_relationsl<br>action_resource_relationship.related_<br>action_resource)<br>(organization_relationshi<br>organization_relationship.related_org<br>organization)<br>(property_definition_relation<br>property_definition_relationship.related_pro<br>property_definition)<br>((product_definition_formation_re<br>product_definition_formation_relationship.related_pro<br>(product_definition_relation<br>product_definition_relationship.related_pro<br>product_definition<br>product_definition.formatior<br>product_definition_formati<br>product_definition_formation.of_p<br>product)<br>(action_relationship<br>action_relationship.related_act<br>action)<br>(property_definition <-<br>property_definition_relationship.relating_p<br>property_definition_relation<br>property_definition_relationship.related_pro<br>property_definition)<br>(action_property <-<br>action_property_relationship.relating_a<br>action_property_relationsh<br>action_property_relationship.related_act<br>action_property)<br>(resource_property <-<br>resource_property_relationship.relating_r |



87

# Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_version_-<br>relationship to item_-<br>version<br>(as related_item)<br><br>(concluded) | | | | resource_property_relationship.related_res<br>resource_property) |

## Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_version_-<br>relationship to item_-<br>version<br>(as relating_item) | PATH | | 1,8,10<br>11 | (applied_document_referer<br>applied_document_reference.ite<br>document_reference_iter<br>(document_reference_item = dc<br>document)<br>(document_reference_item = actio<br>action_resource)<br>(document_reference_item = org<br>organization)<br>(document_reference_item = propert<br>property_definition)<br>(document_reference_item = p<br>product)<br>(document_reference_item =<br>action)<br>(document_reference_item = actio<br>action_property)<br>(document_reference_item = resour<br>resource_property))<br>(action_resource_relationsl<br>action_resource_relationship.relating<br>action_resource)<br>(organization_relationshi<br>organization_relationship.relating_or<br>organization <-<br>organization_assignment.assigned_c<br>organization_assignment =<br>applied_organization_assign<br>applied_organization_assignment.i<br>organized_item<br>(organized_item = docume<br>document)<br>(organized_item = action_res<br>action_resource)<br>(organized_item = organiza<br>organization)<br>(organized_item = property_de |



89

# Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_version_-<br>relationship to item_-<br>version<br>(as relating_item)<br><br>(concluded) | | | | (organized_item = produ<br>product)<br>(organized_item = actior<br>action)<br>(organized_item = action_pro<br>action_property)<br>(organized_item = resource_pr<br>resource_property))<br>(property_definition_relation<br>property_definition_relationship.relating_pr<br>property_definition)<br>((product_definition_formation_re<br>product_definition_formation_relationship.relating_pro<br>(product_definition_relation<br>product_definition_relationship.relating_pr<br>product_definition<br>product_definition.formatior<br>product_definition_formati<br>product_definition_formation.of_p<br>product)<br>(action_relationship<br>action_relationship.relating_ac<br>action)<br>(property_definition <-<br>property_definition_relationship.related_pr<br>property_definition_relation<br>property_definition_relationship.relating_pr<br>property_definition)<br>(action_property <-<br>action_property_relationship.related_a<br>action_property_relationsh<br>action_property_relationship.relating_act<br>action_property)<br>(resource_property <-<br>resource_property_relationship.related_re<br>resource_property_relations<br>resource_property_relationship.relating_res |

## Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| MAKE_FROM_USAGE | make_from_usage_option | 44 | | |
| ORGANIZATION | organization | 41 | | |
| organization_type | organization.name | 41 | | |
| PART | product | 41 | 1 | |
| function_type | product_context.discipline_type | 41 | 1 | product<br>product.frame_of_reference[<br>product_context<br>product_context.discipline_t |
| name | product.name | 41 | 1 | |
| type | product_category.name | 41 | 1 | product <-<br>product_related_product_category.<br>product_related_product_categ<br>product_category<br>product_category.name |
| PART_SYSTEM_-RELATIONSHIP | product_definition_formation_relationship | 41 | | |
| PHYSICAL_UNIT | product_definition_formation | 41 | | |
| physical_unit_type | group.name | 41 | 7,9 | product_definition_formati<br>grouped_item = product_definition<br>grouped_item <-<br>applied_group_assignment.ite<br>applied_group_assignment<br>{group_assignment<br>group_assignment.role -><br>group_role<br>group_role.name = `physical un<br>group_assignment<br>group_assignment.assigned_gr<br>group<br>group.name |
| physical_unit to part (as baseline_part) | PATH | | 1 | product_definition_formati<br>product_definition_formation.of_p<br>product |



91

# Table 2 - Mapping table for item_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| PRODUCT_PROPERTY | property_definition | 41 | | |
| PROGRAM | organization | 41 | | |
| SOURCE_-<br>MAINTENANCE | applied_group_assignment | 208 | 7,9 | {applied_group_assignment<br>group_assignment<br>group_assignment.role -><br>group_role<br>group_role.name = `source maint<br>applied_group_assignment<br>group_assignment |
| source_maintenance_-<br>repair_code | group.name | 41 | 7,9 | applied_group_assignment<br>group_assignment<br>group_assignment.assigned_gr<br>group<br>group.name |
| source_maintenance to<br>assembly_or_-<br>component_usage<br>(as component_usage) | PATH | | 7,9 | applied_group_assignmer<br>applied_group_assignment.item<br>grouped_item<br>grouped_item = assembly_compor<br>assembly_component_usa |
| SUPPLIER | organization | 41 | | |
| code | organization.id | 41 | | |
| type | group.name | 41 | 7,9 | organization<br>grouped_item = organizati<br>grouped_item <-<br>applied_group_assignment.ite<br>applied_group_assignment<br>group_assignment<br>group_assignment.assigned_gr<br>group<br>group.name<br>{(group.name = `manufactu<br>(group.name = `non-manufact |
| SYSTEM | product | 41 | 1 | |

## Table 2 - Mapping table for item_definition UoF (concluded)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| system_type | product_context.discipline_type | 41 | 1 | product<br>product.frame_of_reference[<br>product_context<br>product_context.discipline_t |



93

# Table 3 - Mapping table for item_properties UoF

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| AVAILABILITY | item_property_representation | 208 | | item_property_representatic<br>representation<br>{representation<br>representation.name = `availal |
| availability_value | [measure_with_unit.value_component]<br>[measure_with_unit.unit_component] | 41<br>41 | | item_property_representatic<br>representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `availal<br>representation_item =><br>measure_representation_ite<br>{measure_with_unit =><br>ratio_measure_with_uni<br>measure_with_unit<br>[measure_with_unit.value_con<br>[measure_with_unit.unit_com |
| indicator | descriptive_representation_-<br>item.description | 45 | | item_property_representat<br>representation<br>representation.item ->><br>{representation_item<br>representation_item.name = `availabil<br>representation_item =><br>descriptive_representation_<br>descriptive_representation_item.d |

## Table 3 - Mapping table for item_properties UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| availability to assembly_-<br>or_component_usage<br>(as component_usage) | PATH | | 9 | item_property_representat<br>representation <-<br>property_definition_representation.usec<br>property_definition_represen<br>property_definition_representation.<br>represented_definition<br>represented_definition = property_<br>property_definition<br>property_definition.definitic<br>characterized_definitior<br>characterized_definition = characterized_<br>characterized_product_defir<br>characterized_product_definition = product_c<br>product_definition_relationsl<br>product_definition_usage<br>assembly_component_usa |
| CAPACITY_LEVEL | representation_item | 43 | 8,10 | {representation_item <-<br>representation.items[i]<br>{representation<br>representation.name = `organization c<br>representation <-<br>resource_property_representation.re<br>resource_property_represent<br>resource_property_representation.<br>resource_property<br>resource_property.resourc<br>characterized_resource_defi<br>characterized_resource_definition = a<br>action_resource =><br>organization_action_resourc<br>organization} |
| capacity_level_function | descriptive_representation_-<br>item.description | 45 | | representation_item =><br>descriptive_representation_<br>descriptive_representation_item.d |
| FACILITY_CAPACITY_-<br>LEVEL | representation_item | 43 | | {representation_item<br>representation_item.name = `facility c |



95

# Table 3 - Mapping table for item_properties UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| facility_capacity_level to facility<br>(as subject_facility) | PATH | | 8,10 | representation_item <-<br>representation.items[i]<br>representation <-<br>resource_property_representation.re<br>resource_property_represent<br>resource_property_representation.<br>resource_property<br>resource_property.resourc<br>characterized_resource_defi<br>characterized_resource_definition = a<br>action_resource<br>{action_resource<br>action_resource.kind -><br>action_resource_type<br>action_resource_type.name = `f |
| ITEM_AGE | item_property_representation | 208 | | item_property_representatic<br>{representation<br>representation.name = `item<br>representation |
| age_date_time | date_and_time | 41 | | item_property_representatic<br>representation<br>date_and_time_item = represe<br>date_and_time_item <-<br>applied_date_and_time_assignme<br>applied_date_and_time_assign<br>{date_and_time_assignm<br>date_and_time_assignment.r<br>date_time_role<br>date_time_role.name = `age recor<br>date_and_time_assignme<br>date_and_time_assignment.assigned_d<br>date_and_time |

## Table 3 - Mapping table for item_properties UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| age_defined_state | descriptive_representation_-<br>item.description | 45 | | item_property_representatic<br>representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `age de<br>representation_item =><br>descriptive_representation_<br>descriptive_representation_item.d |
| age_value | [measure_with_unit.value_component]<br>[measure_with_unit.unit_component] | 41<br>41 | | item_property_representatic<br>representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `ag<br>representation_item =><br>measure_representation_ite<br>measure_with_unit<br>[measure_with_unit.value_con<br>[measure_with_unit.unit_com |
| item_age to physical_unit<br>(as related_item) | PATH | | 1 | item_property_representatic<br>representation <-<br>property_definition_representation.usec<br>property_definition_represen<br>property_definition_representation.<br>represented_definition<br>represented_definition = property_<br>property_definition<br>property_definition.definitic<br>characterized_definitior<br>characterized_definition = characterized_<br>characterized_product_defir<br>characterized_product_definition = prc<br>product_definition<br>product_definition.formatic<br>product_definition_format |



97

# Table 3 - Mapping table for item_properties UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| ITEM_CAPACITY_-<br>LEVEL | (item_property_representation)<br>(representation_item) | 208<br>43 | | (item_property_representatic<br>{representation<br>(representation.name = `item repaira<br>(representation.name = `task requiring<br>representation)<br>({representation_item<br>representation_item.name = `facility c |
| item_capacity_level to<br>capacity_level<br>(as level) | PATH | | | (item_property_representatic<br>representation<br>representation.items[i] -<br>representation_item<br>{representation_item<br>representation_item.name = `organizatio<br>(IDENTICAL MAPPIN |
| ITEM_PRODUCT_-<br>PROPERTY | (property_definition)<br>(action_property)<br>(resource_property) | 41<br>49<br>49 | | |
| ITEM_REPAIRABILITY_-<br>LEVEL | item_property_representation | 208 | | item_property_representatic<br>representation<br>{representation<br>representation.name = `item repaira |
| condemnation_percentage | [measure_with_unit.value_component]<br>[measure_with_unit.unit_component] | 41<br>41 | | item_property_representatic<br>representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `conder<br>representation_item =><br>measure_representation_ite<br>{measure_with_unit =><br>ratio_measure_with_uni<br>measure_with_unit<br>[measure_with_unit.value_con<br>[measure_with_unit.unit_com |

## Table 3 - Mapping table for item_properties UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| not_repairable_percentage | [measure_with_unit.value_component]<br>[measure_with_unit.unit_component] | 41<br>41 | | item_property_representatic<br>representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `not rep<br>representation_item =><br>measure_representation_ite<br>{measure_with_unit =><br>ratio_measure_with_uni<br>measure_with_unit<br>[measure_with_unit.value_com<br>[measure_with_unit.unit_com |
| repairable_percentage | [measure_with_unit.value_component]<br>[measure_with_unit.unit_component] | 41<br>41 | | item_property_representatic<br>representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `repai<br>representation_item =><br>measure_representation_ite<br>{measure_with_unit =><br>ratio_measure_with_uni<br>measure_with_unit<br>[measure_with_unit.value_com<br>[measure_with_unit.unit_com |



99

# Table 3 - Mapping table for item_properties UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_repairability_level to assembly_or_-component_usage (as component_usage) | PATH | | 9 | item_property_representatic representation <- property_definition_representation.usec property_definition_represen property_definition_representation. represented_definition represented_definition = property_ property_definition property_definition.definitic characterized_definitior characterized_definition = characterized_ characterized_product_defir characterized_product_definition = product_ product_definition_relationsh product_definition_usage assembly_component_usa |
| TASK_REQUIRING_-CAPACITY_LEVEL | item_property_representation | 208 | | item_property_representatic representation {representation representation.name = `task requiring |

## Table 3 - Mapping table for item_properties UoF (concluded)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| task_requiring_capacity_-<br>level to item_task<br>(as requiring_item_task) | PATH | | 6,11 | item_property_representatic<br>representation <-<br>property_definition_representation.usec<br>property_definition_represen<br>property_definition_representation.<br>represented_definition<br>represented_definition = property_<br>property_definition<br>property_definition.definitic<br>characterized_definitior<br>characterized_definition = characte<br>characterized_object =><br>characterized_applied_action_assi<br>(applied_action_assignme<br>(applied_action_assignmen<br>action_assignment<br>action_assignment.assigned_a<br>action <-<br>action_relationship.?<br>action_relationship) |



101

# Table 4 - Mapping table for supporting_resources UoF

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| APPROVAL_LEVEL | approval | 41 | 2 | |
| approval_identifier | approval.level | 41 | 2 | |
| DOCUMENTATION | document | 41 | | |
| name | document.name | 41 | | |
| type | document_type.product_data_type | 41 | | document<br>document.kind -><br>document_type<br>document_type.product_data |
| documentation to data_-template<br>(as format) | PATH | | | document <-<br>document_relationship.related_c<br>{document_relationship<br>document_relationship.name = `docume<br>document_relationship<br>document_relationship.relating_d<br>document |
| FACILITY | action_resource | 41 | 8,10 | {action_resource<br>action_resource.kind -><br>action_resource_type<br>action_resource_type.name = `f |
| geographic_location | representation | 43 | 8,10 | action_resource<br>characterized_resource_definition = a<br>characterized_resource_defini<br>resource_property.resour<br>resource_property <-<br>resource_property_representatior<br>resource_property_represent<br>resource_property_representation.rep<br>{representation<br>representation.name = `geographic<br>representation |

## Table 4 - Mapping table for supporting_resources UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| type | group.name | 41 | 7,8,9, 10 | action_resource<br>grouped_item = action_resc<br>grouped_item <-<br>applied_group_assignment.it<br>applied_group_assignment<br>{group_assignment<br>group_assignment.role -<br>group_role<br>group_role.name = `facility t<br>group_assignment<br>group_assignment.assigned_g<br>group<br>group.name |
| PACKAGE | action_resource | 41 | 8,10 | {action_resource<br>action_resource.kind -><br>action_resource_type<br>action_resource_type.name = `p |
| package_type | group.name | 41 | 7,8,9, 10 | action_resource<br>grouped_item = action_resc<br>grouped_item <-<br>applied_group_assignment.it<br>applied_group_assignment<br>{group_assignment<br>group_assignment.role -<br>group_role<br>group_role.name = `package<br>group_assignment<br>group_assignment.assigned_g<br>group<br>group.name |
| PERSONNEL | action_resource | 41 | 8,10 | {action_resource<br>action_resource.kind -><br>action_resource_type<br>action_resource_type.name = `pe |



103

# Table 4 - Mapping table for supporting_resources UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| PERSONNEL_SKILL_-<br>LEVEL | resource_property_representation | 49 | | |
| personnel_skill_level to<br>personnel<br>(as person) | PATH | | 8,10 | resource_property_represent<br>resource_property_representation.<br>resource_property<br>resource_property.resourc<br>characterized_resource_defi<br>characterized_resource_definition = a<br>action_resource<br>{action_resource<br>action_resource.kind -><br>action_resource_type<br>action_resource_type.name = `p |
| personnel_skill_level to<br>skill_level<br>(as subject_skill_level) | PATH | | | resource_property_represent<br>resource_property_representation.rep<br>representation<br>{representation<br>representation.name = `skill l |
| SKILL | action_resource | 41 | 8,10 | {action_resource<br>action_resource.kind -><br>action_resource_type<br>action_resource_type.name = |
| skill_description | action_resource.description | 41 | 8,10 | |
| skill_identifier | action_resource.name | 41 | 8,10 | |
| SKILL_LEVEL | representation | 43 | | {representation<br>representation.name = `skill l |
| description | descriptive_representation_-<br>item.description | 45 | | representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `skill leve<br>representation_item =><br>descriptive_representation_<br>descriptive_representation_item.d |

## Table 4 - Mapping table for supporting_resources UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| skill_level to skill<br>(as subject_skill) | PATH | | 8,10 | representation <-<br>resource_property_representation.re<br>resource_property_represent<br>resource_property_representation.<br>resource_property<br>resource_property.resourc<br>characterized_resource_defi<br>characterized_resource_definition = a<br>action_resource |
| SUPPORT_EQUIPMENT | action_resource | 41 | 8,10 | {action_resource<br>action_resource.kind -><br>action_resource_type<br>action_resource_type.name = `suppo |
| install_factor | representation | 43 | 8,10 | action_resource<br>characterized_resource_definition = a<br>characterized_resource_defini<br>resource_property.resour<br>resource_property <-<br>resource_property_representatio<br>resource_property_represent<br>resource_property_representation.rep<br>{representation<br>representation.name = `support equipme<br>representation |
| SUPPORT_RESOURCE | action_resource | 41 | 8,10 | |
| SUPPORT_RESOURCE_-<br>APPROVAL_-<br>AUTHORITY | applied_approval_assignment | 208 | 2,3 | applied_approval_assignme<br>{approval_assignment<br>approval_assignment.role<br>approval_role<br>approval_role.name = `support resource a<br>approval_assignment |
| support_resource_-<br>approval_authority to<br>approval_level<br>(as approval) | PATH | | 2,3 | applied_approval_assignme<br>approval_assignment<br>approval_assignment.assigned_a<br>approval |



105

# Table 4 - Mapping table for supporting_resources UoF (concluded)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| support_resource_-<br>approval_authority to<br>support_resource<br>(as granting_resource) | PATH | | 2,3,8,<br>10 | applied_approval_assignm<br>applied_approval_assignment.ite<br>approved_item<br>approved_item = action_res<br>action_resource |

# Table 5 - Mapping table for task_definition UoF

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| FIELD_MAINTENANCE | executed_action | 41 | 6 | |
| maintenance_type | group.name | 41 | 6,7,9 | executed_action <=<br>action<br>grouped_item = action<br>grouped_item <-<br>applied_group_assignment.it<br>applied_group_assignmen<br>group_assignment<br>group_assignment.assigned_g<br>group<br>group.name<br>{(group.name = `operatior<br>(group.name = `inspectio<br>(group.name = `repair')<br>(group.name)} |
| field_maintenance to<br>item_work_unit_code<br>(as work_unit_code) | PATH | | 6 | executed_action <=<br>action <-<br>action_assignment.assigned_<br>action_assignment =><br>applied_action_assignme<br>applied_action_assignment.iter<br>{action_item<br>action_item = product_definition_<br>product_definition_formati<br>action_item<br>identification_item = action_<br>identification_item <-<br>applied_identification_assignmer<br>applied_identification_assigr<br>{applied_identification_assigr<br>identification_assignmer<br>identification_assignment.rc<br>identification_role<br>identification_role.name = `work |



107

# Table 5 - Mapping table for task_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| ITEM_COORDINATE_-<br>LOCATION | item_property_representation | 208 | | item_property_representatic<br>representation<br>{representation<br>[representation.name = `item coordi<br>[representation.items[i] -<br>representation_item =><br>geometric_representation_ite<br>(cartesian_point)<br>(placement)]} |
| component_location | geometric_representation_item | 42 | | item_property_representatic<br>representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `component<br>representation_item =><br>geometric_representation_i |
| coordinate_system_<br>identifier | representation_context.context_identifier | 43 | | item_property_representatic<br>representation<br>representation.context_of_ite<br>representation_context<br>representation_context.context_ |
| item_coordinate_location<br>to functional_definition_-<br>usage<br>(as component_usage) | PATH | | | item_property_representatic<br>representation <-<br>property_definition_representation.usec<br>property_definition_represen<br>property_definition_representation.<br>represented_definition<br>represented_definition = property_<br>property_definition<br>property_definition.definitic<br>characterized_definition<br>characterized_definition = characterized_<br>characterized_product_defir<br>characterized_product_definition = product_<br>product_definition_relationsl<br>product_definition_usag |

## Table 5 - Mapping table for task_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| ITEM_TASK | (applied_action_assignment)<br><br>(action_relationship) | 208<br><br>41 | 6 | applied_action_assignmen<br>action_assignment |
| item_operability | descriptive_representation_-<br>item.description | 45 | 6,8,10 | (applied_action_assignmen<br>action_assignment<br>action_assignment.assigned_ac<br>(action_relationship<br>action_relationship.relating_ac<br>action<br>characterized_action_definition<br>characterized_action_definiti<br>action_property.definitio<br>action_property <-<br>action_property_representation.<br>action_property_representa<br>action_property_representation.repr<br>representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `item c<br>representation_item =><br>descriptive_representation_<br>descriptive_representation_item.c |
| ITEM_TASK_-<br>AUTHORIZATION | applied_approval_assignment | 208 | 2,3,6 | {applied_approval_assignn<br>applied_approval_assignment.it<br>approved_item<br>(approved_item = applied_action_<br>applied_action_assignme<br>(approved_item = action_relat<br>action_relationship)} |
| item_task_authorization to<br>authorization<br>(as authority) | IDENTICAL MAPPING | | | |
| ITEM_TASK_-<br>FREQUENCY | representation | 43 | | {representation<br>representation.name = `item task f |



109

# Table 5 - Mapping table for task_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| task_frequency | [measure_with_unit.value_component]<br>[measure_with_unit.unit_component] | 41<br>41 | | representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `task<br>representation_item =><br>measure_representation_ite<br>{measure_with_unit =><br>ratio_measure_with_uni<br>measure_with_unit<br>[measure_with_unit.value_con<br>[measure_with_unit.unit_com |
| task_frequency_indicator | type_qualifier.name | 45 | 5 | representation<br>representation.items[i] -<br>representation_item =><br>qualified_representation_i<br>qualified_representation_item.qua<br>value_qualifier<br>value_qualifier = type_qual<br>type_qualifier<br>type_qualifier.name<br>{(type_qualifier.name = `allo<br>(type_qualifier.name = `meas<br>(type_qualifier.name = `predi |
| item_task_frequency to item_task<br>(as subject_item_task) | PATH | | 6,11 | representation <-<br>property_definition_representation.usec<br>property_definition_represen<br>property_definition_representation.<br>represented_definition<br>represented_definition = property_<br>property_definition<br>property_definition.definitic<br>characterized_definitior<br>characterized_definition = characte<br>characterized_object =><br>characterized_applied_action_assi<br>applied_action_assignme |

## Table 5 - Mapping table for task_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| ITEM_TASK_TIME | representation | 43 | | {representation<br>representation.name = `item tas |
| task_time_indicator | type_qualifier.name | 45 | 5 | representation<br>representation.items[i] -<br>representation_item =><br>qualified_representation_i<br>qualified_representation_item.qua<br>value_qualifier<br>value_qualifier = type_qual<br>type_qualifier<br>type_qualifier.name<br>{(type_qualifier.name = `allo<br>(type_qualifier.name = `meas<br>(type_qualifier.name = `predi |
| time_duration | [measure_with_unit.value_component]<br>[measure_with_unit.unit_component] | 41<br>41 | | representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `time<br>representation_item =><br>measure_representation_ite<br>{measure_with_unit =><br>ratio_measure_with_uni<br>measure_with_unit<br>[measure_with_unit.value_con<br>[measure_with_unit.unit_com |



