

INTERNATIONAL  
STANDARD

**ISO**  
**10303-44**

First edition  
1994-12-15

---

---

**Industrial automation systems and  
integration — Product data representation  
and exchange —**

**Part 44:**

Integrated generic resources: Product  
structure configuration

*Systemes d'automatisation industrielle et integration — Représentation  
et échange de données de produits —*

*Partie 44: Ressources génériques intégrées: Configuration de structure de  
produits*



Reference number  
ISO 10303-44:1994(E)

<b>Contents</b>	<b>Page</b>
1 Scope . . . . .	1
2 Normative references . . . . .	2
3 Definitions and abbreviations . . . . .	2
4 Product structure schema . . . . .	4
4.1 Introduction . . . . .	5
4.2 Fundamental concepts and assumptions . . . . .	6
4.3 Product structure schema entity definitions . . . . .	7
4.3.1 alternate_product_relationship . . . . .	7
4.3.2 product_definition_usage . . . . .	8
4.3.3 make_from_usage_option . . . . .	9
4.3.4 make_from_usage_option_group . . . . .	11
4.3.5 assembly_component_usage . . . . .	12
4.3.6 quantified_assembly_component_usage . . . . .	13
4.3.7 next_assembly_usage_occurrence . . . . .	14
4.3.8 specified_higher_usage_occurrence . . . . .	15
4.3.9 promissory_usage_occurrence . . . . .	17
4.3.10 assembly_component_usage_substitute . . . . .	18
5 Product concept schema . . . . .	19
5.1 Introduction . . . . .	19
5.2 Fundamental concepts and assumptions . . . . .	19
5.3 Product concept schema entity definitions: product_concept . . . . .	19
6 Configuration management schema . . . . .	20
6.1 Introduction . . . . .	21
6.2 Fundamental concepts and assumptions . . . . .	22
6.3 Configuration management schema entity definitions . . . . .	22
6.3.1 configuration_item . . . . .	22
6.3.2 configuration_design . . . . .	23
6.3.3 configuration_effectivity . . . . .	24

© ISO 1994  
 All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from the publisher.

International Organization for Standardization  
 Case Postale 56 • CH-1211 Genève 20 • Switzerland

Printed in Switzerland

**Annexes**

A Short names of entities . . . . .	26
B Information object registration . . . . .	27
B.1 Document identification . . . . .	27
B.2 Schema identification . . . . .	27
C Computer-interpretable listings . . . . .	28
D Graphical model representations . . . . .	29
E Examples . . . . .	33
E.1 Introduction . . . . .	33
E.2 Example of a mechanical assembly product . . . . .	33
E.3 Examples of product structure reports . . . . .	33
E.3.1 Example: A numeric, multi-level, indented product structure report . . . . .	35
E.3.2 Example: A labelled occurrence, multi-level, indented product structure report . . . . .	36
E.3.3 Example: A numeric, multi-level, indented, extended product structure report . . . . .	37
E.4 BOM and parts list data structures . . . . .	37
E.4.1 BOM data structure . . . . .	37
E.4.2 Parts list data structure . . . . .	38
E.5 Representation of product structure data structures using Part 44 entities . . . . .	38
E.6 Generating product structure reports . . . . .	45
E.7 Summary . . . . .	45
F Bibliography . . . . .	47
Index . . . . .	48

**Figures**

Figure E.1 - Part 44 example product . . . . .	34
Figure E.2 - Part 44 example product: graphical product structure . . . . .	35
Figure E.3 - BOM data structure for Part 44 example product . . . . .	38
Figure E.4 - Parts list data structure for Part 44 example product . . . . .	39
Figure E.5 - BOM data structure examples using Part 44 entities . . . . .	40
Figure E.6 - Parts list structure examples using Part 44 entities . . . . .	41
Figure E.7 - BOM data structure of Part 44 example product . . . . .	42
Figure E.8 - Simple specified higher usage occurrence . . . . .	43
Figure E.9 - Identifying specific usages with specified higher usage occurrence . . . . .	44
Figure E.10 - Complex specified higher usage occurrence . . . . .	46

**Tables**

Table 1 - Example 8 Make_from_usage_option_group . . . . .	11
Table 2 - Example 9 Make_from_usage_option_group . . . . .	11

## 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 10303-44 was prepared by Technical Committee ISO/TC184, *Industrial automation systems and integration*, Subcommittee SC4, *Industrial data and global manufacturing languages*.

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 21 Implementation methods: Clear text encoding of the exchange structure;
- Part 22 Implementation methods: Standard data access interface;
- Part 31 Conformance testing methodology and framework: General concepts;
- Part 32 Conformance testing methodology and framework: requirements on testing laboratories and clients;
- 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 resources: Materials;
- Part 46 Integrated generic resources: Visual presentation;
- Part 47 Integrated generic resources: Shape variation tolerances;
- Part 49 Integrated generic resources: Process structure and properties;

- Part 101 Integrated application resources: Draughting;
- Part 104 Integrated application resources: Finite element analysis;
- Part 105 Integrated application resources: Kinematics;
- Part 201 Application protocol: Explicit draughting;
- Part 202 Application protocol: Associative draughting;
- Part 203 Application protocol: Configuration controlled design;
- Part 207 Application protocol: Sheet metal die planning and design;
- Part 210 Application protocol: Printed circuit assembly product design data;
- Part 213 Application protocol: Numerical control process plans for machined parts.

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

- Part 11 specifies the description method;
- Parts 21 and 22 specify the implementation methods;
- Parts 31 and 32 specify the conformance testing methodology and framework;
- Parts 41 to 49 specify the integrated generic resources;
- Parts 101 to 105 specify the integrated application resources;
- Parts 201 to 213 specify the application protocols.

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

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

#### **Diskette**

Users should note that this part of ISO 10303 comprises a diskette:

- the short names of entities given in annex A are also included on the diskette;
- the EXPRESS listings (annex C) are provided on the diskette only;
- a method to enable users to report errors in the documentation is given. Full details are provided in the file.

## 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 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 integrated resources series. Major subdivisions of this part of ISO 10303 are:

- the product structure schema;
- the product concept schema;
- the configuration management schema.

The product structure schema:

- defines a product in terms of its composition as a set of constituents or consumed products. A product may be assembled from the constituents or produced by consuming other products, or both;
- defines mechanisms for expressing the composition relationship.

The product concept schema identifies the product concept as a set of specifications for a product derived from analysis of customer needs for the product. It represents the idea of a product based on customer needs and not as it might be designed or built.

The configuration management schema identifies those products participating in the manufacture of another product whose configurations are under the direct control of an organization.

Industrial automation is concerned with the management of the information including the following:

- product structure and its complexity;
- product configuration and its complexity;
- product change.

Product structure is focussed on that aspect of product design defining a product in terms of a nested decomposition into constituents. The product structure schema of this part and the product definition schema of ISO 10303-41 together define the representation of the information that manages the complexity of product structure. An example of the use of the information represented is the generation of bill-of-material reports.

Product configuration is concerned with the specification of manufacturing or assembly plans for specific products. The planning includes specification of the actual constituents of a product which are to be included in a planned unit of production. The configuration management schema and the product structure schema represent the information that manages the configuration of a product. The concept of effectivity is used to manage the configuration of a product.

Change management is involved with the changes over time in a product as new versions of a product are developed. This part of ISO 10303 is concerned with changes that affect the organization of constituents into interrelated structures. The configuration management schema represents information on the structural form of the definition of a product as the product changes and is enhanced during the product life cycle. Representation of information on other aspects of change management is defined in the product definition schema and the change schema of ISO 10303-41.

STANDARDSISO.COM : Click to view the full PDF of ISO 10303-44:1994

This page intentionally left blank

STANDARDSISO.COM : Click to view the full PDF of ISO 10303-44:1994

---

# Industrial automation systems and integration — Product data representation and exchange — Part 44: Integrated generic resources: Product structure configuration

## 1 Scope

This part of ISO 10303 specifies the resources to manage the structure and configuration of a product during its life cycle.

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

- the relationship among the components and assemblies of a product;
- the relationship among products and their components as made by modification of other products;

EXAMPLE 1 - The machining of a product from another product.

- the description of a product as defined by customer needs;
- the management of the structure for configuration of assemblies and components as planned for manufacture;
- the decomposition of a product to support different product life cycle activities;

NOTE 1 - An organization may need to decompose a product into one bill of materials that enumerates each component with respect to the number of that component used in an assembly, and into a second bill of materials that decomposes a product with multiple assemblies into the individual components. See annex E for more examples of different product structure reports that are supported.

- multiple versions of a single product which are equivalent with respect to form, fit, and function.

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

- the relationships among different product definitions for the same product;

EXAMPLE 2 - The relationship of a product definition for a component in a preliminary design to a corresponding product definition for the same component in a detailed design;

- administrative activities of the product life cycle including approvals, security classifications, contractual arrangements, and supplier organizations;

- the change process for a product, including the reason for change and what aspect of a product has changed;
- the decisions made, and their reasons, during the product life cycle;
- the physical connections among components of a product;
- the properties that a product constituent may have;

NOTE 2 - A mechanism is defined in ISO 10303-41 to support the association of properties with components. The actual associations are included in various application protocols which are parts of this International Standard. For example, the details of what a material property is and how it is defined are out of scope, as well as the fact that a component has a material property.

- the information for as-built manufacturing, manufacturing planning, and logistical structure and configurations;
- multiple versions of a single product that are not form, fit, and function equivalent.

NOTE 3 - The concept of versions of a product is defined in ISO 10303-41.

## 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:–<sup>1))</sup>, *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 10303-41:1994, *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 10303-43:1994, *Industrial automation systems and integration — Product data representation and exchange — Part 43: Integrated generic resources: Representation structures.*

---

<sup>1))</sup> To be published.

### 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.

- assembly;
- component;
- product.

#### 3.2 Other definitions

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

**3.2.1 ancestor node:** any node that can be reached from a given node, by successive traversals of links in the reverse direction. For a given node, its ancestor nodes include all parent nodes, all parent nodes of these parent nodes, etc.

**3.2.2 bill-of-material structure (BOM):** a structural description of a product in terms of its nested constituents.

NOTE - See annex E for examples of bill-of-material structures.

**3.2.3 child node:** the node to which a link is pointing.

**3.2.4 constituent:** a subdivision of a product, either a component or an assembly.

**3.2.5 directed acyclic graph (DAG):** a collection of nodes and directed links such that no node is an ancestor (or descendant) of itself.

