



中华人民共和国国家标准

GB/T 16656.44—2008/ISO 10303-44:2000
代替 GB/T 16656.44—1999

工业自动化系统与集成 产品数据表达与交换 第 44 部分：集成通用资源：产品结构配置

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

(ISO 10303-44:2000, IDT)

2008-10-07 发布

2009-04-01 实施

中华人民共和国国家质量监督检验检疫总局 发布
中国国家标准化管理委员会

目 次

前言	I
引言	III
1 范围	1
2 规范性引用文件	1
3 术语、定义和缩略语	2
4 产品结构模式	4
4.1 概述	4
4.2 基本概念与假设	5
4.3 产品结构模式的实体定义	6
5 产品概念	16
5.1 概述	16
5.2 基本概念与假设	16
5.3 产品概念实体定义	17
6 配置管理	20
6.1 概述	21
6.2 基本概念与假设	22
6.3 配置管理类型定义、配置设计项	22
6.4 配置管理实体定义	22
附录 A (规范性附录) 实体短名	26
附录 B (规范性附录) 信息对象注册	27
附录 C (资料性附录) 计算机可解释的清单	28
附录 D (资料性附录) EXPRESS-G 图	29
附录 E (资料性附录) 范例	32
参考文献	48

前 言

GB/T 16656《工业自动化系统与集成 产品数据表达与交换》是一个由多个部分组成的标准,各部分单独出版。GB/T 16656 的所属各部分又组成多个子系列,即:

- 第 1 至第 19 部分规定了描述方法;
- 第 20 至第 29 部分规定了实现方法;
- 第 30 至第 39 部分规定了一致性测试方法与框架;
- 第 40 至第 59 部分规定了集成通用资源;
- 第 100 至第 199 部分规定了集成应用资源;
- 第 200 至第 299 部分规定了应用协议;
- 第 300 至第 399 部分规定了抽象测试套件;
- 第 400 至第 499 部分规定了应用模块;
- 第 500 至第 599 部分规定了应用解释构造;
- 第 1000 至第 1999 部分规定了应用模块。

GB/T 16656《工业自动化系统与集成 产品数据表达与交换》现已发布了以下 26 个部分:

- 第 1 部分:概述与基本原理;
- 第 11 部分:描述方法:EXPRESS 语言参考手册;
- 第 21 部分:实现方法:交换文件结构的纯正文编码;
- 第 31 部分:一致性测试方法与框架:基本概念;
- 第 32 部分:一致性测试方法与框架:对测试实验室和客户的要求;
- 第 34 部分:一致性测试方法与框架:应用协议实现的抽象测试方法;
- 第 41 部分:集成通用资源:产品描述与支持原理;
- 第 42 部分:集成通用资源:几何与拓扑表达;
- 第 43 部分:集成通用资源:表达结构;
- 第 44 部分:集成通用资源:产品结构配置;
- 第 45 部分:集成通用资源:材料;
- 第 46 部分:集成通用资源:可视化显示;
- 第 47 部分:集成通用资源:形状变化公差;
- 第 49 部分:集成通用资源:工艺过程结构和特性;
- 第 101 部分:集成应用资源:绘图;
- 第 105 部分:集成应用资源:运动学;
- 第 201 部分:应用协议:显式绘图;
- 第 202 部分:应用协议:相关绘图;
- 第 203 部分:应用协议:配置控制设计;
- 第 501 部分:应用解释构造:基于边的线框;
- 第 502 部分:应用解释构造:基于壳的线框;
- 第 503 部分:应用解释构造:几何有界的二维线框;
- 第 513 部分:应用解释构造:基本边界表达;
- 第 520 部分:应用解释构造:相关绘图元素;
- 第 1001 部分:应用模块:外观赋值;

——第 1006 部分:应用模块:基础表达。

本部分是 GB/T 16656 的第 44 部分。本部分等同采用 ISO 10303-44:2000,技术内容和编写格式上与 ISO 10303-44:2000 保持一致。仅由于为将其转化为国家标准,根据我国国家标准的制定要求,作了如下编辑性改动:

对于带下划线的用于 EXPRESS 语言描述的各黑体英文实体名、属性名和函数名等,为了既要维护其英文原意又要便于了解其名称代表的意思,在本部分中,当其作为标题出现时,标出了其中文译名;但在正文中,以英文为主,仅在正文中第一次出现或必要时,才将中文译名括起来放在英文原名后。

本部分代替 GB/T 16656.44—1999《工业自动化系统与集成 产品数据表达与交换 第 44 部分:集成通用资源:产品结构配置》。与 GB/T 16656.44—1999 相比主要变化如下:

1) 修改了以下 EXPRESS 说明;

——alternate_product_relationship;
——assembly_component_usage_substitute;
——configuration_design;
——configuration_effectivity;
——configuration_item;
——make_from_usage_option;
——product_concept;
——product_definition_usage;
——quantified_assembly_component_usage;
——specified_higher_usage_occurrence。

2) 增添了以下 EXPRESS 说明:

——assembly_component_usage_substitute_with_ranking;
——concept_feature_operator;
——concept_feature_relationship;
——concept_feature_relationship_with_condition;
——conditional_concept_feature;
——configurable_item;
——configuration_design_item;
——configuration_design_relationship;
——product_concept_feature;
——product_concept_feature_association;
——product_concept_relationship;
——product_concept_occurrence_relationship。

3) 删除了 ISO 前言。

4) 修改了附录 B 中 B.1 文件的标识、B.2.1、B.2.2 和 B.2.3 模式的标识。

5) 修改了附录 C 中规定的实体短名和 EXPRESS 模式下载网址。

本部分的附录 A、附录 B 为规范性附录,附录 C、附录 D 和附录 E 为资料性附录。

本部分由中国机械工业联合会提出。

本部分由全国工业自动化系统与集成标准化技术委员会(SAC/TC 159)归口。

本部分起草单位:中国标准化研究院。

本部分主要起草人:王志强、李文武、洪岩、刘守华、秦光里。

本部分所代替标准的历次版本发布情况为:

——GB/T 16656.44—1999。

引 言

GB/T 16656 是一个计算机可解释的用于产品信息表达和产品数据交换的技术标准。其目的是对产品全生命周期提供一种独立于任何特定系统、能够描述产品数据的中性机制。其描述功能不仅适合于中性文件的交换,而且也是实现和共享产品数据库及存档的基础。在 GB/T 16656.1 中对这些类别进行了说明。本部分属于集成资源类。

本部分的主要包括:

- product_structure_schema(产品结构模式);
- product_concept_schema(产品概念模式);
- configuration_management_schema(配置管理模式)。

product_structure_schema:

- 根据产品的构成把一个产品定义为一个构成部分或所用产品的集合。本部分以特定生命周期阶段或规范视图定义了这些产品,并将这些产品进行关联。产品可由部件组装,或由采购其他产品生成,或采取以上两种方式。

定义表示构成关系的机制。

product_concept_schema 把产品的概念标识为由客户的产品需求分析导出产品的一组规范。它给出基于客户需求的产品的概念,而不是产品的设计或制造。

configuration_management_schema 标识参与另一个产品制造的那些产品,这些配置是在一个组织的直接控制下进行的。

本部分中的模式和 GB/T 16656 其他部分中的定义了集成资源的模式之间的关系用 EXPRESS-G 图说明,如图 1 所示。EXPRESS-G 在 GB/T 16656.11 中定义。application_context_schema, effectivity_schema 和 product_definition_schema 在 GB/T 16656.41 中定义。图 1 所示的模式是集成资源的构成要素。

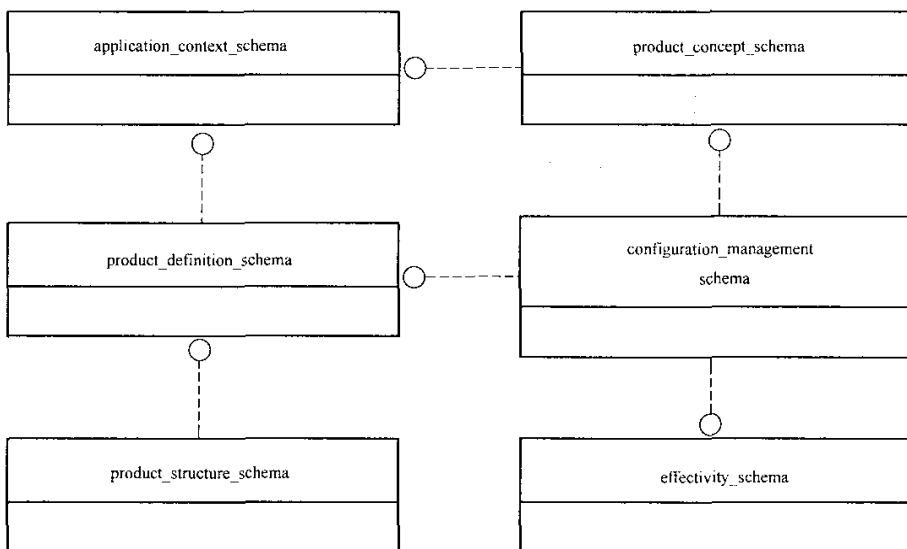


图 1 集成资源中模式的关系

工业自动化与下述内容的信息管理有关：

- 产品结构；
- 产品配置；
- 产品变更。

产品结构关注通过嵌套式分解一个产品成为其组成部分而定义的成品设计方面。本部分的产品结构模式和 GB/T 16656.41 的产品定义模式一起定义了管理产品结构细节的信息表达。

产品配置涉及特定产品的制造规范和装置计划。这个计划含有包括在事先已安排好的生产单元中的一个产品实际组成部分的技术要求。配置管理模式和产品结构模式表达了管理产品配置的信息。有效性的概念适用于产品配置的管理。

变更管理涉及在开发出一个产品的新版本时，产品随着时间而发生的变更，本部分涉及的变化是影响组成部分构成相互关联的结构组织方式的变换。配置管理模式给出在产品变更时产品定义的结构形状的信息和在产品生命周期中增强的信息。变更管理其他方面的信息表达在 GB/T 16656.41 中的产品定义模式和变化模式中定义。

GB/T 16656(ISO 10303)的本部分的这一版本中融入了对前版本所作的向上兼容性修订的部分内容。在以下条件下对 EXPRESS 描述的修订为向上兼容性的：

- 所作修订不会导致 GB/T 16656.21 标准编码要求的变更；这些要求符合尚未修订和已作修订的 EXPRESS 规范。
- 所作修订不会导致 ISO 10303-22 标准数据结构的数据内容的存取软件的变化。
- 所作修订不会导致前版本 GB/T 16656 标准应用协议映象表中本部分内容的映象失效。

工业自动化系统与集成

产品数据表达与交换

第 44 部分:集成通用资源:产品结构配置

1 范围

GB/T 16656 的本部分规定了在产品的生命周期中,管理产品的结构和配置的资源。

下述内容在本部分的范围之内:

——产品的零件和部件间的关系;

——由另一个产品的修改而制成的产品及其零件间的关系;

示例 1:一个产品来自于另一个产品的机械加工。

——根据客户的需求所定义的产品的描述;

——产品技术规格之间的相依性表示交付到顾客手上的产品可能存在差异;

——根据制造规程用于部件和零件配置的结构管理;

——为支持产品生命周期的不同活动的产品分解;

示例 2:一个组织可能需要把一个产品分解成一个物料单,它按一个零件在一个部件中使用的个数列举出每个零件,以及需要把一个产品分解成另一个物料单,它把具有多个部件的产品分解成单个零件。支持不同产品结构报告的更多例子见附录 E。

——与形状、安装及功能等价的单个产品的多种版本。

下述内容不在本部分的范围之内:

——同一个产品的不同产品定义间的关系;

注 1:同一产品的不同定义间的关系可参见 GB/T 16656.41 中 product_definition_schema(产品定义模式)有关内容。

示例 3:对于初步设计中某零件的产品定义与详细设计中同一零件的对应产品定义的关系。

——包括批准、密级、合同协议,及供应商组织的产品生命周期的管理活动;

——产品变更过程,包括变更的理由及产品在哪方面进行了变更;

——在产品生命周期中的决策及其理由;

——产品零件间的物理连接;

——产品组成部分可能具有的特性;

注 2:在 GB/T 16656.41 中定义了支持特性与零件联系的机制,实际的联系包括在各应用协议中。例如,一种材料的特性是什么和如何定义它的细节,以及一个零件具有某种材料特性的事实都超出本标准的范围。

——单一产品(非自由形状、适应性及功能等效)的多个版本。

注 3:产品版本的概念在 GB/T 16656.41 中定义。

2 规范性引用文件

下列文件中的条款通过 GB/T 16656 的本部分的引用而成为本部分的条款。凡是注日期的引用文件,其随后所有的修改单(不包括勘误的内容)或修订版均不适用于本部分,然而,鼓励根据本部分达成协议的各方研究是否可使用这些文件的最新版本。凡是不注日期的引用文件,其最新版本适用于本部分。

GB/T 16262.1 信息技术 抽象语法记法一(ASN.1) 第 1 部分:基本记法规范 (GB/T 16262.1—2006,ISO/IEC 8824-1:2002,IDT)

GB/T 16656.1 工业自动化系统与集成 产品数据的表达与交换 第1部分:概述与基本原理 (GB/T 16656.1—2008, ISO 10303-1:1994, MOD)

GB/T 16656.11 工业自动化系统与集成 产品数据表达与交换 第11部分:描述方法:EXPRESS 语言参考手册

GB/T 16656.41 工业自动化系统与集成 产品数据表达和交换 第41部分:集成通用资源:产品描述与支持原理 (GB/T 16656.41-1999, idt ISO 10303-41:1994)

3 术语、定义和缩略语

3.1 GB/T 16656.1 中定义的术语

GB/T 16656.1 中确立的下列术语和定义适用于本部分。

- 装配件 (assembly);
- 构件 (component);
- 产品 (product)。

3.2 GB/T 16656.41 中定义的术语

GB/T 16656.41 中确立的下列术语和定义适用于本部分。

- 通用理解协议 (agreement of common understanding)
- 加注的 EXPRESS 模式 (annotated EXPRESS schema)
- 产品类型 (type of product)

3.3 其他术语和定义

下列术语和定义适用于本部分。

3.3.1

前驱节点 ancestor node

从一给定节点出发,按逆方向对链的连续遍历所能够达到的任何节点。对于一个给定节点,它的前驱节点包括全部父节点,这些父节点的全部父节点,等等。

3.3.2

物料单数据结构 bill-of-material (BOM) data structure

描述物料单结构的数据元素的图。

注:物料单数据结构的例子参见附录 E。

3.3.3

物料单结构 bill-of-material (BOM) structure

按照产品所嵌套的组成部分对产品的一种结构描述。

注:物料单结构的例子参见附录 E。

3.3.4

子节点 child node

一个链指向的节点。

3.3.5

配置项 configuration item

产品的组成,要么是部件,要么是组装件。

3.3.6

后继节点 descendent node

从一给定点出发,对链的连续遍历所能够达到的任何节点。对于一个给定节点,它的后继节点包括子节点、这些子节点的全部子节点,等等。

3.3.7

有向非循环图 directed acyclic graph; DAG

节点和有向链的一个集合,致使其没有一个节点是它自己的前辈(或后辈)。

3.3.8

有效性 effectivity

表明一个产品被允许、批准或许可用于另一个产品的一种特性。

3.3.9

形状、安装和功能 form, fit, and function

形状是该产品的样式,安装是该产品与另一产品连接的方法,功能是该产品具有的用途。

3.3.10

叶节点 leaf node

没有子节点的节点。

3.3.11

链 link

在一个有向非循环图中,从一个节点到另一个节点的单向关系。

3.3.12

批 lot; batch

作为一个单位来处理的多个产品的集合。

示例: 3 000 捆纱锭可被分为不同的组。每组纱锭都会浸入各自的红色染料桶。这样的组就可视为批和赋值后的批号。标识批号的目的是为了区分因颜色微小改变而引起的,分属不同批号的几捆纱锭。顾客希望他购买的几捆纱锭具有相同的批号,以确保颜色的一致性。

3.3.13

节点 node

有向非循环图中的一个元素,其通过链与另一个这样的元素相连接。

3.3.14

父节点 parent node

一个链的开始节点。

3.3.15

零件表数据结构 parts list data structure

描述零件表结构的实例的图。

注: 参见附录 E 给出的零件表数据结构的示例。

3.3.16

零件表结构 parts list structure

按照产品组成部分的所有不同用途的层次结构对产品的一种结构描述。

注: 零件表结构的例子参见附录 E。

3.3.17

约定作用 promissory use

试图使用一个部件中的一个组成部分。

3.3.18

根节点 root node

没有父节点的节点。

3.3.19

树 tree

有向非循环图的一种有约束的类型,即在其中只有一个根节点,且每个节点最多有一个父节点。

3.4 缩略语

以下缩略语适用于本部分:

BOM(bill-of-material)物料单

CM(configuration management)配置管理

DAG(directed acyclic graph)有向非循环图

4 产品结构模式

下面的 EXPRESS 声明开始描述 product_structure_schema(产品结构模式),并表明了必要的外部引用。

EXPRESS 描述:

*)

SCHEMA product_structure_schema;

REFERENCE FROM product_definition_schema (
product,
product_definition,
product_definition_relationship,
acyclic_product_definition_relationship);

REFERENCE FROM measure_schema
(measure_with_unit);

REFERENCE FROM support_resource_schema
(identifier,label,text);

(*

注1:上面引用的模式可在 GB/T 16656 的下列部分中找到:

product_definition_schema GB/T 16656.41(ISO 10303-41);

measure_schema GB/T 16656.41(ISO 10303-41);

support_resource_schema GB/T 16656.41(ISO 10303-41)。

注2:参见附录 D 给出了的该模式采用 EXPRESS-G 图的图形表达。

注3:本部分规定的不带注释和解释性文本的 EXPRESS 模式列表可从互联网上得到,参见附录 C。

4.1 概述

product_structure_schema 的讨论对象是下述两个定义间的关系:

——为构成其他产品而被装配的产品;

——为构成其他产品而要变更的产品。

定义这些关系如同在 GB/T 16656.41 的 product_definition_schema 中所说明的 product_definition (产品定义)间关系的规范一样。此外,在一个产品结构中使用的任何产品都可能有的透于该应用所规定的替代产品。总之,把这些关系称为是一个产品结构。

产品结构定义一个产品能表示为由各组成部分构成的不同方法,在构成一个产品的各组成部分间建立产品结构关系。产品结构是产品定义的一个方面。

本部分中产品定义间的这些关系,用 GB/T 16656.41 的产品定义模式中所定义的 product_definition_relationship(产品定义关系)实体的子类型来表示它们。本模式中定义的 product_definition_relationship 的子类型对 GB/T 16656.41 中的超类型规定了附加的约束和意义。

注:图2为 GB/T 16656.41 中 product_definition_schema 结构关系的一个局部图,此图描述了该模式定义实体的子型结构。图中没有将模式定义实体属性特征和 product_definition 实体显示出来。所有该模式实体及其属性特征参见完整图:图 D.1。

产品结构的概念适用于一个产品版本结构的多种定义。不同的定义对应于机构在产品开发生命周期中对产品结构定义的不同要求。

示例：一个组织可能为发布设计工程生命活动而定义一个物料清单，还可能为发布制造工程活动而定义一个物料清单。

本模式支持由另一个产品构成一个产品的概念。这个概念涉及一个产品与应用该产品去生成一个新产品的过程的结果间的关系。

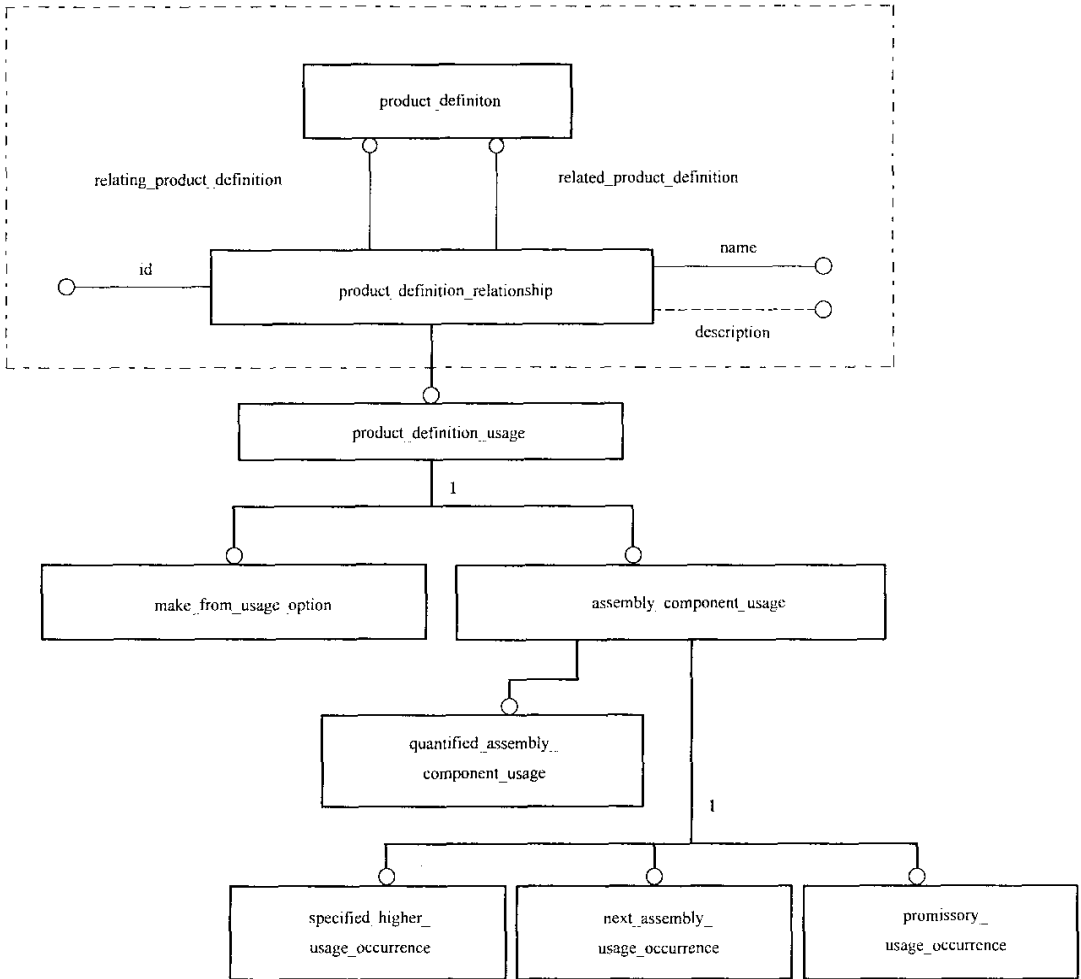


图 2 本部分的产品结构实体的关系

4.2 基本概念与假设

以下概念和假设适用于本部分。

——产品结构通过有向非循环图(DAG)来建模。在这些模型中，节点表示产品的定义，而有向链表示组成的关系。在本模式中，节点对应于 product_definition(产品定义)实体，链对应于 assembly_component_usage 实体。