111

# Table 5 - Mapping table for task_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_task_time to item_-task<br>(as subject_item_task) | PATH | | 6,11 | representation <-<br>property_definition_representation.usec<br>property_definition_represen<br>property_definition_representation.<br>represented_definition<br>represented_definition = property.<br>property_definition<br>property_definition.definitic<br>characterized_definitior<br>characterized_definition = characte<br>characterized_object =><br>characterized_applied_action_assi<br>applied_action_assignme |
| ITEM_WORK_UNIT_-CODE | applied_identification_assignment | 208 | | {applied_identification_assign<br>identification_assignmer<br>identification_assignment.rc<br>identification_role<br>identification_role.name = `work |
| work_unit_code_identifier | identification_assignment.assigned_id | 41 | | |

## Table 5 - Mapping table for task_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| item_work_unit_code to item_version (as system_or_part) | PATH | | 1,8,10 | applied_identification_assigr applied_identification_assignment identification_item (identification_item = docu document) (identification_item = action_r action_resource) (identification_item = organi organization) (identification_item = property_ property_definition) (identification_item = product_definit product_definition_format product_definition_formation.of_ product) (identification_item = acti action) (identification_item = action_p action_property) (identification_item = resource_ resource_property) |
| RECOMMENDED_-SUPPORT_RESOURCE | action_resource | 41 | 8,10 | |



113

# Table 5 - Mapping table for task_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| recommended_quantity | [measure_with_unit.value_component]<br>[measure_with_unit.unit_component] | 41<br>41 | 8,10 | action_resource<br>characterized_resource_definition = a<br>characterized_resource_defini<br>resource_property.resour<br>resource_property <-<br>resource_property_representatior<br>resource_property_represent<br>resource_property_representation.rep<br>{representation<br>(representation.name = `recommended re<br>(representation.name = `charact<br>representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `recomme<br>representation_item =><br>measure_representation_ite<br>measure_with_unit<br>[measure_with_unit.value_con<br>[measure_with_unit.unit_com |

## Table 5 - Mapping table for task_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| technical_evaluation_-<br>priority | compound_representation_item | 43 | 8,10 | action_resource<br>characterized_resource_definition = a<br>characterized_resource_defini<br>resource_property.resour<br>resource_property <-<br>resource_property_representatior<br>resource_property_represent<br>resource_property_representation.rep<br>representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `technical ev<br>representation_item =><br>{compound_representation_<br>compound_representation_item.iten<br>compound_item_definiti<br>compound_item_definition = set_repre<br>[set_representation_iten<br>set_representation_item[1]: = repres<br>{representation_item<br>representation_item.name =<br>representation_item =><br>descriptive_representation_i<br>[set_representation_iten<br>set_representation_item[2]: = repres<br>{representation_item<br>representation_item.name =<br>representation_item =><br>measure_representation_ite<br>measure_with_unit]<br>[set_representation_iten<br>set_representation_item[3]: = repres<br>{representation_item<br>representation_item.name =<br>representation_item =><br>descriptive_representation_it<br>compound_representation_ |



115

# Table 5 - Mapping table for task_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| recommended_support_-<br>resource to item_task<br>(as requiring_item_task) | PATH | | 6,8,10 | action_resource<br>action_resource.usage[1]<br>supported_item<br>supported_item = actio<br>action <-<br>action_assignment.assigned_<br>action_assignment =><br>applied_action_assignme |
| recommended_support_-<br>resource to support_-<br>resource<br>(as resource) | PATH | | 8,10 | action_resource <-<br>action_resource_relationship.relati<br>{action_resource_relations<br>action_resource_relationship.name = `recom<br>action_resource_relations<br>action_resource_relationship.related<br>action_resource |
| REFERENCE_ACTIVITY | action_relationship | 41 | | |
| reference_activity to task<br>(as referenced_task) | PATH | | | action_relationship<br>action_relationship.related_ac<br>action |
| reference_activity to task<br>(as referencing_task) | PATH | | | action_relationship<br>action_relationship.relating_ac<br>action |
| RESOURCE_ACTIVITY_-<br>LOCATION | (action_property_representation)<br>(resource_property_representation) | 49<br>49 | | |
| resource_activity_location<br>to item_coordinate_-<br>location<br>(as location) | PATH | | | (action_property_represent<br>action_property_representation.repre<br>(resource_property_represen<br>resource_property_representation.rep<br>{representation<br>representation.name = `item coordin<br>representation =><br>item_property_representat |

## Table 5 - Mapping table for task_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| resource_activity_location to recommended_-support_resource (as recommended_-resource) | PATH | | 8,10 | (action_property_representa<br>action_property_representation.p<br>action_property<br>action_property.definition<br>characterized_action_defini<br>characterized_action_definition<br>action<br>supported_item = actio<br>supported_item <-<br>action_resource.usage[1<br>action_resource)<br>(resource_property_represen<br>resource_property_representation.<br>resource_property<br>resource_property.resourc<br>characterized_resource_defi<br>characterized_resource_definition = a<br>action_resource) |
| TASK | action | 41 | | |
| acceptance_criteria | action_directive.comment | 41 | | action<br>action.chosen_method -<br>action_method <-<br>action_request_solution.me<br>action_request_solutio<br>action_request_solution.requ<br>versioned_action_request<br>action_directive.requests<br>action_directive<br>action_directive.comme |



117

# Table 5 - Mapping table for task_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| consideration | action_directive.analysis | 41 | | action<br>action.chosen_method -<br>action_method <-<br>action_request_solution.me<br>action_request_solutior<br>action_request_solution.requ<br>versioned_action_request<br>action_directive.requests<br>action_directive<br>action_directive.analysis |
| name | action.name | 41 | | |
| procedure | action.description | 41 | | |
| purpose | action_directive.description | 41 | | action<br>action.chosen_method -<br>action_method <-<br>action_request_solution.me<br>action_request_solutior<br>action_request_solution.requ<br>versioned_action_request<br>action_directive.requests<br>action_directive<br>action_directive.descripti |
| TASK_CONDITION | item_property_representation | 208 | | item_property_representatic<br>{representation<br>representation.name = `task cor<br>representation |
| condition_description | descriptive_representation_-<br>item.description | 45 | | item_property_representatic<br>representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `conditio<br>representation_item =><br>descriptive_representation_<br>descriptive_representation_item.d |

## Table 5 - Mapping table for task_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| condition_identifier | descriptive_representation_-<br>item.description | 45 | | item_property_representatic<br>representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `conditi<br>representation_item =><br>descriptive_representation_<br>descriptive_representation_item.d |
| task_condition to item_-<br>task<br>(as requiring_item_task) | PATH | | 6 | item_property_representatic<br>representation <-<br>action_property_representation.rep<br>action_property_representa<br>action_property_representation.p<br>action_property<br>action_property.definition<br>characterized_action_defini<br>characterized_action_definition<br>action <-<br>action_assignment.assigned_<br>action_assignment =><br>applied_action_assignme |
| TASK_EXECUTION | executed_action | 41 | 6 | |



119

# Table 5 - Mapping table for task_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| run_identifier | identification_assignment.assigned_id | 41 | 6 | executed_action <=<br>action <-<br>action_assignment.assigned_<br>action_assignment =><br>applied_action_assignme<br>identification_item = applied_action<br>identification_item <-<br>applied_identification_assignmer<br>applied_identification_assignn<br>{identification_assignme<br>identification_assignment.rc<br>identification_role<br>identification_role.name = `run i<br>identification_assignmer<br>identification_assignment.assig |
| task execution to<br>item_task<br>(as requiring_item_task) | PATH | | 6 | executed_action <=<br>action <-<br>action_assignment.assigned_<br>action_assignment =><br>applied_action_assignme |
| TASK_EXECUTION_-<br>CONTEXT_DEFINITION | executed_action | 41 | 6 | |
| task_execution_context_-<br>definition to end_item_-<br>effectivity<br>(as context_relationship) | PATH | | 6 | executed_action <=<br>action <-<br>action_assignment.assigned_<br>action_assignment =><br>applied_action_assignme<br>applied_action_assignment.iter<br>action_item<br>action_item = product_definition_<br>product_definition_effectiv |

## Table 5 - Mapping table for task_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| task_execution_context_-<br>definition to functional_-<br>definition_usage<br>(as context_task) | PATH | | 6 | executed_action <=<br>action <-<br>action_assignment.assigned_<br>{action_assignment<br>action_assignment.role -<br>action_role<br>action_role.name = `context<br>action_assignment =><br>applied_action_assignme<br>applied_action_assignment.iter<br>action_item<br>action_item = product_definitic<br>product_definition_usag |
| TASK_EXECUTION_-<br>STATUS | action_status | 41 | | |
| execution_status_value | action_status.status | 41 | | |
| status | action_status.status | 41 | | |



121

# Table 5 - Mapping table for task_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| status_date | [date.year_component] | 41 | | action_status |
| | [calendar_date.month_component] | 41 | | date_and_time_item = action_ |
| | [calendar_date.day_component] | 41 | | date_and_time_item <- |
| | [local_time.hour_component] | 41 | | applied_date_and_time_assignme |
| | [local_time.minute_component] | 41 | | applied_date_and_time_assign |
| | [local_time.second_component] | 41 | | {date_and_time_assignm |
| | | | | date_and_time_assignment.r |
| | | | | date_time_role |
| | | | | date_time_role.name = `status |
| | | | | date_and_time_assignme |
| | | | | date_and_time_assignment.assigned_d |
| | | | | [date_and_time |
| | | | | date_and_time.date_compon |
| | | | | date |
| | | | | date.year_component] |
| | | | | [date_and_time |
| | | | | date_and_time.date_compon |
| | | | | date => |
| | | | | calendar_date |
| | | | | calendar_date.month_compo |
| | | | | [date_and_time |
| | | | | date_and_time.date_compon |
| | | | | date => |
| | | | | calendar_date |
| | | | | calendar_date.day_compon |
| | | | | [date_and_time |
| | | | | date_and_time.time_compon |
| | | | | local_time |
| | | | | local_time.hour_compone |
| | | | | [date_and_time |
| | | | | date_and_time.time_compon |
| | | | | local_time |
| | | | | local_time.minute_compon |
| | | | | [date_and_time |
| | | | | date_and_time.time_compon |
| | | | | local_time |
| | | | | local_time.second_compon |

## Table 5 - Mapping table for task_definition UoF (continued)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| task_execution_status to task_execution (as executed_task) | PATH | | 6 | action_status<br>action_status.assigned_actic<br>executed_action |
| TASK_EXECUTION_-SUPPORT_RESOURCE | action_resource.usage | 41 | | |
| actual_quantity | [measure_with_unit.value_component]<br>[measure_with_unit.unit_component] | 41<br>41 | 8,10 | action_resource.usage<br>action_resource<br>characterized_resource_definition = a<br>characterized_resource_defini<br>resource_property.resour<br>resource_property <-<br>resource_property_representatior<br>resource_property_represent<br>resource_property_representation.rep<br>representation<br>representation.items[i] -<br>{representation_item<br>representation_item.name = `actua<br>representation_item =><br>measure_representation_ite<br>measure_with_unit<br>[measure_with_unit.value_con<br>[measure_with_unit.unit_com |
| task_execution_support_-resource to support_-resource (as used_resource) | PATH | | 8,10 | action_resource.usage<br>action_resource |
| task_execution_support_-resource to task_execution (as executed_task) | PATH | | 6,8,10 | action_resource.usage<br>action_resource<br>action_resource.usage[1]<br>supported_item<br>supported_item = actior<br>action =><br>executed_action |



123

# Table 5 - Mapping table for task_definition UoF (concluded)

| Application element | AIM element | Source | Rules | Reference path |
|---|---|---|---|---|
| TASK_SKILL_LEVEL | representation_relationship | 43 | | {representation_relationsl representation_relationship.name = `task cc |
| task_skill_level to skill_-level (as subject_skill_level) | PATH | | | representation_relationsh representation_relationship.re {representation representation.name = `skill l representation |
| task_skill_level to task_-condition (as condition) | PATH | | | representation_relationsh representation_relationship.re {representation representation.name = `task coı representation => item_property_representat |

The following rules are referenced in the preceding table:

1) application_context_requires_ap_definition
2) approval_is_assigned
3) authorized_approval
4) dependent_instantiable_person
5) dependent_instantiable_type_qualifier
6) executed_action_is_assigned
7) group_is_assigned
8) selected_action_resource
9) source_maintenance_assembly_component_usage
10) subtype_exclusive_action_resource
11) subtype_mandatory_characterized_object

## 5.2 AIM EXPRESS short listing

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

EXPRESS specification:

```
*)
SCHEMA life_cycle_change_management;

USE FROM action_schema                                          -- ISO 10303-41
  (action,
   action_directive,
   action_method,
   action_relationship,
   action_request_solution,
   action_resource,
   action_resource_relationship,
   action_status,
   directed_action,
   executed_action,
   versioned_action_request);

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

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

```
USE FROM configuration_management_schema                        -- ISO 10303-41
  (configuration_effectivity);

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

USE FROM document_schema                                        -- ISO 10303-41
  (document,
   document_relationship,
   document_usage_constraint);

USE FROM effectivity_schema                                     -- ISO 10303-41
  (dated_effectivity,
   lot_effectivity,
   serial_numbered_effectivity);

USE FROM geometry_schema                                        -- ISO 10303-42
  (axis1_placement,
   axis2_placement_2d,
   axis2_placement_3d,
   cartesian_point,
   geometric_representation_context,
   geometric_representation_item,
   point);

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_and_time_assignment,
   date_assignment,
   document_reference,
   group_assignment,
   identification_assignment,
   organization_assignment);

USE FROM material_property_definition_schema                    -- ISO 10303-45
  (property_definition_relationship);

USE FROM measure_schema                                         -- ISO 10303-41
  (amount_of_substance_measure,
   amount_of_substance_measure_with_unit,
   amount_of_substance_unit,
   context_dependent_measure,
   context_dependent_unit,
   conversion_based_unit,
   count_measure,
   derived_unit,
   electric_current_measure_with_unit,
   electric_current_unit,
   global_unit_assigned_context,
   length_measure,
   length_measure_with_unit,
```

```
   length_unit,
   mass_measure,
   mass_measure_with_unit,
   mass_unit,
   named_unit,
   numeric_measure,
   parameter_value,
   positive_length_measure,
   positive_ratio_measure,
   ratio_measure,
   ratio_measure_with_unit,
   ratio_unit,
   si_unit,
   time_measure,
   time_measure_with_unit,
   time_unit,
   volume_measure,
   volume_measure_with_unit,
   volume_unit);

USE FROM person_organization_schema                          -- ISO 10303-41
  (organization,
   organization_relationship,
   person);

USE FROM process_property_schema                             -- ISO 10303-49
  (action_property,
   action_property_relationship,
   resource_property,
   resource_property_relationship);

USE FROM process_property_representation_schema              -- ISO 10303-49
  (action_property_representation,
   resource_property_representation);

USE FROM product_definition_schema                           -- ISO 10303-41
  (product,
   product_category,
   product_category_relationship,
   product_definition,
   product_definition_effectivity,
   product_definition_formation,
   product_definition_formation_relationship,
   product_definition_formation_with_specified_source,
   product_definition_relationship,
   product_definition_with_associated_documents,
   product_related_product_category);

USE FROM product_property_definition_schema                  -- ISO 10303-41
  (characterized_object,
   product_definition_shape,
   property_definition);

USE FROM product_property_representation_schema              -- ISO 10303-41
  (property_definition_representation);

USE FROM product_structure_schema                            -- ISO 10303-44
  (assembly_component_usage,
   make_from_usage_option,
```

```
   next_assembly_usage_occurrence,
   product_definition_usage,
   promissory_usage_occurrence,
   quantified_assembly_component_usage,
   specified_higher_usage_occurrence);

USE FROM qualified_measure_schema                                -- ISO 10303-45
  (descriptive_representation_item,
   measure_representation_item,
   qualified_representation_item,
   type_qualifier);

USE FROM representation_schema                                   -- ISO 10303-43
  (compound_representation_item,
   representation,
   representation_context,
   representation_item,
   representation_relationship,
   set_representation_item);
(*
```

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 |
| configuration_management_schema | ISO 10303-44 |
| date_time_schema | ISO 10303-41 |
| document_schema | ISO 10303-41 |
| effectivity_schema | ISO 10303-41 |
| geometry_schema | ISO 10303-42 |
| group_schema | ISO 10303-41 |
| management_resources_schema | ISO 10303-41 |
| material_property_definition_schema | ISO 10303-45 |
| measure_schema | ISO 10303-41 |
| person_organization_schema | ISO 10303-41 |
| process_property_schema | ISO 10303-49 |
| process_property_representation_schema | ISO 10303-49 |
| 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 |

## 5.2.1 AIM EXPRESS short listing types

### 5.2.1.1 action_item

An **action_item** identifies the **action_resource**, **document**, **organization**, **product**, **product_definition**, **product_definition_effectivity**, **product_definition_formation**, **product_definition_usage**, **property_definition**, or **resource_property** that is associated with an **action**.

EXPRESS specification:

```
*)
TYPE action_item = SELECT
    (action_resource,
     document,
     organization,
     product,
     product_definition,
     product_definition_effectivity,
     product_definition_formation,
     product_definition_usage,
     property_definition,
     resource_property);
END_TYPE;
(*
```

### 5.2.1.2 action_request_item

An **action_request_item** identifies the **product_definition** that is associated with an **action_request**.

EXPRESS specification:

```
*)
TYPE action_request_item = SELECT
    (product_definition);
END_TYPE;
(*
```

### 5.2.1.3 approved_item

An **approved_item** identifies the **action_relationship**, **action_resource**, **applied_action_assignment**, **applied_organization_assignment**, or **organization_relationship** that is associated with an **approval**.

EXPRESS specification:

```
*)
TYPE approved_item = SELECT
    (action_relationship,
     action_resource,
     applied_action_assignment,
     applied_organization_assignment,
     organization_relationship);
END_TYPE;
(*
```

### 5.2.1.4 date_and_time_item

A **date_and_time_item** identifies the **action**, **action_property**, **action_resource**, **action_status**, **document**, **organization**, **product**, **property_definition**, **representation**, or **resource_property** that is associated with a **date_and_time**.

EXPRESS specification:

```
*)
TYPE date_and_time_item = SELECT
    (action,
     action_property,
     action_resource,
     action_status,
     document,
     organization,
     product,
     property_definition,
     representation,
     resource_property);
END_TYPE;
(*
```

### 5.2.1.5 dated_item

A **dated_item** identifies the **action**, **action_property**, **action_resource**, **document**, **organization**, **product**, **property_definition**, or **resource_property** that is associated with a **date**.

EXPRESS specification:

```
*)
TYPE dated_item = SELECT
    (action,
     action_property,
     action_resource,
     document,
     organization,
     product,
     property_definition,
     resource_property);
END_TYPE;
(*
```

### 5.2.1.6 document_reference_item

A **document_reference_item** identifies the **action**, **action_property**, **action_resource**, **document**, **organization**, **product**, **property_definition** or **resource_property** that is associated with a **document**.

EXPRESS specification:

```
*)
TYPE document_reference_item = SELECT
    (action,
     action_property,
     action_resource,
     document,
     organization,
     product,
     property_definition,
     resource_property);
END_TYPE;
(*
```

### 5.2.1.7 grouped_item

A **grouped_item** identifies the **action**, **action_resource**, **applied_action_assignment**, **assembly_component_-usage**, **document**, **organization**, **product**, **product_definition_formation**, or **property_definition** that is associated with a **group**.

EXPRESS specification:

```
*)
TYPE grouped_item = SELECT
    (action,
     action_resource,
     applied_action_assignment,
     assembly_component_usage,
     document,
     organization,
     product,
     product_definition_formation,
     property_definition);
END_TYPE;
(*
```

### 5.2.1.8 identification_item

An **identification_item** identifies the **action**, **action_item**, **action_property**, **action_resource**, **applied_-action_assignment**, **approval_assignment**, **document**, **next_assembly_usage_occurrence**, **organization**, **product_definition_formation**, **promissory_usage_occurrence**, **property_definition**, **resource_property**, or **specified_higher_usage_occurrence** that is associated with an **identification_assignment.**

EXPRESS specification:

```
*)
TYPE identification_item = SELECT
    (action,
     action_item,
     action_property,
     action_resource,
     applied_action_assignment,
     approval_assignment,
     document,
     next_assembly_usage_occurrence,
     organization,
```

```
     product_definition_formation,
     promissory_usage_occurrence,
     property_definition,
     resource_property,
     specified_higher_usage_occurrence);
END_TYPE;
(*
```

### 5.2.1.9 organized_item

EXPRESS specification:

An **organized_item** identifies the **action**, **action_property**, **action_resource**, **document**, **organization**, **product**, **property_definition**, or **resource_property** that is associated with an **organization**.