**3.2.6 descendent node:** any node that can be reached from a given node, by successive traversals of links. For a given node, its descendent nodes include all children nodes, all children nodes of these children nodes, etc.

**3.2.7 effectivity:** a characteristic that indicates when a product is allowed, approved or permitted to be used in another product.

**3.2.8 form, fit, and function of a product:** the form is the shape of the product, the fit is the way the product interfaces with other products, and the function is the purpose that the product serves.

**3.2.9 leaf node:** a node that has no children.

**3.2.10 link:** a uni-directional relationship from one node to another node within a directed acyclic graph.

**3.2.11 lot:** a collection of actual units treated as a single unit.

**3.2.12 node:** an element of a directed acyclic graph, connected to other such elements by links.

**3.2.13 parent node:** the node from which a link is initiated.

**3.2.14 parts list structure:** a structural description of a product in terms of a hierarchy of all distinct usages of its constituents.

NOTE - See annex E for cases of parts list structures.

**3.2.15 promissory use:** the intention to use a constituent in an assembly.

**3.2.16 root node:** a node that has no parents.

**3.2.17 tree:** a restricted type of directed acyclic graph in which there is only one root node, and in which each node has at most one parent.

### 3.3 Abbreviations

For the purposes of this part of ISO 10303, the following abbreviations apply.

BOM bill-of-material.

CM configuration management.

DAG directed acyclic graph.

## 4 Product structure schema

The following EXPRESS declaration begins the **product\_structure\_schema** and identifies the necessary external references.

EXPRESS specification:

```

*)
SCHEMA product_structure_schema;

    REFERENCE FROM product_definition_schema
        (product,
         product_definition_relationship,
         acyclic_product_definition_relationship);

    REFERENCE FROM measure_schema
        (measure_with_unit);

    REFERENCE FROM support_resource_schema
        (identifier, label, text);
(*)

```

NOTE - The schemas referenced above can be found in the following parts of ISO 10303:

product_definition_schema	ISO 10303-41
measure_schema	ISO 10303-41
support_resource_schema	ISO 10303-41

## 4.1 Introduction

The subject of the **product\_structure\_schema** is the relationship between the definitions of:

- products which are assembled to make other products;
- products which are changed to make other products.

These relationships are defined as specialisations of relationships among **product\_definitions** as specified in the **product\_definition\_schema** of ISO 10303-41. In addition, any product that is used in a product structure may have alternative products specified for that use. Collectively these relationships are referred to as a product structure.

The product structure defines the different methods by which a product can be represented as being made up of constituents. Product structure relationships are established among the constituents that make up a product. Product structure is an aspect of product definition.

The relationships between product definitions are specializations of the general relationship among product definitions in the **product\_definition\_schema** of ISO 10303-41. They are represented in this International Standard using a subtype of the **product\_definition\_relationship** entity as defined in the product definition schema of ISO 10303-41. The subtypes of **product\_definition\_relationship** defined in this schema establish additional constraints and meanings to the supertype in ISO 10303-41.

The product structure concept applies to multiple definitions of the structure of a single product version. The different definitions correspond to different organization requirements for defining the structure of a product during the life cycle of the product's development.

EXAMPLE 3 - An organization may define a bill-of-material for both a design engineering release life cycle activity, as well as for a manufacturing engineering activity.

This schema supports the concept of making a product from another product. This concept deals with the relationship between a product and the results of a process applied to that product which produces a new product.

The **product\_structure\_schema** develops the following specific concepts for use in representing the various forms of product structure:

- the **alternate\_product\_relationship** is used to convey the information that two products have a relationship whereby one product may be used in the place of another product;
- the **make\_from\_usage\_option** represents the fact that any actual unit of one design can be manufactured by consuming or modifying an actual unit of another design;

NOTE 1 - Typically the consumed product is referred to as stock or raw material.

– the **make\_from\_usage\_option\_group** is used to represent one specific combination of products that can be made from a single product;

NOTE 2 - Typically the single product is referred to as stock or raw material.

– the **assembly\_component\_usage** establishes relationships between designs within one of four different subtypes:

a) The **quantified\_assembly\_component\_usage** represents the relationship between a constituent and an assembly. For discrete constituents, several occurrences of the constituent are represented by the single constituent and the quantity representing the number of occurrences of it. For non-discrete constituents the quantity represents a unit of measure other than a unitless number;

b) The **next\_assembly\_usage\_occurrence** represents the relationship between an occurrence of a constituent and its immediate assembly;

c) The **specified\_higher\_usage\_occurrence** represents the relationship between a specific use of a constituent with respect to a non-immediate/non-parent ancestor assembly within the product structure;

d) The **promissory\_usage\_occurrence** represents the relationship between a constituent and an ancestor assembly within an overall product structure without any specification of the intermediate assemblies being represented.

## 4.2 Fundamental concepts and assumptions

Product structures are modelled by directed acyclic graphs (DAG). In these models, nodes represent product definitions, and the directed links represent composed-of relationships. In this schema, nodes correspond to **product\_definition** entities and the links correspond to **assembly\_component\_usage** entities.

NOTE 1 - For a detailed discussion of graph theory, see [1].

Many forms of product structure can be represented using this schema. Two product structures of special utility are bill-of-material and parts list structures.

NOTE 2 - Examples and diagrams of the manner in which the entities of this part may be used to represent product structures are included in annex E.

A parts list structure is a specific form of a bill-of-material that can be represented by a tree. A bill-of-material structure may require a more general DAG.

For a general product structure, in order to identify the usage of any constituent within an assembled product, it is necessary to identify the path between the assembled product and the constituent. The **specified\_higher\_usage\_occurrence** entity provides this capability.

## 4.3 Product structure schema entity definitions

### 4.3.1 alternate\_product\_relationship

The alternate product relationship is used to convey the information that two products have a relationship whereby one product, the **alternate** product, may be used in the place of another product, the **base** product.

When one product is an alternate for another product it is understood that there is no interest within the organization to keep track of which product, the base or any alternates specified, is used as a particular instance of the base product within a product structure.

NOTE - An organization may track design changes for a base part, and establish effectivity conditions for the use of that base part in various assemblies to be manufactured. The use of an alternate product implies that an organization does not specify any particular version of the alternate product or establish effectivities relating to it.

The two products specified in the entity are equivalent with respect to form, fit, and function, as determined by the organization.

EXAMPLE 4 - Two screws of the same size are products. One screw has a cross head and the other has a straight head. These head shapes are properties of the respective products. Within a particular organization, the two screws are considered equivalent with respect to form, fit, and function: they both have sufficiently close physical shape, they take up the same space when used, and they both serve to fasten two things together. Thus, one of these two screws could be considered to be an alternate part for the other screw. A different organization might consider the forms to be different.

The use of the **alternate\_product\_relationship** entity shall be defined in the context of the product structure of the base product. The relationship established by the use of **alternate\_product\_relationship** need not be symmetric. If B is an alternate product for A, it is not required that A is also an alternate product for B.

#### EXPRESS specification:

```

*)
ENTITY alternate_product_relationship;
  name      : label;
  definition: text;
  alternate : product;
  base      : product;
  basis     : text;
UNIQUE
  UR1: alternate, base;
WHERE
  WR1: alternate :<>: base;
END_ENTITY;
(*

```

Attribute definitions:

**alternate:** a product that may be used in place of the base product.

**base:** a product for which another product may be used as an alternate.

**basis:** a text description to specify the rationale and domain of applicability of the alternate product.

Formal propositions:

**UR1:** The combination of the **alternate** product and the **base** product shall be unique.

**WR1:** The instance for the **alternate** shall not be the same as the instance of the **base** attribute.

### 4.3.2 product\_definition\_usage

The **product\_definition\_usage** is a specialization of the **product\_definition\_relationship** entity of ISO 10303-41 for use within the context of product structure definition and management. This specialization adds meaning to the two attributes: **relating\_product\_definition**, **related\_product\_definition**.

The subtypes of this entity represent different kinds of product structure relationships between the referenced pair of **product\_definitions**. One subtype, **make\_from\_usage\_option**, represents the relationship between a product and another product, where one product is made from the other. The other subtype, **assembly\_component\_usage**, represents the relationship between an assembly and one of its constituents.

EXPRESS specification:

```

*)
ENTITY product_definition_usage
  SUPERTYPE OF (ONEOF (make_from_usage_option,
                      assembly_component_usage))
  SUBTYPE OF (product_definition_relationship);
UNIQUE
  UR1: SELF\product_definition_relationship.id,
        SELF\product_definition_relationship.relating_product_definition,
        SELF\product_definition_relationship.related_product_definition;
WHERE
  WR1: acyclic_product_definition_relationship
        (SELF,
         [SELF\product_definition_relationship.related_product_definition],
         'PRODUCT_STRUCTURE_SCHEMA.PRODUCT_DEFINITION_USAGE.' +
         'RELATED_PRODUCT_DEFINITION');
END_ENTITY;
(*

```

Attribute definitions:

**SELF\product\_definition\_relationship.id:** An identifier for a usage of a **product\_definition**. It is used to distinguish between two instances of **product\_definition\_usage** where the pair of **product\_definition** attributes are the same.

EXAMPLE 5 - If four identical bolts are used to attach two plates, there may be a need to identify one specific bolt for some purpose. It needs to be torqued to a greater degree than the rest. The **id** attribute then is used to identify this specific bolt's requirement, even though all four bolt **product\_definition\_usages** will have the same attribute pair of **product\_definitions**.

**SELF\product\_definition\_relationship.relying\_product\_definition**: a **product\_definition** that is made from or serves as the assembly for the **related\_product\_definition**.

**SELF\product\_definition\_relationship.related\_product\_definition**: a **product\_definition** from which the **relying\_product\_definition** is made or which is the component in the **relying\_product\_definition** assembly.

Formal propositions:

**UR1**: The inherited **id**, **relying\_product\_definition**, and **related\_product\_definition** uniquely identify an instance of **product\_definition\_usage**.

**WR1**: The graph structure of **product\_definition** nodes and **product\_definition\_usage** links shall be acyclic. Each **product\_definition** shall not be a descendant of itself in the graph structure.

NOTE - This rule uses the function **acyclic\_product\_definition\_relationship** defined in ISO 10303-41.

### 4.3.3 make\_from\_usage\_option

The **make\_from\_usage\_option** identifies that a product is made from another product through machining or some other unspecified process.