注 1：对图论的详细讨论见参考文献[1]。

——使用本模式可表示产品结构的多种形式，特别实用的两个产品结构是物料单和零件表结构。

注 2：附录 E 中含有这种方法的一些例子和原理图。其中，本标准的实体可用于表示产品结构。

——零件表结构是物料单的一种特殊形式，其可用一棵树来表示，而物料单结构则要求更一般的有向非循环图。

——对一般的产品结构而言,为了在一个已装配好的产品中标识任何组成部分的应用,还必须标识已装配好的产品与该组成部分间的路径,specified_higher_usage_occurrence 实体即提供这种能力。

4.3 产品结构模式的实体定义

4.3.1 替代产品关系

alternate_product_relationship (替代产品关系)两个产品之间的联系,从而使一个产品(替代产品)可以用来取代另一个产品(基本产品)。

如果一个产品是另一个产品的替代产品时,可以理解为没有必要关注在一个组织中记录的是基本产品还是任何指定的替代产品,它们都有作产品结构中基本产品的一个具体的实例。

注:一个组织可以记录一个基本零件的设计变更,以及建立在各个要制造的部件中应用该基本零件的有效性条件。

作为基础产品的装配组合关系,alternate_product_relationship 规定了用于替代基础产品及其产品结构的替代产品的完整产品结构。

示例 1:两种相同尺寸的螺杆都是产品,一种是十字头的,一种是一字头的。两者头的形状是相应产品的特性。在一个具体的组织中,这两种螺杆就形状、安装及功能而言可认为是等价的;两者具有十分接近的形状,使用时它们占用相同的空间,并且它们都用来把两件东西紧固在一起。这样,这两种螺杆中的一种就可以认为是另一种的替代件。不同的组织可能认为它们具有不同的形状。

由 alternate_product_relationship 建立起来的产品关系不具有对称性。假如 B 是 A 的替代产品,则并不意味着 A 也是 B 的替代产品。

EXPRESS 描述:

*)

ENTITY alternate_product_relationship;

name:label;

definition:OPTIONAL text;

alternate:product;

base:product;

basis: text;

UNIQUE

URL:alternate,base;

WHERE

WR1:alternate:<>;base;

END_ENTITY;

(*

属性定义:

name(名称):识别 alternate_product_relationship 的 label(标签)。

definition(定义):alternate_product_relationship 特征化文本。对于其属性赋值不必作硬性规定。

alternate(替代产品):可以用来代替基本产品的产品。

base(基本产品)可用另一个产品作为替代产品的产品。

basis(基本原理):说明替代产品适用性的基本原理和领域的文本描述。

示例 2:在例 1 中所定义的两个螺栓的 alternate_product_relationship 中,基本属性的赋值可为“head shape”(头部形状),而定义属性赋值则为“alternate for use as fastener in engine assembly”(发动机装配紧固件替代品)。

形式限制:

URL:替代产品与基本产品的组合应是唯一的。

WR1:alternate 属性的实例不应与 base 属性实例相同。

4.3.2 部件构件应用

assembly_component_usage(部件构件应用)把一个组成部分同它的部件联系起来。assembly_component_usage 实体是 product_definition_usage 实体的子类型,product_definition_usage 实体在下列三种产品结构之一中建立 product_definition 间的关系:

——物料单(BOM)结构;

注1:在BOM结构中,product_definition 实体代表节点,且 next_assembly_usage_occurrence 或者 quantified_assembly_component_usage 实体代表链接。

——零件表结构;

注2:在零件表结构中,product_definition 实体代表根节点。next_assembly_usage_occurrence 实体代表结构的每一中间级的节点。specified_higher_usage_occurrence 实体代表链接到更高级别的结构。

——约定使用结构。

注3:在约定使用结构中,product_definition 实体代表节点,且 promissory_usage_occurrence 实体代表节点间的链接。

注4:考虑到本实体在 GB/T 16656 其他部分内容中扩展的需要,对本实体的超类子句只作部分明确规定。如果 GB/T 16656 本部分内容中定义的子型分类齐全,则本实体的超类子句能够完全读出。

EXPRESS 描述:

(*)

```
ENTITY assembly_component_usage
  SUPERTYPE OF (ONE OF(next_assembly_usage_occurrence,
                        specified_higher_usage_occurrence,
                        promissory_usage_occurrence))
  SUBTYPE OF (product_definition_usage)
  reference_designator: OPTIONAL identifier;
END_ENTITY;
```

(*)

属性定义:

SELF\product_definition_relationship. relating_product_definition: 是一个部件,related_product_definition 是它的组成部分。

SELF\product_definition_relationship. related_product_definition: 是一个组成部分,relating_product_definition 是它的父部件。

reference_designator(引用指示符),一种特殊的识别编码,它识别在结构图中、清单中、图表中或者在一件实际设备上,将 related_product_definition 作为 relating_product_definition 组件的使用方式。其属性赋值不必作硬性规定。

注5:在使用或者专论本实体的加注的 EXPRESS 模式中,reference_designator 属性可为命令性或单一限定性,或者两种属性均具。

4.3.3 部件构件应用的代用件

assembly_component_usage_substitute(部件构件应用的代用件)规定在给定部件的环境中,一个组成部分可用作另一个组成部分的代用件。

注1:装配的前后关系则要依据在后继 relating_product_definition 属性中参考同一 product_definition 的有关基础产品及其替代品属性的 assembly_component_usage 的实际情况来作规定。

代用件组成部分的实例不要求占有同样的空间关系或同样的数量。代用件组成部分也不要求与被代用的组成部分的形状、装配和功能是等价的。

本实体仅定义一种代用方法。在一给定的环境中,如果规定 A 为 B 的代用件,那么这不等于设定

B 为 A 的代用件。

注 2: assembly_component_usage_substitute 实体能够在—个较低层的组成部分建立新版本时用于避免所有较高层的部件的重标识。

EXPRESS 描述:

*)

ENTITY assembly_component_usage_substitute;

name: label;

definition: OPTIONAL text;

base: assembly_component_usage;

substitute: assembly_component_usage;

UNIQUE

UR1: base, substitute;

WHERE

WR1: base.relatinq_product_definition: =:

substitute.relatinq_product_definition;

WR2: base: <>: substitute;

END_ENTITY;

(*

属性定义:

name(名字): 识别 assembly_component_usage_substitute 的 label。

definition(描述): 表征 assembly_component_usage_substitute 的 text, 该值不必硬性规定。

base(基本件): 是一个可以应用代用件的 assembly_component_usage。

substitute(代用件): 是可以用来替代基本件的 assembly_component_usage。

形式限制:

UR1: 基本件与代用件属性的组合应是唯一的。

WR1: 基础产品与替代产品属性的 relating_product_definition 属性均需参考同—种装配 product_definition。

WR2: 基本件与代用件属性不应是同样的实例。

4.3.4 带级别的部件构建的代用件

assembly_component_usage_substitute_with_ranking(带级别的部件构建的代用件)是 assembly_component_usage_substitute 的子类, 这种 assembly_component_usage_substitute 规定在给定的装配前后关系中的所有替代品中, 其中—个替代品为取代另—替代品而使用这种特别组件的相对优先级值。

EXPRESS 描述:

*)

ENTITY assembly_component_usage_substitute_with_ranking

SUBTYPE OF (assembly_component_usage_substitute);

ranking: INTEGER;

ranking_rationale: text;

END_ENTITY;

(*

属性定义:

ranking(级别): 是一个进行优先使用分级的整数, 它表示在具有相同基础产品赋值的所有 assembly_component_usage_substitute_with_ranking 实例中, 替代性 assembly_component_usage 的优先使

用级别。对于级别值这一整数,只有将它与具有相同的基本 `assembly_component_usage` 的 `assembly_component_usage_substitute_with_ranking` 对应值作比较时才有意义。它是一个相对级别数值,而不是一个绝对 `ranking` 数值。低级别数值表示对替代性 `assembly_component_usage` 具有更高的优先级,而高级别值则表示更低的优先级。

`ranking_rationale`(级别基本原理说明):描述分级基本原理的文字。

示例:创造这些 `ranking_rationale` 实例,要耗费较高的成本和较长的提前期。

4.3.5 由应用选择制成

`make_from_usage_option`(由应用选择制成)是 `product_definition_usage` 的子类,其中,一个产品是对另一个产品进行加工的产物。

示例 1:加工、电镀以及弯曲成形是制作另一产品的工艺过程。

注 1:在某种情况下,一个产品是由另一个产品经过一系列工艺过程而制成的,这些中间成品将用 `make_from_usage_option` 实体关联起来。

注 2:一个要修改的产品可能是一个部件。通常,`assembly_component_usage` 不同于 `make_from_usage_option`,在那里,部件的组成部分不做任何修改地用于该部件。

注 3:`product_definition` 既可为多种 `make_from_usage_option` 关系的 `relating_product_definition`,也可为多种 `make_from_usage_option` 关系的 `related_product_definition`。而且,文中列举了多个 `make_from_usage_option` 实例,这些实例都参考 `product_definition` 的同一对 `relating_product_definition` 与 `related_product_definition`。

示例 2:考虑到由铸件或锻件加工而成的轴的情况。轴、铸件和锻件这三者都由单独的 `product_definition` 实例来表示。存在有 `make_from_usage_option` 实体的两个实例,一个位于 `relating_product_definition` 轴和 `related_product_definition` 锻件之间,另一个位于 `relating_product_definition` 轴和 `related_product_definition` 铸件之间。