```
*)
TYPE organized_item = SELECT
    (action,
     action_property,
     action_resource,
     document,
     organization,
     product,
     property_definition,
     resource_property);
END_TYPE;
(*
```

## 5.2.2 AIM EXPRESS short listing entities

## 5.2.2.1 AIM EXPRESS short listing entity definitions

### 5.2.2.1.1 applied_action_assignment

An **applied_action_assignment** assigns an **action** to one or more **action_resource**, **document**, **organization**, **product**, **product_definition**, **product_definition_effectivity**, **product_definition_formation**, **product_-definition_usage**, **property_definition**, or **resource_property**s.

EXPRESS specification:

```
*)
ENTITY applied_action_assignment
  SUBTYPE OF (action_assignment);
  items : SET [1:?] OF action_item;
WHERE
  WR1: (NOT (SELF.role.name = 'change from')) OR
       (SIZEOF (USEDIN (SELF.assigned_action.chosen_method,
       'LIFE_CYCLE_CHANGE_MANAGEMENT.ACTION_REQUEST_SOLUTION.METHOD')) = 1);
  WR2: (NOT (SELF.role.name = 'change from')) OR
       (SIZEOF (QUERY (it <* SELF.items |
       NOT (SIZEOF (USEDIN (it, 'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'PROPERTY_DEFINITION.DEFINITION') +
```

```
       USEDIN (it, 'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'RESOURCE_PROPERTY.RESOURCE') +
       USEDIN (it, 'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'ACTION_PROPERTY.DEFINITION')) = 1))) = 0);
END_ENTITY;
(*
```

Attribute definitions:

**items:** the set of **action_resource**, **document**, **organization**, **product**, **product_definition**, **product_-definition_effectivity**, **product_definition_formation**, **product_definition_usage**, **property_definition**, or **resource_property**s that are assigned to an **action.**

Formal propositions:

**WR1:** If the **applied_action_assignment** has a **role** with a **name** of `change from', it shall reference an **action** with a **chosen_method** that is referenced by exactly one **action_request_solution**.

**WR2:** If the **applied_action_assignment** has a **role** with a **name** of `change from', its items shall be referenced by exactly one **property_definition**, **resource_property**, or **action_property**.

Associated global rules:

— executed_action_is_assigned (see 5.2.3.6).

### 5.2.2.1.2 applied_action_request_assignment

An **applied_action_request_assignment** assigns an **action_request** to one or more **product_definition**s.

EXPRESS specification:

```
*)
ENTITY applied_action_request_assignment
  SUBTYPE OF (action_request_assignment);
  items : SET [1:?] OF action_request_item;
END_ENTITY;
(*
```

Attribute definitions:

**items:** the set of **product_definition**s that are assigned to an **action_request**.

### 5.2.2.1.3 applied_approval_assignment

An **applied_approval_assignment** assigns an **approval** to one or more **action_relationship**, **action_resource**, **applied_action_assignment**, **applied_organization_assignment**, or **organization_relationship**s.

EXPRESS specification:

```
*)
ENTITY applied_approval_assignment
  SUBTYPE OF (approval_assignment);
  items : SET [1:?] OF approved_item;
END_ENTITY;
(*
```

Attribute definitions:

**items:** the set of **action_relationship**, **action_resource**, **applied_action_assignment**, **applied_organization_-assignment**, or **organization_relationship**s to which an **approval** is assigned.

Associated global rules:

— authorized_approval (see 5.2.3.3);

— approval_is_assigned (see 5.2.3.2).

### 5.2.2.1.4 applied_date_and_time_assignment

An **applied_date_and_time_assignment** assigns a **date_and_time** to one or more **action**, **action_property**, **action_resource**, **action_status**, **document**, **organization**, **product**, **property_definition**, **representation**, or **resource_property**s.

EXPRESS specification:

```
*)
ENTITY applied_date_and_time_assignment
  SUBTYPE OF (date_and_time_assignment);
  items : SET [1:?] OF date_and_time_item;
END_ENTITY;
(*
```

Attribute definitions:

**items:** the set of **action**, **action_property**, **action_resource**, **action_status**, **document**, **organization**, **product**, **property_definition**, **representation**, or **resource_property**s to which a **date_and_time** is assigned.

### 5.2.2.1.5 applied_date_assignment

An **applied_date_assignment** assigns a **date** to one or more **action**, **action_property**, **action_resource**, **document**, **organization**, **product**, **property_definition**, or **resource_property**s.

EXPRESS specification:

```
*)
ENTITY applied_date_assignment
  SUBTYPE OF (date_assignment);
```

```
  items : SET [1:?] OF dated_item;
END_ENTITY;
(*
```

Attribute definitions:

**items**: the set of **action**, **action_property**, **action_resource**, **document**, **organization**, **product**, **property_-definition**, or **resource_property**s to which the **date** is assigned.

### 5.2.2.1.6 applied_document_reference

An **applied_document_reference** assigns a reference to a **document** to one or more **action**, **action_property**, **action_resource**, **document**, **organization**, **product**, **property_definition** or **resource_property**s.

EXPRESS specification:

```
*)
ENTITY applied_document_reference
  SUBTYPE OF (document_reference);
  items : SET [1:?] OF document_reference_item;
END_ENTITY;
(*
```

Attribute definitions:

**items:** the set of **action**, **action_property**, **action_resource**, **document**, **organization**, **product**, **property_-definition** or **resource_property**s to which the reference to a **document** is being assigned.

### 5.2.2.1.7 applied_group_assignment

An **applied_group_assignment** assigns one or more **action**, **action_resource**, **applied_action_assignment**, **assembly_component_usage**, **document**, **organization**, **product**, **product_definition_formation**, or **property_definition**s to a **group**.

EXPRESS specification:

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

Attribute definitions:

**items:** the set of **action**, **action_resource**, **applied_action_assignment**, **assembly_component_usage**, **document**, **organization**, **product**, **product_definition_formation**, or **property_definition**s that are assigned to the **group**.

Associated global rules:

— group_is_assigned (see 5.2.3.7);

— source_maintenance_assembly_component_usage (see 5.2.3.9).

### 5.2.2.1.8 applied_identification_assignment

An **applied_identification_assignment** assigns an identifier to one or more **action**, **action_item**, **action_property**, **action_resource**, **applied_action_assignment**, **approval_assignment**, **document**, **next_assembly_usage_occurrence**, **organization**, **product_definition_formation**, **promissory_usage_occurrence**, **property_definition**, **resource_property**, or **specified_higher_usage_occurrence**s.

EXPRESS specification:

```
*)
ENTITY applied_identification_assignment
  SUBTYPE OF (identification_assignment);
  items : SET [1:?] OF identification_item;
END_ENTITY;
(*
```

Attribute definitions:

**items:** the set of **action**, **action_item**, **action_property**, **action_resource**, **applied_action_assignment**, **approval_assignment**, **document**, **next_assembly_usage_occurrence**, **organization**, **product_definition_formation**, **promissory_usage_occurrence**, **property_definition**, **resource_property**, or **specified_higher_usage_occurrence**s to which the identifier is assigned.

### 5.2.2.1.9 applied_organization_assignment

An **applied_organization_assignment** assigns an organization to one or more **action**, **action_property**, **action_resource**, **document**, **organization**, **product**, **property_definition**, or **resource_property**s.

EXPRESS specification:

```
*)
ENTITY applied_organization_assignment
  SUBTYPE OF (organization_assignment);
  items : SET [1:?] OF organized_item;
END_ENTITY;
(*
```

Attribute definitions:

**items:** the set of **action**, **action_property**, **action_resource**, **document**, **organization**, **product**, **property_definition**, or **resource_property**s to which the **organization** is assigned.

### 5.2.2.1.10 characterized_applied_action_assignment

A **characterized_applied_action_assignment** is an **applied_action_assignment** and a **characterized_object** that associates property values to the usage of an **action**.

EXPRESS specification:

```
*)
ENTITY characterized_applied_action_assignment
  SUBTYPE OF (characterized_object, action_assignment);
WHERE
  WR1: SIZEOF (QUERY (pd <* USEDIN (SELF, 'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       NOT (pdr.used_representation.name IN ['task requiring capacity level',
       'item task time', 'item task frequency']))) = 0))) >= 1;
END_ENTITY;
(*
```

Formal propositions:

**WR1:** Every **characterized_applied_action_assignment** shall be represented by at least one **representation** with a **name** of either `task requiring capacity level', `item task time', or `item task frequency'.

### 5.2.2.1.11 document_action_resource

A **document_action_resource** is a **document** and an **action_resource**. The **document_action_resource** shall have an anomaly defined for it.

```
*)
ENTITY document_action_resource
  SUBTYPE OF (document, action_resource);
WHERE
  WR1: SIZEOF (QUERY (rp <* USEDIN (SELF, 'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'RESOURCE_PROPERTY.RESOURCE') |
       rp.name = 'item anomaly')) >= 1;
END_ENTITY;
(*
```

Formal propositions:

**WR1:** Each **document_action_resource** shall have at least one item anomaly defined for it.

### 5.2.2.1.12 item_property_representation

An **item_property_definition** is a representation of a property that is associated with an item version or item task.

EXPRESS specification:

```
*)
ENTITY item_property_representation
  SUBTYPE OF (representation);
WHERE
  WR1: SELF.name IN ['item coordinate location', 'availability indicator',
       'item age', 'item repairability level',
       'task requiring capacity level', 'item task frequency',
       'item task time', 'task condition'];
  WR2: (NOT (SELF.NAME = 'item coordinate location')) OR
       (SIZEOF (QUERY (pdr <* USEDIN (SELF, 'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'PROPERTY_DEFINITION_REPRESENTATION.USED_REPRESENTATION') |
       'LIFE_CYCLE_CHANGE_MANAGEMENT.PRODUCT_DEFINITION_USAGE' IN
       TYPEOF (pdr.definition.definition))) = 1);
  WR3: (NOT (SELF.NAME = 'availability indicator')) OR
       (SIZEOF (QUERY (pdr <* USEDIN (SELF, 'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'PROPERTY_DEFINITION_REPRESENTATION.USED_REPRESENTATION') |
       'LIFE_CYCLE_CHANGE_MANAGEMENT.ASSEMBLY_COMPONENT_USAGE' IN
       TYPEOF (pdr.definition.definition))) = 1);
  WR4: (NOT (SELF.NAME = 'item age')) OR
       (SIZEOF (QUERY (pdr <* USEDIN (SELF, 'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'PROPERTY_DEFINITION_REPRESENTATION.USED_REPRESENTATION') |
       'LIFE_CYCLE_CHANGE_MANAGEMENT.PRODUCT_DEFINITION' IN
       TYPEOF (pdr.definition.definition))) = 1);
  WR5: (NOT (SELF.NAME = 'item repairability level')) OR
       (SIZEOF (QUERY (pdr <* USEDIN (SELF, 'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'PROPERTY_DEFINITION_REPRESENTATION.USED_REPRESENTATION') |
       'LIFE_CYCLE_CHANGE_MANAGEMENT.ASSEMBLY_COMPONENT_USAGE' IN
       TYPEOF (pdr.definition.definition))) = 1);
  WR6: (NOT (SELF.NAME = 'task requiring capacity level')) OR
       (SIZEOF (QUERY (pdr <* USEDIN (SELF, 'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'PROPERTY_DEFINITION_REPRESENTATION.USED_REPRESENTATION') |
       'LIFE_CYCLE_CHANGE_MANAGEMENT.CHARACTERIZED_APPLIED_ACTION_ASSIGNMENT'
       IN TYPEOF (pdr.definition.definition))) = 1);
  WR7: (NOT (SELF.NAME = 'item task frequency')) OR
       (SIZEOF (QUERY (pdr <* USEDIN (SELF, 'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'PROPERTY_DEFINITION_REPRESENTATION.USED_REPRESENTATION') |
       'LIFE_CYCLE_CHANGE_MANAGEMENT.CHARACTERIZED_APPLIED_ACTION_ASSIGNMENT'
       IN TYPEOF (pdr.definition.definition))) = 1);
  WR8: (NOT (SELF.NAME = 'item task time')) OR
       (SIZEOF (QUERY (pdr <* USEDIN (SELF, 'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'PROPERTY_DEFINITION_REPRESENTATION.USED_REPRESENTATION') |
       'LIFE_CYCLE_CHANGE_MANAGEMENT.CHARACTERIZED_APPLIED_ACTION_ASSIGNMENT'
       IN TYPEOF (pdr.definition.definition))) = 1);
  WR9: (NOT (SELF.NAME = 'task condition')) OR
       (SIZEOF (QUERY (act_apr <* QUERY (apr <* USEDIN (SELF,
       'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'ACTION_PROPERTY_REPRESENTATION.REPRESENTATION') |
       'LIFE_CYCLE_CHANGE_MANAGEMENT.ACTION'
       IN TYPEOF (apr.property.definition)) |
       SIZEOF (QUERY (aa <* USEDIN (act_apr.property.definition,
       'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'ACTION_ASSIGNMENT.ASSIGNED_ACTION') |
       'LIFE_CYCLE_CHANGE_MANAGEMENT.APPLIED_ACTION_ASSIGNMENT' IN
       TYPEOF (aa))) = 1)) = 1);
END_ENTITY;
(*
```

Formal propositions:

**WR1:** The **item_property_representation** shall have a **name** of `item coordinate location', `availability indicator', `item age', `item repairability level', `task requiring capacity level', `item task frequency', `item task time', or `task condition'.

**WR2:** If the **item_property_representation** has a **name** of `item coordinate location', it shall be the representation of exactly one **property_definition** of a **product_definition_usage**.

**WR3:** If the **item_property_representation** has a **name** of `availability indicator', it shall be the representation of exactly one **property_definition** of an **assembly_component_usage**.

**WR4:** If the **item_property_representation** has a **name** of `item age', it shall be the representation of exactly one **property_definition** of a **product_definition**.

**WR5:** If the **item_property_representation** has a **name** of `item repairability level', it shall be the representation of exactly one **property_definition** of an **assembly_component_usage**.

**WR6:** If the **item_property_representation** has a **name** of `task requiring capacity level', it shall be the representation of exactly one **property_definition** of a **characterized_applied_action_assignment**.

**WR7:** If the **item_property_representation** has a **name** of `item task frequency', it shall be the representation of exactly one **property_definition** of a **characterized_applied_action_assignment**.

**WR8:** If the **item_property_representation** has a **name** of `item task time', it shall be the representation of exactly one **property_definition** of a **characterized_applied_action_assignment**.

**WR9:** If the **item_property_representation** has a **name** of `task condition', it shall be the representation of exactly one **action_property** of an **action** that is the **assigned_action** for exactly one **action_assignment**.

### 5.2.2.1.13 organization_action_resource

An **organized_action_resource** is an **organization** and an **action_resource** that has property values associated with it.

EXPRESS specification:

```
*)
ENTITY organization_action_resource
  SUBTYPE OF (organization, action_resource);
WHERE
  WR1: SIZEOF (QUERY (pd <* USEDIN (SELF, 'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'PROPERTY_DEFINITION.DEFINITION') |
       NOT (SIZEOF (QUERY (pdr <* USEDIN (pd,
       'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') |
       NOT (pdr.used_representation.name IN ['organization capacity level',
       'item capacity level', 'task requiring capacity level']))) = 0))) >= 1;
```

```
END_ENTITY;
(*
```

Formal propositions:

**WR1:** Every **organization_action_resource** shall be represented by at least one **representation** with a name of `organization capacity level', `item capacity level', or `task requiring capacity level'.

## 5.2.2.2 AIM EXPRESS short listing imported entity modifications

### 5.2.2.2.1 action_resource

The base definition of the **action_resource** 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_resource** entity:

— selected_action_resource (see 5.2.3.8);

— subtype_exclusive_action_resource (see 5.2.3.10).

### 5.2.2.2.2 action_resource_relationship

The base definition of the **action_resource_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_resource_relationship** entity:

— selected_action_resource (see 5.2.3.8).

### 5.2.2.2.3 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 5.2.3.1).

#### 5.2.2.2.4 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 5.2.3.1).

#### 5.2.2.2.5 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 rule defined in this part of ISO 10303 applies to the **approval** entity:

— approval_is_assigned (see 5.2.3.2).

#### 5.2.2.2.6 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:

— authorized_approval (see 5.2.3.3).

#### 5.2.2.2.7 assembly_component_usage

The base definition of the **assembly_component_usage** entity is given in ISO 10303-44. The following modifications apply to this part of ISO 10303.

Associated global rules:

— source_maintenance_assembly_component_usage (see 5.2.3.9).

Informal propositions:

**IP1:** Every **assembly_component_usage** that satisfies the requirement for an assembly_or_component_usage shall have at most one **property_definition** that has at most one **representation** containing at most one **representation_item** with a **name** `conversion factor'.

### 5.2.2.2.8 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 5.2.3.11).

### 5.2.2.2.9 document

The base definition of the **document** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Informal propositions:

**IP1:** Every **document** that is a documentation shall be the **relating_document** in at least one **document_-relationship** in which the **related_document** is a data template.

### 5.2.2.2.10 executed_action

The base definition of the **executed_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 **executed_action** entity:

— executed_action_is_assigned (see 5.2.3.6).

Informal propositions:

**IP1:** An **executed_action** that is a task execution shall be referenced as the **assigned_action** by exactly one **action_status**.

### 5.2.2.2.11 group

The base definition of the **group** 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 **group** entity:

— group_is_assigned (see 5.2.3.7).

### 5.2.2.2.12 person

The base definition of the **person** 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 **person** entity:

— dependent_instantiable_person (see 5.2.3.4).

### 5.2.2.2.13 representation_item

The base definition of the **representation_item** entity is given in ISO 10303-43. The following modifications apply to this part of ISO 10303.

Informal propositions:

**IP1:** Every **representation_item** that specifies a value for facility capacity level shall be used in exactly one **representation** that represents exactly one **resource_property** of an **action_resource** with a **kind name** of `facility'.

### 5.2.2.2.14 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 5.2.3.5).

### 5.2.2.2.15 versioned_action_request

The base definition of the **versioned_action_request** entity is given in ISO 10303-41. The following modifications apply to this part of ISO 10303.

Informal propositions:

**IP1:** A **versioned_action_request** that specifies the disposition of an item anomaly shall have exactly one **applied_approval_assignment** referencing it in its set of **items** with an **approval_role name** of `support resource approval authority'.

**IP2:** A **versioned_action_request** that specifies the disposition of an item anomaly shall have exactly one **action_request_solution** specified for it that references an **action_method** that is the **chosen_method** for exactly one **executed_action** to specify the disposition task.

## 5.2.3 EXPRESS short listing rules

### 5.2.3.1 application_context_requires_ap_definition

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 IN apd.application)
       AND
       (apd.application_interpreted_model_schema_name =
       'life_cycle_change_management'))) = 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 `life_cycle_change_management' as its **application_interpreted_model_schema_name**.

### 5.2.3.2 approval_is_assigned

Every approval shall be associated with product data by at least one **applied_approval_assignment**.

EXPRESS specification:

```
*)
RULE approval_is_assigned FOR (approval, applied_approval_assignment);
WHERE
  WR1: SIZEOF (QUERY (app <* approval |
       NOT (SIZEOF (QUERY (aaa <* applied_approval_assignment |
       app :=: aaa.assigned_approval)) >= 1))) = 0;
END_RULE;
(*
```

Argument definitions:

**approval:** the set of all instances of **approval**.

**applied_approval_assignment:** the set of all instances of **applied_approval_assignment**.

Formal propositions:

**WR1:** Every instance of **approval** shall be referenced as the **assigned_approval** by at least one **applied_-approval_assignment**.

### 5.2.3.3 authorized_approval

Every **applied_approval_assignment** shall have exactly one **approval_person_organization** associated with it.

EXPRESS specification:

```
*)
RULE authorized_approval FOR (applied_approval_assignment,
     approval_person_organization);
WHERE
  WR1: SIZEOF (QUERY (aaa <* applied_approval_assignment |
       NOT (SIZEOF (QUERY (apo <* approval_person_organization |
       aaa.assigned_approval :=: apo.authorized_approval)) =1))) = 0;
END_RULE;
(*
```

Argument definitions:

**applied_approval_assignment:** the set of all instances of **applied_approval_assignment**.

**approval_person_organization:** the set of all instances of **approval_person_organization**.

Formal propositions:

**WR1:** For each **applied_approval_assignment** there shall be exactly one **approval_person_organization** associated with the **assigned_approval**.

### 5.2.3.4 dependent_instantiable_person

Every instance of **person** shall be dependent upon its usage to support the definition of another entity.

EXPRESS specification:

```
*)
RULE dependent_instantiable_person FOR (person);
WHERE
  WR1: SIZEOF (QUERY (p <* person |
       NOT (SIZEOF (USEDIN (p, '')) >=1))) = 0;
END_RULE;
(*
```

Argument definitions:

**person:** the set of all instances of **person**.

Formal propositions:

**WR1:** For each instance of **person**, there shall be a reference to the **person** instance from an attribute of another entity.

### 5.2.3.5 dependent_instantiable_type_qualifier

Every instance of **type_qualifier** shall be dependent upon its usage to support the definition of 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;
(*
```

Argument definitions:

**type_qualifier:** the set of all instances of **type_qualifier**.

Formal propositions:

**WR1:** For each instance of **type_qualifier**, there shall be a reference to the **type_qualifier** instance from an attribute of another entity.

### 5.2.3.6 executed_action_is_assigned

Every **executed_action** shall be assigned by at least one **applied_action_assignment**.

EXPRESS specification:

```
*)
RULE executed_action_is_assigned FOR (executed_action,
     applied_action_assignment);
WHERE
  WR1: SIZEOF (QUERY (ea <* executed_action |
       NOT (SIZEOF (QUERY (aga <* applied_action_assignment |
       ea :=: aga.assigned_action)) >= 1))) = 0;
END_RULE;
(*
```

Argument definitions:

**executed_action:** the set of all instances of **executed_action**.

**applied_action_assignment:** the set of all instances of **applied_action_assignment**.

Formal propositions:

**WR1:** Each **executed_action** shall be the **assigned_action** in at least one **applied_action_assignment**.

### 5.2.3.7 group_is_assigned

Every **group** shall be assigned by at least one **applied_group_assignment**.

EXPRESS specification:

```
*)
RULE group_is_assigned FOR (group, applied_group_assignment);
WHERE
  WR1: SIZEOF (QUERY (grp <* group |
       NOT (SIZEOF (QUERY (aga <* applied_group_assignment |
       grp :=: aga.assigned_group)) >= 1))) = 0;
END_RULE;
(*
```

Argument definitions:

**group:** the set of all instances of **group**.

**applied_group_assignment:** the set of all instances of **applied_group_assignment**.

Formal propositions:

**WR1:** Each **group** shall be the **assigned_group** in at least one **applied_group_assignment**.

### 5.2.3.8 selected_action_resource

An **action_resource_relationship** shall specify a single **action_resource** as a recommendation selection for a particular **action_resource**.

EXPRESS specification:

```
*)
RULE selected_action_resource FOR (action_resource,
     action_resource_relationship);
WHERE
  WR1: SIZEOF (QUERY (ar <* action_resource |
       NOT (SIZEOF (QUERY (sar <* QUERY (arr <* action_resource_relationship |
       arr.name = 'recommendation selection') |
       ar :=: sar.relating_resource)) <= 1))) = 0;
END_RULE;
(*
```

Argument definitions:

**action_resource:** the set of all instances of **action_resource**.

**action_resource_relationship:** the set of all instances of **action_resource_relationship**.

Formal propositions:

**WR1:** Every **action_resource** shall be the **relating_resource** in at most one **action_resource_relationship** with a **name** of `recommendation selection'.

### 5.2.3.9 source_maintenance_assembly_component_usage

Every assembly_component_usage shall be an item in the **items** of most one **applied_group_assignment**.

EXPRESS specification:

```
*)
RULE source_maintenance_assembly_component_usage FOR
(assembly_component_usage,
     applied_group_assignment);
WHERE
  WR1: SIZEOF (QUERY (acu <* assembly_component_usage |
       NOT (SIZEOF (QUERY (aga <* applied_group_assignment |
       SIZEOF (QUERY (it <* aga.items |
       acu :=: it)) = 1)) <= 1)))= 0;
END_RULE;
(*
```

Argument definitions:

**assembly_component_usage:** the set of all instances of **assembly_component_usage**.

**applied_group_assignment:** the set of all instances of **applied_group_assignment**.

Formal propositions:

**WR1:** Every instance of **assembly_component_usage** shall be in the set of **items** of at most one **applied_-group_assignment**.

### 5.2.3.10 subtype_exclusive_action_resource

A **action_resource** shall also be an instance of **characterized_applied_action_assignment**.

EXPRESS specification:

```
*)
RULE subtype_exclusive_action_resource FOR (action_resource);
WHERE
  WR1: SIZEOF (QUERY (ar <* action_resource |
       NOT (SIZEOF (['LIFE_CYCLE_CHANGE_MANAGEMENT.DOCUMENT_ACTION_RESOURCE',
       'LIFE_CYCLE_CHANGE_MANAGEMENT.ORGANIZATION_ACTION_RESOURCE'] *
       TYPEOF(ar)) <= 1))) = 0;
END_RULE;
(*
```

Argument definitions:

**action_resource:** the set of all instances of **action_resource**.

Formal propositions:

**WR1:** An **action_resource** may be at most one instance of **document_action_resource** or **organization_-action_resource**.

### 5.2.3.11 subtype_mandatory_characterized_object

A **characterized_object** shall also be an instance of **characterized_applied_action_assignment**.

EXPRESS specification:

```
*)
RULE subtype_mandatory_characterized_object FOR (characterized_object);
WHERE
  WR1: SIZEOF (QUERY (co <* characterized_object |
       NOT ('LIFE_CYCLE_CHANGE_MANAGEMENT.' +
       'CHARACTERIZED_APPLIED_ACTION_ASSIGNMENT' IN TYPEOF(co)))) = 0;
END_RULE;
(*
```

Argument definitions:

**characterized_object:** the set of all instances of **characterized_object**.

Formal propositions:

**WR1:** Every **characterized_object** shall be an instance of **characterized_applied_action_assignment**.

```
*)
END_SCHEMA;
(*
```

# 6 Conformance requirements

Conformance to this part of ISO 10303 includes satisfying the requirements stated in this part, the requirements of the implementation method(s) supported, and the relevant requirements of the normative references.

An implementation shall support at least one of the following implementation methods:

— ISO 10303-21.

Requirements with respect to implementation methods-specific requirements 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 a number of options that may be supported by an implementation. These options have been grouped into the following conformance classes:

— Class 1 - Life cycle product change process.

Support for a particular conformance class requires support of all the options specified in this class. This part specifies only a single conformance class; therefore, all options for the class must be supported.

Conformance to a particular class requires that all AIM elements defined as part of that class be supported. Since this part specifies only a single conformance class, all options for the class must be supported.

NOTE - ISO 10303-308:—[2] defines the abstract test suite (ATS) to be used in the assessment of conformance. ISO 10303-32:—[1] describes the conformance assessment process.

[2] To be published.