In situations in which a product is made from another product using a sequence of processes, the intermediate products will be related using the **make\_from\_usage\_option** entity.

A product to be modified can be an assembly.

NOTE 1 - Generally, the **assembly\_component\_usage** differs from the **make\_from\_usage\_option** in that the constituents of an assembly are used in the assembly without any change.

A **make\_from\_usage\_option** applies to designs, represented by **product\_definitions**, rather than the actual units of the designs. A **make\_from\_usage\_option** relationship is independent of any specific manufactured instances of actual units and is represented by the attribute references, inherited from the supertype entities, to the **relying\_product\_definition** and **related\_product\_definition**.

A **product\_definition** may be the **relying\_product\_definition** of many **make\_from\_usage\_option** relationships, and a **product\_definition** may be the **related\_product\_definition** of many **make\_from\_usage\_option** relationships. Further, there may be multiple **make\_from\_usage\_option** instances referencing the same **relying\_product\_definition** and **related\_product\_definition** pair of **product\_definitions**.

EXAMPLE 6 - Consider the case of a shaft which can be machined from either a casting or a forging. All three, the shaft, the forging and the casting, are represented by separate instances of **product\_definitions**. Two instances of the **make\_from\_usage\_option** entity exist, one between the **relying\_product\_definition**

shaft and the **related\_product\_definition** forging, the other between the **relating\_product\_definition** shaft and the **related\_product\_definition** casting.

EXPRESS specification:

```

*)
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;
(*

```

Attribute definitions:

**SELF\product\_definition\_relationship.relying\_product\_definition:** a **product\_definition** made from the **related\_product\_definition** product.

**SELF\product\_definition\_relationship.related\_product\_definition:** a **product\_definition** from which the **relating\_product\_definition** is made.

**ranking:** an integer which ranks the preference for use of the **related\_product\_definition** input **product\_definition** among all **make\_from\_usage\_option** instances with the same value for the inherited **relating\_product\_definition** attribute. This is a positive integer value that only has meaning when comparing it with corresponding values for **make\_from\_usage\_options** sharing the same **relating\_product\_definition** **product\_definition**. It is a relative ranking value, not an absolute ranking. A lower value indicates a higher preference for the **related\_product\_definition** **product\_definition**, and a higher value indicates a lower preference.

NOTE 2 - Special care is required when assigning these values. If different organizations use different ranges of values, and if populated data sets from these organizations are merged, and multiple **make\_from\_usage\_options** from both organizations then exist in the merged file for a single **relating\_product\_definition** **product\_definition**, then non-comparable values for this attribute may result.

**ranking\_rationale:** text which describes the rationale used for the ranking.

EXAMPLE 7 - Examples of **ranking\_rationale** are cost and long lead time.

**quantity:** the number of physical instances of the **relating\_product\_definition** **product\_definition** that can be made from one unit of a **related\_product\_definition** **product\_definition**.

Formal propositions:

**WR1:** The value of **ranking** shall be positive.

Informal propositions:

**IP1:** If `make_from_usage_option.quantity.value_component` is represented by a NUMBER then its value shall be greater than 0.

**4.3.4 make\_from\_usage\_option\_group**

The `make_from_usage_option_group` is used to indicate that several different products may be made from a single product. A `make_from_usage_option_group` instance shall represent a collection of `make_from_usage_option` instances that defines one possible combination of `product_definition` that can be made from another single `product_definition`.

The single product constraint is represented by the fact that the `related_product_definition` `product_definitions` for all of the instances of the `make_from_usage_options` within a single `make_from_usage_option_group` collection shall be the same.

EXAMPLE 8 - Suppose a bar stock, represented by `product_definition` D, can be cut twice so as to create three `relating_product_definition` products, X, Y, and Z. The `make_from_usage_option_group` representing this situation would aggregate three `make_from_usage_options` as shown in table 1:

Table 1 - Example 8 Make\_from\_usage\_option\_group

id	relating_product_definition	related_product_definition	quantity
1	X	D	1
2	Y	D	1
3	Z	D	1

It is possible to use a product to make more than one combination of other products.

EXAMPLE 9 - The bar stock `product_definition` D may also be cut to produce two `product_definitions` X and one `product_definition` T. In this case the corresponding `make_from_usage_option_group` would aggregate the following `make_from_usage_options`, shown in table 2:

Table 2 - Example 9 Make\_from\_usage\_option\_group

id	relating_product_definition	related_product_definition	quantity
2	X	D	2
1	T	D	1

A single `make_from_usage_option` instance may be part of multiple `make_from_usage_option_groups`.

EXPRESS specification:

```

*)
ENTITY make_from_usage_option_group;
  members : SET [2:?] OF make_from_usage_option;
WHERE
  WR1: SIZEOF (QUERY (example <* members |
    example.related_product_definition
    ::= members[1].related_product_definition)) =SIZEOF(members);
END_ENTITY;
(*

```

Attribute definitions:

**members:** a collection of **make\_from\_usage\_option** instances whose **relating\_product\_definition** instances can be made from the same **related\_product\_definition**.

Formal propositions:

**WR1:** All instances of **members** shall share the same instance of **product\_definition** for their respective **related\_product\_definition**.

### 4.3.5 assembly\_component\_usage

The **assembly\_component\_usage** relates a constituent to its assembly. The **assembly\_component\_usage** entity is a subtype of the **product\_definition\_usage** entity that establishes a relationship between **product\_definitions** within one of the following three product structures:

- bill-of-material (BOM) structure;
- parts list structure;
- promissory use structure.

The **assembly\_component\_usage** entity has four subtypes specified in this part of ISO 10303:

- **quantified\_assembly\_component\_usage**;
- **next\_assembly\_usage\_occurrence**;
- **specified\_higher\_usage\_occurrence**;
- **promissory\_usage\_occurrence**.

The **quantified\_assembly\_component\_usage** represents the relationship between a constituent and an assembly where, for discrete constituents, several occurrences of the constituent are represented by the single constituent and a quantity representing the number of occurrences of it. The quantity represents a unit of measure other than a unitless number for non-discrete constituents. The **next\_assembly\_usage\_occurrence** represents a relationship between a component and its immediate assembly in a product structure. The **specified\_higher\_usage\_occurrence** shall be used to represent the explicit relationship

between a descendent component and any ancestor higher-level assembly. The **promissory\_usage\_occurrence** shall be used to represent intended relationships between a lower-level constituent and a higher-level assembly, when intermediate constituents and their relationships are yet undetermined.

In a BOM structure, **product\_definition** entities represent nodes and **next\_assembly\_usage\_occurrence** or **quantified\_assembly\_component\_usage** entities represent links.

In a parts list tree structure, a **product\_definition** entity represents the root node. The **next\_assembly\_usage\_occurrence** entities represent nodes at each intermediate level of the structure. The **specified\_higher\_usage\_occurrence** entities enable links to higher-levels of the structure.

In a promissory use structure, **product\_definition** entities represent nodes, and **promissory\_usage\_occurrence** entities represent links between the nodes.

NOTE 1- In order to allow for extensions to this entity in other parts of ISO 10303, the SUPERTYPE clause of this entity is only partially explicit. If the subtypes defined in this part of ISO 10303 were a complete categorization, the SUPERTYPE clause of this entity could properly read:

```
SUPERTYPE OF (quantified_assembly_component_usage ANDOR
               ONEOF (next_assembly_usage_occurrence,
                    specified_higher_usage_occurrence,
                    promissory_usage_occurrence))
```

EXPRESS specification:

```
*)
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;
(*
```

Attribute definitions:

**SELF\product\_definition\_relationship.relying\_product\_definition:** an assembly for which the **related\_product\_definition** is its constituent.

**SELF\product\_definition\_relationship.related\_product\_definition:** a constituent for which the **relying\_product\_definition** is its parent assembly.

**reference\_designator:** identifier for the **assembly\_component\_usage**, in addition to the **id** attribute inherited from the **product\_definition\_relationship**.

NOTE 2- The reference designator attribute may be constrained to be unique by an application protocol.

### 4.3.6 quantified assembly component usage

The **quantified\_assembly\_component\_usage** establishes the relationship between an assembly and one of its constituents when there is a need to specify the quantity of the child constituent used in the assembly.

NOTE - Generally, for production planning or material planning purposes, several occurrences of a constituent are lumped together, and a quantity is specified to account for the several occurrences. A typical example would be the specifying of an occurrence of a rivet used for joining aeroplane structures and denoting the number of such rivets used on the entire aeroplane. If each of the occurrences of the rivets used is to be specified, then the **next\_assembly\_usage\_occurrence** entity may be used. As many instances of the **next\_assembly\_usage\_occurrence** as the number of occurrences of the rivets will exist in this case.

EXPRESS specification:

```
*)
ENTITY quantified_assembly_component_usage
  SUBTYPE OF (assembly_component_usage);
  quantity : measure_with_unit;
END_ENTITY;
(*
```

Attribute definitions:

**SELF\product\_definition\_relationship.relatng\_product\_definition:** an assembly for which the **related\_product\_definition** is its constituent, and where the quantity of the constituent is specified.

**SELF\product\_definition\_relationship.related\_product\_definition:** a constituent for which the **relating\_product\_definition** is its parent assembly, and where the quantity of the constituent is specified.

**quantity:** a measure of how many or how much of the constituent is used in the assembly.

Informal propositions:

**IP1:** If **quantified\_assembly\_component\_usage.quantity.value\_component** is represented by a NUMBER then its value shall be greater than 0.

### 4.3.7 next\_assembly\_usage\_occurrence

The **next\_assembly\_usage\_occurrence** is the relationship between a child constituent and its immediate parent assembly in a product structure. It represents the use of individual occurrences of constituents. The use of the same constituent may be represented by another distinct **next\_assembly\_usage\_occurrence** instance for the purpose of assigning a position and orientation for the constituent.

#### NOTES

1 - This position and orientation information is not provided by any of the entities of this part of ISO 10303 but by means of using the appropriate capabilities defined in ISO 10303-42 and ISO 10303-43. The representation of a constituent occurrence in an assembly may be linked to mechanisms in ISO 10303-42 and ISO 10303-43 by means of a **product\_definition\_shape** entity defined in ISO 10303-41. The position and orientation of a constituent with respect to its assembly would be computed using a **transformation** defined in ISO 10303-43.

2 - An application algorithm can derive an indented parts list for a product by sequentially tracing through a structure of **next\_assembly\_usage\_occurrence** instances. A similar algorithm can be used to calculate the position and orientation of each occurrence of every constituent relative to its higher-level assemblies within a BOM.