EXPRESS 描述:

```
* )
ENTITY make_from_usage_option
  SUBTYPE OF (product_definition_usage);
  ranking:INTEGER;
  ranking_rationale:text;
  quantity:measure_with_unit;
WHERE
  WR1:(NOT ('NUMBER' IN TYPEOF(quantity, value_component)))
  OR (quantity, value_component > 0);
END_ENTITY;
```

(*

属性定义:

`SELF\product_definition_relationship, relating_product_definition`:由 `related_product_definition` 产品制成的一个 `product_definition`。

`SELF\product_definition_relationship, related_product_definition`:由它制成 `relating_product_definition` 的一个 `product_definition`。

`ranking`(级别):是一个进行优先使用分级的整数,它表示在具有相同的后继 `relating_product_definition` 属性赋值的所有 `make_from_usage_option` 实例中, `related_product_definition` 的优先使用级别。对于级别值这一整数,只有将它与具有相同 `relating_product_definition` 的 `make_from_usage_option` 对应值作比较时才有意义。它是一个相对级别数值,而不是一个绝对 `ranking` 数值。低级别数值表示对 `related_product_definition` 具有更高的优先级,而高级别值则表示更低的优先级。

`ranking_rationale`(级别原理说明):描述分级基本原理的文字。

示例 3:`ranking_rationale` 的例子如成本及研制周期的长短。

quantity(数量): 能够从 related_product_definition 衍生出来的 relating_product_definition 的数量。

形式限制:

WR1: 如果用数字表示组件数量, 其值要大于零。

4.3.6 由应用选择组制成

make_from_usage_option_group(由应用选择组制成)是 make_from_usage_option 实例组成的集合, 它们定义了将一些 product_definition 组合起来的一种可能性, 这些 product_definition 能组成另一个单独的 product_definition。

注 1: make_from_usage_option_group 用于指明几个不同的产品可以由单一的产品制成。

为了表达在单一 make_from_usage_option_group 中的单一产品的约束, 所有 make_from_usage_options 实例的 related_product_definition 应该是相同的。

示例 1: 假设由 product_definition D 表示的一个棒材可以切割两次, 使之产生三个 relating_product_definition 产品 X、Y 和 Z, 那么, 表示这种情况的 make_from_usage_option_group 应集合这三个 make_from_usage_option。如表 1 所示:

表 1 示例 1 的 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

注 2: 使用一个产品去制造多于一个的其他产品组合是可能的。

示例 2: 棒材 product_definition D 也可以切割产生两个 product_definition X 和一个 product_definition T。在这种情况下, 相应的 make_from_usage_option_group 应集合示于表 2 中的 make_from_usage_option。

表 2 示例 2 的 make_from_usage_option_group

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

注 3: 一个单独的 make_from_usage_option 实例可以是多个 make_from_usage_option_group 的一部分。

EXPRESS 描述:

```

*)
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;

```

(*

属性定义:

members(成员): 至少包含两个 make_from_usage_option 实例的一个集合, 实例中的 relating_product_definition 实例可以由同一 related_product_definition 产生。

形式限制:

WR1: 所有项集(members)中的实例在各自 related_product_definition 属性说明中都要引用同一 product_definition 实例。

4.3.7 邻接部件应用具体值

`next_assembly_usage_occurrence`(邻接部件应用具体值)是一种说明产品结构中子组件与其直接上一级装配作业之间关系的 `assembly_component_usage`。

注1: 一个 `next_assembly_usage_occurrence` 实例代表在装配作业中某一组件的单次出现。同一个组件的每一特定用途都可由另一不同的 `next_assembly_usage_occurrence` 的实例来表达,以便指明该组件的属性信息,例如:组件的位置和方向等。属性信息不是由本部分内容中的实体提供,而是由 GB/T 16656.42^[2]、GB/T 16656.43^[3]以及 GB/T 16656.41 中定义的指定性能来提供。对于在一次装配作业中某一组件出现一次的表示方法,可通过定义在 GB/T 16656.41 的 `product_property_definition_schema` 中的 `property_definition` 实体而与 GB/T 16656.42^[2]与 GB/T 16656.43^[3]中的机制联系起来。

示例: 一次装配作业中某一组件的位置和方向信息可通过 GB/T 16656.43^[3]中 `representation_schema` 定义的 `transformation` 来进行计算。

注2: 通过对 `next_assembly_usage_occurrence` 实例结构的序贯式跟踪,应用算法便可获得一个按级缩进排列的(`indented`)产品部件清单。如果将 BOM 与 GB/T 16656.41、GB/T 16656.42^[2]、GB/T 16656.43^[3]中定义的属性信息结构一起配合使用,则类似的算法可以用来计算每个组件在 BOM 中进行更高级别装配作业中的每次出现的位置与方向。

注3: 通过将 `next_assembly_usage_occurrence` 和 `quantified_assembly_component_usage`(见 4.3.11)复杂实例组合起来,可规定出部件中所用组成的数目。

EXPRESS 描述:

*)

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

(*

属性定义:

`SELF\product_definition_relationship.related_product_definition`: 是一个部件, `related_product_definition` 是它的直接组成部分。

`SELF\product_definition_relationship.related_product_definition`: 是一个组成部分, `relating_product_definition` 是它的上一级部件。

4.3.8 产品定义具体值关系

`product_definition_occurrence_relationship`(产品定义具体值关系)是 `product_definition` 与 `assembly_component_usage` 之间的一种结合,其中,由组件具体值的属性所引用的 `product_definition` 实际上是由 `occurrence_usage` 的 `related_product_definition` 属性所引用的组件 `product_definition` 的一个特殊实例,而这个 `occurrence_usage` 则在装配 `product_definition` 中由 `relating_product_definition` 属性所引用。

注: 对于装配组件的特殊实例的识别可用 `assembly_component_usage` 实体的 `reference_designator` 属性来进行定义。

示例: 在轿车的产品结构定义中,要对底盘和车轮作出 `product_definition` 实体的实例方面的定义。这样一来,就可以创建四个 `assembly_component_usage` 的实例来分别表达底盘上的四个车轮,其中 `reference_designator` 的赋值分别为“左前(left front)”、“右前(right front)”、“左后(left rear)”、“右后(right rear)”。为表达右后车轮的特定信息,则要创建一个能够与特定信息联系起来的右后车轮 `product_definition`。为确定这个 `product_definition` 表达了 `assembly_component_usage` 实例所说明的车轮(其中, `assembly_component_usage` 实例的 `reference_designator` 属性赋值为“右后(right rear)”),则要创建 `product_concept_occurrence_relationship` 以便将 `product_definition` 与 `assembly_component_usage` 联系起来。

EXPRESS 描述:

*)

ENTITY product_definition_occurrence_relationship;

name:label;

description:OPTIONAL text;

occurrence:product_definition;

occurrence_usage:assembly_component_usage;

WHERE

WR1:occurrence_usage.relatng_product_definition;<>:occurrence;

WR2:occurrence_usage.related_product_definition;<>:occurrence;

WR3:occurrence.formation;=;

occurrence_usage.related_product_definition.formation;

END_ENTITY;

(*

属性定义:

name(名称):标识 product_definition_occurrence_relationship 的 label。

description(描述):表征 product_definition_occurrence_relationship 的 text。该值不必硬性规定。

occurrence(出现):是 product_definition,它表达了产品特定实例。

occurrence_usage(出现用法):是 assembly_component_usage,它表达了部件中组件的特定用法。

形式限制:

WR1:在由 occurrence_usage 属性引用的 assembly_component_usage 中,occurrence 属性引用的 product_definition 不应为部件 product_definition。

WR2:在由 occurrence_usage 属性引用的 assembly_component_usage 中,occurrence 属性引用的 product_definition 不应为组件 product_definition。

WR3:occurrence 属性引用的 product_definition 和表达 assembly_component_usage 的组成的 product_definition 应该是相同 product_definition_formation 的定义。

4.3.9 产品定义应用

product_definition_usage(产品定义应用)是一种规定两个 product_definitions 间定向联系的 product_definition_relationship,在这两个 product_definitions 中,将 related_product_definition 使用在 relating_product_definition 前后关系中。product_definition_usage 建立了一种阐明在相关产品的前后关系中使用被相关产品的关系。而这种用法要建立在描述两个产品的整个生命周期和前后关系应用中。

示例 1:由于在设计阶段期内,有时产品会包含比建立的装配树更多的中间级别,因此要创建多个装配树。在这种情况下,组件及包含组件的装配件的设计阶段关系可用生产期内的数个中间级别关系来替代。

示例 2:给定一个产品,可能用于对另一个产品起支持作用。可将螺丝刀确定为给定产品整个维护生命周期内的一种资源。在这种情况下,这种螺丝刀与产品之间被确定为维护活动的支持资源关系就得以说明。

EXPRESS 描述:

*)

ENTITY product_definition_usage

SUPERTYPE OF (ONEOF (make_from_usage_option,
assembly_component_usage))

SUBTYPE OF (product_definition_relationship);

UNIQUE

URI: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');
END_ENTITY;
( *

```

SELF\product_definition_relationship. id;是用于 product_definition 应用的标识符。它用于区分 product_definition_usage 一两个实例,其中一对 product_definition 属性是相同的。

示例 3:当应用四个相同的螺栓去连接两个平板时,为了某种目的可能需要标识一个特定的螺栓,它比其余的螺栓需要更大程度的扭转,则继承 id(标识符)属性可用来标识这个特定螺栓的要求,尽管这四个螺栓的 product_definition_usage 全部有同样的属性对 product_definition。

形式限制:

UR1:继承 id 组合、relating_product_definition、related_product_definition 的属性均为单一的。

WR1:如同 product_definition_usage 图中的 relating_product_definition 一样,每个 product_definition_usage 都不会将相同的 product_definition 的 relating_product_definition 实例与 related_product_definition 或者其任何参考对象形成的后继关系联系到一起。

注:此规则具有 GB/T 16656.41 中 product_definition_schema 定义的 acyclic_product_definition_relationship 的功用。

4.3.10 约定应用具体值

promissory_usage_occurrence(约定应用具体值)是一种 assembly_component_usage,其中 related_product_definition 存在于 relating_product_definition 的装配树中,不管它们之间有多少中间级别。

注:当产品结构定义不完全时,可以使用 promissory_usage_occurrence。在这种情况下,有可能会将约定应用具体值装配与组件联系在一起,从而达到在装配中使用组件的目的。

EXPRESS 描述:

```

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

```

4.3.11 定量的部件构件应用

quantified_assembly_component_usage(定量的部件构件应用)是一种在需要规定装配中的子组件数量时确立装配与其一个组件之间关系的 assembly_component_usage。

注:通常,对生产计划或材料计划来说,一个组成部分的若干个具体值要合起来,并且要规定一个数量,以指明有若干个具体值。典型的例子是用于飞机结构连接的铆钉具体的规定及指示用于整架飞机的这种铆钉的个数。如果所使用的铆钉的每个具体值都要规定,则可以使用 next_assembly_usage_occurrence 实体。在这种情况下,铆钉具体值的个数将与 next_assembly_usage_occurrence 的实例一样多。

EXPRESS 描述:

```

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

```

WHERE

WR1:(NOT ('NUMBER' IN TYPEOF(quantity.value_component)))

OR (quantity.value_component > 0);

END_ENTITY;

(*

属性定义:

SELF\product_definition_relationship.relatng_product_definition:是一个部件,related_product_definition 是它的组成部分,并且在此边指定了该组成部分的数量。

SELF\product_definition_relationship.related_product_definition:是一个组成部分,relating_product_definition 是它的父部件,并且在此也指定了该组成部分的数量。

quantity(数量):用于该部件的该组成部分个数或多少的度量。

形式限制:

WR1:如果用数字表示组件数量,其值要大于零。

4.3.12 给定的高层应用具体值

specified_higher_usage_occurrence(给定的高层应用具体值)是一个 assembly_component_usage,它规定一个组件和一个装配件之间的关系,如果装配件不是它的直接上层。

要规定的 specified_higher_usage_occurrence 的组成部分与部件间的关系应通过继承属性 relating_product_definition 和 related_product_definition 的关系来捕获。

在 specified_higher_usage_occurrence 的主实例中,两个属性(upper_usage(上层应用)和 next_usage(下层应用)将分别规定 next_assembly_usage_occurrence 和 assembly_component_usage,它们一起将提供规定 specified_higher_usage_occurrence 的从组成部分到该部件的路径的定义。为了保证 next_assembly_usage_occurrence 和 assembly_component_usage 一起构成 specified_higher_usage_occurrence 所要求的整个路径,要求由 upper_usage 引用的 assembly_component_usage 实体的 related_product_definition 属性的实例与由 next_usage 引用的 next_assembly_usage_occurrence 实体的 relating_product_definition 属性的实例是相同的。next_assembly_usage_occurrence 的 related_product_definition 属性应与要规定的 specified_higher_usage_occurrence 的 related_product_definition 属性的实例相同。由属性 upper_usage 引用的 assembly_component_usage 实体的 relating_product_definition 属性应与要规定的 specified_higher_usage_occurrence 的 relating_product_definition 属性的实例相同。

如果由 upper_usage 属性引用的 assembly_component_usage 不是一个 next_assembly_usage_occurrence,则它将是一个 specified_higher_usage_occurrence。这个 specified_higher_usage_occurrence 应具有如上一段描述所定义的 upper_usage 和 next_usage 属性,以规定主 specified_higher_usage_occurrence 的更进一步的路径。这个递归特性应继续到 upper_usage 属性引用是 next_assembly_usage_occurrence 的一个 assembly_component_usage 实体为止。此时,即利用其组成部分和部件关系及它们之间的整个路径两者完全地规定了主 specified_higher_usage_occurrence。

为了完整地规定一个 specified_higher_usage_occurrence,还应定义全部必要的 assembly_component_usage 实例。

specified_higher_usage_occurrence 实体支持零件表树结构的表示。典型地,它适用于定义零件表的一些部分,即在一个部件中包含与某些特性相联的特定组成部分。

注 1: specified_higher_usage_occurrence 用于定义在一个装配中包含了一个特殊组件的部件清单的某一部分,在清单中的这部分部件中,要将其某些特性进行组合,或者清单中的这部分用于不止一个装配结构中。

注 2: 附录 E 含有该实体的例子及描述它与本模式其他实体间的关系的图。

EXPRESS 描述:

*)

```

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) OR
        (SIZEOF (QUERY (pdr < * USEDIN
          (upper_usage. related_product_definition,
            'PRODUCT_DEFINITION_SCHEMA. PRODUCT_DEFINITION_RELATION-
            SHIP. ' + 'RELATED_PRODUCT_DEFINITION')
          pdr. relating_product_definition; :=:
            next_usage. relating_product_definition)) = 1);
  WR5: SIZEOF ([ 'PRODUCT_STRUCTURE_SCHEMA. NEXT_ASSEMBLY_USAGE_
    OCCURRENCE', 'PRODUCT_STRUCTURE_SCHEMA. SPECIFIED_HIGHER_
    USAGE_OCCURRENCE' ]
    * TYPEOF(upper_usage)) = 1;
END_ENTITY;

```

(*

属性定义:

SELF\product_definition_relationship. relating_product_definition; specified_higher_usage_occurrence 的部件产品定义的继承属性。

SELF\product_definition_relationship. relating_product_definition; specified_higher_usage_occurrence 的组成部分产品定义的继承属性。

upper_usage(上层应用):是该 specified_higher_usage_occurrence 与 relating_product_definition 属性实例相同的且由属性 next_usage 引用的 next_assembly_usage_occurrence 的 relating_product_definition 与 related_product_definition 属性实例相同的一个 assembly_component_usage。

next_usage(下层应用):是该 specified_higher_usage_occurrence 与 relating_product_definition 属性实例相同的且由属性 upper_usage 的 related_product_definition 属性引用的产品定义与属性 relating_product_definition 引用的产品定义实例相同的一个 next_assembly_usage_occurrence。

注 3: 说明这两个属性实例的原理见图 E. 7。

形式限制:

UR1: upper_usage 与 next_usage 属性的组合应是唯一的。

WR1: specified_higher_usage_occurrence 的实例不应与 upper_usage 的实例相同。

WR2: 此 specified_higher_usage_occurrence relating_product_definition 属性(如:装配)将参考 upper_usage 的 relating_product_definition(如:装配)的同一 product_definition 实例。

WR3:此 specified_higher_usage_occurrence related_product_definition 属性(如:组件)将参考 next_usage 的 related_product_definition 的同一 product_definition 实例。

WR4: upper_usage 的 related_product_definition(如:组件)将参考 next_usage 的 relating_product_definition(如:装配)的同一 product_definition 实例。

WR5: upper_usage 的属性类型或者为 next_assembly_usage_occurrence, 或者为 specified_higher_usage_occurrence。

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

5 产品概念

下面的 EXPRESS 声明开始描述 product_concept_schema(产品概念模式), 并标明了必要的外部引用。

EXPRESS 描述:

```
* )
SCHEMA product_concept_schema;
    REFERENCE FROM support_resource_schema
        (text, label, identifier);
    REFERENCE FROM application_context_schema
        (product_concept_context);
(*
```

注 1: 上面所引用的模式可在 GB/T 16656 的下列部分中找到:

support_resource_schema	GB/T 16656. 41(ISO 10303-41);
application_context_schema	GB/T 16656. 41(ISO 10303-41)。

注 2: 用 EXPRESS-G 符号表示的此模式的图解说明请参见附录 D。

注 3: 本部分内容中说明的完整 EXPRESS 模式的清单, 可从互联网上得到——参见附录 C, 其中无任何注释或者其他解释性文字。

5.1 概述

product_concept_schema 的含义就是根据顾客需要来定义的产品理念, 即: 产品概念。一个产品概念可以在定义产品之前就存在。一个产品概念确定了一个独立的产品特征或性能的集合。一个产品概念也具有特别为其确定的产品特征。对于一个特定产品而言, 可以规定其特征条件以便控制其相关性, 并为特定产品概念定义有效的产品变量。

示例: 一辆汽车可以具有两种类型的确定为产品特征的发动机。一个为低功率发动机; 另一个为高功率发动机。对于同一辆汽车而言, 空调装置也可以确定为产品特征。条件规定: 如果存在空调装置, 则需要高功率发动机。

5.2 基本概念与假设

以下概念与假设适用于本部分:

- 一个产品概念确定一个由生产商定义的可交付产品以满足顾客需要。一个产品概念常常被用来确定对产品特征或者性能的选择。
- 产品概念可以由几个配置项来标识。
- 没有实际产品或者产品版本, 产品概念仍可存在。
- 产品概念要具有由生产商定义的产品特征以满足顾客要求。
- 对产品特征所需要进行的相互依赖性组合要规定一套规则。

注: 用本部分内容的实体来表示产品特征组合所采用的方式的有关例子与图解都包含在附录 E 中。

5.3 产品概念实体定义

5.3.1 概念特征运算符

concept_feature_operator(概念特征运算符)定义了约束两个 product_concept_feature 之间关系的一种运算。

示例：“与(AND)”、“非(NOT)”与“或(OR)”均为 concept_feature_operator 名称。

注1：运算的含义与名称属性的许可值的清单可通过每种使用或者专用实体的加注的 EXPRESS 模式加以说明，或者通过共享此信息的各当事方达成的共同理解的基础上加以说明。

注2：附录 E 中包含了该实体的例子以及描述这个实体与其他该模式实体之间关系的图形。

EXPRESS 描述：

*)

ENTITY concept_feature_operator ;

name:label;

description:OPTIONAL text;

END_ENTITY;

(*

属性定义：

name(名称):识别 concept_feature_operator 的 label。

注3：名称指令操作。

description(描述):该文本以 concept_feature_operator 为特征。对属性赋值不必作硬性规定。

5.3.2 概念特征关系

concept_feature_relationship(概念特征关系)是两个 product_concept_feature 之间的一种组合。

注：该组合的含义可通过每种使用或者专用实体的加注的 EXPRESS 模式加以说明，或者通过共享此信息的各当事方达成的共同理解的基础上加以说明。

EXPRESS 描述：

*)

ENTITY concept_feature_relationship;

name :label;

description:OPTIONAL text;

relating_product_concept_feature :product_concept_feature;

related_product_concept_feature :product_concept_feature;

END_ENTITY;

(*

属性定义：

name(名称):识别 product_concept_feature_关系的 label。

description(描述):该文本以 product_concept_feature_关系为特征。对属性赋值不必作硬性规定。

relating_product_concept_feature(相关产品概念特征):作为组合一部分的 product_concept_feature 之一。

related_product_concept_feature(被相关产品概念特征):另一种 product_concept_feature 组合的一部分。如果在这种关系中,其中一个元素取决于另一种元素,则其属性为依赖性。

5.3.3 带条件的概念特征关系

concept_feature_relationship_with_condition(带条件的概念特征关系)是 concept_feature_rela-

tionship 的子类,其中,条件运算符表征 related_product_concept_feature 与 relating_product_concept_feature 之间的关系。

注 1: 如果规定为 concept_feature_feature,则 concept_feature_relationship_with_condition 可对 product_concept_feature 的使用定义一个约束。通过运用 concept_feature_relationship_with_condition 来为 product_concept_feature 构建复杂条件,以便将各个 concept_feature_feature 彼此联系起来。

注 2: 附录 E 中包含了该实体的例子以及描述这个实体与其他该模式实体之间关系的图形。

EXPRESS 描述:

*)