# Annex A
(normative)

# AIM EXPRESS expanded listing

The following EXPRESS is the expanded form of the short form schema given in 5.2. In the event of any discrepancy between the short form and this expanded listing, the expanded listing shall be used.

```
*)
SCHEMA life_cycle_change_management;

  TYPE action_item = SELECT
    (action_resource,
     document,
     organization,
     product,
     product_definition,
     product_definition_effectivity,
     product_definition_formation,
     product_definition_usage,
     property_definition,
     resource_property);
  END_TYPE; -- action_item

  TYPE action_request_item = SELECT
    (product_definition);
  END_TYPE; -- action_request_item

  TYPE ahead_or_behind = ENUMERATION OF
    (ahead,
     behind);
  END_TYPE; -- ahead_or_behind

  TYPE amount_of_substance_measure = REAL;
  END_TYPE; -- amount_of_substance_measure

  TYPE approved_item = SELECT
    (action_relationship,
     action_resource,
     applied_action_assignment,
     applied_organization_assignment,
     organization_relationship);
  END_TYPE; -- approved_item

  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 characterized_action_definition = SELECT
  (action,
   action_method,
   action_relationship);
END_TYPE; -- characterized_action_definition

TYPE characterized_definition = SELECT
  (characterized_object,
   characterized_product_definition,
   shape_definition);
END_TYPE; -- characterized_definition

TYPE characterized_product_definition = SELECT
  (product_definition,
   product_definition_relationship);
END_TYPE; -- characterized_product_definition

TYPE characterized_resource_definition = SELECT
  (action_resource,
   action_resource_relationship);
END_TYPE; -- characterized_resource_definition

TYPE compound_item_definition = SELECT
  (set_representation_item);
END_TYPE; -- compound_item_definition

TYPE configuration_design_item = SELECT
  (product_definition,
   product_definition_formation);
END_TYPE; -- configuration_design_item

TYPE context_dependent_measure = REAL;
END_TYPE; -- context_dependent_measure

TYPE count_measure = NUMBER;
END_TYPE; -- count_measure

TYPE date_and_time_item = SELECT
  (action,
   action_property,
   action_resource,
   action_status,
   document,
   organization,
   product,
   property_definition,
   representation,
   resource_property);
END_TYPE; -- date_and_time_item

TYPE date_time_select = SELECT
  (date,
   local_time,
   date_and_time);
END_TYPE; -- date_time_select
```

```
TYPE dated_item = SELECT
  (action,
   action_property,
   action_resource,
   document,
   organization,
   product,
   property_definition,
   resource_property);
END_TYPE; -- dated_item

TYPE day_in_month_number = INTEGER;
END_TYPE; -- day_in_month_number

TYPE dimension_count = INTEGER;
WHERE
  wr1: (SELF > 0);
END_TYPE; -- dimension_count

TYPE document_reference_item = SELECT
  (action,
   action_property,
   action_resource,
   document,
   organization,
   product,
   property_definition,
   resource_property);
END_TYPE; -- document_reference_item

TYPE expression_operand = SELECT
  (measure_with_unit);
END_TYPE; -- expression_operand

TYPE grouped_item = SELECT
  (action,
   action_resource,
   applied_action_assignment,
   assembly_component_usage,
   document,
   organization,
   product,
   product_definition_formation,
   property_definition);
END_TYPE; -- grouped_item

TYPE hour_in_day = INTEGER;
WHERE
  wr1: ((0 <= SELF) AND (SELF < 24));
END_TYPE; -- hour_in_day

TYPE identification_item = SELECT
  (action,
   action_item,
   action_property,
   action_resource,
```

```
    applied_action_assignment,
    approval_assignment,
    document,
    next_assembly_usage_occurrence,
    organization,
    product_definition_formation,
    promissory_usage_occurrence,
    property_definition,
    resource_property,
    specified_higher_usage_occurrence);
END_TYPE; -- identification_item

TYPE identifier = STRING;
END_TYPE; -- identifier

TYPE label = STRING;
END_TYPE; -- label

TYPE length_measure = REAL;
END_TYPE; -- length_measure

TYPE mass_measure = REAL;
END_TYPE; -- mass_measure

TYPE measure_value = SELECT
  (length_measure,
   mass_measure,
   time_measure,
   amount_of_substance_measure,
   volume_measure,
   ratio_measure,
   parameter_value,
   numeric_measure,
   context_dependent_measure,
   positive_length_measure,
   positive_ratio_measure,
   count_measure);
END_TYPE; -- measure_value

TYPE minute_in_hour = INTEGER;
WHERE
  wr1: ((0 <= SELF) AND (SELF <= 59));
END_TYPE; -- minute_in_hour

TYPE month_in_year_number = INTEGER;
WHERE
  wr1: ((1 <= SELF) AND (SELF <= 12));
END_TYPE; -- month_in_year_number

TYPE numeric_measure = NUMBER;
END_TYPE; -- numeric_measure

TYPE organized_item = SELECT
  (action,
   action_property,
```

```
   action_resource,
   document,
   organization,
   product,
   property_definition,
   resource_property);
END_TYPE; -- organized_item

TYPE parameter_value = REAL;
END_TYPE; -- parameter_value

TYPE person_organization_select = SELECT
  (person,
   organization);
END_TYPE; -- person_organization_select

TYPE positive_length_measure = length_measure;
WHERE
  wr1: (SELF > 0);
END_TYPE; -- positive_length_measure

TYPE positive_ratio_measure = ratio_measure;
WHERE
  wr1: (SELF > 0);
END_TYPE; -- positive_ratio_measure

TYPE property_or_shape_select = SELECT
  (property_definition,
   shape_definition);
END_TYPE; -- property_or_shape_select

TYPE ratio_measure = REAL;
END_TYPE; -- ratio_measure

TYPE represented_definition = SELECT
  (property_definition);
END_TYPE; -- represented_definition

TYPE second_in_minute = REAL;
WHERE
  wr1: ((0 <= SELF) AND (SELF <= 60));
END_TYPE; -- second_in_minute

TYPE set_representation_item = SET [1:?] OF representation_item;
END_TYPE; -- set_representation_item

TYPE shape_definition = SELECT
  (product_definition_shape);
END_TYPE; -- shape_definition

TYPE si_prefix = ENUMERATION OF
  (exa,
   peta,
   tera,
   giga,
   mega,
```

```
    kilo,
    hecto,
    deca,
    deci,
    centi,
    milli,
    micro,
    nano,
    pico,
    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 = ENUMERATION OF
    (made,
    bought,
    not_known);
  END_TYPE; -- source

  TYPE supported_item = SELECT
    (action_directive,
    action,
    action_method);
  END_TYPE; -- supported_item
```

```
TYPE text = STRING;
END_TYPE; -- text

TYPE time_measure = REAL;
END_TYPE; -- time_measure

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
  (type_qualifier);
END_TYPE; -- value_qualifier

TYPE vector_or_direction = SELECT
  (vector,
   direction);
END_TYPE; -- vector_or_direction

TYPE volume_measure = REAL;
END_TYPE; -- volume_measure

TYPE year_number = INTEGER;
END_TYPE; -- year_number

ENTITY action;
    id            : identifier;
    name          : label;
    description   : text;
    chosen_method : action_method;
END_ENTITY; -- action

ENTITY action_assignment
  ABSTRACT SUPERTYPE;
    assigned_action : action;
    role            : action_role;
END_ENTITY; -- action_assignment

ENTITY action_directive;
    name        : label;
    description : text;
    analysis    : text;
    comment     : text;
    requests    : SET [1:?] OF versioned_action_request;
END_ENTITY; -- action_directive

ENTITY action_method;
    name        : label;
    description : text;
    consequence : text;
```

```
    purpose     : text;
END_ENTITY; -- action_method

ENTITY action_property;
    name        : label;
    description : text;
    definition  : characterized_action_definition;
END_ENTITY; -- action_property

ENTITY action_property_relationship;
    name                      : label;
    description               : text;
    relating_action_property : action_property;
    related_action_property  : action_property;
  WHERE
    wr1: (relating_action_property :<>: related_action_property);
END_ENTITY; -- action_property_relationship

ENTITY action_property_representation;
    name           : label;
    description    : text;
    property       : action_property;
    representation : representation;
END_ENTITY; -- action_property_representation

ENTITY action_relationship;
    name            : label;
    description     : text;
    relating_action : action;
    related_action  : action;
END_ENTITY; -- action_relationship

ENTITY action_request_assignment
  ABSTRACT SUPERTYPE;
    assigned_action_request : versioned_action_request;
    role                    : action_request_role;
END_ENTITY; -- action_request_assignment

ENTITY action_request_role;
    name        : label;
    description : text;
END_ENTITY; -- action_request_role

ENTITY action_request_solution;
    name        : label;
    description : text;
    method      : action_method;
    request     : versioned_action_request;
END_ENTITY; -- action_request_solution

ENTITY action_resource;
    name        : label;
    description : text;
    usage       : SET [1:?] OF supported_item;
    kind        : action_resource_type;
```

```
END_ENTITY; -- action_resource

ENTITY action_resource_relationship;
    name              : label;
    description       : text;
    relating_resource : action_resource;
    related_resource  : action_resource;
END_ENTITY; -- action_resource_relationship

ENTITY action_resource_type;
    name : label;
END_ENTITY; -- action_resource_type

ENTITY action_role;
    name        : label;
    description : text;
END_ENTITY; -- action_role

ENTITY action_status;
    status          : label;
    assigned_action : executed_action;
END_ENTITY; -- action_status

ENTITY amount_of_substance_measure_with_unit
  SUBTYPE OF (measure_with_unit);
  WHERE
    wr1: ('LIFE_CYCLE_CHANGE_MANAGEMENT.AMOUNT_OF_SUBSTANCE_UNIT' IN
             TYPEOF(SELF\measure_with_unit.unit_component));
END_ENTITY; -- amount_of_substance_measure_with_unit

ENTITY amount_of_substance_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 = 1) AND (SELF\
             named_unit.dimensions.luminous_intensity_exponent = 0));
END_ENTITY; -- amount_of_substance_unit

ENTITY application_context;
    name        : label;
    description : text;
  INVERSE
    context_elements : SET [1:?] OF application_context_element FOR
                          frame_of_reference;
END_ENTITY; -- application_context

ENTITY application_context_element
  SUPERTYPE OF (ONEOF (product_context,product_definition_context,
      product_concept_context));
    name               : label;
    frame_of_reference : SET [1:?] OF 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                             : SET [1:?] OF
                                                    application_context;
  END_ENTITY; -- application_protocol_definition

  ENTITY applied_action_assignment
    SUBTYPE OF (action_assignment);
      items : SET [1:?] OF action_item;
    WHERE
      wr1: ((NOT (SELF.role.name = 'change from')) OR (SIZEOF(USEDIN(SELF.

assigned_action.chosen_method,'LIFE_CYCLE_CHANGE_MANAGEMENT.ACTION_REQUEST_SOL
UTION.METHOD'))
               = 1));
      wr2: ((NOT (SELF.role.name = 'change from')) OR (SIZEOF(
               QUERY ( it <* SELF.items | (NOT (SIZEOF(USEDIN(it,
               'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
               'PROPERTY_DEFINITION.DEFINITION') + USEDIN(it,
               'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
               'RESOURCE_PROPERTY.RESOURCE') + USEDIN(it,
               'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
               'ACTION_PROPERTY.DEFINITION')) = 1)) )) = 0));
  END_ENTITY; -- applied_action_assignment

  ENTITY applied_action_request_assignment
    SUBTYPE OF (action_request_assignment);
      items : SET [1:?] OF action_request_item;
  END_ENTITY; -- applied_action_request_assignment

  ENTITY applied_approval_assignment
    SUBTYPE OF (approval_assignment);
      items : SET [1:?] OF approved_item;
  END_ENTITY; -- applied_approval_assignment

  ENTITY applied_date_and_time_assignment
    SUBTYPE OF (date_and_time_assignment);
      items : SET [1:?] OF date_and_time_item;
  END_ENTITY; -- applied_date_and_time_assignment

  ENTITY applied_date_assignment
    SUBTYPE OF (date_assignment);
      items : SET [1:?] OF dated_item;
  END_ENTITY; -- applied_date_assignment

  ENTITY applied_document_reference
    SUBTYPE OF (document_reference);
      items : SET [1:?] OF document_reference_item;
  END_ENTITY; -- applied_document_reference

  ENTITY applied_group_assignment
```

```
  SUBTYPE OF (group_assignment);
    items : SET [1:?] OF grouped_item;
END_ENTITY; -- applied_group_assignment

ENTITY applied_identification_assignment
  SUBTYPE OF (identification_assignment);
    items : SET [1:?] OF identification_item;
END_ENTITY; -- applied_identification_assignment

ENTITY applied_organization_assignment
  SUBTYPE OF (organization_assignment);
    items : SET [1:?] OF organized_item;
END_ENTITY; -- applied_organization_assignment

ENTITY approval;
    status : approval_status;
    level  : label;
END_ENTITY; -- approval

ENTITY approval_assignment
  ABSTRACT SUPERTYPE;
    assigned_approval : approval;
    role              : approval_role;
END_ENTITY; -- approval_assignment

ENTITY approval_date_time;
    date_time      : date_time_select;
    dated_approval : approval;
    role           : approval_date_time_role;
END_ENTITY; -- approval_date_time

ENTITY approval_date_time_role;
    name        : label;
    description : text;
END_ENTITY; -- approval_date_time_role

ENTITY approval_person_organization;
    person_organization : person_organization_select;
    authorized_approval : approval;
    role                : approver_role;
END_ENTITY; -- approval_person_organization

ENTITY approval_role;
    name        : label;
    description : text;
END_ENTITY; -- approval_role

ENTITY approval_status;
    name : label;
END_ENTITY; -- approval_status

ENTITY approver_role;
    name        : label;
    description : text;
END_ENTITY; -- approver_role
```

```
ENTITY assembly_component_usage
  SUPERTYPE OF (ONEOF (next_assembly_usage_occurrence,
      specified_higher_usage_occurrence,promissory_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),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 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

ENTITY characterized_applied_action_assignment
  SUBTYPE OF (characterized_object, action_assignment);
  WHERE
```

```
    wr1: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
             'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
             'PROPERTY_DEFINITION.DEFINITION') | (NOT (SIZEOF(
             QUERY ( pdr <* USEDIN(pd,'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
             'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (NOT (pdr
             .used_representation.name IN [
             'task requiring capacity level','item task time',
             'item task frequency'])) )) = 0)) )) >= 1);
END_ENTITY; -- characterized_applied_action_assignment

ENTITY characterized_object;
    name        : label;
    description : text;
END_ENTITY; -- characterized_object

ENTITY compound_representation_item
  SUBTYPE OF (representation_item);
    item_element : compound_item_definition;
END_ENTITY; -- compound_representation_item

ENTITY configuration_design;
    name          : label;
    description   : text;
    configuration : configuration_item;
    design        : configuration_design_item;
  UNIQUE
    ur1 : configuration, design;
END_ENTITY; -- configuration_design

ENTITY configuration_effectivity
  SUBTYPE OF (product_definition_effectivity);
    configuration : configuration_design;
  UNIQUE
    ur1 : configuration, usage, id;
  WHERE
    wr1: ('LIFE_CYCLE_CHANGE_MANAGEMENT.PRODUCT_DEFINITION_USAGE' IN
             TYPEOF(SELF\product_definition_effectivity.usage));
END_ENTITY; -- configuration_effectivity

ENTITY configuration_item;
    id           : identifier;
    name         : label;
    description  : OPTIONAL text;
    item_concept : product_concept;
    purpose      : OPTIONAL label;
  UNIQUE
    ur1 : id;
END_ENTITY; -- configuration_item

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 coordinated_universal_time_offset;
    hour_offset   : hour_in_day;
    minute_offset : OPTIONAL minute_in_hour;
    sense         : ahead_or_behind;
END_ENTITY; -- coordinated_universal_time_offset

ENTITY date
  SUPERTYPE OF (calendar_date);
    year_component : year_number;
END_ENTITY; -- date

ENTITY date_and_time;
    date_component : date;
    time_component : local_time;
END_ENTITY; -- date_and_time

ENTITY date_and_time_assignment
  ABSTRACT SUPERTYPE;
    assigned_date_and_time : date_and_time;
    role                   : date_time_role;
END_ENTITY; -- date_and_time_assignment

ENTITY date_assignment
  ABSTRACT SUPERTYPE;
    assigned_date : date;
    role          : date_role;
END_ENTITY; -- date_assignment

ENTITY date_role;
    name        : label;
    description : text;
END_ENTITY; -- date_role

ENTITY date_time_role;
    name        : label;
    description : text;
END_ENTITY; -- date_time_role

ENTITY dated_effectivity
  SUBTYPE OF (effectivity);
    effectivity_start_date : date_and_time;
    effectivity_end_date   : OPTIONAL date_and_time;
END_ENTITY; -- dated_effectivity

ENTITY derived_unit;
    name     : label;
    elements : SET [1:?] OF derived_unit_element;
  WHERE
    wr1: ((SIZEOF(elements) > 1) OR ((SIZEOF(elements) = 1) AND (
              elements[1].exponent <> 1)));
END_ENTITY; -- derived_unit
```

```
ENTITY derived_unit_element;
    unit     : named_unit;
    exponent : REAL;
END_ENTITY; -- derived_unit_element

ENTITY descriptive_representation_item
  SUBTYPE OF (representation_item);
    description : text;
END_ENTITY; -- descriptive_representation_item

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 directed_action
  SUBTYPE OF (executed_action);
    directive : action_directive;
END_ENTITY; -- directed_action

ENTITY direction
  SUBTYPE OF (geometric_representation_item);
    direction_ratios : LIST [2:3] OF REAL;
  WHERE
    wr1: (SIZEOF(QUERY ( tmp <* direction_ratios | (tmp <> 0) )) > 0);
END_ENTITY; -- direction

ENTITY document;
    id          : identifier;
    name        : label;
    description : text;
    kind        : document_type;
END_ENTITY; -- document

ENTITY document_action_resource
  SUBTYPE OF (document, action_resource);
  WHERE
    wr1: (SIZEOF(QUERY ( rp <* USEDIN(SELF,
             'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
             'RESOURCE_PROPERTY.RESOURCE') | (rp.name = 'item anomaly') ))
             >= 1);
END_ENTITY; -- document_action_resource

ENTITY document_reference
  ABSTRACT SUPERTYPE;
    assigned_document : document;
    source            : label;
    role              : document_role;
END_ENTITY; -- document_reference

ENTITY document_relationship;
```

```
    name              : label;
    description       : text;
    relating_document : document;
    related_document  : document;
END_ENTITY; -- document_relationship

ENTITY document_role;
    name        : label;
    description : text;
END_ENTITY; -- document_role

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 effectivity
  SUPERTYPE OF (ONEOF (serial_numbered_effectivity,dated_effectivity,
      lot_effectivity));
    id          : identifier;
    name        : label;
    description : text;
END_ENTITY; -- effectivity

ENTITY electric_current_measure_with_unit
  SUBTYPE OF (measure_with_unit);
  WHERE
    wr1: ('LIFE_CYCLE_CHANGE_MANAGEMENT.ELECTRIC_CURRENT_UNIT' IN
            TYPEOF(SELF\measure_with_unit.unit_component));
END_ENTITY; -- electric_current_measure_with_unit

ENTITY electric_current_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 = 1) 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; -- electric_current_unit

ENTITY executed_action
  SUBTYPE OF (action);
  INVERSE
    status_of_action : SET [1:?] OF action_status FOR assigned_action;
END_ENTITY; -- executed_action
```

```
  ENTITY geometric_representation_context
    SUBTYPE OF (representation_context);
      coordinate_space_dimension : dimension_count;
  END_ENTITY; -- geometric_representation_context

  ENTITY geometric_representation_item
    SUPERTYPE OF (ONEOF (point,direction,vector,placement))
    SUBTYPE OF (representation_item);
    DERIVE
      dim : dimension_count := dimension_of(SELF);
    WHERE
      wr1: (SIZEOF(QUERY ( using_rep <* using_representations(SELF) | (
               NOT
('LIFE_CYCLE_CHANGE_MANAGEMENT.GEOMETRIC_REPRESENTATION_CONTEXT'
               IN TYPEOF(using_rep.context_of_items))) )) = 0);
  END_ENTITY; -- geometric_representation_item

  ENTITY global_unit_assigned_context
    SUBTYPE OF (representation_context);
      units : SET [1:?] OF unit;
  END_ENTITY; -- global_unit_assigned_context

  ENTITY group;
      name        : label;
      description : text;
  END_ENTITY; -- group

  ENTITY group_assignment
    ABSTRACT SUPERTYPE;
      assigned_group : group;
      role           : group_role;
  END_ENTITY; -- group_assignment

  ENTITY group_relationship;
      name           : label;
      description    : text;
      relating_group : group;
      related_group  : group;
  END_ENTITY; -- group_relationship

  ENTITY group_role;
      name        : label;
      description : text;
  END_ENTITY; -- group_role

  ENTITY identification_assignment
    ABSTRACT SUPERTYPE;
      assigned_id : identifier;
      role        : identification_role;
  END_ENTITY; -- identification_assignment

  ENTITY identification_role;
      name        : label;
      description : text;
  END_ENTITY; -- identification_role
```

```
  ENTITY item_property_representation
    SUBTYPE OF (representation);
    WHERE
      wr1: (SELF.name IN ['item coordinate location',
               'availability indicator','item age',
               'item repairability level','task requiring capacity level',
               'item task frequency','item task time','task condition']);
      wr2: ((NOT (SELF.name = 'item coordinate location')) OR (SIZEOF(
               QUERY ( pdr <* USEDIN(SELF,'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
                'PROPERTY_DEFINITION_REPRESENTATION.USED_REPRESENTATION')
                | ('LIFE_CYCLE_CHANGE_MANAGEMENT.PRODUCT_DEFINITION_USAGE'
               IN TYPEOF(pdr.definition.definition)) )) = 1));
      wr3: ((NOT (SELF.name = 'availability indicator')) OR (SIZEOF(
               QUERY ( pdr <* USEDIN(SELF,'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
                'PROPERTY_DEFINITION_REPRESENTATION.USED_REPRESENTATION')
                | ('LIFE_CYCLE_CHANGE_MANAGEMENT.ASSEMBLY_COMPONENT_USAGE'
               IN TYPEOF(pdr.definition.definition)) )) = 1));
      wr4: ((NOT (SELF.name = 'item age')) OR (SIZEOF(QUERY ( pdr <*
               USEDIN(SELF,'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
               'PROPERTY_DEFINITION_REPRESENTATION.USED_REPRESENTATION') |
               ('LIFE_CYCLE_CHANGE_MANAGEMENT.PRODUCT_DEFINITION' IN
               TYPEOF(pdr.definition.definition)) )) = 1));
      wr5: ((NOT (SELF.name = 'item repairability level')) OR (SIZEOF(
               QUERY ( pdr <* USEDIN(SELF,'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
                'PROPERTY_DEFINITION_REPRESENTATION.USED_REPRESENTATION')
                | ('LIFE_CYCLE_CHANGE_MANAGEMENT.ASSEMBLY_COMPONENT_USAGE'
               IN TYPEOF(pdr.definition.definition)) )) = 1));
      wr6: ((NOT (SELF.name = 'task requiring capacity level')) OR (
               SIZEOF(QUERY ( pdr <* USEDIN(SELF,
               'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
               'PROPERTY_DEFINITION_REPRESENTATION.USED_REPRESENTATION') |

('LIFE_CYCLE_CHANGE_MANAGEMENT.CHARACTERIZED_APPLIED_ACTION_ASSIGNMENT'
               IN TYPEOF(pdr.definition.definition)) )) = 1));
      wr7: ((NOT (SELF.name = 'item task frequency')) OR (SIZEOF(
               QUERY ( pdr <* USEDIN(SELF,'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
                'PROPERTY_DEFINITION_REPRESENTATION.USED_REPRESENTATION')
                |
('LIFE_CYCLE_CHANGE_MANAGEMENT.CHARACTERIZED_APPLIED_ACTION_ASSIGNMENT'
               IN TYPEOF(pdr.definition.definition)) )) = 1));
      wr8: ((NOT (SELF.name = 'item task time')) OR (SIZEOF(
               QUERY ( pdr <* USEDIN(SELF,'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
                'PROPERTY_DEFINITION_REPRESENTATION.USED_REPRESENTATION')
                |
('LIFE_CYCLE_CHANGE_MANAGEMENT.CHARACTERIZED_APPLIED_ACTION_ASSIGNMENT'
               IN TYPEOF(pdr.definition.definition)) )) = 1));
      wr9: ((NOT (SELF.name = 'task condition')) OR (SIZEOF(
               QUERY ( act_apr <* QUERY ( apr <* USEDIN(SELF,
               'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
               'ACTION_PROPERTY_REPRESENTATION.REPRESENTATION') | (
               'LIFE_CYCLE_CHANGE_MANAGEMENT.ACTION' IN TYPEOF(apr.property
               .definition)) ) | (SIZEOF(QUERY ( aa <* USEDIN(act_apr.
               property.definition,'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
               'ACTION_ASSIGNMENT.ASSIGNED_ACTION') | (
               'LIFE_CYCLE_CHANGE_MANAGEMENT.APPLIED_ACTION_ASSIGNMENT' IN
```

```
            TYPEOF(aa)) )) = 1) )) = 1));
END_ENTITY; -- item_property_representation

ENTITY length_measure_with_unit
  SUBTYPE OF (measure_with_unit);
  WHERE
    wr1: ('LIFE_CYCLE_CHANGE_MANAGEMENT.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 local_time;
    hour_component   : hour_in_day;
    minute_component : OPTIONAL minute_in_hour;
    second_component : OPTIONAL second_in_minute;
    zone             : coordinated_universal_time_offset;
  WHERE
    wr1: valid_time(SELF);
END_ENTITY; -- local_time

ENTITY lot_effectivity
  SUBTYPE OF (effectivity);
    effectivity_lot_id   : identifier;
    effectivity_lot_size : measure_with_unit;
END_ENTITY; -- lot_effectivity

ENTITY make_from_usage_option
  SUBTYPE OF (product_definition_usage);
    ranking           : INTEGER;
    ranking_rationale : text;
    quantity          : measure_with_unit;
  WHERE
    wr1: (ranking > 0);
END_ENTITY; -- make_from_usage_option

ENTITY mass_measure_with_unit
  SUBTYPE OF (measure_with_unit);
  WHERE
    wr1: ('LIFE_CYCLE_CHANGE_MANAGEMENT.MASS_UNIT' IN TYPEOF(SELF\
            measure_with_unit.unit_component));
END_ENTITY; -- mass_measure_with_unit

ENTITY mass_unit
  SUBTYPE OF (named_unit);
```

```
    WHERE
      wr1: ((SELF\named_unit.dimensions.length_exponent = 0) AND (SELF\
                named_unit.dimensions.mass_exponent = 1) 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; -- mass_unit

  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,mass_measure_with_unit,
        time_measure_with_unit,electric_current_measure_with_unit,
        amount_of_substance_measure_with_unit,volume_measure_with_unit,
        ratio_measure_with_unit));
      value_component : measure_value;
      unit_component  : unit;
    WHERE
      wr1: valid_units(SELF);
  END_ENTITY; -- measure_with_unit

  ENTITY named_unit
    SUPERTYPE OF (ONEOF (si_unit,conversion_based_unit,
        context_dependent_unit) ANDOR ONEOF (length_unit,mass_unit,
        time_unit,electric_current_unit,amount_of_substance_unit,
        volume_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 organization;
      id          : OPTIONAL identifier;
      name        : label;
      description : text;
  END_ENTITY; -- organization

  ENTITY organization_action_resource
    SUBTYPE OF (organization, action_resource);
    WHERE
      wr1: (SIZEOF(QUERY ( pd <* USEDIN(SELF,
                'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
                'PROPERTY_DEFINITION.DEFINITION') | (NOT (SIZEOF(
                QUERY ( pdr <* USEDIN(pd,'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
                'PROPERTY_DEFINITION_REPRESENTATION.DEFINITION') | (NOT (pdr
                .used_representation.name IN ['organization capacity level',
                'item capacity level','task requiring capacity level'])) ))
                = 0)) )) >= 1);
```

```
END_ENTITY; -- organization_action_resource

ENTITY organization_assignment
  ABSTRACT SUPERTYPE;
    assigned_organization : organization;
    role                  : organization_role;
END_ENTITY; -- organization_assignment

ENTITY organization_relationship;