3 - When this entity is used to establish the location of a constituent relative to its assembly, it is referenced by a **product\_definition\_shape** entity as defined in ISO 10303-41.

EXPRESS specification:

```
*)
ENTITY next_assembly_usage_occurrence
  SUBTYPE OF (assembly_component_usage);
END_ENTITY;
(*
```

Attribute definitions:

**SELF\product\_definition\_relationship.relatng\_product\_definition:** an assembly for which the **related\_product\_definition** is its immediate constituent.

**SELF\product\_definition\_relationship.related\_product\_definition:** a constituent for which the **relating\_product\_definition** is its immediate parent assembly.

#### 4.3.8 specified\_higher\_usage\_occurrence

The **specified\_higher\_usage\_occurrence** specifies the relationship between a constituent and an assembly where the assembly is not the immediate parent for the constituent.

If a **specified\_higher\_usage\_occurrence** is specified, the entire path between the constituent and the assembly is also identified using successive instances of **specified\_higher\_usage\_occurrence**. Successive instances of **specified\_higher\_usage\_occurrence** identify all the intermediate constituent and assembly relationships that exist between the assembly and its constituent specified by the primary **specified\_higher\_usage\_occurrence**.

The relationship between the constituent and the assembly of the **specified\_higher\_usage\_occurrence** to be specified is captured by the relationship of the inherited attributes **relating\_product\_definition** and **related\_product\_definition**.

The two attributes (**upper\_usage** and **next\_usage**) within the primary instance of the entity **specified\_higher\_usage\_occurrence** will respectively specify the **next\_assembly\_usage\_occurrence** and an **assembly\_component\_usage** which together will provide the definition of the path from the constituent to the assembly for which the **specified\_higher\_usage\_occurrence** is being specified. To ensure that the **next\_assembly\_usage\_occurrence** and the **assembly\_component\_usage** together constitute the entire path desired for the **specified\_higher\_usage\_occurrence**, it is required that the instance of the **related\_product\_definition** attribute of the **assembly\_component\_usage** entity referenced by the **upper\_usage** be the same as the instance of the **relating\_product\_definition** attribute of the **next\_assembly\_usage\_occurrence** entity referenced by **next\_usage**. The attribute **related\_product\_definition** of the **next\_assembly\_usage\_occurrence** shall be the same instance of the attribute **related\_product\_definition** of the **specified\_higher\_usage\_occurrence** being specified. The attribute **relating\_product\_definition** of

the **assembly\_component\_usage** entity referenced by the attribute **upper\_usage** shall be the same instance as the attribute **relating\_product\_definition** of the **specified\_higher\_usage\_occurrence** being specified.

If the **assembly\_component\_usage** referenced by the attribute **upper\_usage** is not a **next\_assembly\_usage\_occurrence** it will be a **specified\_higher\_usage\_occurrence**. This **specified\_higher\_usage\_occurrence** shall have its attributes **upper\_usage** and **next\_usage** defined as described in the previous paragraph to specify further the path of the primary **specified\_higher\_usage\_occurrence**. This recursive specification shall continue until the attribute **upper\_usage** references an **assembly\_component\_usage** entity that is a **next\_assembly\_usage\_occurrence**. At this point, the primary **specified\_higher\_usage\_occurrence** is fully specified both in terms of its constituents/assembly relationship and the entire path between them.

In order to be able to completely specify a **specified\_higher\_usage\_occurrence** all the necessary **assembly\_component\_usage** instances shall have been defined.

The **specified\_higher\_usage\_occurrence** entity supports the representation of parts list tree structures. Typically, it is used to define portions of parts lists that contain a specific constituent within an assembly for which certain properties are to be associated.

NOTE 1 - Annex E contains examples of this entity and figures which depict the relationships between it and the other entities of this schema.

#### EXPRESS specification:

```

*)
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.relying_product_definition
      := upper_usage.relying_product_definition;
  WR3: SELF\product_definition_relationship.related_product_definition
      := next_usage.related_product_definition;
  WR4: upper_usage.related_product_definition :=
      next_usage.relying_product_definition;
  WR5: NOT ('PRODUCT_STRUCTURE_SCHEMA.PROMISSORY_USAGE_OCCURRENCE' IN
      TYPEOF(upper_usage));
END_ENTITY;
(*)

```

#### Attribute definitions:

**upper\_usage:** An **assembly\_component\_usage** that is the same instance of the attribute **relating\_product\_definition** as this **specified\_higher\_usage\_occurrence** and the same instance of the attribute **related\_product\_definition** as the **relating\_product\_definition** of the **next\_assembly\_usage\_occurrence** referenced by the attribute **next\_usage**.

**next\_usage:** A **next\_assembly\_usage\_occurrence** that is the same instance of the attribute **related\_product\_definition** as this **specified\_higher\_usage\_occurrence** and the same instance of the product definition referenced by the attribute **relating\_product\_definition** as the product definition referenced by the attribute **related\_product\_definition** of the attribute **upper\_usage**.

NOTE 2 - See figure E.6 for diagrams illustrating instances of these two attributes.

**SELF\product\_definition\_relationship.relying\_product\_definition:** the inherited attribute for the assembly product definition of the **specified\_higher\_usage\_occurrence**.

**SELF\product\_definition\_relationship.related\_product\_definition:** the inherited attribute for the constituent product definition of the **specified\_higher\_usage\_occurrence**.

Formal propositions:

**UR1:** The combination of the **upper\_usage** and **next\_usage** attributes shall be unique.

**WR1:** The instance of **specified\_higher\_usage\_occurrence** shall not be the same as the instance of **upper\_usage**.

**WR2:** The **relating\_product\_definition** (i.e., assembly) of the **specified\_higher\_usage\_occurrence** shall be the same instance product definition as **relating\_product\_definition** (i.e., assembly) for the **upper\_usage**.

**WR3:** The **related\_product\_definition** (i.e., constituent) of the **specified\_higher\_usage\_occurrence** shall be the same instance of product definition as the **related\_product\_definition** for the **next\_usage**.

**WR4:** The **related\_product\_definition** (i.e., constituent) for the **upper\_usage** shall be the same instance of **product\_definition** as the **relating\_product\_definition** (i.e., assembly) for the **next\_usage**.

**WR5:** The type of the **upper\_usage** attribute shall not be **promissory\_usage\_occurrence**.

### 4.3.9 promissory\_usage\_occurrence

The **promissory\_usage\_occurrence** is the intention to use the constituent **product\_definition** in an assembly **product\_definition**. It is used when the product structure is not completely defined. In such a situation, it is still possible to relate an assembly to a constituent to capture the intent that the constituent will be used.

EXPRESS specification:

```
*)
ENTITY promissory_usage_occurrence
  SUBTYPE OF (assembly_component_usage);
END_ENTITY;
(*
```

Attribute definitions:

**SELFproduct\_definition\_relationship.relatng\_product\_definition:** an assembly for which the **related\_product\_definition** is a constituent, and the details of the product structure are not completely defined.

**SELFproduct\_definition\_relationship.related\_product\_definition:** a constituent for which the **relating\_product\_definition** is an assembly, and the details of the product structure are not completely defined.

### 4.3.10 assembly\_component\_usage\_substitute

The **assembly\_component\_usage\_substitute** specifies that one constituent can be used as a substitute for another within a given assembly context.

The instance of the substitute constituent does not require the same spatial relationship or the same quantity. A substitute constituent does not require equivalent form, fit, and function of the constituent for which it is a substitute.

This entity defines one-way substitution only. Within a given context, if A is specified as a substitute for B, B is not assumed to be a substitute for A, unless explicitly stated so in another instance of the entity.

The **assembly\_component\_usage\_substitute** entity can be used to eliminate the re-identification of all higher-level assemblies when a new version of a lower-level constituent is created.

EXPRESS specification:

```

*)
ENTITY assembly_component_usage_substitute;
  name      : label;
  definition: text;
  base      : assembly_component_usage;
  substitute : assembly_component_usage;
UNIQUE
  UR1: base, substitute;
WHERE
  WR1: base.relatng_product_definition :=
      substitute.relatng_product_definition;
  WR2: base :<>: substitute;
END_ENTITY;
(*

```

Attribute definitions:

**base:** an **assembly\_component\_usage** for which the **substitute** may be used.

**substitute:** an **assembly\_component\_usage** which may be used for the **base**.

Formal Propositions:

**UR1:** The combination of the **base** and **substitute** attributes shall be unique.

**WR1:** The value of the **relating\_product\_definition** attribute of both the **base** and the **substitute** attributes shall be the same; i.e., they should refer to the same assembly **product\_definition**.

**WR2:** The **base** and **substitute** attributes shall not be the same instance.

```
*)
END_SCHEMA; -- product_structure
(*
```

## 5 Product concept schema

The following EXPRESS declaration begins the **product\_concept\_schema** and identifies the necessary external references.

EXPRESS specification:

```
*)
SCHEMA product_concept_schema;

  REFERENCE FROM support_resource_schema
    (text, label, identifier);

  REFERENCE FROM application_context_schema
    (product_concept_context);
(*
```

NOTE - The schemas referenced above can be found in the following parts of ISO 10303:

support_resource_schema	ISO 10303-41
application_context_schema	ISO 10303-41

### 5.1 Introduction

The subject of the **product\_concept\_schema** is the idea of a product as defined by customer needs, i.e., a product concept. A product concept may exist before a product has been defined. A product concept identifies a selection of product features or capabilities.

### 5.2 Fundamental concepts and assumptions

The following characteristics of a product concept are assumed:

- A product concept identifies a deliverable product as perceived by the customer. A product concept is often used to identify a selection of product features or capabilities.

- A product concept may be composed of several configuration items (see 6.3.1).
- A product concept may exist without a product or product version.

### 5.3 Product concept schema entity definitions: **product\_concept**

The **product\_concept** is the idea of a product as defined by customer needs. The **product\_concept** and its features may be identified as configuration items to control their manufacture.

NOTE 1 - A **product\_concept** will often correspond to the highest level item(s) manufactured by an organization for a customer. It may be characterized by a set of product features identified by the customers or derived from customers' needs. The definition of product concepts is often driven by marketing.

EXAMPLE 10 - If an organization manufactures cars and engines for cars, the cars will be represented by **product\_concept** instances. If another organization manufactures engines for cars, then the engines will be represented as **product\_concept** in that organization.