```
ENTITY concept_feature_relationship_with_condition
SUBTYPE OF (concept_feature_relationship);
conditional_operator:concept_feature_operator; END_ENTITY;
```

(*

属性定义:

conditional_operator(条件运算符):用于 related_product_concept_feature 与 relating_product_concept_feature 之间的 concept_feature_operator。

5.3.4 条件概念特征

conditional_concept_feature(条件概念特征)是一个 product_concept_feature 它被运算符所约束的两个 product_concept_feature 的组合。

注 1: product_concept_features 的复杂组合可使用 conditional_concept_feature 构造,作为 concept_feature_relationship_with_condition 的运算域。

注 2: 附录 E 包含该实体和描述它和本模式的其他实体间关系的图的例子。

EXPRESS 描述:

*)

```
ENTITY conditional_concept_feature
SUBTYPE OF (product_concept_feature);
condition:concept_feature_relationship_with_condition;
END_ENTITY;
```

(*

属性定义:

condition:是为 conditional_concept_feature 创造条件的 concept_feature_relationship_with_condition。

5.3.5 产品概念

product_concept(产品概念)是如同 GB/T 16656.41—1999 中 3.3.4 所说明的产品的一个类型,该产品由生产商定义以满足潜在或者实际客户的要求。

注 1: product_concept 常常遵循由消费者组织制定的最高级别的条款。其主要特征就是具有由消费者确定或者源自消费者需求的一组产品特征。产品概念的定义通常来自市场和消费者需求以及趋势预测。

示例 1: 如果一个组织制造汽车和汽车发动机,则这种汽车用 product_concept 实例来表示。如果另一个组织生产汽车发动机,则发动机在该组织中将作为 product_concept 来表示。

EXPRESS 描述:

*)

```
ENTITY product_concept;
id:identifier;
name:label;
description:OPTIONAL text;
```

```

    market_context;product_concept_context;
UNIQUE
    UR1:id;
END_ENTITY;

```

(*

属性定义:

id(标识符): product_concept 实例的标识符。

示例 2: id 可以是一个销售模型号。

name(名称): 标识 product_concept 的 label。

description(描述): 表征 product_concept 的 text, 该值不必硬性规定。

注 2: 这个属性可用于传递 product_concept 的用途、功能, 及所选择的特点。

market_context(市场相关环境): 在其中定义 product_concept 的市场环境。

注 3: product_concept 的参考框架不限于市场环境。为了和 GB/T 16656 规定的 product_concept 保持兼容, market_context 属性的名字没有变化。

注 4: market_context 属性的值能用带注释的 EXPRESS 模式约束, 这些模式使用或特化该实体数据类型, 或者在共享此信息的参与者间达成共识。

形式限制:

UR1:id 属性值应是唯一的。

5.3.6 产品概念特征

product_concept_feature(产品概念特征)要确定一个特征, 从而可以区分属于一个或多个 product_concept 的产品间的变化。

示例: “豪华装备”就是一个区分标准价位、低价位和顶级轿车的特征。

注: product_concept_feature 的分类可通过 GB/T 16656.41 中定义的将产品数据收集成组的机制进行。

EXPRESS 描述:

(*

```

ENTITY product_concept_feature;

```

```

    id:identifier;

```

```

    name:label;

```

```

    description:OPTIONAL text;

```

```

END_ENTITY;

```

(*

属性定义:

id(标识符): 此标识符可区别 product_concept_feature。

name(名称): 识别 product_concept_feature 的标识符。

description(描述): 表征 product_concept_feature 的文本。对属性赋值不必作硬性规定。

5.3.7 产品概念特征联系

product_concept_feature_association(产品概念特征联系)将 product_concept_feature 与 product_concept 联系起来。由于 product_concept_feature 与 product_concept 有关, 所以它可以定义 configuration_item 的内容。

注: 该组合的含义可通过每种使用或者专用实体的加注的 EXPRESS 模式加以说明, 或者通过共享此信息的各当事方达成的共同理解的基础上加以说明。

EXPRESS 描述:

(*

```

ENTITY product_concept_feature_association;

```

```
name:label;  
description:OPTIONAL text;  
concept:product_concept;  
feature:product_concept_feature;  
END_ENTITY;
```

(*

属性定义:

name(名称):识别 product_concept_feature_association 的标识符。

description(描述):该文本以 product_concept_feature_association 为特征。对属性赋值不必作硬性规定。

concept(概念):进行组合的 product_concept。

feature(特征):进行组合的 product_concept_feature。

5.3.8 产品概念关系

product_concept_relationship(产品概念关系)是两个 product_concept 之间的一种组合。

注:该组合的含义可通过每种使用或者专用实体的加注的 EXPRESS 模式加以说明,或者通过共享此信息的各当事方达成的共同理解的基础上加以说明。

EXPRESS 描述:

*)

```
ENTITY product_concept_relationship;  
name:label;  
description:OPTIONAL text;  
relating_product_concept:product_concept;  
related_product_concept:product_concept;  
END_ENTITY;
```

(*

属性定义:

name(名称):识别 product_concept_relationship 的标识符。

description(描述):该文本以 product_concept_relationship 为特征。对属性赋值不必作硬性规定。

relating_product_concept_feature(相关产品概念特征):作为组合一部分的 product_concept 之一。

related_product_concept_feature(被相关产品概念特征):另一种 product_concept,组合的一部分。

如果在这种关系中,其中一个元素取决于另一种元素,则其属性为依赖性。

*)

```
END_SCHEMA; — product_concept_schema
```

(*

6 配置管理

下面的 EXPRESS 声明开始描述 configuration_management_schema(配置管理模式),并标明了必要的外部引用。

EXPRESS 描述:

*)

```
SCHEMA configuration_management_schema;
```

```
REFERENCE FROM product_definition_schema
    (product_definition,
     product_definition_formation,
     product_definition_effectivity);
```

```
REFERENCE FROM product_structure_schema
    (product_definition_usage);
```

```
REFERENCE FROM product_concept_schema
    (product_concept,
     product_concept_feature_association);
```

```
REFERENCE FROM basic_attribute_schema
    (get_description_value,
     get_name_value);
```

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

(*)

注 1: 上面所引用的模式可在 GB/T 16656 的下列部分中找到:

product_definition_schema	GB/T 16656.41(ISO 10303-41);
product_property_representation_schema	GB/T 16656.41(ISO 10303-41);
product_structure_schema	见第 4 章;
product_concept_schema	见第 5 章;
basic_attribute_schema	GB/T 16656.41(ISO 10303-41);
support_resource_schema	GB/T 16656.41(ISO 10303-41);

注 2: 用 EXPRESS_G 标符表达的这种模式的图解说明请参见附录 D。

注 3: 本部分规定的完整 EXPRESS 模式的清单,可从互联网上得到。参见附录 C,其中无任何注释或者其他解释性文字。

6.1 概述

configuration_management 主要内容包括项目识别以及需要进行管理的结构。需要管理的项目规定为 configuration_item。对进行配置管理的组织中的顾客而言,项目是直观的。如果管理的项目为一个产品,则这种模式会促进适当产品版本与 configuration_item 之间组合关系的建立。

本模式还提供记录所建立的联系的能力,致使联系的跟踪成为可能。

配置管理的功能用 configuration_item_configuration_design(配置设计),及 configuration_effectivity(配置的有效性)来实现。

本部分的配置管理模式包括下述概念:

- 配置项及由它们形成一个组成部分的有关 product_concept(产品概念)的标识。
- 实现配置项的 product_definition 或者 product_definition_formation(产品定义生成)的标识。
- 为建立一个 configuration_item 的产品适当版本的联系。这种联系称为 configuration_effectivity。

应用 configuration_effectivity 有三种方法,即:

- serial_numbered_effectivity(序号的有效性),这种配置的有效性建立在所制造产品的编顺序

号实例的基础上。

——dated_effectivity(日期的有效性),这种配置的有效性建立在所要制造产品的日期实例的基础上。

——lot_effectivity(批的有效性),这种配置的有效性建立在所制造产品的批实例的基础。

注: serial_numbered_effectivity, dated_effectivity 和 lot_effectivity 的定义在 GB/T 16656.41 中给出。

6.2 基本概念与假设

以下概念与假设适用于本部分:

- 配置管理可用于要设计或者制造的产品或者产品部件。
- 一个组织确定的那些产品在它的配置管理的控制之下,这些产品便成为该组织的配置项。这些配置项是作为低层组成部分有效性管理关键点的高层功能要素。
- 产品概念可根据一套不变的配置项目确定对产品的客户视图,或者根据定义的客户挑选对象来配置。

6.3 配置管理类型定义:配置设计项

configuration_design_item(配置设计项)类型表示用于制造与 configuration_item 相联系的实际装置的候选设计方案。

```

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

```

6.4 配置管理实体定义

6.4.1 可配置项

configurable_item(可配置项)是一种以一组为用于 product_concept 之中而已经确定的 product_concept_feature 为特征的 configuration_item。由 product_concept_feature_association 规定的 product_concept 可以为继承 item_concept 属性所参考的同一 product_concept,或者与继承 item_concept 属性所参考的 product_concept 有关。

示例:订购轿车的顾客规定所订购轿车的样式及其某些特征,如:颜色为“红”、装饰为“灰色皮革”、“天窗”以及“铝边框”。订购的轿车可由 configuration_item 来表示。订购轿车的样式可由作为 item_concept 的 product_concept 来表示。订购轿车的每一个特征都可以由一个 product_concept_feature 来表示。

EXPRESS 描述:

```

*)
ENTIFY configurable_item
    SUBTYPE OF (configuration_item);
    item_concept_feature: SET[1:?] OF product_concept_feature_association;
END_ENTIFY;
(*

```

属性定义:

item_concept_feature(项概念特征):与 configuration_item 相联系的 product_concept_feature_association 实例。

6.4.2 配置设计

configuration_design(配置设计)把一个配置项和试图实现该项的产品设计联系起来。因此,配置设计实体表示一个配置项与产品定义生成的联系,以规定一项设计是对应某个特定的配置项。

注 1: 在任何实际单元计划之前及在任何设计细节建立之前,由组织建立这个联系。

EXPRESS 描述:

```

* )
ENTITY configuration_design;
    configuration: configuration_item;
    design: configuration_design_item;
DERIVE
    name: label; := get_name_value (SELF);
    description: text; := get_description_value (SELF);
UNIQUE
    UR1: configuration, design;
WHERE
    WR1: SIZEOF (USEDIN (SELF, 'BASIC_ATTRIBUTE_SCHEMA.' +
        'NAME_ATTRIBUTE.NAMED_ITEM')) <= 1;
    WR2: SIZEOF (USEDIN (SELF, 'BASIC_ATTRIBUTE_SCHEMA.' +
        'DESCRIPTION_ATTRIBUTE.DESCRIBED_ITEM')) <= 1;
END_ENTITY;
( *

```

属性定义:

configuration(配置项): 一个配置项, 规定了与该配置项相关联的作为实际制造单元候选的一个 product_definition_formation。

design(设计): 表示一项设计的 product_definition_formation, 它是与 configuration 属性相关联的、用于实际制造单元的候选者。

name(名称): 标识 configuration_design 的 label。

注 2: 该属性附加于 configuration_design 并和 GB/T 16656.44 的规定保持向上兼容。

description(描述): 表征 configuration_design 的 text, 该属性的值不必硬性规定。

注 3: 该属性附加于 configuration_design 并和 GB/T 16656.44 的规定保持向上兼容。

形式限制:

UR1: configuration 属性与 design 属性的组合应是唯一的。

WR1: 至少在一个 name_attribute 中, 每个 configuration_design 都应为 named_item。

注 4: name_attribute 数据类型在 GB/T 16656.41 中的 basic_attribute_schema 中定义。

WR2: 至少在一个 description_attribute 中, 每个 configuration_design 都应为 described_item。

注 5: description_attribute 数据类型在 GB/T 16656.41 中的 basic_attribute_schema 中定义。

注 6: 包含 basic_attribute_schema 中定义的全体实体数据类型的模板在 GB/T 16656.41 的附录 E 中描述。

6.4.3 配置的有效性

configuration_effectivity (配置的有效性) 定义在配置控制下的项的公用有效性属性。一个组织使用 configuration_effectivity 实体去规定 product_definition_usage 的有效性。

示例: 一个用户可能需要规定一个对 configuration_item 是有效的某个 product_definition_usage。一个 147.1 kw 的发动机从某个日期开始是有效的。在 configuration_effectivity 实体中, 对 147.1 kw 发动机的这个信息的获得先于任何生成计划的实际存在。

注: 在 GB/T 16656.41 中定义了 effectivity 的另一个子类型。这个实体的实例也可以是 serial_numbered_effectivity, dated_effectivity 或 lot_effectivity 的实例。

使用 serial_numbered_effectivity 或 lot_effectivity 定义 product_definition_usage 的适用性的情况下, 提供了序列号或批号相关环境的产品由 configuration_design 规定。

EXPRESS 描述:

*)

```
ENTITY configuration_effectivity
    SUBTYPE OF (product_definition_effectivity);
    configuration: configuration_design;
```

UNIQUE

```
    UR1: configuration,
        usage,
        id;
```

WHERE

```
    WR1: 'PRODUCT_STRUCTURE_SCHEMA.PRODUCT_DEFINITION_USAGE' IN
        TYPEOF (SELF\product_definition_effectivity.usage);
```

END_ENTITY;

(*

属性定义:

configuration(配置):对配置有效性的 configuration_design, configuration_effectivity 是有效的。

形式限制:

UR1: configuration 属性、usage 属性及 identification 属性的组合应是唯一的。

WR1: 该 usage 应是一个 product_definition_usage。

非形式限制:

IP1: 被 usage 引用的 product_definition_usage 的 relating_product_definition 应该是 product_definition 的同一个实例, 因为它被一个 configuration_design 的 design 属性引用, 或者, 它应该与 assembly_component_usage 实例树中实例相关。

6.4.4 配置项

configuration_item(配置项)是一个产品或一个产品具体组成部分的标识, 它是为了配置管理而表示的, 并且被视为配置管理过程中的单独单元。

注1: 一个 configuration_item 可由一个 product_concept, 全部 product_concept 或者它的某些组成部分转变而来。

注2: 一个 configuration_item 的建立可能先于对应的 product_definition_formation 的存在。

注3: configuration_item 与 product_definition_formation 间的联系用 configuration_design 来建立。

EXPRESS 描述:

*)