    name                  : label;
    description           : text;
    relating_organization : organization;
    related_organization  : organization;
END_ENTITY; -- organization_relationship

ENTITY organization_role;
    name        : label;
    description : text;
END_ENTITY; -- organization_role

ENTITY person;
    id            : identifier;
    last_name     : OPTIONAL label;
    first_name    : OPTIONAL label;
    middle_names  : OPTIONAL LIST [1:?] OF label;
    prefix_titles : OPTIONAL LIST [1:?] OF label;
    suffix_titles : OPTIONAL LIST [1:?] OF label;
  WHERE
    wr1: (EXISTS(last_name) OR EXISTS(first_name));
END_ENTITY; -- person

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 point
  SUPERTYPE OF (cartesian_point)
  SUBTYPE OF (geometric_representation_item);
END_ENTITY; -- point

ENTITY product;
    id                 : identifier;
    name               : label;
    description        : text;
    frame_of_reference : SET [1:?] OF product_context;
END_ENTITY; -- product

ENTITY product_category;
    name        : label;
    description : OPTIONAL text;
END_ENTITY; -- product_category

ENTITY product_category_relationship;
```

```
    name         : label;
    description  : text;
    category     : product_category;
    sub_category : product_category;
  WHERE
    wr1: acyclic_product_category_relationship(SELF,[SELF.sub_category]);
END_ENTITY; -- product_category_relationship

ENTITY product_concept;
    id              : identifier;
    name            : label;
    description     : text;
    market_context  : product_concept_context;
  UNIQUE
    ur1 : id;
END_ENTITY; -- product_concept

ENTITY product_concept_context
  SUBTYPE OF (application_context_element);
    market_segment_type : label;
END_ENTITY; -- product_concept_context

ENTITY product_context
  SUBTYPE OF (application_context_element);
    discipline_type : label;
END_ENTITY; -- product_context

ENTITY product_definition;
    id                  : identifier;
    name                : label;
    description         : text;
    formation           : product_definition_formation;
    frame_of_reference  : product_definition_context;
END_ENTITY; -- product_definition

ENTITY product_definition_context
  SUBTYPE OF (application_context_element);
    life_cycle_stage : label;
END_ENTITY; -- product_definition_context

ENTITY product_definition_effectivity
  SUBTYPE OF (effectivity);
    usage : product_definition_relationship;
  UNIQUE
    ur1 : usage, id;
END_ENTITY; -- product_definition_effectivity

ENTITY product_definition_formation;
    id          : identifier;
    description : text;
    of_product  : product;
  UNIQUE
    ur1 : id, of_product;
END_ENTITY; -- product_definition_formation
```

```
ENTITY product_definition_formation_relationship;
    id                                    : identifier;
    name                                  : label;
    description                           : text;
    relating_product_definition_formation : product_definition_formation;
    related_product_definition_formation  : product_definition_formation;
END_ENTITY; -- product_definition_formation_relationship

ENTITY product_definition_formation_with_specified_source
  SUBTYPE OF (product_definition_formation);
    make_or_buy : source;
END_ENTITY; -- product_definition_formation_with_specified_source

ENTITY product_definition_relationship;
    id                          : identifier;
    name                        : label;
    description                 : 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: (NOT ('LIFE_CYCLE_CHANGE_MANAGEMENT.SHAPE_DEFINITION' IN
             TYPEOF(SELF\property_definition.definition)));
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],
             'LIFE_CYCLE_CHANGE_MANAGEMENT.PRODUCT_DEFINITION_USAGE.' +
             'RELATED_PRODUCT_DEFINITION');
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 product_related_product_category
  SUBTYPE OF (product_category);
    products : SET [1:?] OF product;
END_ENTITY; -- product_related_product_category

ENTITY promissory_usage_occurrence
  SUBTYPE OF (assembly_component_usage);
END_ENTITY; -- promissory_usage_occurrence
```

```
ENTITY property_definition;
    name        : label;
    description : 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;
    name                : label;
    description         : text;
    definition          : represented_definition;
    used_representation : representation;
END_ENTITY; -- property_definition_representation

ENTITY qualified_representation_item
  SUBTYPE OF (representation_item);
    qualifiers : SET [1:?] OF value_qualifier;
  WHERE
    wr1: (SIZEOF(QUERY ( temp <* qualifiers | (
             'LIFE_CYCLE_CHANGE_MANAGEMENT.PRECISION_QUALIFIER' IN
             TYPEOF(temp)) )) < 2);
END_ENTITY; -- qualified_representation_item

ENTITY quantified_assembly_component_usage
  SUBTYPE OF (assembly_component_usage);
    quantity : measure_with_unit;
END_ENTITY; -- quantified_assembly_component_usage

ENTITY ratio_measure_with_unit
  SUBTYPE OF (measure_with_unit);
  WHERE
    wr1: ('LIFE_CYCLE_CHANGE_MANAGEMENT.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 representation;
```

```
    id               : identifier;
    name             : label;
    description      : text;
    items            : SET [1:?] OF representation_item;
    context_of_items : representation_context;
END_ENTITY; -- representation

ENTITY representation_context;
    context_identifier : identifier;
    context_type       : text;
  INVERSE
    representations_in_context : SET [1:?] OF representation FOR
                                     context_of_items;
END_ENTITY; -- representation_context

ENTITY representation_item;
    name : label;
  WHERE
    wr1: (SIZEOF(using_representations(SELF)) > 0);
END_ENTITY; -- representation_item

ENTITY representation_relationship;
    name        : label;
    description : text;
    rep_1       : representation;
    rep_2       : representation;
END_ENTITY; -- representation_relationship

ENTITY resource_property;
    name        : label;
    description : text;
    resource    : characterized_resource_definition;
END_ENTITY; -- resource_property

ENTITY resource_property_relationship;
    name                        : label;
    description                 : text;
    relating_resource_property : resource_property;
    related_resource_property  : resource_property;
  WHERE
    wr1: (relating_resource_property :<>: related_resource_property);
END_ENTITY; -- resource_property_relationship

ENTITY resource_property_representation;
    name           : label;
    description    : text;
    property       : resource_property;
    representation : representation;
END_ENTITY; -- resource_property_representation

ENTITY serial_numbered_effectivity
  SUBTYPE OF (effectivity);
    effectivity_start_id : identifier;
    effectivity_end_id   : OPTIONAL identifier;
END_ENTITY; -- serial_numbered_effectivity
```

```
ENTITY si_unit
  SUBTYPE OF (named_unit);
    prefix : OPTIONAL si_prefix;
    name   : si_unit_name;
  DERIVE
    SELF\named_unit.dimensions : dimensional_exponents :=
                                     dimensions_for_si_unit(SELF.name);
END_ENTITY; -- si_unit

ENTITY specified_higher_usage_occurrence
  SUBTYPE OF (assembly_component_usage);
    upper_usage : assembly_component_usage;
    next_usage  : next_assembly_usage_occurrence;
  UNIQUE
    ur1 : upper_usage, next_usage;
  WHERE
    wr1: (SELF :<>: upper_usage);
    wr2: (SELF\product_definition_relationship.
             relating_product_definition :=: upper_usage.
             relating_product_definition);
    wr3: (SELF\product_definition_relationship.
             related_product_definition :=: next_usage.
             related_product_definition);
    wr4: (upper_usage.related_product_definition :=: next_usage.
             relating_product_definition);
    wr5: (NOT (
             'LIFE_CYCLE_CHANGE_MANAGEMENT.PROMISSORY_USAGE_OCCURRENCE'
             IN TYPEOF(upper_usage)));
END_ENTITY; -- specified_higher_usage_occurrence

ENTITY time_measure_with_unit
  SUBTYPE OF (measure_with_unit);
  WHERE
    wr1: ('LIFE_CYCLE_CHANGE_MANAGEMENT.TIME_UNIT' IN TYPEOF(SELF\
            measure_with_unit.unit_component));
END_ENTITY; -- time_measure_with_unit

ENTITY time_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 = 1) 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; -- time_unit

ENTITY type_qualifier;
    name : label;
END_ENTITY; -- type_qualifier

ENTITY vector
```

```
  SUBTYPE OF (geometric_representation_item);
    orientation : direction;
    magnitude   : length_measure;
  WHERE
    wr1: (magnitude >= 0);
END_ENTITY; -- vector

ENTITY versioned_action_request;
    id          : identifier;
    version     : label;
    purpose     : text;
    description : text;
END_ENTITY; -- versioned_action_request

ENTITY volume_measure_with_unit
  SUBTYPE OF (measure_with_unit);
  WHERE
    wr1: ('LIFE_CYCLE_CHANGE_MANAGEMENT.VOLUME_UNIT' IN TYPEOF(SELF\
              measure_with_unit.unit_component));
END_ENTITY; -- volume_measure_with_unit

ENTITY volume_unit
  SUBTYPE OF (named_unit);
  WHERE
    wr1: ((SELF\named_unit.dimensions.length_exponent = 3) 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; -- volume_unit

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 IN apd.
            application) AND (apd.
            application_interpreted_model_schema_name =
            'life_cycle_change_management')) )) = 1)) )) = 0);

END_RULE; -- application_context_requires_ap_definition

RULE approval_is_assigned FOR (approval, applied_approval_assignment);

WHERE
  wr1: (SIZEOF(QUERY ( app <* approval | (NOT (SIZEOF(QUERY ( aaa <*
            applied_approval_assignment | (app :=: aaa.assigned_approval) ))
            >= 1)) )) = 0);

END_RULE; -- approval_is_assigned

RULE authorized_approval FOR (applied_approval_assignment,
```

```
            approval_person_organization);

  WHERE
    wr1: (SIZEOF(QUERY ( aaa <* applied_approval_assignment | (NOT (
            SIZEOF(QUERY ( apo <* approval_person_organization | (aaa.
            assigned_approval :=: apo.authorized_approval) )) = 1)) )) = 0);

  END_RULE; -- authorized_approval

  RULE compatible_dimension FOR (cartesian_point, direction,
            representation_context, geometric_representation_context);

  WHERE
    wr1: (SIZEOF(QUERY ( x <* cartesian_point | (SIZEOF(QUERY ( y <*
            geometric_representation_context | (item_in_context(x,y) AND (
            HIINDEX(x.coordinates) <> y.coordinate_space_dimension)) )) >
            0) )) = 0);
    wr2: (SIZEOF(QUERY ( x <* direction | (SIZEOF(QUERY ( y <*
            geometric_representation_context | (item_in_context(x,y) AND (
            HIINDEX(x.direction_ratios) <> y.coordinate_space_dimension)) ))
            > 0) )) = 0);

  END_RULE; -- compatible_dimension

  RULE dependent_instantiable_person FOR (person);

  WHERE
    wr1: (SIZEOF(QUERY ( p <* person | (NOT (SIZEOF(USEDIN(p,'')) >= 1)) ))
            = 0);

  END_RULE; -- dependent_instantiable_person

  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 executed_action_is_assigned FOR (executed_action,
            applied_action_assignment);

  WHERE
    wr1: (SIZEOF(QUERY ( ea <* executed_action | (NOT (SIZEOF(
            QUERY ( aga <* applied_action_assignment | (ea :=: aga.
            assigned_action) )) >= 1)) )) = 0);

  END_RULE; -- executed_action_is_assigned

  RULE group_is_assigned FOR (group, applied_group_assignment);

  WHERE
    wr1: (SIZEOF(QUERY ( grp <* group | (NOT (SIZEOF(QUERY ( aga <*
            applied_group_assignment | (grp :=: aga.assigned_group) )) >=
```

```
          1)) )) = 0);

END_RULE; -- group_is_assigned

RULE selected_action_resource FOR (action_resource,
          action_resource_relationship);

WHERE
  wr1: (SIZEOF(QUERY ( ar <* action_resource | (NOT (SIZEOF(
          QUERY ( sar <* QUERY ( arr <* action_resource_relationship | (
          arr.name = 'recommendation selection') ) | (ar :=: sar.
          relating_resource) )) <= 1)) )) = 0);

END_RULE; -- selected_action_resource

RULE source_maintenance_assembly_component_usage FOR (
          assembly_component_usage, applied_group_assignment);

WHERE
  wr1: (SIZEOF(QUERY ( acu <* assembly_component_usage | (NOT (SIZEOF(
          QUERY ( aga <* applied_group_assignment | (SIZEOF(
          QUERY ( it <* aga.items | (acu :=: it) )) = 1) )) <= 1)) )) =
          0);

END_RULE; -- source_maintenance_assembly_component_usage

RULE subtype_exclusive_action_resource FOR (action_resource);

WHERE
  wr1: (SIZEOF(QUERY ( ar <* action_resource | (NOT (SIZEOF([
          'LIFE_CYCLE_CHANGE_MANAGEMENT.DOCUMENT_ACTION_RESOURCE',
          'LIFE_CYCLE_CHANGE_MANAGEMENT.ORGANIZATION_ACTION_RESOURCE'] *
           TYPEOF(ar)) <= 1)) )) = 0);

END_RULE; -- subtype_exclusive_action_resource

RULE subtype_mandatory_characterized_object FOR (characterized_object);

WHERE
  wr1: (SIZEOF(QUERY ( co <* characterized_object | (NOT ((
          'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
          'CHARACTERIZED_APPLIED_ACTION_ASSIGNMENT') IN TYPEOF(co))) ))
          = 0);

END_RULE; -- subtype_mandatory_characterized_object

FUNCTION acyclic_product_category_relationship(
            relation: product_category_relationship;
            children: SET OF product_category
    ): LOGICAL;

  LOCAL
    i              : INTEGER;
    x              : SET OF product_category_relationship;
    local_children : SET OF product_category;
  END_LOCAL;
```

```
    REPEAT i := 1 TO HIINDEX(children) BY 1;
      IF relation.category :=: children[i] THEN
        RETURN(FALSE);
      END_IF;
    END_REPEAT;
    x := bag_to_set(USEDIN(relation.category,
        'LIFE_CYCLE_CHANGE_MANAGEMENT.' +
        'PRODUCT_CATEGORY_RELATIONSHIP.SUB_CATEGORY'));
    local_children := children + relation.category;
    IF SIZEOF(x) > 0 THEN
      REPEAT i := 1 TO HIINDEX(x) BY 1;
        IF NOT acyclic_product_category_relationship(x[i],local_children)
             THEN
          RETURN(FALSE);
        END_IF;
      END_REPEAT;
    END_IF;
    RETURN(TRUE);

  END_FUNCTION; -- acyclic_product_category_relationship

  FUNCTION acyclic_product_definition_relationship(
               relation: product_definition_relationship;
               relatives: SET OF product_definition;
               specific_relation: STRING
      ): LOGICAL;

    LOCAL
      i               : INTEGER;
      x               : SET OF product_definition_relationship;
      local_relatives : SET OF product_definition;
    END_LOCAL;
    REPEAT i := 1 TO HIINDEX(relatives) BY 1;
      IF relation.relating_product_definition :=: relatives[i] THEN
        RETURN(FALSE);
      END_IF;
    END_REPEAT;
    x := bag_to_set(USEDIN(relation.relating_product_definition,
        specific_relation));
    local_relatives := relatives + relation.relating_product_definition;
    IF SIZEOF(x) > 0 THEN
      REPEAT i := 1 TO HIINDEX(x) BY 1;
        IF NOT acyclic_product_definition_relationship(x[i],
            local_relatives,specific_relation) THEN
          RETURN(FALSE);
        END_IF;
      END_REPEAT;
    END_IF;
    RETURN(TRUE);

  END_FUNCTION; -- acyclic_product_definition_relationship

  FUNCTION bag_to_set(
               the_bag: BAG OF GENERIC:intype
      ): SET OF GENERIC:intype;
```

```
  LOCAL
    i       : INTEGER;
    the_set : SET OF GENERIC:intype := [];
  END_LOCAL;
  IF SIZEOF(the_bag) > 0 THEN
    REPEAT i := 1 TO HIINDEX(the_bag) BY 1;
      the_set := the_set + the_bag[i];
    END_REPEAT;
  END_IF;
  RETURN(the_set);

END_FUNCTION; -- bag_to_set

FUNCTION build_2axes(
             ref_direction: direction
    ): LIST [2:2] OF direction;

  LOCAL
    u : LIST [2:2] OF direction;
  END_LOCAL;
  u[1] := NVL(normalise(ref_direction),direction([1,0]));
  u[2] := orthogonal_complement(u[1]);
  RETURN(u);

END_FUNCTION; -- build_2axes

FUNCTION build_axes(
             axis, ref_direction: direction
    ): LIST [3:3] OF direction;

  LOCAL
    u : LIST [3:3] OF direction;
  END_LOCAL;
  u[3] := NVL(normalise(axis),direction([0,0,1]));
  u[1] := first_proj_axis(u[3],ref_direction);
  u[2] := normalise(cross_product(u[3],u[1])).orientation;
  RETURN(u);

END_FUNCTION; -- build_axes

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.direction_ratios[1] := (v1[2] * v2[3]) - (v1[3] * v2[2]);
      res.direction_ratios[2] := (v1[3] * v2[1]) - (v1[1] * v2[3]);
      res.direction_ratios[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.orientation := res;
        result.magnitude := SQRT(mag);
      ELSE
        result.orientation := arg1;
        result.magnitude := 0;
      END_IF;
      RETURN(result);
    END;
  END_IF;

END_FUNCTION; -- cross_product

FUNCTION derive_dimensional_exponents(
             x: unit
    ): dimensional_exponents;

  LOCAL
    i      : INTEGER;
    result : dimensional_exponents := dimensional_exponents(0,0,0,0,0,0,
               0);
  END_LOCAL;
  IF 'LIFE_CYCLE_CHANGE_MANAGEMENT.DERIVED_UNIT' IN TYPEOF(x) THEN
    REPEAT i := LOINDEX(x.elements) TO HIINDEX(x.elements) BY 1;
      result.length_exponent := result.length_exponent + (x.elements[i].
          exponent * x.elements[i].unit.dimensions.length_exponent);
      result.mass_exponent := result.mass_exponent + (x.elements[i].
          exponent * x.elements[i].unit.dimensions.mass_exponent);
      result.time_exponent := result.time_exponent + (x.elements[i].
          exponent * x.elements[i].unit.dimensions.time_exponent);
      result.electric_current_exponent := result.
          electric_current_exponent + (x.elements[i].exponent * x.
          elements[i].unit.dimensions.electric_current_exponent);
      result.thermodynamic_temperature_exponent := result.
          thermodynamic_temperature_exponent + (x.elements[i].exponent *
          x.elements[i].unit.dimensions.
          thermodynamic_temperature_exponent);
      result.amount_of_substance_exponent := result.
          amount_of_substance_exponent + (x.elements[i].exponent * x.
          elements[i].unit.dimensions.amount_of_substance_exponent);
      result.luminous_intensity_exponent := result.
          luminous_intensity_exponent + (x.elements[i].exponent * x.
          elements[i].unit.dimensions.luminous_intensity_exponent);
    END_REPEAT;
  ELSE
    result := x.dimensions;
```

```
  END_IF;
  RETURN(result);

END_FUNCTION; -- derive_dimensional_exponents

FUNCTION dimension_of(
             item: geometric_representation_item
    ): dimension_count;

  LOCAL
    x : SET OF representation;
    y : representation_context;
  END_LOCAL;
  x := using_representations(item);
  y := x[1].context_of_items;
  RETURN(y\geometric_representation_context.coordinate_space_dimension);

END_FUNCTION; -- dimension_of

FUNCTION dimensions_for_si_unit(
             n: si_unit_name
    ): dimensional_exponents;
  CASE n OF
    metre         :         RETURN(dimensional_exponents(1,0,0,0,0,0,0));
    gram          :         RETURN(dimensional_exponents(0,1,0,0,0,0,0));
    second        :         RETURN(dimensional_exponents(0,0,1,0,0,0,0));
    ampere        :         RETURN(dimensional_exponents(0,0,0,1,0,0,0));
    kelvin        :         RETURN(dimensional_exponents(0,0,0,0,1,0,0));
    mole          :         RETURN(dimensional_exponents(0,0,0,0,0,1,0));
    candela       :         RETURN(dimensional_exponents(0,0,0,0,0,0,1));
    radian        :         RETURN(dimensional_exponents(0,0,0,0,0,0,0));
    steradian     :         RETURN(dimensional_exponents(0,0,0,0,0,0,0));
    hertz         :         RETURN(dimensional_exponents(0,0,-1,0,0,0,0));
    newton        :         RETURN(dimensional_exponents(1,1,-2,0,0,0,0));
    pascal        :         RETURN(dimensional_exponents(-1,1,-2,0,0,0,0));
    joule         :         RETURN(dimensional_exponents(2,1,-2,0,0,0,0));
    watt          :         RETURN(dimensional_exponents(2,1,-3,0,0,0,0));
    coulomb       :         RETURN(dimensional_exponents(0,0,1,1,0,0,0));
    volt          :         RETURN(dimensional_exponents(2,1,-3,-1,0,0,0));
    farad         :         RETURN(dimensional_exponents(-2,-1,4,1,0,0,0));
    ohm           :         RETURN(dimensional_exponents(2,1,-3,-2,0,0,0));
    siemens       :         RETURN(dimensional_exponents(-2,-1,3,2,0,0,0));
    weber         :         RETURN(dimensional_exponents(2,1,-2,-1,0,0,0));
    tesla         :         RETURN(dimensional_exponents(0,1,-2,-1,0,0,0));
    henry         :         RETURN(dimensional_exponents(2,1,-2,-2,0,0,0));
    degree_celsius  :         RETURN(dimensional_exponents(0,0,0,0,1,0,0));
    lumen         :         RETURN(dimensional_exponents(0,0,0,0,0,0,1));
    lux           :         RETURN(dimensional_exponents(-2,0,0,0,0,0,1));
    becquerel     :         RETURN(dimensional_exponents(0,0,-1,0,0,0,0));
    gray          :         RETURN(dimensional_exponents(2,0,-2,0,0,0,0));
    sievert       :         RETURN(dimensional_exponents(2,0,-2,0,0,0,0));
    END_CASE;

END_FUNCTION; -- dimensions_for_si_unit

FUNCTION 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 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)) OR (NOT EXISTS(arg)) OR (arg.dim <> 3) THEN
    x_axis := ?;
  ELSE
    z_axis := normalise(z_axis);
    IF NOT EXISTS(arg) THEN
      IF z_axis <> direction([1,0,0]) THEN
        v := direction([1,0,0]);
      ELSE
        v := direction([0,1,0]);
      END_IF;
    ELSE
      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_axis);
    x_axis := vector_difference(v,x_vec).orientation;
    x_axis := normalise(x_axis);
  END_IF;
  RETURN(x_axis);

END_FUNCTION; -- first_proj_axis

FUNCTION item_in_context(
             item: representation_item;
             cntxt: representation_context
    ): BOOLEAN;

  LOCAL
    i : INTEGER;
    y : BAG OF representation_item;
  END_LOCAL;
  IF SIZEOF(USEDIN(item,
      'LIFE_CYCLE_CHANGE_MANAGEMENT.REPRESENTATION.ITEMS') * cntxt.
      representations_in_context) > 0 THEN
    RETURN(TRUE);
  ELSE
    y := QUERY ( z <* USEDIN(item,'') | (
        'LIFE_CYCLE_CHANGE_MANAGEMENT.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 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 'LIFE_CYCLE_CHANGE_MANAGEMENT.VECTOR' IN TYPEOF(arg) THEN
        BEGIN
          vec := arg;
          v := arg.orientation;
          IF arg.magnitude = 0 THEN
            RETURN(?);
          ELSE
            vec.magnitude := 1;
          END_IF;
        END;
      ELSE
        v := arg;
      END_IF;
      mag := 0;
      REPEAT i := 1 TO ndim BY 1;
        mag := mag + (v.direction_ratios[i] * v.direction_ratios[i]);
      END_REPEAT;
      IF mag > 0 THEN
        mag := SQRT(mag);
        REPEAT i := 1 TO ndim BY 1;
          v.direction_ratios[i] := v.direction_ratios[i] / mag;
        END_REPEAT;
        IF 'LIFE_CYCLE_CHANGE_MANAGEMENT.VECTOR' IN TYPEOF(arg) THEN
          vec.orientation := v;
          result := vec;
        ELSE
          result := v;
        END_IF;
      ELSE
        RETURN(?);
      END_IF;
    END_IF;
    RETURN(result);

  END_FUNCTION; -- normalise

  FUNCTION orthogonal_complement(
               vec: direction
      ): direction;

    LOCAL
      result : direction;
    END_LOCAL;
    IF (vec.dim <> 2) OR (NOT EXISTS(vec)) THEN
      RETURN(?);
```

```
  ELSE
    result.direction_ratios[1] := -vec.direction_ratios[2];
    result.direction_ratios[2] := vec.direction_ratios[1];
    RETURN(result);
  END_IF;

END_FUNCTION; -- orthogonal_complement

FUNCTION 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
    result := ?;
  ELSE
    IF 'LIFE_CYCLE_CHANGE_MANAGEMENT.VECTOR' IN TYPEOF(vec) THEN
      v := vec.orientation;
      mag := scalar * vec.magnitude;
    ELSE
      v := vec;
      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.orientation := normalise(v);
    result.magnitude := mag;
  END_IF;
  RETURN(result);

END_FUNCTION; -- scalar_times_vector

FUNCTION using_representations(
             item: representation_item
    ): SET OF representation;

  LOCAL
    results            : SET OF representation;
    i                  : INTEGER;
    intermediate_items : SET OF representation_item;
    result_bag         : BAG OF representation;
  END_LOCAL;
  results := [];
  result_bag := USEDIN(item,
      'LIFE_CYCLE_CHANGE_MANAGEMENT.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 := QUERY ( z <* bag_to_set(USEDIN(item,'')) | (
        'LIFE_CYCLE_CHANGE_MANAGEMENT.REPRESENTATION_ITEM' IN TYPEOF(z)) );
    IF SIZEOF(intermediate_items) > 0 THEN
      REPEAT i := 1 TO HIINDEX(intermediate_items) BY 1;
        results := results + using_representations(intermediate_items[i]);
      END_REPEAT;
    END_IF;
    RETURN(results);

  END_FUNCTION; -- using_representations

  FUNCTION valid_calendar_date(
                date: calendar_date
      ): LOGICAL;
    IF NOT ((1 <= date.day_component) AND (date.day_component <= 31))
         THEN
      RETURN(FALSE);
    END_IF;
    CASE date.month_component OF
      4        :         RETURN((1 <= date.day_component) AND (date.
            day_component <= 30));
      6        :         RETURN((1 <= date.day_component) AND (date.
            day_component <= 30));
      9        :         RETURN((1 <= date.day_component) AND (date.
            day_component <= 30));
      11       :         RETURN((1 <= date.day_component) AND (date.
            day_component <= 30));
      2        :         BEGIN
          IF leap_year(date.year_component) THEN
            RETURN((1 <= date.day_component) AND (date.day_component <= 29));
          ELSE
            RETURN((1 <= date.day_component) AND (date.day_component <= 28));
          END_IF;
        END;
      OTHERWISE  :         RETURN(TRUE);
      END_CASE;

  END_FUNCTION; -- valid_calendar_date

  FUNCTION valid_time(
                time: local_time
      ): BOOLEAN;
    IF EXISTS(time.second_component) THEN
      RETURN(EXISTS(time.minute_component));
    ELSE
      RETURN(TRUE);
    END_IF;

  END_FUNCTION; -- valid_time

  FUNCTION valid_units(
                m: measure_with_unit
```

```
  ): BOOLEAN;
IF 'LIFE_CYCLE_CHANGE_MANAGEMENT.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 'LIFE_CYCLE_CHANGE_MANAGEMENT.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 'LIFE_CYCLE_CHANGE_MANAGEMENT.TIME_MEASURE' IN TYPEOF(m.
    value_component) THEN
  IF derive_dimensional_exponents(m.unit_component) <>
      dimensional_exponents(0,0,1,0,0,0,0) THEN
    RETURN(FALSE);
  END_IF;
END_IF;
IF 'LIFE_CYCLE_CHANGE_MANAGEMENT.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 'LIFE_CYCLE_CHANGE_MANAGEMENT.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 'LIFE_CYCLE_CHANGE_MANAGEMENT.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 'LIFE_CYCLE_CHANGE_MANAGEMENT.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 'LIFE_CYCLE_CHANGE_MANAGEMENT.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 'LIFE_CYCLE_CHANGE_MANAGEMENT.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 'LIFE_CYCLE_CHANGE_MANAGEMENT.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 'LIFE_CYCLE_CHANGE_MANAGEMENT.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 'LIFE_CYCLE_CHANGE_MANAGEMENT.RATIO_MEASURE' IN TYPEOF(m.
        value_component) THEN
      IF derive_dimensional_exponents(m.unit_component) <>
          dimensional_exponents(0,0,0,0,0,0,0) THEN
        RETURN(FALSE);
      END_IF;
    END_IF;
    IF 'LIFE_CYCLE_CHANGE_MANAGEMENT.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 'LIFE_CYCLE_CHANGE_MANAGEMENT.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
      result := ?;
    ELSE
      BEGIN
        IF 'LIFE_CYCLE_CHANGE_MANAGEMENT.VECTOR' IN TYPEOF(arg1) THEN
          mag1 := arg1.magnitude;
          vec1 := arg1.orientation;
        ELSE
          mag1 := 1;
          vec1 := arg1;
        END_IF;
        IF 'LIFE_CYCLE_CHANGE_MANAGEMENT.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;
        REPEAT i := 1 TO ndim BY 1;
          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.magnitude := SQRT(mag);
          result.orientation := res;
        ELSE
          result.magnitude := 0;
          result.orientation := vec1;
        END_IF;
      END;
    END_IF;
    RETURN(result);

  END_FUNCTION; -- vector_difference

END_SCHEMA; -- life_cycle_change_management
(*
```