#### EXPRESS specification:

```

*)
ENTITY product_concept;
  id          : identifier;
  name        : label;
  description  : text;
  market_context : product_concept_context;
UNIQUE
  UR1: id;
END_ENTITY;
(*

```

#### Attribute definitions:

**id:** an identifier for a **product\_concept** instance.

EXAMPLE 11 - The **id** could be a sales model number.

**name:** a descriptive identifier for **product\_concept**.

**description:** a text description of the **product\_concept**.

NOTE 2 - This attribute can be used to convey the purpose, functionality, and selected features for the **product\_concept**.

**market\_context:** a market context in which the **product\_concept** is defined.

NOTE 3 - The values for this attribute will be defined in application protocols.

Formal propositions:

**UR1:** The value of the **id** attribute shall be unique.

```
*)
END_SCHEMA; -- product_concept_schema
(*
```

## 6 Configuration management schema

The following EXPRESS declaration begins the **configuration\_management\_schema** and identifies the necessary references.

EXPRESS specification:

```
*)
SCHEMA configuration_management_schema;

    REFERENCE FROM product_definition_schema
        (product_definition_formation,
         product_definition_effectivity);

    REFERENCE FROM product_property_representation_schema
        (relatives_of_product_definitions);

    REFERENCE FROM product_concept_schema (product_concept);

    REFERENCE FROM product_structure_schema
        (product_definition_usage);

    REFERENCE FROM support_resource_schema
        (text, label, identifier, bag_to_set);
(*
```

NOTE - The schemas referenced above can be found in the following parts of ISO 10303:

product_structure_schema	clause 4 of this part of ISO 10303
product_property_representation_schema	ISO 10303-41
product_concept_schema	clause 5 of this part of ISO 10303
product_definition_schema	ISO 10303-41
support_resource_schema	ISO 10303-41
effectivity_schema	ISO 10303-41

### 6.1 Introduction

The subject of the **configuration\_management\_schema** is the establishment of the association of appropriate versions of products to plan the manufacture of products. The product that is planned for manufacture is referred to as the **configuration\_item**. It is usually visible to customers of the organization that does the configuration management.

This schema also provides the capability to record the associations established to enable the tracking of the associations.

The functionality of configuration management is achieved using the entities **configuration\_item**, **configuration\_design**, and **configuration\_effectivity**.

The configuration management schema of this part of ISO 10303 contains the following concepts:

- The identification of the **configuration\_items** and the respective **product\_concepts** of which they form a constituent.
- The identification of a **product\_definition\_formation** that realizes the **configuration\_item**.
- The association of the appropriate versions of a product to build a **configuration\_item**. This association is referred to as **configuration\_effectivity**.
- There are three ways to apply **configuration\_effectivity**. They are:
  - a) **serial\_numbered\_effectivity**, where the **configuration\_effectivity** is based on serial numbered instances of manufactured products.
  - b) **dated\_effectivity**, where the **configuration\_effectivity** is based on dates when instances of the product are manufactured.
  - c) **lot\_effectivity**, where the **configuration\_effectivity** is based on instances of lots of products manufactured.

NOTE - The definitions of **serial\_numbered\_effectivity**, **dated\_effectivity**, and **lot\_effectivity** are given in ISO 10303-41, clause 21.

## 6.2 Fundamental concepts and assumptions

The following are the fundamental concepts related to configuration management:

- Configuration management within an organization may be applied to products and parts of products to be manufactured by that organization. Configuration of vendor-supplied parts supplied by a vendor is not included;

EXAMPLE 12 - A specific motor is supplied by a vendor. The purchasing organization does not consider individual parts of the motor for configuration management.

- An organization determines which products are to be under its configuration management control. These products become the configuration items of the organization. These are higher-level functional elements which act as the focal points for managing the effectivity of lower-level constituents.

## 6.3 Configuration management schema entity definitions

### 6.3.1 configuration\_item

A **configuration\_item** is used to manage the composition of constituents for actual units of manufacture. All configuration management within an organization is done using these **configuration\_items**.

A **configuration\_item** may be an entire **product\_concept** or some portion thereof.

A **configuration\_item** can be established prior to the existence of a corresponding **product\_definition\_formation**.

The association between a **configuration\_item** and a corresponding **product\_definition\_formation** is established using a **configuration\_design**.

A **configuration\_item** is associated with a single **product\_concept**.

EXPRESS specification:

```

*)
ENTITY configuration_item;
  id          : identifier;
  name       : label;
  description : OPTIONAL text;
  item_concept : product_concept;
  purpose    : OPTIONAL label;
UNIQUE
  UR1: id;
END_ENTITY;
(*

```

Attribute definitions:

**id:** an identifier for the **configuration\_item** instance.

**name:** a descriptive identifier for a **configuration\_item**.

**description:** a text description of the **configuration\_item**.

NOTE - The text description typically identifies the specification and function of the **configuration\_item**.

**item\_concept:** a **product\_concept** associated with the **configuration\_item**.

**purpose:** a descriptive label providing a reason to create the **item\_concept**.

Formal propositions:

**UR1:** The value of the **id** attribute shall be unique.

### 6.3.2 configuration\_design

The **configuration\_design** relates a configuration-controlled item and a product design intended to implement that item. Thus, the **configuration\_design** entity represents the association of a **configuration\_item** with a **product\_definition\_formation** to specify that the corresponding design is for the specific **configuration\_item**.

NOTE - Organizations establish this association before any actual units are planned and before any details of the design have been established.

EXPRESS specification:

```

*)
ENTITY configuration_design;
  configuration : configuration_item;
  design       : product_definition_formation;
UNIQUE
  UR1: configuration, design;
END_ENTITY;
(*

```

Attribute definitions:

**configuration:** a **configuration\_item** which specifies a **product\_definition\_formation** as a candidate for manufacturing actual units associated with the **configuration\_item**.

**design:** a **product\_definition\_formation** representing a design which is a candidate for use in manufacturing actual units associated with the **configuration** attribute.

Formal propositions:

**UR1:** The combination of the **configuration** attribute and the **design** attribute shall be unique.

### 6.3.3 configuration\_effectivity

The **configuration\_effectivity** defines common effectivity attributes for items under configuration control. The **configuration\_effectivity** entity is used by an organization to specify effectivity of **product\_definition\_usages**.

EXAMPLE 13 - A user may want to specify that certain **product\_definition\_usages** are to be effective for a **configuration\_item**. A 200 HP engine is to be effective starting on a certain date. This information is captured prior to any production plans existing for the 200 HP engine in a **configuration\_effectivity** entity.

NOTE - Other subtypes of **effectivity** are defined in ISO 10303-41. An instance of this entity may also be an instance of either a **serial\_numbered\_effectivity**, a **dated\_effectivity**, or a **lot\_effectivity**.

EXPRESS specification:

```

*)
ENTITY configuration_effectivity
  SUBTYPE OF (product_definition_effectivity);
  configuration : configuration_design;
UNIQUE
  UR1: configuration,
      usage,
      id;
WHERE
  WR1: SELF\product_definition_effectivity.usage.relativing_product_definition IN
      USEDIN (SELF.configuration.design,
              'PRODUCT_DEFINITION_SCHEMA.' +
              'PRODUCT_DEFINITION.FORMATION') +
      relatives_of_product_definitions ( bag_to_set (USEDIN
              (SELF.configuration.design,
              'PRODUCT_DEFINITION_SCHEMA.' +
              'PRODUCT_DEFINITION.FORMATION') ) ,
              'PRODUCT_STRUCTURE_SCHEMA.' +

```

```
        'PRODUCT_DEFINITION_USAGE');  
WR2: 'PRODUCT_STRUCTURE_SCHEMA.PRODUCT_DEFINITION_USAGE' IN  
      TYPEOF (SELF\product_definition_effectivity.usage);  
END_ENTITY;  
(*
```

Attribute definitions:

**configuration:** a **configuration\_design** for which the **configuration\_effectivity** is effective.

Formal propositions:

**UR1:** The combination of the **configuration** attribute, the **usage** attribute, and the **identification** attribute shall be unique.

**WR1:** The **usage** shall refer to a constituent of the **product\_definition\_formation** referenced by the **configuration**.

**WR2:** The **usage** shall be a **product\_definition\_usage**.

```
*)  
END_SCHEMA; -- configuration_management_schema  
(*
```

STANDARDSISO.COM : Click to view the full PDF of ISO 10303-44:1994

## Annex A (normative)

### Short names of entities

Table A.1 provides the short names of entities specified in this part of ISO 10303. Requirements on the use of the short names are found in the implementation methods included in ISO 10303.

**Table A.1 - Short names of entities**

Entity Name	Short Name
ALTERNATE_PRODUCT_RELATIONSHIP	ALPRRL
ASSEMBLY_COMPONENT_USAGE	ASCMUS
ASSEMBLY_COMPONENT_USAGE_SUBSTITUTE	ACUS
CONFIGURATION_DESIGN	CNFDSG
CONFIGURATION_EFFECTIVITY	CNFEFF
CONFIGURATION_ITEM	CNFITM
MAKE_FROM_USAGE_OPTION	MFUO
MAKE_FROM_USAGE_OPTION_GROUP	MFUOG
NEXT_ASSEMBLY_USAGE_OCCURRENCE	NAUO
PRODUCT_CONCEPT	PRDCNC
PRODUCT_DEFINITION_USAGE	PRDFUS
PROMISSORY_USAGE_OCCURRENCE	PRUSOC
QUANTIFIED_ASSEMBLY_COMPONENT_USAGE	QACU
SPECIFIED_HIGHER_USAGE_OCCURRENCE	SHUO

## Annex B (normative)

### Information object registration

#### B.1 Document identification

In order to provide for unambiguous identification of this International Standard as an information object in an open system, it is assigned the object identifier

{ iso standard 10303 part(44) version(1) }

The meaning of this value is defined in ISO/IEC 8824-1 and is further described in ISO 10303-1.

#### B.2 Schema identification

ISO 10303-1 further describes how ISO/IEC 8824-1 can be used to identify individual schemas. This part of ISO 10303 contains three such schemas, and each is assigned a unique object identifier to provide for unambiguous identification of the schema in an open information system.

##### B.2.1 product\_structure\_schema identification

The `product_structure_schema` (see clause 4) is assigned the object identifier

{ iso standard 10303 part(44) version(1) object(1) product-structure-schema(1) }

##### B.2.2 product\_concept\_schema identification

The `product_concept_schema` (see clause 5) is assigned the object identifier

{ iso standard 10303 part(44) version(1) object(1) product-concept-schema(2) }

##### B.2.3 configuration\_management\_schema identification

The `configuration_management_schema` (see clause 6) is assigned the object identifier

{ iso standard 10303 part(44) version(1) object(1) configuration-management-schema(3) }

**Annex C**  
(informative)

**Computer-interpretable listings**

This annex provides a listing of the short names with their corresponding entity names and EXPRESS constructs specified in this part of ISO 10303. No text or annotation is included. This annex is provided only in computer-interpretable form.

NOTE - The information provided in computer-interpretable form is informative; the normative text is that contained in the body of this part of ISO 10303.

STANDARDSISO.COM : Click to view the full PDF of ISO 10303-44:1994

**Annex D**  
(informative)

**Graphical model representations**

The EXPRESS-G representation for the schemas defined in subclauses 4 through 6 of this part of ISO 10303 are provided in the following figures.

STANDARDSISO.COM : Click to view the full PDF of ISO 10303-44:1994

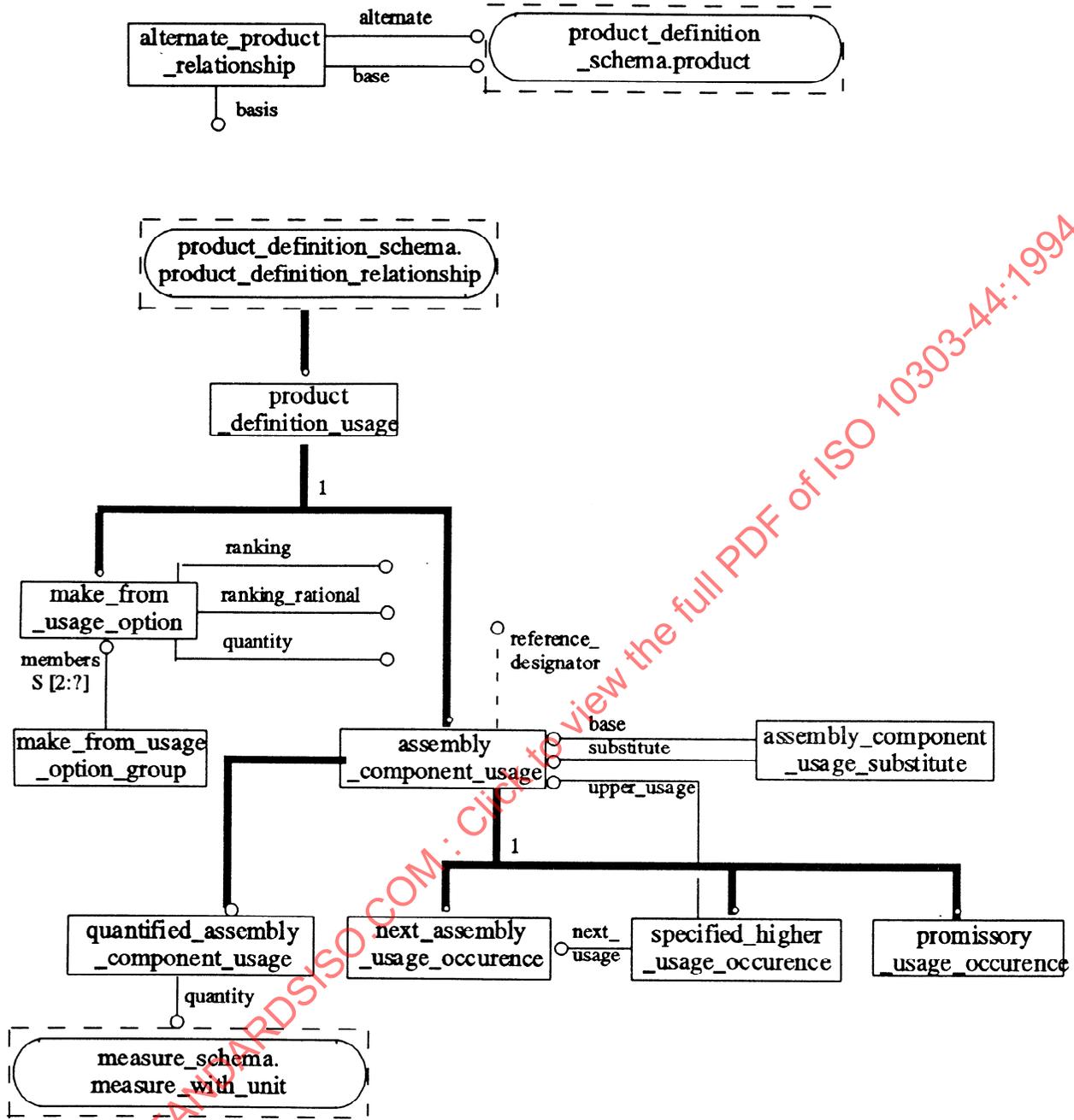


Figure D.1: product\_structure\_schema - EXPRESS-G diagram 1 of 1

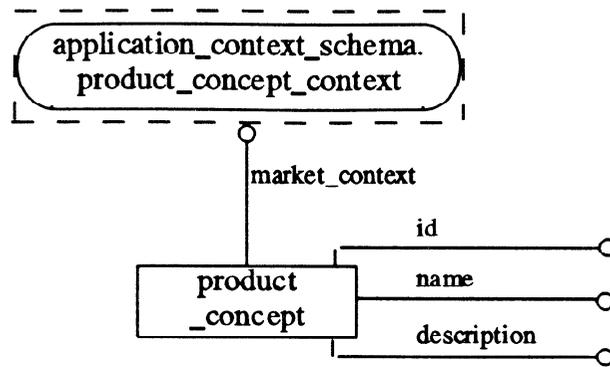


Figure D.2: product\_concept\_schema - EXPRESS-G diagram 1 of 1

STANDARDSISO.COM : Click to view the full PDF of ISO 10303-44:1994

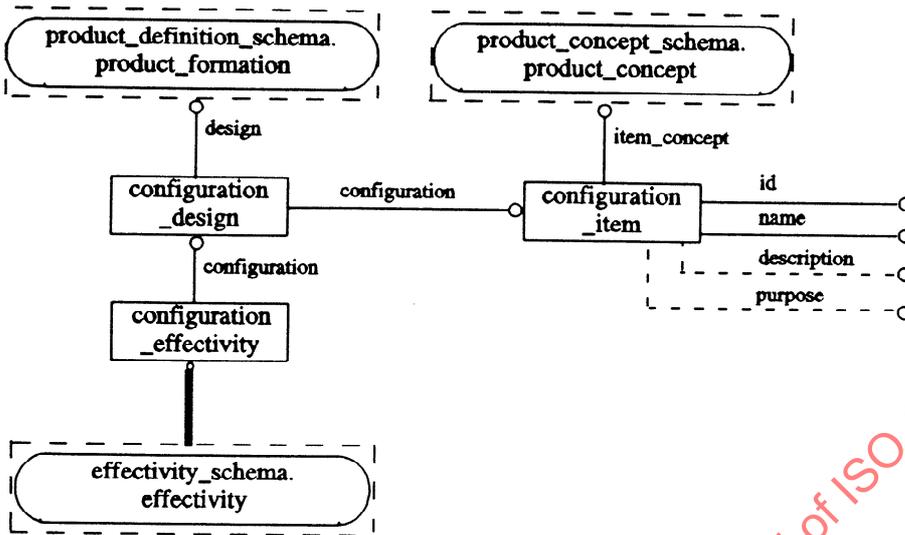


Figure D.3: configuration\_management\_schema - EXPRESS-G diagram 1 of 1

STANDARDSISO.COM : Click to view the full PDF of ISO 10303-44:1994

## **Annex E**

(informative)

### **Examples**

#### **E.1 Introduction**

Product structure reports are used by industry to describe the composition and structure of a product. Most products are assembled from other assemblies and components. A product structure report provides details about what subassemblies and components are used.

NOTE - See definitions in clause 3.

In this annex, examples of product structure reports are provided. In addition, the possible use of this part of ISO 10303 to generate these reports is outlined. The focus is on the **product\_structure** schema.

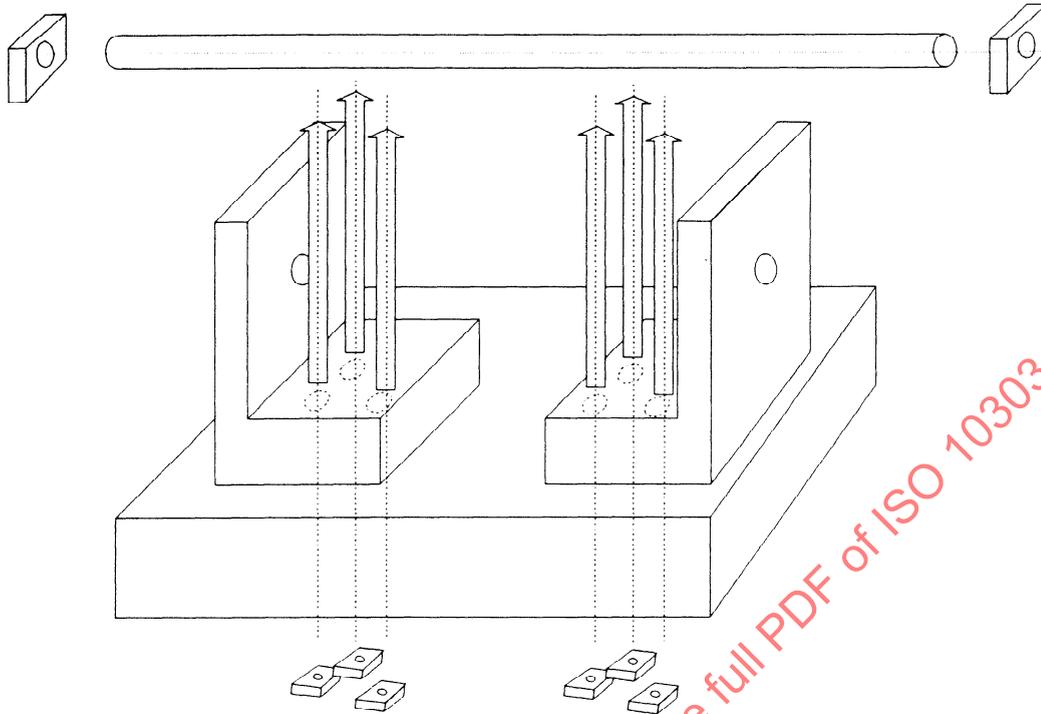
#### **E.2 Example of a mechanical assembly product**

In this section, a hypothetical mechanical assembly product called the "part 44 example product" is described, both in diagrams and in words. This assembly forms the basis for explaining different types of product structure reports, e.g., BOM and parts list, in the next section. This product is modelled using the various entities defined earlier in this part of ISO 10303. See figure E.1 for the 3D drawing of the "part 44 example product", and figure E.2 for a diagrammatic representation of the product structure.