# Annex B
(normative)

## AIM short names of entities

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 - Short names of entities

| Entity names | Short names |
|---|---|
| ACTION | ACTION |
| ACTION_ASSIGNMENT | ACTASS |
| ACTION_DIRECTIVE | ACTDRC |
| ACTION_METHOD | ACTMTH |
| ACTION_PROPERTY | ACTPRP |
| ACTION_PROPERTY_RELATIONSHIP | ACPRRL |
| ACTION_PROPERTY_REPRESENTATION | ACPRRP |
| ACTION_RELATIONSHIP | ACTRLT |
| ACTION_REQUEST_ASSIGNMENT | ACRQAS |
| ACTION_REQUEST_ROLE | ACRQRL |
| ACTION_REQUEST_SOLUTION | ACRQSL |
| ACTION_RESOURCE | ACTRSR |
| ACTION_RESOURCE_RELATIONSHIP | ACRSRL |
| ACTION_RESOURCE_TYPE | ACRSTY |
| ACTION_ROLE | ACTRL |
| ACTION_STATUS | ACTSTT |
| AMOUNT_OF_SUBSTANCE_MEASURE_WITH_UNIT | AOSMWU |
| AMOUNT_OF_SUBSTANCE_UNIT | AOSU |
| APPLICATION_CONTEXT | APPCNT |

# Table B.1 - Short names of entities (continued)

| Entity names | Short names |
|---|---|
| APPLICATION_CONTEXT_ELEMENT | APCNEL |
| APPLICATION_PROTOCOL_DEFINITION | APPRDF |
| APPLIED_ACTION_ASSIGNMENT | APACAS |
| APPLIED_ACTION_REQUEST_ASSIGNMENT | AARA |
| APPLIED_APPROVAL_ASSIGNMENT | APAPAS |
| APPLIED_DATE_AND_TIME_ASSIGNMENT | ADATA |
| APPLIED_DATE_ASSIGNMENT | APDTAS |
| APPLIED_DOCUMENT_REFERENCE | APDCRF |
| APPLIED_GROUP_ASSIGNMENT | APGRAS |
| APPLIED_IDENTIFICATION_ASSIGNMENT | APIDAS |
| APPLIED_ORGANIZATION_ASSIGNMENT | APORAS |
| APPROVAL | APPRVL |
| APPROVAL_ASSIGNMENT | APPASS |
| APPROVAL_DATE_TIME | APDTTM |
| APPROVAL_DATE_TIME_ROLE | ADTR |
| APPROVAL_PERSON_ORGANIZATION | APPROR |
| APPROVAL_ROLE | APPRL |
| APPROVAL_STATUS | APPSTT |
| APPROVER_ROLE | APP0 |
| ASSEMBLY_COMPONENT_USAGE | ASCMUS |
| AXIS1_PLACEMENT | AX1PLC |
| AXIS2_PLACEMENT_2D | A2PL2D |
| AXIS2_PLACEMENT_3D | A2PL3D |
| CALENDAR_DATE | CLNDT |

## Table B.1 - Short names of entities (continued)

| Entity names | Short names |
|---|---|
| CARTESIAN_POINT | CRTPNT |
| CHARACTERIZED_APPLIED_ACTION_ASSIGNMENT | CAAA |
| CHARACTERIZED_OBJECT | CHROBJ |
| COMPOUND_REPRESENTATION_ITEM | CMRPIT |
| CONFIGURATION_DESIGN | CNFDSG |
| CONFIGURATION_EFFECTIVITY | CNFEFF |
| CONFIGURATION_ITEM | CNFITM |
| CONTEXT_DEPENDENT_UNIT | CNDPUN |
| CONVERSION_BASED_UNIT | CNBSUN |
| COORDINATED_UNIVERSAL_TIME_OFFSET | CUTO |
| DATE | DATE |
| DATE_AND_TIME | DTANTM |
| DATE_AND_TIME_ASSIGNMENT | DATA |
| DATE_ASSIGNMENT | DTASS |
| DATE_ROLE | DTRL |
| DATE_TIME_ROLE | DTTMRL |
| DATED_EFFECTIVITY | DTDEFF |
| DERIVED_UNIT | DRVUNT |
| DERIVED_UNIT_ELEMENT | DRUNEL |
| DESCRIPTIVE_REPRESENTATION_ITEM | DSRPIT |
| DIMENSIONAL_EXPONENTS | DMNEXP |
| DIRECTED_ACTION | DRCACT |
| DIRECTION | DRCTN |
| DOCUMENT | DCMNT |

# Table B.1 - Short names of entities (continued)

| Entity names | Short names |
|---|---|
| DOCUMENT_ACTION_RESOURCE | DCACRS |
| DOCUMENT_REFERENCE | DCMRFR |
| DOCUMENT_RELATIONSHIP | DCMRLT |
| DOCUMENT_ROLE | DCMRL |
| DOCUMENT_TYPE | DCMTYP |
| DOCUMENT_USAGE_CONSTRAINT | DCUSCN |
| EFFECTIVITY | EFFCTV |
| ELECTRIC_CURRENT_MEASURE_WITH_UNIT | ECMWU |
| ELECTRIC_CURRENT_UNIT | ELCRUN |
| EXECUTED_ACTION | EXCACT |
| GEOMETRIC_REPRESENTATION_CONTEXT | GMRPCN |
| GEOMETRIC_REPRESENTATION_ITEM | GMRPIT |
| GLOBAL_UNIT_ASSIGNED_CONTEXT | GUAC |
| GROUP | GROUP |
| GROUP_ASSIGNMENT | GRPASS |
| GROUP_RELATIONSHIP | GRPRLT |
| GROUP_ROLE | GRPRL |
| IDENTIFICATION_ASSIGNMENT | IDNASS |
| IDENTIFICATION_ROLE | IDNRL |
| ITEM_PROPERTY_REPRESENTATION | ITPRRP |
| LENGTH_MEASURE_WITH_UNIT | LMWU |
| LENGTH_UNIT | LNGUNT |
| LOCAL_TIME | LCLTM |
| LOT_EFFECTIVITY | LTEFF |

## Table B.1 - Short names of entities (continued)

| Entity names | Short names |
|---|---|
| MAKE_FROM_USAGE_OPTION | MFUO |
| MASS_MEASURE_WITH_UNIT | MMWU |
| MASS_UNIT | MSSUNT |
| MEASURE_REPRESENTATION_ITEM | MSRPIT |
| MEASURE_WITH_UNIT | MSWTUN |
| NAMED_UNIT | NMDUNT |
| NEXT_ASSEMBLY_USAGE_OCCURRENCE | NAUO |
| ORGANIZATION | ORGNZT |
| ORGANIZATION_ACTION_RESOURCE | ORACRS |
| ORGANIZATION_ASSIGNMENT | ORGASS |
| ORGANIZATION_RELATIONSHIP | ORGRLT |
| ORGANIZATION_ROLE | ORGRL |
| PERSON | PERSON |
| PLACEMENT | PLCMNT |
| POINT | POINT |
| PRODUCT | PRDCT |
| PRODUCT_CATEGORY | PRDCTG |
| PRODUCT_CATEGORY_RELATIONSHIP | PRCTRL |
| PRODUCT_CONCEPT | PRDCNC |
| PRODUCT_CONCEPT_CONTEXT | PRCNCN |
| PRODUCT_CONTEXT | PRDCNT |
| PRODUCT_DEFINITION | PRDDFN |
| PRODUCT_DEFINITION_CONTEXT | PRDFCN |
| PRODUCT_DEFINITION_EFFECTIVITY | PRDFEF |

# Table B.1 - Short names of entities (continued)

| Entity names | Short names |
|---|---|
| PRODUCT_DEFINITION_FORMATION | PRDFFR |
| PRODUCT_DEFINITION_FORMATION_RELATIONSHIP | PDFR |
| PRODUCT_DEFINITION_FORMATION_WITH_SPECIFIED_SOURCE | PDFWSS |
| PRODUCT_DEFINITION_RELATIONSHIP | PRDFRL |
| PRODUCT_DEFINITION_SHAPE | PRDFSH |
| PRODUCT_DEFINITION_USAGE | PRDFUS |
| PRODUCT_DEFINITION_WITH_ASSOCIATED_DOCUMENTS | PDWAD |
| PRODUCT_RELATED_PRODUCT_CATEGORY | PRPC |
| PROMISSORY_USAGE_OCCURRENCE | PRUSOC |
| PROPERTY_DEFINITION | PRPDFN |
| PROPERTY_DEFINITION_RELATIONSHIP | PRDFR |
| PROPERTY_DEFINITION_REPRESENTATION | PRDFRP |
| QUALIFIED_REPRESENTATION_ITEM | QLRPIT |
| QUANTIFIED_ASSEMBLY_COMPONENT_USAGE | QACU |
| RATIO_MEASURE_WITH_UNIT | RMWU |
| RATIO_UNIT | RTUNT |
| REPRESENTATION | RPRSNT |
| REPRESENTATION_CONTEXT | RPRCNT |
| REPRESENTATION_ITEM | RPRITM |
| REPRESENTATION_RELATIONSHIP | RPRRLT |
| RESOURCE_PROPERTY | RSRPRP |
| RESOURCE_PROPERTY_RELATIONSHIP | RSPRRL |
| RESOURCE_PROPERTY_REPRESENTATION | RSPRRP |

## Table B.1 - Short names of entities (concluded)

| Entity names | Short names |
|---|---|
| SERIAL_NUMBERED_EFFECTIVITY | SRNMEF |
| SI_UNIT | SUNT |
| SPECIFIED_HIGHER_USAGE_OCCURRENCE | SHUO |
| TIME_MEASURE_WITH_UNIT | TMWU |
| TIME_UNIT | TMUNT |
| TYPE_QUALIFIER | TYPQLF |
| VECTOR | VECTOR |
| VERSIONED_ACTION_REQUEST | VRACRQ |
| VOLUME_MEASURE_WITH_UNIT | VMWU |
| VOLUME_UNIT | VLMUNT |

# Annex C
(normative)

## Implementation method-specific requirements

The implementation method defines what types of exchange behaviour are required with respect to this part of ISO 10303. Conformance to this part of ISO 10303 shall be realized in an exchange structure. The file format shall be encoded according to the syntax and EXPRESS language mapping defined in ISO 10303-21 and the AIM defined in annex A of thispart of ISO 10303. The header of the exchange structure shall identify the use of this part of ISO 10303 by the schema name `life_cycle_product_change_process'.

# Annex D
(normative)

# Protocol Implementation Conformance Statement proforma

This clause lists the optional elements of this part of ISO 10303. An implementation may chose to support any combination of this 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 implementor and may be used in preparation for conformance testing by a testing laboratory. The completed PICS proforma is referred to as a PICS.

One conformance class is identified in this part of ISO 10303. A conforming implementation shall support at least one conformance class. Each class specifies a subset of the AIM constructs in this part of ISO 10303. These classes are detailed in clause 6 of this part of ISO 10303.

Questions:

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:_____

3. Please indicate the classes for which conformance is claimed:

    — Class 1:_____

# Annex E
(normative)

# Information object registration

## E.7 Document identification

To provide for unambiguous identification of an information object in an open system, the object identifier

{ iso standard 10303 part(208) version(-1) }

is assigned to this part of ISO 10303. The meaning of this value is defined in ISO/IEC 8824-1, and is described in ISO 10303-1.

## E.8 Schema identification

To provide for unambiguous identfication of the schema specifications given in this application protocol plant_-spatial_configuration in an open information system, object identifiers are assigned as follows:

{ iso standard 10303 part(208) version(-1) object(1) life-cycle-product-change-process-schema(1) }

is assigned to the **life_cycle_product_change_process** expanded schema (see annex A).

{ iso standard 10303 part(208) version(-1) object(1) life-cycle-product-change-process-schema(2) }

is assigned to the **life_cycle_product_change_process** short form schema (see 5.2).

The meaning of these values is defined in ISO/IEC 8824-1, and is described in ISO 10303-1.

# Annex F
(informative)

# Application activity model

The application activity model (AAM) is provided as an aid to 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 application activity model is given in figures F.2 through F.6. Activities and data flows that are out of scope are marked with an asterisk.

## F.9 Application activity model definitions and abbreviations

The following terms are used in the application activity model. Terms marked with an asterisk are outside the scope of this application protocol.

The definitions given in this annex do not supersede the definitions given in the main body of the text.

The viewpoint of the AAM is the individual(s) managing the product change process.

**F.9.1 administer and perform planning (A2)*:** Determines the conjunction with engineering and manufacturing the schedules for change criteria and for implementing any corrective actions to prevent the reoccurrence of the item anomaly(s). Both discrepant item and enhancement items require planning to determine priority and resources for implementation of the change.

**F.9.2 analyze item requiring change (A321):** This activity analyzes the item anomaly(s) associated with the product to be changed and looks at the product design, fabrication and support requirements, as well as previous change history data to determine the reason and scope of the change.

**F.9.3 approve change procedure (A323):** Involves the approval and sign-off of the entire change proposal. If problems with the change or corrective procedures are uncovered, the procedures are sent back to the responsible parties requesting that appropriate changes be made.

**F.9.4 approved change procedure:** A change procedure that has had a favorable review and subsequently achieved a status of approved. An approved process or set of tasks that will be used to make changes to one or more baseline products.

**F.9.5 approved corrective action:** A corrective action that has had a favorable review and subsequently achieved a status of approved. An approved set of strategies to be taken to prevent the reason for a change from recurring.

**F.9.6 approved change request:** A change request that has had a favorable review and subsequently achieved a status of approved.

**F.9.7 authorization:** The act of conferring authority, permission, approval to proceed for a given action or set of actions.

**F.9.8 baseline product:** The original, unchanged description of the item requiring change. Used to serve as a base for measurement or comparison of a given product item.

**F.9.9 baseline product documentation subset:** The set of information about the item requiring change which is used to support the development of the change solution.

**F.9.10 change reason and scope:** The results of analysis about the item anomaly(s) associated with the product to be changed that defines the cause and direction of the change need.

**F.9.11 change request:** A statement seeking a change in an item to correct an anomaly and prevent future recurrences of the anomaly.

**F.9.12 change schedule:** A definitive production plan, or procedure, allotting work to be performed within specified time frames, ensuring the completion of the desired change.

**F.9.13 change solution:** The method or process of solving a problem in exchanging for or replacing by another, usually of the same kind or category

**F.9.14 changed product:** A baseline product that has undergone a configuration controlled transformation or modification as the result of a structured series of events.

**F.9.15 collect relevant product documentation (A31):** This activity involves gathering and reviewing the pertinent product documentation to determine the cause or reason of the issue, concern, or non conformance.

**F.9.16 conceptualized change procedure:** A set of strategies or actions to be taken to make changes to one or more baseline products.

**F.9.17 determine skill level and number of personnel (A41):** This process determines who is needed to perform the approved change and corrective action procedures.