The "part 44 example product" represents a simple mechanical assembly consisting of three major constituents: a large metal plate forming the base for attaching two identical L-bracket assemblies. Each of the two L-bracket assemblies is bolted to the base using three nuts and bolts and using three holes drilled into the plate. The L-brackets themselves have a hole drilled in them to support a rod assembly which consists of a threaded rod and nuts attached at each end of the rod. The nuts used on the threaded rod and in the L-bracket assembly are of the same size. This product design has a structure consisting of a number of assemblies, some of which also contain assemblies, i.e., sub-assemblies, as well as a number of individual components.

#### **E.3 Examples of product structure reports**

Product structure reports can decompose an assembly to many levels of details. The details can cover the degree of decomposition, the quantities of subassemblies and components used, the position and orientation of the subassemblies and components, and the products used to make other products. The levels of detail include:



**Figure E.1 - Part 44 example product**

– degree of decomposition;

a) single-level: a single assembly and its immediate constituent sub-assemblies or components;

b) multi-level: one or more assemblies in which at least one subassembly is further decomposed;

– type of decomposition;

a) extended: where the products used to make one or more components are identified;

b) exploded: one assembly that is decomposed one level at a time with all subassemblies described ultimately in terms of their components;

c) flattened: one assembly described in terms of its components; no subassemblies are described;

d) numeric: for a given level of decomposition, each subassembly and component is described only once with its quantity or amount;

e) occurrence: for a given level of decomposition, every subassembly and component is described, even if it is identical with another subassembly or component;

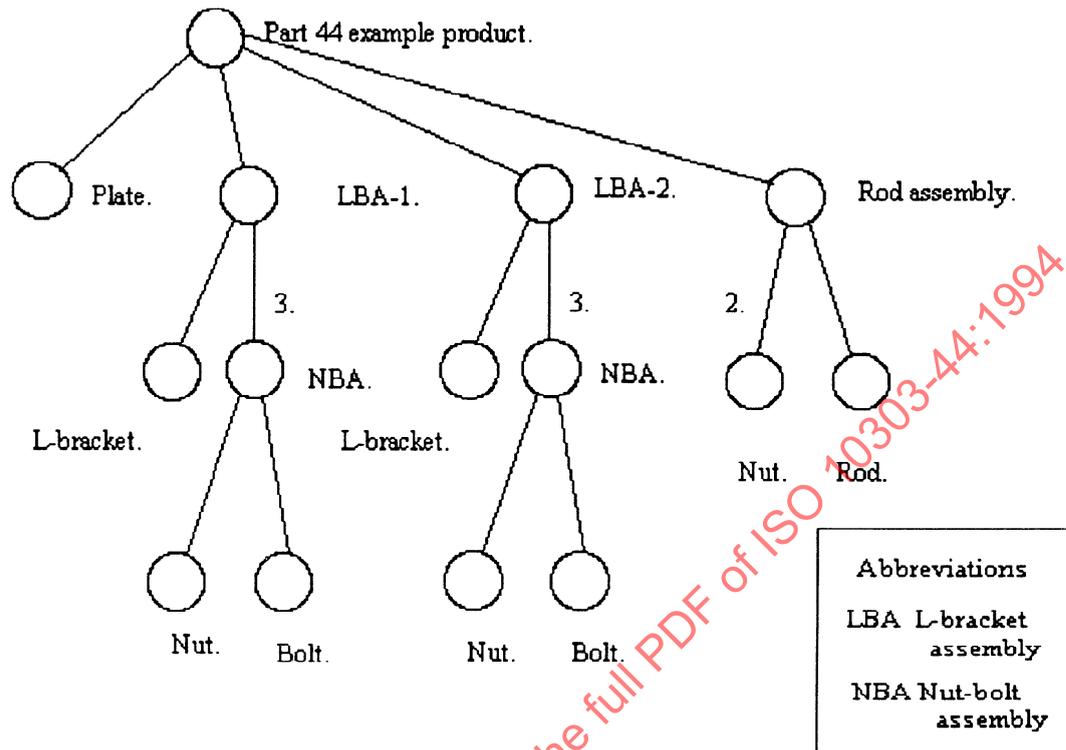


Figure E.2 - Part 44 example product: graphical product structure

- f) labelled occurrence: an occurrence decomposition in which each subassembly and component is labelled;
  - g) positioned: each constituent is associated with its position and orientation relative to the coordinate system of its immediate assembly.
- style of decomposition: each level of decomposition may be indented with respect to the previous level.

Examples of typical reports having a mixture of the characteristics enumerated above are now presented using the "part 44 example product".

### E.3.1 Example: A numeric, multi-level, indented product structure report

Each assembly, subassembly, and component is placed on a new line. A numeric value, indicating the count or amount of the constituent, may be present. The nesting of the levels is indicated by indentation. Constituents at the same level have the same indentation. This report shows that the "part 44 example product" has two L-bracket assembly units and that each unit contains three nut-bolt assemblies. The rod assembly contains two nuts and a rod.

## Part 44 Example Product

Plate

L-bracket assembly (2)

L-bracket

Nut-bolt assembly (3)

Nut

Bolt

Rod-assembly

Rod

Nut (2)

**E.3.2 Example: A labelled occurrence, multi-level, indented product structure report**

In this report, the structure is decomposed to the lowest level of the product structure. A label is assigned to each subassembly and component, down to the lowest level, e.g., Nut-1-1. With this form, all subassemblies and components can be referenced uniquely within the complete product structure and can be distinguished from one another. One can retrieve Nut-2-1 (the first nut on the second L-bracket assembly) and differentiate it from Nut-rod-2 (the second nut on the rod assembly.)

## Part 44 Example Product

Plate-1: Plate

L-bracket-assembly-1: L-bracket assembly

L-bracket-1-1: L-bracket

Nut-bolt-assembly-1-1: Nut-bolt assembly

Nut-1-1: Nut

Bolt-1-1: Bolt

Nut-bolt-assembly-1-2: Nut-bolt assembly

Nut-1-2: Nut

Bolt-1-2: Bolt

Nut-bolt-assembly-1-3: Nut-bolt assembly

Nut-1-3: Nut

Bolt-1-3: Bolt

L-bracket-assembly-2: L-bracket assembly

L-bracket-2-1: L-bracket

Nut-bolt-assembly-2-1: Nut-bolt assembly

Nut-2-1: Nut

Bolt-2-1: Bolt

Nut-bolt-assembly-2-2: Nut-bolt assembly

Nut-2-2: Nut

Bolt-2-2: Bolt

Nut-bolt-assembly-2-3: Nut-bolt assembly

Nut-2-3: Nut

Bolt-2-3: Bolt

Rod-assembly-1: Rod-assembly

Rod-1: Rod

Nut-Rod-1: Nut

Nut-Rod-2: Nut

### E.3.3 Example: A numeric, multi-level, indented, extended product structure report

This report includes the number of each constituent. This report is extended because it references the product from which some of the components are made; e.g., the sheet metal used to make the plate.

#### Part 44 Example Product

- Plate
  - Sheet metal
- L-bracket assembly (2)
  - L-bracket
    - Sheet metal
  - Nut-bolt assembly (3)
    - Nut
    - Bolt
- Rod-assembly
  - Rod
    - Bar stock
  - Nut (2)

## E.4 BOM and parts list data structures

In order for a compliant application to generate the information normally expected in the various product structure reports in the last section, it will need the information to be represented in data structures for physical storage. This example introduces two data structures that may be used to represent this information. They are the BOM data structure and the Parts List data structure, which are defined in E.4.1 and E.4.2.

### E.4.1 BOM data structure

The *BOM data structure* is used to represent the assembly aspects of product structure. It represents the assembly structure by using only the **product\_definition** and **product\_definition\_usage** entities, and only one instance of a single product definition is included for each kind of product that participates in the structure. Although there are eight nuts specified in the "part 44 example product" above, the BOM data structure representing this structure would only have one **product\_definition** entity instance representing a nut and this single entity represents all eight bolts. (See figure E.3)

The BOM data structure does support the representation of multiple relationships between two specific **product\_definitions**. This representation uses multiple **product\_definition\_usage** entities whose attributes point to the two instances of **product\_definitions**. To indicate that there are three nut-bolt assemblies in the "part 44 example product", the data structure holds only one instance of the nut-bolt assembly, but three **product\_definition\_usage** entity instances connecting this one instance to the parent assembly.

A ramification of this BOM data structure is that each **product\_definition** entity's components are indistinguishable within any assembly which includes it. Thus, although the L-bracket assembly contains three nut-bolt assemblies, each in turn containing a nut and a bolt, there is no way to distinguish any of the individual bolts or nuts within the L-bracket assembly.

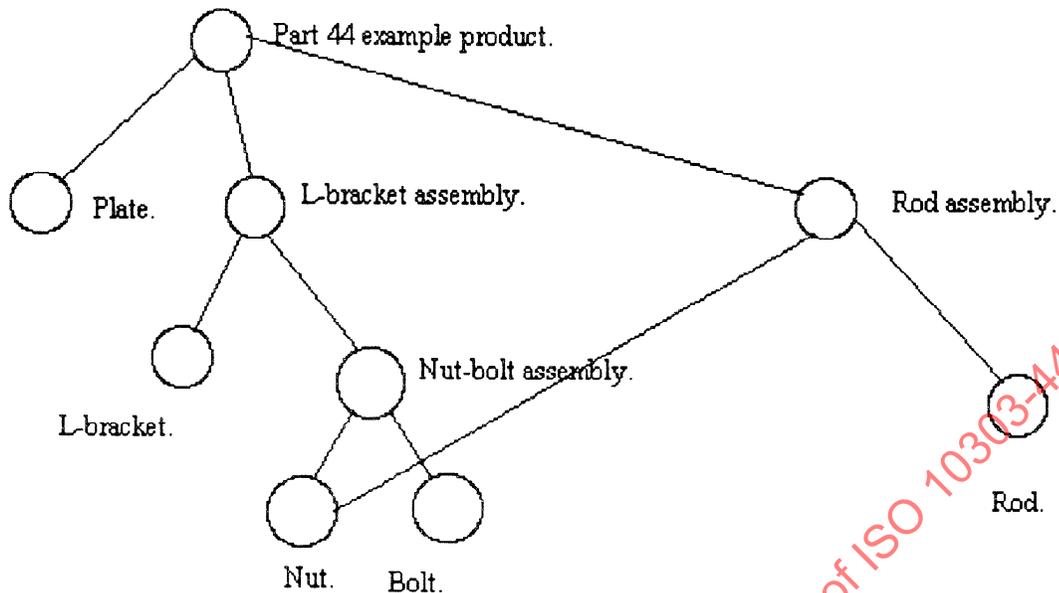


Figure E.3 - BOM data structure for Part 44 example product

#### E.4.2 Parts list data structure

The parts list data structure individualizes the relationship between lower-level parts of the product structure with higher-level assemblies in which they are contained.

This identification uses multiple instances of the same product within the data structure at a single time. When there are two L-bracket assemblies within the "part 44 example product", the parts list data structure would include two different L-bracket instances which then permits the unique identification of the respective components of the L-bracket assemblies within the context of the overall "part 44 example product". This parts list data structure can be seen in figure E.4, where the L-bracket assemblies, and the nut-bolt assemblies are individually represented.

#### E.5 Representation of product structure data structures using Part 44 entities

BOM and parts list data structures are mapped into a combination of **product\_definition**, and **product\_definition\_usage** (or subtypes) entities. For pure BOM data structures, only the **quantified\_assembly\_component\_usage**, **next\_assembly\_usage\_occurrence** or **make\_from\_usage\_option** subtypes of **product\_definition\_usage** need to be used, since the lower-level components are used only as "black boxes" to the parent assembly component. The "part 44 example product" does have two L-bracket assemblies, but from a BOM data structure point of view, both of these L-bracket assemblies are the same and it is not possible to distinguish their internal structure.

See figure E.5 for a subset of the BOM data structure. Figure E.5 and figure E.6 define the symbology for the entities in this part of ISO 10303 which is used for the remainder of the figures in this annex. Rectangles represent the **product\_definition** entities, and circles represent entities whose type is some

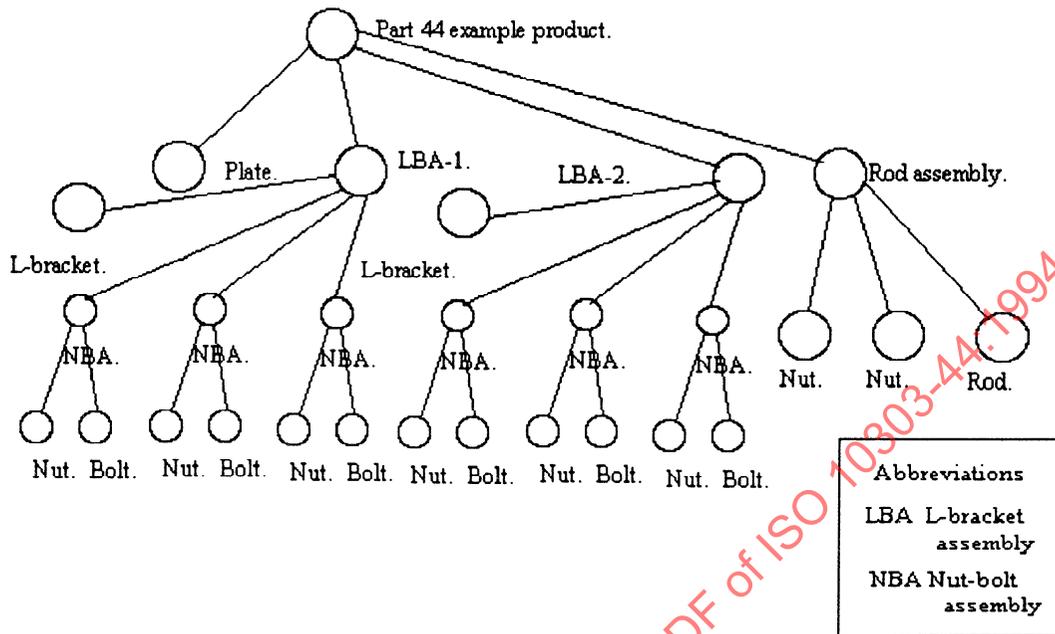


Figure E.4 - Parts list data structure for Part 44 example product

subtype of the **product\_definition\_usage** entity. The circles are labelled with the first letter of the specific subtype. In figure E.5, there are instances of both the **next\_assembly\_usage\_occurrence** (label is "N") and **make\_from\_usage\_option** (label is "M") subtypes of **product\_definition\_usage**.

For parts list data structures, however, the additional **specified\_higher\_usage\_occurrence** subtype of **product\_definition\_usage** is used in order to make explicit the specific usage of a particular lower-level component within the overall context of a higher assembly. A parts list data structure can specifically identify a sub-component of a lower assembly as distinguished from a similar sub-component of another lower assembly. This is possible because of the additional context information that is provided by the **specified\_higher\_usage\_occurrence** subtype of **product\_definition\_usage**, i.e., which specific immediate parent assembly and which specific top-level assembly. For instance, it is possible to identify the bolt on the third nut-bolt assembly of the second L-bracket assembly.

In figure E.6, the additional information needed to represent the parts list data structure is represented by the additional subtype of **product\_definition\_usage**, namely the **specified\_higher\_usage\_occurrence** entity, represented symbolically by the circle with label "S". Here, the **specified\_higher\_usage\_occurrence** entity allows us to reference one specific nut-bolt assembly on only one of the two L-bracket assemblies. This is accomplished by having the **specified\_higher\_usage\_occurrence** entity relate to only one of the two usages of the L-bracket assembly as represented by the **next\_assembly\_usage\_occurrence** entity.

Note that even though the parts list data structure is implemented using the entities of this part of ISO 10303, it is done without having to create individual instances of the lower-level components. Rather, the additional identifying information is supplied through the existence of the intermediate **specified\_higher\_usage\_occurrence** and **next\_assembly\_usage\_occurrence** entities. Figure E.7 presents the

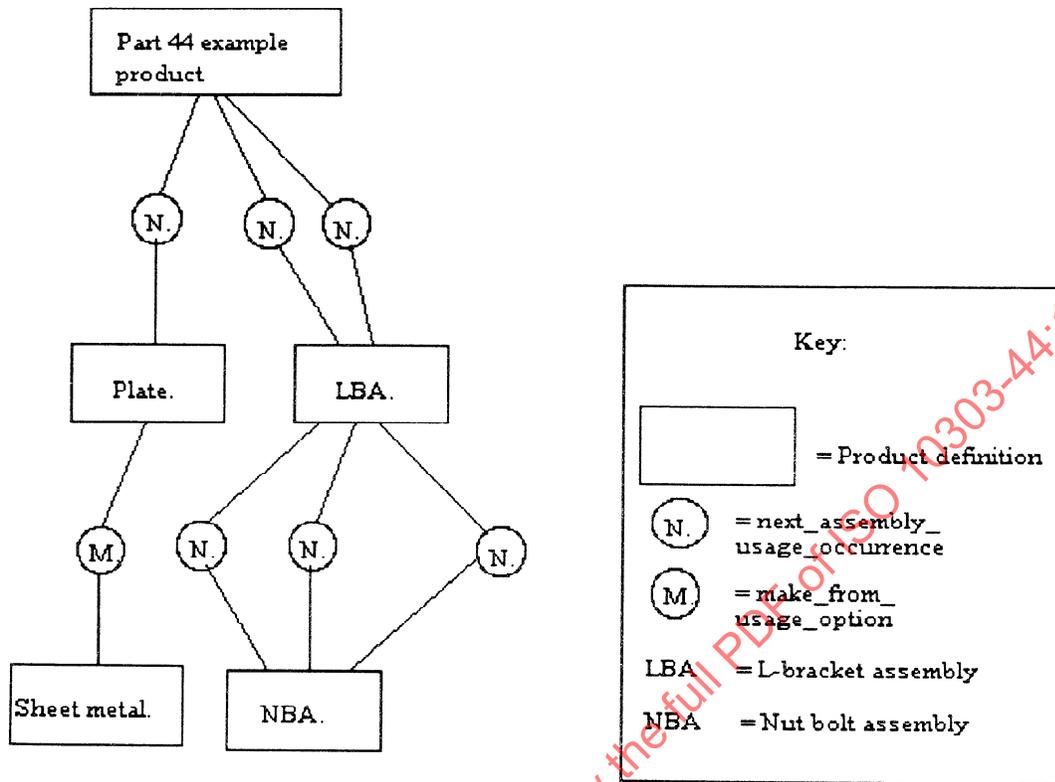


Figure E.5 - BOM data structure examples using Part 44 entities

complete model for the "part 44 example product" in a BOM data structure using this part of ISO 10303. Note that individual **next\_assembly\_usage\_occurrence** entities connect the multiple uses of a component within a higher assembly, rather than using only the count attribute of a single **next\_assembly\_usage\_occurrence** entity.

In figure E.9, figure E.9 and figure E.9, the representation of a number of lower-level components (Nuts, Bolts, Assemblies) is presented within the context of higher-level assemblies as they would be represented using the entities of this part of ISO 10303 entity. In each of these figures, the original BOM data structure is included within the large box; the additional entities needed to support the parts list aspects of the examples are placed outside the box, with their attributes referencing the original BOM.

In particular, figure E.9 highlights the use of a **specified\_higher\_usage\_occurrence** entity which represents how the Rod is used within the "part 44 example product" as specifically included in a Rod Assembly entity. This is an example of a two-level component inclusion and as such, both the **upper** and **next** attributes of the **specified\_higher\_usage\_occurrence** entity refer to **next\_assembly\_usage\_occurrence** components in the BOM structure.

In figure E.9, two different cases are represented. In one case, two **specified\_higher\_usage\_occurrence** entities, located towards the top of the diagram, are included to represent the two nuts which are specifically used within the Rod assembly part of the "part 44 example product". Note that the extra relationship lines to the top-level "part 44 example product" definition and the low-level Nut definition