**F.9.18 develop change procedure (A322):** With the reason and scope of the change known, a detail set of actions or steps are defined which define the procedure for changing the product and, if required, any corrective actions.

**F.9.19 develop solution (A32):** This activity involves reviewing historical records, the product design, fabrication and support requirements, the explanation for the cause of the non conformance, and the consideration of strategies to be taken to prevent the reason for a change from recurring (i.e., corrective action). This knowledge is used to develop an acceptable change procedure and if required devise a set of actions to be taken to prevent the need for the change from recurring. A corrective action is typically only required for product changes resulting from discrepancies as opposed to enhancement needs and is applied to all physical instances of the item requiring change.

**F.9.20 disapproved change request:** A change request that has had an unfavorable review and subsequently not achieved a status of approved.

**F.9.21 failed product with inspection report:** An item that did not pass an inspection operation and the report documenting the results of the inspection operation.

**F.9.22 failure and Change History:** A narrative of chronological events for an item requiring change that describes the entire non conformance and change history of the item.

**F.9.23 finalize change and product release (A44):** This process involves the signing off of all discrepancy list and maintenance documents prior to product release.

**F.9.24 identify and analyze change criteria (A3):** This activity involves receiving the approved change request and forwarding it to the appropriate technical personnel. Detail analysis is then performed on determining the changes that need to be made as well as determining what needs to be done to prevent reoccurrence of an item requiring change.

**F.9.25 identify products to be changed (A1):** Analyzes item anomalies to determine if the item requiring change meets specifications and contract requirements. The need for the change must be verified, the change request formalized, and finally approved or disapproved within this activity. Both discrepant item and enhancement items require program management analysis to determine criticality of the needed change.

**F.9.26 implement change (A4):** This activity involves using the necessary support resources in supporting the maintenance and/or repair procedures identified in the approved change and corrective action procedures.

**F.9.27 implementation schedule*:** A production plan allotting work to be done and specifying deadlines to provide a definitive plan or procedure to ensure the fulfillment thereof.

**F.9.28 organizational experience*:** One who has detailed knowledge in the structure and workings of a given enterprise.

**F.9.29 operational experience*:** One who has a comprehensive understanding in the use and techniques for operating a given product item.

**F.9.30 perform change inspection (A43):** This activity involves the inspection process of ensuring that the item requiring change complies with all the approved change and corrective action as well as existing design, fabrication and support requirements.

**F.9.31 perform change operation (A42):** This process involves the actual performance of all tasks necessary to perform the approved change and corrective action procedures pertaining to the item requiring change.

**F.9.32 perform product change management (A0):** Anomalies are in the form of product flaws or issues and concerns. The identified anomalies apply to one or more configured items requiring change. These configured items are further classified as either discrepant items or enhancement items.

First the change need is verified, then the requirements for the change are defined, and the activities of planning, performance, and verification are performed. In addition, any corrective actions for preventing the reoccurrence of the need for the change to the configured item are defined and performed.

**F.9.33 personnel requirement:** The specification of a need for one or more personnel of a given skill capability to be employed by or active in a given task or set of tasks.

**F.9.34 planning experience*:** One who has the responsibility for defining a detailed scheme, program, or method, worked out in advance to accomplish a specified plan; a plan of attack.

**F.9.35 product and associated process documentation:** The supporting reference documents or records pertaining to and describing a series of products, actions, or functions that bring about an end or result that will integrate actions, or products produced by humans, mechanical efforts, or by natural processes.

**F.9.36 product maintenance documentation:** The documents or references supplied for the work of keeping a product in proper condition.

**F.9.37 product specification*:** The supporting documents or records that describe the purpose, scope/bounds, and function of a product item.

**F.9.38 product with passing inspection report:** An item that passed an inspection operation and the report documenting the results of the inspection operation.

**F.9.39 products to be changed:** A description of anything produced by human or mechanical effort or by a natural process that will undergo an alteration or transformation or replacement by another product item.

**F.9.40 proposed corrective action:** A set of unevaluated strategies or actions to be taken to prevent the reason for an item change from recurring.

**F.9.41 technical product knowledge*:** One who has familiarity, awareness, or understanding gained through experience or study in a systematic procedure by which a complex or scientific task is accomplished with anything produced by human or mechanical effort or by a natural process.

**F.9.42 unapproved change procedure:** A change procedure that has not had a favorable review and has not achieved a status of approved.

**F.9.43 unapproved corrective action:** A corrective action that has not had a favorable review and has not achieved a status of approved.

**F.9.44 unapproved change proposal:** A description of an activity that will allow the processing of an end item without an authorization or not required to be in the approval cycle.

**F.9.45 validate change action (A33):** This activity validates that the actions applied to the item requiring change are acceptable and all issues, problems, or concerns raised about the product are resolved. If any unresolved issues,

problems, or concerns existed, the appropriate responsible personnel are notified and instructed to develop resolutions.

**F.9.46 validation failure report:** A report indicating that the actions applied to an item requiring change are not acceptable for resolving all issues, problems, or concerns raised about the product are resolved.

**F.9.47 validation report:** A report that validates that the actions applied to the item requiring change are acceptable and all issues, problems, or concerns raised about the product are resolved. A set of documentation specifying that the product has undergone an evaluation which determined that the product is within a given characteristic envelope.

## F.2 Application activity model diagrams

The application activity model diagrams are given in figures F.2 through F.6. The graphical form of the application activity model is presented in the IDEF0 activity modelling format. Activities and data flows that are out of scope are marked with asterisks.

Figure F.1 describes the basic notation used in IDEF0 modelling. Each activity may be decomposed to provide more detail. If an activity has been decomposed, a separate diagram is included.

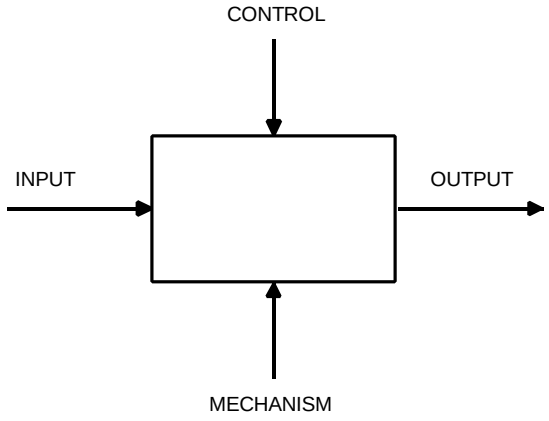


**Figure F.1 - IDEF0 basic notation**

As with any IDEF0 model, the AAM is dependent on a particular viewpoint and purpose. The purpose of the AAM is to describe the exchange of process plant spatial configuration information and design, fabrication, and maintenance information for process plant piping systems.

Contains Data for

Postscript Only.

**Figure F.2 - A-0: Perform product change management (context)**

Contains Data for

Postscript Only.

**Figure F.3 - A0: Perform product change management**

Contains Data for

Postscript Only.

**Figure F.4 - A3: Identify and analyze change criteria**

Contains Data for

Postscript Only.

**Figure F.5 - A32: Develop solution**

Contains Data for

Postscript Only.

**Figure F.6 - A4: Implement change**

# 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.12. The application reference model is a graphical representation of the structure and constraints of the application objects specified in clause 4. The graphical form of the application reference model is presented in EXPRESS-G. The application reference model is independent from any implementation method.

Contains Data for

Postscript Only.

**Figure G.1 - ARM diagram 1 of 12**

Contains Data for

Postscript Only.

**Figure G.2 - ARM diagram 2 of 12**

Contains Data for

Postscript Only.

**Figure G.3 - ARM diagram 3 of 12**

Contains Data for

Postscript Only.

**Figure G.4 - ARM diagram 4 of 12**

Contains Data for

Postscript Only.

**Figure G.5 - ARM diagram 5 of 12**

Contains Data for

Postscript Only.

**Figure G.6 - ARM diagram 6 of 12**

Contains Data for

Postscript Only.

**Figure G.7 - ARM diagram 7 of 12**

Contains Data for

Postscript Only.

**Figure G.8 - ARM diagram 8 of 12**

Contains Data for

Postscript Only.

**Figure G.9 - ARM diagram 9 of 12**

Contains Data for

Postscript Only.

**Figure G.10 - ARM diagram 10 of 12**

Contains Data for

Postscript Only.

**Figure G.11 - ARM diagram 11 of 12**

Contains Data for

Postscript Only.

**Figure G.12 - ARM diagram 12 of 12**

# Annex H
(informative)

# AIM EXPRESS-G

Figures H.1 through H.32 correspond tothe AIM EXPRESS expanded listing given in annex A. The figures use the EXPRESS-G graphical notation for the EXPRESS language. EXPRESS-G is defined in annex A of ISO 10303-11.

Contains Data for

Postscript Only.

**Figure H.1 - AIM EXPRESS-G diagram 1 of 28**

Contains Data for

Postscript Only.

**Figure H.2 - AIM EXPRESS-G diagram 2 of 28**

Contains Data for

Postscript Only.

**Figure H.3 - AIM EXPRESS-G diagram 3 of 28**

Contains Data for

Postscript Only.

**Figure H.4 - AIM EXPRESS-G diagram 4 of 28**

Contains Data for

Postscript Only.

**Figure H.5 - AIM EXPRESS-G diagram 5 of 28**

Contains Data for

Postscript Only.

**Figure H.6 - AIM EXPRESS-G diagram 6 of 28**

Contains Data for

Postscript Only.

**Figure H.7 - AIM EXPRESS-G diagram 7 of 28**

Contains Data for

Postscript Only.

**Figure H.8 - AIM EXPRESS-G diagram 8 of 28**

Contains Data for

Postscript Only.

**Figure H.9 - AIM EXPRESS-G diagram 9 of 28**

Contains Data for

Postscript Only.

**Figure H.10 - AIM EXPRESS-G diagram 10 of 28**

Contains Data for

Postscript Only.

**Figure H.11 - AIM EXPRESS-G diagram 11 of 28**

Contains Data for

Postscript Only.

**Figure H.12 - AIM EXPRESS-G diagram 12 of 28**

Contains Data for

Postscript Only.

**Figure H.13 - AIM EXPRESS-G diagram 13 of 28**

Contains Data for

Postscript Only.

**Figure H.14 - AIM EXPRESS-G diagram 14 of 28**

Contains Data for

Postscript Only.

**Figure H.15 - AIM EXPRESS-G diagram 15 of 28**

Contains Data for

Postscript Only.

**Figure H.16 - AIM EXPRESS-G diagram 16 of 28**

Contains Data for

Postscript Only.

**Figure H.17 - AIM EXPRESS-G diagram 17 of 28**

Contains Data for

Postscript Only.

**Figure H.18 - AIM EXPRESS-G diagram 18 of 28**

Contains Data for

Postscript Only.

**Figure 19 - AIM EXPRESS-G diagram 19 of 28**

Contains Data for

Postscript Only.

**Figure H.20 - AIM EXPRESS-G diagram 20 of 28**

Contains Data for

Postscript Only.

**Figure H.21 - AIM EXPRESS-G diagram 21 of 28**

Contains Data for

Postscript Only.

**Figure H.22 - AIM EXPRESS-G diagram 22 of 28**

Contains Data for

Postscript Only.

**Figure H.23 - AIM EXPRESS-G diagram 23 of 28**

Contains Data for

Postscript Only.

**Figure H.24 - AIM EXPRESS-G diagram 24 of 28**

Contains Data for

Postscript Only.

**Figure H.25 - AIM EXPRESS-G diagram 25 of 28**

Contains Data for

Postscript Only.

**Figure - AIM EXPRESS-G diagram 26 of 28**

Contains Data for

Postscript Only.

**Figure H.27 - AIM EXPRESS-G diagram 27 of 28**

Contains Data for

Postscript Only.

**Figure H.28 - AIM EXPRESS-G diagram 28 of 28**

# Annex J
(informative)

# AIM EXPRESS listing

This annex provides a listing of the table of short names and a listing of the EXPRESS specified in the AIM of this part of ISO 10303. No text or annotation is included.This annex is provided only in computer-interpretable form.

NOTE - The information provided on this diskette is informative; the normative text is that contained in the body of this part of ISO 10303.

# Annex K
(informative)

# Bibliography

[1] *IDEF0 Federal Information Processing Standards Publication 183, Integration Defintion for Functional Modeling (IDEF0)*, FIPS PUB 183, National Institute of Standards and Technology, December, 1993.

# Index

AAM . . . . . 5
Abstract test suite . . . . . 4
Action
AIM diagrams . . . . . 238
AIM EXPRESS listing entities . . . . . 158
Action_assignment
AIM diagrams . . . . . 243
AIM EXPRESS listing entities . . . . . 158
Action_directive
AIM diagrams . . . . . 244
AIM EXPRESS listing entities . . . . . 158
Action_item
AIM diagrams . . . . . 243
AIM EXPRESS listing types . . . . . 152
mapping table . . . . . 58, 69, 72, 107, 120, 121
Action_method
AIM diagrams . . . . . 238
AIM EXPRESS listing entities . . . . . 158
Action_property
AIM diagrams . . . . . 238
AIM EXPRESS listing entities . . . . . 158
Action_property_relationship
AIM EXPRESS listing entities . . . . . 158
Action_property_representation
AIM diagrams . . . . . 238
AIM EXPRESS listing entities . . . . . 159
Action_relationship
AIM diagrams . . . . . 238
AIM EXPRESS listing entities . . . . . 159
Action_request_assignment
AIM diagrams . . . . . 244
AIM EXPRESS listing entities . . . . . 159
Action_request_item
AIM diagrams . . . . . 244
AIM EXPRESS listing types . . . . . 152
mapping table . . . . . 66
Action_request_role
AIM diagrams . . . . . 244
AIM EXPRESS listing entities . . . . . 159
Action_request_solution
AIM diagrams . . . . . 244
AIM EXPRESS listing entities . . . . . 159
Action_resource

AIM diagrams . . . . . 237
AIM EXPRESS listing entities . . . . . 159
Action_resource_relationship
AIM diagrams . . . . . 237
AIM EXPRESS listing entities . . . . . 159
Action_resource_type
AIM diagrams . . . . . 237
AIM EXPRESS listing entities . . . . . 159
Action_role
AIM diagrams . . . . . 243
AIM EXPRESS listing entities . . . . . 159
Action_status
AIM diagrams . . . . . 238
AIM EXPRESS listing entities . . . . . 159
Acyclic_product_category_relationship
AIM EXPRESS listing functions . . . . . 179
Acyclic_product_definition_relationship
AIM EXPRESS listing functions . . . . . 179
Ahead_or_behind
AIM diagrams . . . . . 249
AIM EXPRESS listing types . . . . . 152
AIC . . . . . 5
AIM . . . . . 5
Amount_of_substance_measure
AIM diagrams . . . . . 248
AIM EXPRESS listing types . . . . . 152
Amount_of_substance_measure_with_unit
AIM diagrams . . . . . 251
AIM EXPRESS listing entities . . . . . 160
Amount_of_substance_unit
AIM diagrams . . . . . 252
AIM EXPRESS listing entities . . . . . 160
AP . . . . . 5
Application . . . . . 3
Application activity model . . . . . 3
Application interpreted model . . . . . 3
Application protocol . . . . . 3
Application reference model . . . . . 3
Application_context
AIM diagrams . . . . . 226
AIM EXPRESS listing entities . . . . . 160
Application_context_element
AIM diagrams . . . . . 226
AIM EXPRESS listing entities . . . . . 160
Application_context_requires_ap_definition
AIM EXPRESS listing rules . . . . . 177

AIM EXPRESS short listing rules . . . . . 144
Application_protocol_definition
AIM diagrams . . . . . 226
AIM EXPRESS listing entities . . . . . 160
Applied_action_assignment
AIM diagrams . . . . . 243
AIM EXPRESS listing entities . . . . . 160
AIM EXPRESS short listing entities . . . . . 132
mapping table . . . . . 54, 55, 58, 67-69, 71-73, 75, 101, 107, 109, 110, 112, 116, 119-121
Applied_action_request_assignment
AIM diagrams . . . . . 244
AIM EXPRESS listing entities . . . . . 161
AIM EXPRESS short listing entities . . . . . 133
mapping table . . . . . 66
Applied_approval_assignment
AIM diagrams . . . . . 232
AIM EXPRESS listing entities . . . . . 161
AIM EXPRESS short listing entities . . . . . 133
mapping table . . . . . 67, 75, 76, 79, 105, 106, 109
Applied_date_and_time_assignment
AIM diagrams . . . . . 249
AIM EXPRESS listing entities . . . . . 161
AIM EXPRESS short listing entities . . . . . 134
mapping table . . . . . 82, 96, 122
Applied_date_assignment
AIM diagrams . . . . . 249
AIM EXPRESS listing entities . . . . . 161
AIM EXPRESS short listing entities . . . . . 134
mapping table . . . . . 81
Applied_document_reference
AIM diagrams . . . . . 230
AIM EXPRESS listing entities . . . . . 161
mapping table . . . . . 79, 86, 87, 89
Applied_group_assignment
AIM diagrams . . . . . 228
AIM EXPRESS listing entities . . . . . 161
mapping table . . . . . 54, 55, 57, 70, 71, 83, 84, 91-93, 103, 107
Applied_identification_assignment
AIM diagrams . . . . . 229
AIM EXPRESS listing entities . . . . . 161
AIM EXPRESS-G diagram . . . . . 229
mapping table . . . . . 74, 75, 85, 107, 112, 113, 120
Applied_organization_assignment
AIM diagrams . . . . . 231
AIM EXPRESS listing entities . . . . . 161
AIM EXPRESS-G diagram . . . . . 231

mapping table . . . . . 75, 79, 80, 89
Approval
AIM diagrams . . . . . 232
AIM EXPRESS listing entities . . . . . 161
Approval_assignment
AIM diagrams . . . . . 232
AIM EXPRESS listing entities . . . . . 161
Approval_date_time
AIM diagrams . . . . . 232
AIM EXPRESS listing entities . . . . . 161
Approval_date_time_role
AIM diagrams . . . . . 232
AIM EXPRESS listing entities . . . . . 162
Approval_is_assigned
AIM EXPRESS listing rules . . . . . 177
Approval_level
application assertion . . . . . 50
application object . . . . . 11
ARM diagrams . . . . . 224
mapping table . . . . . 102
Approval_person_organization
AIM diagrams . . . . . 232
AIM EXPRESS listing entities . . . . . 162
Approval_role
AIM diagrams . . . . . 232
AIM EXPRESS listing entities . . . . . 162
Approval_status
AIM diagrams . . . . . 232
AIM EXPRESS listing entities . . . . . 162
Approved_item
AIM diagrams . . . . . 232
AIM EXPRESS listing types . . . . . 152
mapping table . . . . . 67, 75, 79, 106, 109
Approver_role
AIM diagrams . . . . . 232
AIM EXPRESS listing entities . . . . . 162
ARM . . . . . 5
Assembly_component_usage
AIM diagrams . . . . . 240
AIM EXPRESS listing entities . . . . . 162
Assembly_or_component_usage
application assertion . . . . . 45, 48, 50
application object . . . . . 11
ARM diagrams . . . . . 222
mapping table . . . . . 74
ATS . . . . . 5

Attribute_type
AIM diagrams . . . . . 226
AIM EXPRESS listing types . . . . . 152
Authorization
application assertion . . . . . 45, 47, 48
application object . . . . . 12
ARM diagrams . . . . . 214
mapping table . . . . . 75
Authorized_approval
AIM EXPRESS listing rules . . . . . 177
Availability
application assertion . . . . . 45
application object . . . . . 13
ARM diagrams . . . . . 218
mapping table . . . . . 94
Axis1_placement
AIM diagrams . . . . . 245
AIM EXPRESS listing entities . . . . . 162
Axis2_placement
AIM diagrams . . . . . 245
AIM EXPRESS listing types . . . . . 152
Axis2_placement_2d
AIM diagrams . . . . . 245
AIM EXPRESS listing entities . . . . . 162
Axis2_placement_3d
AIM diagrams . . . . . 245
AIM EXPRESS listing entities . . . . . 162
Bag_to_set
AIM EXPRESS listing functions . . . . . 180
Build_2axes
AIM EXPRESS listing functions . . . . . 180
Build_axes
AIM EXPRESS listing functions . . . . . 181
Calendar_date
AIM diagrams . . . . . 249
AIM EXPRESS listing entities . . . . . 163
Capacity_level
application assertion . . . . . 47
application object . . . . . 14
ARM diagrams . . . . . 218
mapping table . . . . . 95
Cartesian_point
AIM diagrams . . . . . 246
AIM EXPRESS listing entities . . . . . 163
Change_definition
unit of functionality . . . . . 7

Characterized_action_definition
AIM diagrams . . . . . 238
AIM EXPRESS listing types . . . . . 153
Characterized_applied_action_assignment
AIM diagrams . . . . . 243
AIM EXPRESS listing entities . . . . . 163
mapping table . . . . . 101, 110, 112
Characterized_definition
AIM diagrams . . . . . 243
AIM EXPRESS listing types . . . . . 153
Characterized_object
AIM diagrams . . . . . 243
AIM EXPRESS listing entities . . . . . 163
Characterized_product_definition
AIM diagrams . . . . . 234
AIM EXPRESS listing types . . . . . 153
Characterized_resource_definition
AIM diagrams . . . . . 237
AIM EXPRESS listing types . . . . . 153
Compatible_dimension
AIM EXPRESS listing rules . . . . . 177
Compound_item_definition
AIM diagrams . . . . . 247
AIM EXPRESS listing types . . . . . 153
Compound_representation_item
AIM diagrams . . . . . 247
AIM EXPRESS listing entities . . . . . 163
Configuration_design
AIM diagrams . . . . . 233
AIM EXPRESS listing entities . . . . . 163
Configuration_design_item
AIM diagrams . . . . . 233
AIM EXPRESS listing types . . . . . 153
Configuration_effectivity
AIM diagrams . . . . . 239
AIM EXPRESS listing entities . . . . . 163
Configuration_item
AIM diagrams . . . . . 233
AIM EXPRESS listing entities . . . . . 164
Conformance class . . . . . 3
Conformance testing . . . . . 4
Context_dependent_measure
AIM diagrams . . . . . 248
AIM EXPRESS listing types . . . . . 153
Context_dependent_unit
AIM diagrams . . . . . 252

AIM EXPRESS listing entities . . . . . 164
Conversion_based_unit
AIM diagrams . . . . . 252
AIM EXPRESS listing entities . . . . . 164
Conversion_factor
application assertion . . . . . 45
application object . . . . . 14
ARM diagrams . . . . . 222
mapping table . . . . . 77
Coordinated_universal_time_offset
AIM diagrams . . . . . 249
AIM EXPRESS listing entities . . . . . 164
Count_measure
AIM diagrams . . . . . 248
AIM EXPRESS listing types . . . . . 153
Cross_product
AIM EXPRESS listing functions . . . . . 181
Data_template
application assertion . . . . . 45
application object . . . . . 14
ARM diagrams . . . . . 215
mapping table . . . . . 77
Date
AIM diagrams . . . . . 249
AIM EXPRESS listing entities . . . . . 164
Date_and_time
AIM diagrams . . . . . 249
AIM EXPRESS listing entities . . . . . 164
Date_and_time_assignment
AIM diagrams . . . . . 249
AIM EXPRESS listing entities . . . . . 164
Date_and_time_item
AIM diagrams . . . . . 250
AIM EXPRESS listing types . . . . . 153
mapping table . . . . . 82, 96, 122
Date_assignment
AIM diagrams . . . . . 249
AIM EXPRESS listing entities . . . . . 164
Date_role
AIM diagrams . . . . . 249
AIM EXPRESS listing entities . . . . . 164
Date_time_role
AIM diagrams . . . . . 249
AIM EXPRESS listing entities . . . . . 165
Date_time_select
AIM diagrams . . . . . 249

AIM EXPRESS listing types . . . . . 153
Dated_effectivity
AIM diagrams . . . . . 239
AIM EXPRESS listing entities . . . . . 165
Dated_item
AIM diagrams . . . . . 250
AIM EXPRESS listing types . . . . . 153
mapping table . . . . . 81
Day_in_month
AIM diagrams . . . . . 249
Day_in_month_number
AIM EXPRESS listing types . . . . . 154
Dependent_instantiable_person
AIM EXPRESS listing rules . . . . . 178
Dependent_instantiable_type_qualifier
AIM EXPRESS listing rules . . . . . 178
Derive_dimensional_exponents
AIM EXPRESS listing functions . . . . . 181
Derived_unit
AIM diagrams . . . . . 252
AIM EXPRESS listing entities . . . . . 165
Derived_unit_element
AIM diagrams . . . . . 252
AIM EXPRESS listing entities . . . . . 165
Descriptive_representation_item
AIM diagrams . . . . . 247
AIM EXPRESS listing entities . . . . . 165
Dimension_count
AIM diagrams . . . . . 236
AIM EXPRESS listing types . . . . . 154
Dimension_of
AIM EXPRESS listing functions . . . . . 182
Dimensional_exponents
AIM diagrams . . . . . 253
AIM EXPRESS listing entities . . . . . 165
Dimensions_for_si_unit
AIM EXPRESS listing functions . . . . . 182
Directed_action
AIM diagrams . . . . . 238
AIM EXPRESS listing entities . . . . . 165
Direction
AIM diagrams . . . . . 246
AIM EXPRESS listing entities . . . . . 165
Discrepant_item
application object . . . . . 15
ARM diagrams . . . . . 216

mapping table . . . . . 54
Document
AIM diagrams . . . . . 230
AIM EXPRESS listing entities . . . . . 165
Document_action_resource
AIM diagrams . . . . . 230
AIM EXPRESS listing entities . . . . . 166
mapping table . . . . . 61, 62, 64
Document_reference
AIM diagrams . . . . . 230
AIM EXPRESS listing entities . . . . . 166
Document_reference_item
AIM diagrams . . . . . 230
AIM EXPRESS listing types . . . . . 154
mapping table . . . . . 89
Document_relationship
AIM diagrams . . . . . 230
AIM EXPRESS listing entities . . . . . 166
Document_role
AIM diagrams . . . . . 230
AIM EXPRESS listing entities . . . . . 166
Document_type
AIM diagrams . . . . . 230
AIM EXPRESS listing entities . . . . . 166
Document_usage_constraint
AIM diagrams . . . . . 230
AIM EXPRESS listing entities . . . . . 166
Documentation
application assertion . . . . . 45
application object . . . . . 15
ARM diagrams . . . . . 220
mapping table . . . . . 102
Dot_product
AIM EXPRESS listing functions . . . . . 183
Effectivity
AIM diagrams . . . . . 239
AIM EXPRESS listing entities . . . . . 166
Electric_currenct_measure_with_unit
AIM diagrams . . . . . 251
Electric_current_measure_with_unit
AIM EXPRESS listing entities . . . . . 166
Electric_current_unit
AIM diagrams . . . . . 252
AIM EXPRESS listing entities . . . . . 166
End_item_effectivity
application assertion . . . . . 45, 46, 51

application object . . . . . 16
ARM diagrams . . . . . 222
mapping table . . . . . 77
Enhancement_item
application object . . . . . 16
ARM diagrams . . . . . 216
mapping table . . . . . 55
Enterprise
application assertion . . . . . 45
application object . . . . . 16
ARM diagrams . . . . . 219
mapping table . . . . . 78
Executed_action
AIM diagrams . . . . . 238
AIM EXPRESS listing entities . . . . . 167
Executed_action_is_assigned
AIM EXPRESS listing rules . . . . . 178
Expression_operand
AIM diagrams . . . . . 251
AIM EXPRESS listing types . . . . . 154
Facility
application assertion . . . . . 46
application object . . . . . 16
ARM diagrams . . . . . 220
mapping table . . . . . 102
Facility_capacity_level
application assertion . . . . . 46
application object . . . . . 17
ARM diagrams . . . . . 218
mapping table . . . . . 95
Field_maintenance
application assertion . . . . . 46
application object . . . . . 17
ARM diagrams . . . . . 221
mapping table . . . . . 107
First_proj_axis
AIM EXPRESS listing functions . . . . . 183
Functional_definition
application assertion . . . . . 46
application object . . . . . 18
ARM diagrams . . . . . 215
mapping table . . . . . 78
Functional_definition_usage
application assertion . . . . . 45-47, 51
application object . . . . . 18
ARM diagrams . . . . . 222

mapping table . . . . . 78
Geometric_representation_context
AIM diagrams . . . . . 236
AIM EXPRESS listing entities . . . . . 167
Geometric_representation_item
AIM diagrams . . . . . 246
AIM EXPRESS listing entities . . . . . 167
Global_unit_assigned_context
AIM diagrams . . . . . 236
AIM EXPRESS listing entities . . . . . 167
Group
AIM diagrams . . . . . 228
AIM EXPRESS listing entities . . . . . 167
Group_assignment
AIM diagrams . . . . . 228
AIM EXPRESS listing entities . . . . . 167
Group_is_assigned
AIM EXPRESS listing rules . . . . . 178
Group_relationship
AIM diagrams . . . . . 228
AIM EXPRESS listing entities . . . . . 167
Group_role
AIM diagrams . . . . . 228
AIM EXPRESS listing entities . . . . . 167
Grouped_item
AIM diagrams . . . . . 228
AIM EXPRESS listing types . . . . . 154
mapping table . . . . . 54, 55, 57, 70, 71, 83, 84, 91-93, 103, 107
Hour_in_day
AIM diagrams . . . . . 249
AIM EXPRESS listing types . . . . . 154
ICOM . . . . . 6
Id . . . . . 5
Identification_assignment
AIM EXPRESS listing entities . . . . . 168
AIM EXPRESS-G diagram . . . . . 229
Identification_item
AIM diagrams . . . . . 229
AIM EXPRESS listing types . . . . . 154
mapping table . . . . . 74, 75, 85, 107, 113, 120
Identification_role
AIM diagrams . . . . . 229
AIM EXPRESS listing entities . . . . . 168
Identifier
AIM EXPRESS listing types . . . . . 155
Implementation method . . . . . 3

Implementation under test . . . . . 4
Indentification_assignment
AIM diagrams . . . . . 229
Integrated resource . . . . . 3
Item_age
application assertion . . . . . 46
application object . . . . . 19
ARM diagrams . . . . . 218
mapping table . . . . . 96
Item_anomaly
application assertion . . . . . 46, 48
application object . . . . . 20
ARM diagrams . . . . . 223
mapping table . . . . . 55
Item_anomaly_disposition
application assertion . . . . . 46, 47
application object . . . . . 21
ARM diagrams . . . . . 223
mapping table . . . . . 65
Item_capacity_level
application assertion . . . . . 47
application object . . . . . 22
ARM diagrams . . . . . 218
mapping table . . . . . 98
Item_change
application assertion . . . . . 47
application object . . . . . 22
ARM diagrams . . . . . 216
mapping table . . . . . 67
Item_coordinate_location
application assertion . . . . . 47, 50
application object . . . . . 23
ARM diagrams . . . . . 217
mapping table . . . . . 108
Item_definition
unit of functionality . . . . . 7
Item_documentation_relationship
application object . . . . . 24
ARM diagrams . . . . . 214
mapping table . . . . . 79
Item_flaw
application assertion . . . . . 47
application object . . . . . 24
ARM diagrams . . . . . 223
mapping table . . . . . 70
Item_flaw_classification

application assertion . . . . . 47
application object . . . . . 24
ARM diagrams . . . . . 223
mapping table . . . . . 70
Item_in_context
AIM EXPRESS listing functions . . . . . 184
Item_issue_or_concern
application object . . . . . 24
ARM diagrams . . . . . 223
mapping table . . . . . 71
Item_product_property
application object . . . . . 25
ARM diagrams . . . . . 218
mapping table . . . . . 98
Item_program
application assertion . . . . . 47
application object . . . . . 25
ARM diagrams . . . . . 214
mapping table . . . . . 79
Item_properties
unit of functionality . . . . . 8
Item_property_representation
AIM diagrams . . . . . 242
AIM EXPRESS listing entities . . . . . 168
mapping table . . . . . 94-101, 108, 116, 118, 119, 124
Item_repairability_level
application assertion . . . . . 48
application object . . . . . 25
ARM diagrams . . . . . 218
mapping table . . . . . 98
Item_requiring_change
application assertion . . . . . 47, 48, 50
application object . . . . . 26
ARM diagrams . . . . . 216
mapping table . . . . . 71
Item_responsibility
application assertion . . . . . 48
application object . . . . . 27
ARM diagrams . . . . . 215
mapping table . . . . . 79
Item_task
application assertion . . . . . 48-51
application object . . . . . 27
ARM diagrams . . . . . 214
mapping table . . . . . 109
Item_task_authorization

application assertion . . . . . 48
application object . . . . . 28
ARM diagrams . . . . . 214
mapping table . . . . . 109
Item_task_frequency
application assertion . . . . . 48
application object . . . . . 28
ARM diagrams . . . . . 213
mapping table . . . . . 109
Item_task_time
application assertion . . . . . 48
application object . . . . . 28
ARM diagrams . . . . . 213
mapping table . . . . . 111
Item_version
application assertion . . . . . 46-49
application object . . . . . 29
ARM diagrams . . . . . 215
mapping table . . . . . 80
Item_version_relationship
application assertion . . . . . 49
application object . . . . . 30
ARM diagrams . . . . . 214
mapping table . . . . . 86
Item_work_unit_code
application assertion . . . . . 46, 49
application object . . . . . 31
ARM diagrams . . . . . 221
mapping table . . . . . 112
Label
AIM EXPRESS listing types . . . . . 155
Leap_year
AIM EXPRESS listing functions . . . . . 185
Length_measure
AIM diagrams . . . . . 248
AIM EXPRESS listing types . . . . . 155
Length_measure_with_unit
AIM diagrams . . . . . 251
AIM EXPRESS listing entities . . . . . 169
Length_unit
AIM diagrams . . . . . 252
AIM EXPRESS listing entities . . . . . 169
Local_time
AIM diagrams . . . . . 249
AIM EXPRESS listing entities . . . . . 169
Lot_effectivity

AIM diagrams . . . . . 239
AIM EXPRESS listing entities . . . . . 169
Make_from_usage
application object . . . . . 31
ARM diagrams . . . . . 222
mapping table . . . . . 91
Make_from_usage_option
AIM diagrams . . . . . 240
AIM EXPRESS listing entities . . . . . 169
Mass_measure
AIM diagrams . . . . . 248
AIM EXPRESS listing types . . . . . 155
Mass_measure_with_unit
AIM diagrams . . . . . 251
AIM EXPRESS listing entities . . . . . 169
Mass_unit
AIM diagrams . . . . . 252
AIM EXPRESS listing entities . . . . . 170
Measure_representation_item
AIM diagrams . . . . . 251
AIM EXPRESS listing entities . . . . . 170
Measure_value
AIM diagrams . . . . . 248
AIM EXPRESS listing types . . . . . 155
Measure_with_unit
AIM diagrams . . . . . 251
AIM EXPRESS listing entities . . . . . 170
Minute_in_hour
AIM diagrams . . . . . 249
AIM EXPRESS listing types . . . . . 155
Month_in_year_number
AIM diagrams . . . . . 249
AIM EXPRESS listing types . . . . . 155
Named_unit
AIM diagrams . . . . . 252
AIM EXPRESS listing entities . . . . . 170
Next_assembly_usage_occurrence
AIM diagrams . . . . . 240
AIM EXPRESS listing entities . . . . . 170
Normalise
AIM EXPRESS listing functions . . . . . 185
Numeric_measure
AIM diagrams . . . . . 248
AIM EXPRESS listing types . . . . . 155
Organization
AIM diagrams . . . . . 231

AIM EXPRESS listing entities . . . . . 170
application assertion . . . . . 48
application object . . . . . 32
ARM diagrams . . . . . 219
mapping table . . . . . 91
Organization_action_resource
AIM diagrams . . . . . 231
AIM EXPRESS listing entities . . . . . 170
mapping table . . . . . 61, 62, 64, 95
Organization_assignment
AIM diagrams . . . . . 231
AIM EXPRESS listing entities . . . . . 171
Organization_relationship
AIM diagrams . . . . . 231
AIM EXPRESS listing entities . . . . . 171
Organization_role
AIM diagrams . . . . . 231
AIM EXPRESS listing entities . . . . . 171
Organized_item
AIM diagrams . . . . . 230
AIM EXPRESS listing types . . . . . 155
mapping table . . . . . 80, 89
Orthogonal_complement
AIM EXPRESS listing functions . . . . . 186
Package
application object . . . . . 32
ARM diagrams . . . . . 220
mapping table . . . . . 103
Parameter_value
AIM diagrams . . . . . 248
AIM EXPRESS listing types . . . . . 155
Part
application assertion . . . . . 49
application object . . . . . 32
ARM diagrams . . . . . 215
mapping table . . . . . 91
Part_system_relationship
application object . . . . . 33
ARM diagrams . . . . . 214
mapping table . . . . . 91
Person
AIM diagrams . . . . . 231
AIM EXPRESS listing entities . . . . . 171
Person_organization_select
AIM diagrams . . . . . 231
AIM EXPRESS listing types . . . . . 156

Personnel
application assertion . . . . . 49
application object . . . . . 33
ARM diagrams . . . . . 220
mapping table . . . . . 103
Personnel_skill_level
application assertion . . . . . 49
application object . . . . . 33
ARM diagrams . . . . . 213
mapping table . . . . . 103
Physical_unit
application assertion . . . . . 46, 49
application object . . . . . 34
ARM diagrams . . . . . 215
mapping table . . . . . 91
PICS . . . . . 6
Placement
AIM diagrams . . . . . 245
AIM EXPRESS listing entities . . . . . 171
Point
AIM diagrams . . . . . 246
AIM EXPRESS listing entities . . . . . 171
Positive_length_measure
AIM diagrams . . . . . 248
AIM EXPRESS listing types . . . . . 156
Positive_ratio_measure
AIM diagrams . . . . . 248
AIM EXPRESS listing types . . . . . 156
Product . . . . . 3
AIM diagrams . . . . . 226
AIM EXPRESS listing entities . . . . . 171
Product data . . . . . 3
Product_category
AIM diagrams . . . . . 227
AIM EXPRESS listing entities . . . . . 171
Product_category_relationship
AIM diagrams . . . . . 227
AIM EXPRESS listing entities . . . . . 172
Product_concept
AIM diagrams . . . . . 226
AIM EXPRESS listing entities . . . . . 172
Product_concept_context
AIM diagrams . . . . . 226
AIM EXPRESS listing entities . . . . . 172
Product_context
AIM diagrams . . . . . 226

AIM EXPRESS listing entities . . . . . 172
Product_definition
AIM diagrams . . . . . 234
AIM EXPRESS listing entities . . . . . 172
Product_definition_context
AIM diagrams . . . . . 226
AIM EXPRESS listing entities . . . . . 172
Product_definition_effectivity
AIM diagrams . . . . . 239
AIM EXPRESS listing entities . . . . . 172
Product_definition_formation
AIM diagrams . . . . . 233
AIM EXPRESS listing entities . . . . . 172
Product_definition_formation_relationship
AIM diagrams . . . . . 233
AIM EXPRESS listing entities . . . . . 172
Product_definition_formation_with_specified_source
AIM diagrams . . . . . 233
AIM EXPRESS listing entities . . . . . 173
Product_definition_relationship
AIM diagrams . . . . . 234
AIM EXPRESS listing entities . . . . . 173
Product_definition_shape
AIM diagrams . . . . . 235
AIM EXPRESS listing entities . . . . . 173
Product_definition_usage
AIM diagrams . . . . . 240
AIM EXPRESS listing entities . . . . . 173
Product_definition_with_associated_documents
AIM diagrams . . . . . 234
AIM EXPRESS listing entities . . . . . 173
Product_property
application object . . . . . 34
ARM diagrams . . . . . 215
mapping table . . . . . 92
Product_related_product_category
AIM diagrams . . . . . 227
AIM EXPRESS listing entities . . . . . 173
Program
application object . . . . . 34
ARM diagrams . . . . . 219
mapping table . . . . . 92
Promissory_usage_occurrence
AIM diagrams . . . . . 240
AIM EXPRESS listing entities . . . . . 173
Property_definition

AIM diagrams . . . . . 235
AIM EXPRESS listing entities . . . . . 173
Property_definition_relationship
AIM EXPRESS listing entities . . . . . 174
Property_definition_representation
AIM diagrams . . . . . 241
AIM EXPRESS listing entities . . . . . 174
Property_or_shape_select
AIM diagrams . . . . . 235
AIM EXPRESS listing types . . . . . 156
Protocol information and conformance statement . . . . . 4
Qualified_representation_item
AIM diagrams . . . . . 247
AIM EXPRESS listing entities . . . . . 174
Quantified_assembly_component_usage
AIM diagrams . . . . . 240
AIM EXPRESS listing entities . . . . . 174
Ratio_measure
AIM diagrams . . . . . 248
AIM EXPRESS listing types . . . . . 156
Ratio_measure_with_unit
AIM diagrams . . . . . 251
AIM EXPRESS listing entities . . . . . 174
Ratio_unit
AIM diagrams . . . . . 252
AIM EXPRESS listing entities . . . . . 174
Recommended_support_resource
application assertion . . . . . 49, 50
application object . . . . . 34
ARM diagrams . . . . . 217
mapping table . . . . . 113
Reference_activity
application assertion . . . . . 49
application object . . . . . 35
ARM diagrams . . . . . 216
mapping table . . . . . 116
Related_change
application assertion . . . . . 50
application object . . . . . 36
ARM diagrams . . . . . 216
mapping table . . . . . 73
Representation
AIM diagrams . . . . . 242
AIM EXPRESS listing entities . . . . . 174
Representation_context
AIM diagrams . . . . . 236

AIM EXPRESS listing entities . . . . . 174
Representation_item
AIM diagrams . . . . . 247
AIM EXPRESS listing entities . . . . . 175
Representation_relationship
AIM diagrams . . . . . 242
AIM EXPRESS listing entities . . . . . 175
Represented_definition
AIM diagrams . . . . . 235
AIM EXPRESS listing types . . . . . 156
Resource_activity_location
application assertion . . . . . 50
application object . . . . . 36
ARM diagrams . . . . . 217
mapping table . . . . . 116
Resource_property
AIM diagrams . . . . . 237
AIM EXPRESS listing entities . . . . . 175
Resource_property_relationship
AIM EXPRESS listing entities . . . . . 175
Resource_property_representation
AIM diagrams . . . . . 242
AIM EXPRESS listing entities . . . . . 175
Scalar_times_vector
AIM EXPRESS listing functions . . . . . 186
Second_in_minute
AIM diagrams . . . . . 249
AIM EXPRESS listing types . . . . . 156
Selected_action_resource
AIM EXPRESS listing rules . . . . . 178
Serial_numbered_effectivity
AIM diagrams . . . . . 239
AIM EXPRESS listing entities . . . . . 175
Set_representation_item
AIM diagrams . . . . . 247
AIM EXPRESS listing types . . . . . 156
Shape_definition
AIM diagrams . . . . . 235
AIM EXPRESS listing types . . . . . 156
Si_prefix
AIM diagrams . . . . . 252
AIM EXPRESS listing types . . . . . 156
Si_unit
AIM diagrams . . . . . 252
AIM EXPRESS listing entities . . . . . 175
Si_unit_name

AIM diagrams . . . . . 252
AIM EXPRESS listing types . . . . . 157
Skill
application assertion . . . . . 50
application object . . . . . 36
ARM diagrams . . . . . 220
mapping table . . . . . 104
Skill_level
application assertion . . . . . 49-51
application object . . . . . 37
ARM diagrams . . . . . 213
mapping table . . . . . 104
Source
AIM diagrams . . . . . 233
AIM EXPRESS listing types . . . . . 157
Source_maintenance
application assertion . . . . . 50
application object . . . . . 37
ARM diagrams . . . . . 222
mapping table . . . . . 92
Source_maintenance_assembly_component_usage
AIM EXPRESS listing rules . . . . . 178
Specified_higher_usage_occurrence
AIM diagrams . . . . . 240
AIM EXPRESS listing entities . . . . . 175
Subtype_exclusive_action_resource
AIM EXPRESS listing rules . . . . . 179
Subtype_mandatory_characterized_object
AIM EXPRESS listing rules . . . . . 179
Supplier
application object . . . . . 38
ARM diagrams . . . . . 219
mapping table . . . . . 92
Support_equipment
application object . . . . . 38
ARM diagrams . . . . . 220
mapping table . . . . . 105
Support_resource
application assertion . . . . . 49-51
application object . . . . . 39
ARM diagrams . . . . . 220
mapping table . . . . . 105
Support_resource_approval_authority
application assertion . . . . . 46, 50
application object . . . . . 39
ARM diagrams . . . . . 224

mapping table . . . . . 105
Supported_item
AIM diagrams . . . . . 237
AIM EXPRESS listing types . . . . . 157
Supporting_resources
unit of functionality . . . . . 9
System
application object . . . . . 39
ARM diagrams . . . . . 215
mapping table . . . . . 93
Task
application assertion . . . . . 49
application object . . . . . 40
ARM diagrams . . . . . 216
mapping table . . . . . 117
Task_condition
application assertion . . . . . 50, 52
application object . . . . . 41
ARM diagrams . . . . . 213
mapping table . . . . . 118
Task_definition
unit of functionality . . . . . 10
Task_execution
application assertion . . . . . 47, 48, 51
application object . . . . . 41
ARM diagrams . . . . . 221
mapping table . . . . . 119
Task_execution_context_definition
application assertion . . . . . 51
application object . . . . . 42
ARM diagrams . . . . . 221
mapping table . . . . . 120
Task_execution_status
application assertion . . . . . 51
application object . . . . . 42
ARM diagrams . . . . . 221
mapping table . . . . . 121
Task_execution_support_resource
application assertion . . . . . 51
application object . . . . . 43
ARM diagrams . . . . . 221
mapping table . . . . . 123
Task_requiring_capacity_level
application assertion . . . . . 51
application object . . . . . 44
ARM diagrams . . . . . 218

mapping table . . . . . 100
Task_skill_level
application assertion . . . . . 51, 52
application object . . . . . 44
ARM diagrams . . . . . 213
mapping table . . . . . 124
Text
AIM EXPRESS listing types . . . . . 157
Time_measure
AIM diagrams . . . . . 248
AIM EXPRESS listing types . . . . . 157
Time_measure_with_unit
AIM diagrams . . . . . 251
AIM EXPRESS listing entities . . . . . 176
Time_unit
AIM diagrams . . . . . 252
AIM EXPRESS listing entities . . . . . 176
Trimming_select
AIM diagrams . . . . . 246
AIM EXPRESS listing types . . . . . 157
Type_qualifier
AIM diagrams . . . . . 247
AIM EXPRESS listing entities . . . . . 176
Unit
AIM diagrams . . . . . 252
AIM EXPRESS listing types . . . . . 157
Unit of functionality . . . . . 3
UoF . . . . . 6
Using_representations
AIM EXPRESS listing functions . . . . . 186
Valid_calendar_date
AIM EXPRESS listing functions . . . . . 187
Valid_time
AIM EXPRESS listing functions . . . . . 187
Valid_units
AIM EXPRESS listing functions . . . . . 188
Value_qualifier
AIM diagrams . . . . . 247
AIM EXPRESS listing types . . . . . 158
Vector
AIM diagrams . . . . . 246
AIM EXPRESS listing entities . . . . . 176
Vector_difference
AIM EXPRESS listing functions . . . . . 189
Vector_or_direction
AIM diagrams . . . . . 246

AIM EXPRESS listing types. . . . . .158
Versioned_action_request
AIM diagrams . . . . . .244
AIM EXPRESS listing entities . . . . . .176
Volume_measure
AIM diagrams . . . . . .248
AIM EXPRESS listing types. . . . . .158
Volume_measure_with_unit
AIM diagrams . . . . . .251
AIM EXPRESS listing entities . . . . . .176
Volume_unit
AIM diagrams . . . . . .252
AIM EXPRESS listing entities . . . . . .177
Year_number
AIM diagrams . . . . . .249
AIM EXPRESS listing types. . . . . .158

1.

2.

3.

4.