```
ENTITY configuration_item;
    id: identifier;
    name: label;
    description: OPTIONAL text;
    item_concept: product_concept;
    purpose: OPTIONAL label;
```

END_ENTITY;

(*

属性定义:

id(标识符): 配置项实例的标识符。

name(名称): 配置项的描述标识符。

description(描述): 配置项的正文描述。

注 4: 典型正文描述标识配置项的说明或功能。

item_concept(项概念): 与配置项相关联的 product_concept。

Purpose(用途): 提供项概念建立理由的描述标签。

6.4.5 配置项关系

configuration_item_relationship(配置项关系)为两个 configuration_item 之间的一种关系。

注: 该组合的含义可通过每种使用或者专用实体的注解 EXPRESS 模式加以说明, 或者通过共享此信息的各当事方达成的共同理解的基础上加以说明。

EXPRESS specification:

*)

ENTITY configuration_item_relationship;

Name :label;

description :OPTIONAL text;

relating_configuration_item :configuration_item;

related_configuration_item :configuration_item;

END_ENTITY;

(*

属性定义:

name(名称): 识别 configuration_design_relationship 的标识符。

description(描述): 该文本以 configuration_design_relationship 为特征。对属性赋值不必作硬性规定。

relating_configuration_item(相关配置项): 作为组合一部分的 configuration_item 之一。

related_configuration_item(被相关配置项): 另一种 configuration_item, 组合的一部分。如果在这种关系中, 其中一个元素取决于另一种元素, 则其属性为依赖性。

*)

END_SCHEMA; configuration_management_schema

(*

附 录 A
(规范性附录)
实 体 短 名

表 A.1 提供了本部分中规定的实体的短名。使用短名的要求可在 GB/T 16656 所包括的实现方法中找到。

注：可从互联网上得到实体名。

<<http://www.mcl.nist.gov/div826/subject/apdc/snr/>>。

表 A.1 实体短名

实体名	短名
alternate_product_relationship	ALPRRL
assembly_component_usage	ASCMUS
assembly_component_usage_substitute	ACUS
assembly_component_usage_substitute_with_ranking	ACUSWR
concept_feature_operator	CNFTOP
concept_feature_relationship	CNFTRL
concept_feature_relationship_with_condition	CFRWC
conditional_concept_feature	CNCNFT
configurable_item	CNFO
configuration_design	CNFDSG
configuration_effectivity	CNFEFF
configuration_item	CNFITM
configuration_item_relationship	CNITRL
make_from_usage_option	MFUO
make_from_usage_option_group	MFUOG
next_assembly_usage_occurrence	NAUO
product_concept	PRDCNC
product_concept_feature	PRCNFT
product_concept_feature_association	PCFA
product_concept_relationship	PRCNRL
product_definition_occurrence_relationship	PRDFOR
product_definition_usage	PRDFUS
promissory_usage_occurrence	PRUSOC
quantified_assembly_component_usage	QACU
specified_higher_usage_occurrence	SHUO

附录 B
(规范性附录)
信息对象注册

B.1 文件的标识

为了在开放系统中对信息对象提供无二义性的标识,分配给本部分的对象标识符为:

{ iso standard 10303 part(44) version(3) }

这个值的意义在 GB/T 16262.1 中定义,且在 GB/T 16656.1 中描述。

B.2 模式的标识**B.2.1 product_structure_schema 的标识**

为了在开放的信息系统中对 product_structure_schema identification 提供无二义性的标识,分配给 product_structure_schema identification 模式(见第 4 章)的对象标识符为:

{ iso standard 10303 part(44) version(3) object(1) product-structure-schema(1) }

这个值的意义在 GB/T 16262.1 中定义,且在 GB/T 16656.1 中描述。

B.2.2 product_concept_schema 的标识

为了在开放的信息系统中对 product_concept_schema 提供无二义性的标识,分配给 product_concept_schema 模式(见第 5 章)的对象标识符为:

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

这个值的意义在 GB/T 16262.1 中定义,且在 GB/T 16656.1 中描述。

B.2.3 configuration_management_schema 的标识

为了在开放的信息系统中对 configuration_management_schema 提供无二义性的标识,分配给 configuration_management_schema 模式(见第 6 章)的对象标识符为:

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

这个值的意义在 GB/T 16262.1 中定义,且在 GB/T 16656.1 中描述。

附 录 C
(资料性附录)
计算机可解释的清单

本附录引用了本部分规定 EXPRESS 实体名和相应短名的清单。本附录还引用了本部分规定的 EXPRESS 模式的清单,不包括注释或解释性正文。这些清单以计算机可识别的形式提供,并可从以下 URL 找到:

短名:<http://www.mel.nist.gov/div826/subject/apde/snr/>

EXPRESS 模式:<http://www.mel.nist.gov/step/parts/part044c2/is/>

如果访问这些站点有困难,可直接从 sc4sec@cme.nist.gov 联系 ISO 中央秘书处或 ISO TC 184/SC4 秘书处(sc4sec@cme.nist.gov)。

注:以上 URL 以计算机可识别的方式提供的信息是资料性的。本部分包含的信息是规范性的。

附录 D
(资料性附录)
EXPRESS-G 图

本附录中的图 D.1~图 D.3 对应于本部分中的 EXPRESS 模式。该图对 EXPRESS 语言使用了 EXPRESS-G 的图形表示法。EXPRESS-G 在 GB/T 16656.11 的附录 D 中定义。

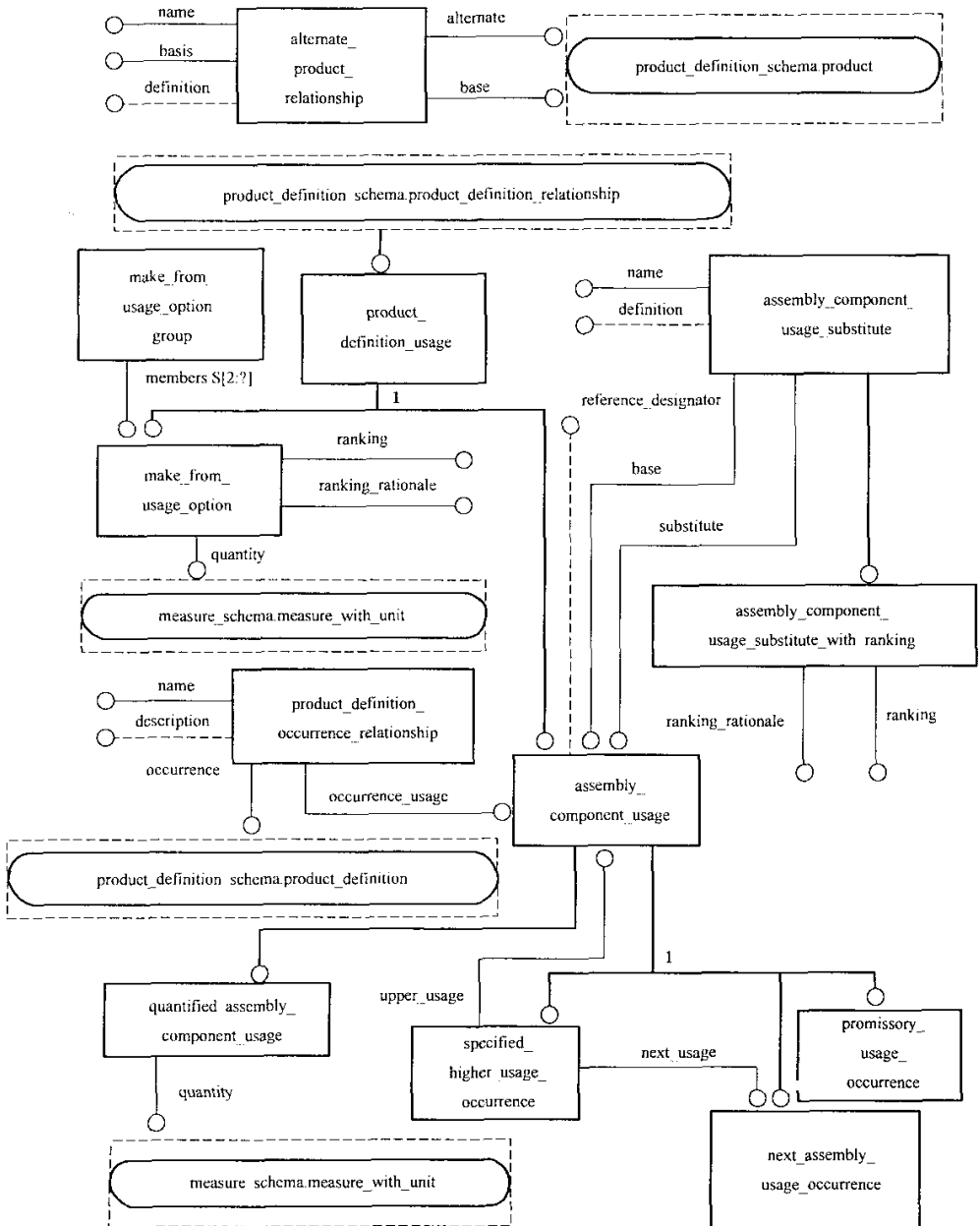


图 D.1 product_structure_schema 的 EXPRESS-G 图

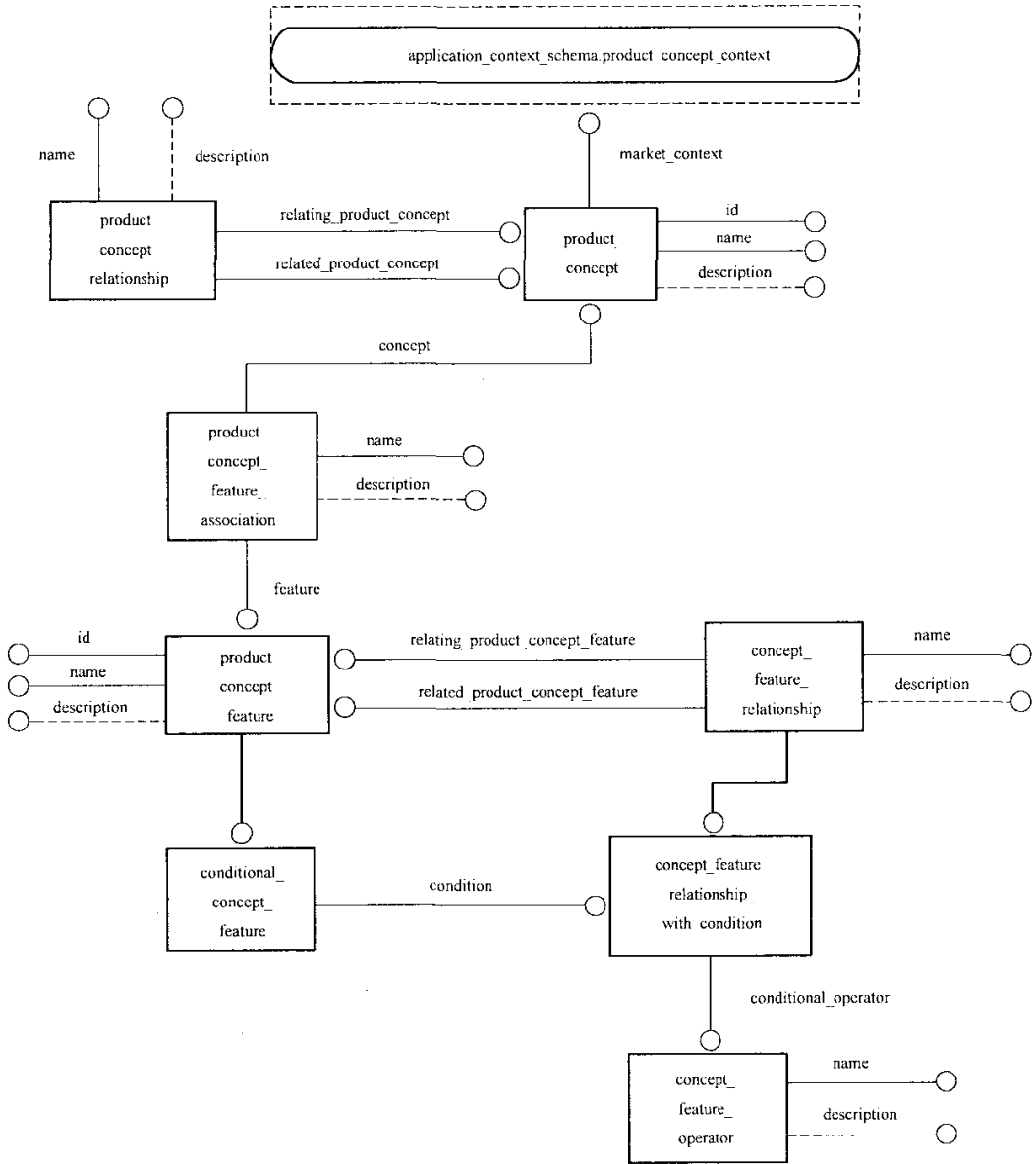


图 D.2 product_concept_schema 的 EXPRESS-G 图

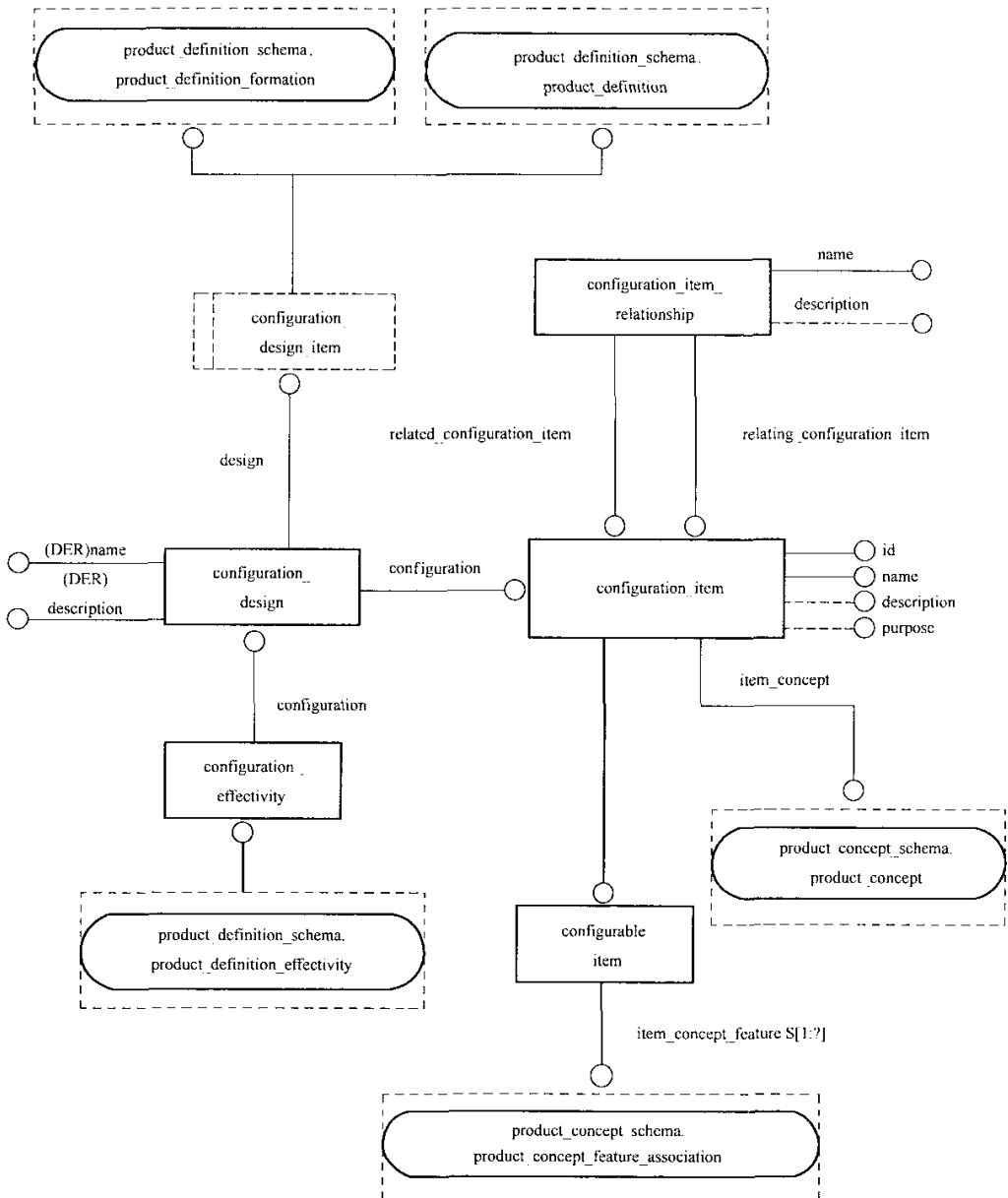


图 D.3 configuration_management_schema 的 EXPRESS-G 图 (3/3)

附 录 E
(资料性附录)
范 例

E.1 产品结构

工业界常用产品结构报告描述一个产品的组成和结构。大多数产品是用一些部件和构件装配而成的。产品结构报告提供关于所使用的部件和构件的细节。

注：见第 3 章中的定义。

本附录给出了产品结构报告的几个例子。此外，还概述了利用本部分去生成这些报告的可能性，其重点是 product_structure_schema 模式。

E.1.1 一个机械装配产品的范例

本条用图和文字两种方法描述了一个称为“第 44 部分的范例产品”的假想的机械装配产品。这个装配构成了说明不同类型的产品结构报告(如下一段中的 BOM 和零件表)的基础。这个产品的建模使用了本标准前面所定义的各种实体。这个范例产品的三维图样见图 E.1, 该产品结构的图示见图 E.2。

该范例产品表示由三个主要组成部分构成的一个简单的机械装配，即：一个作为底板的大金属板，及两个固定在底板上的同样的 L 型支架部件。L 型支架部件的每一个都使用三个螺母和螺杆，并通过在底板上钻出的三个孔固定两端的螺母构成。螺杆和 L 型支架中所使用的螺母的尺寸是相同的。这个产品的设计具有这样一个结构，即它由几个部件(这些部件中有的还包含部件，即子部件)以及若干单个零件构成。

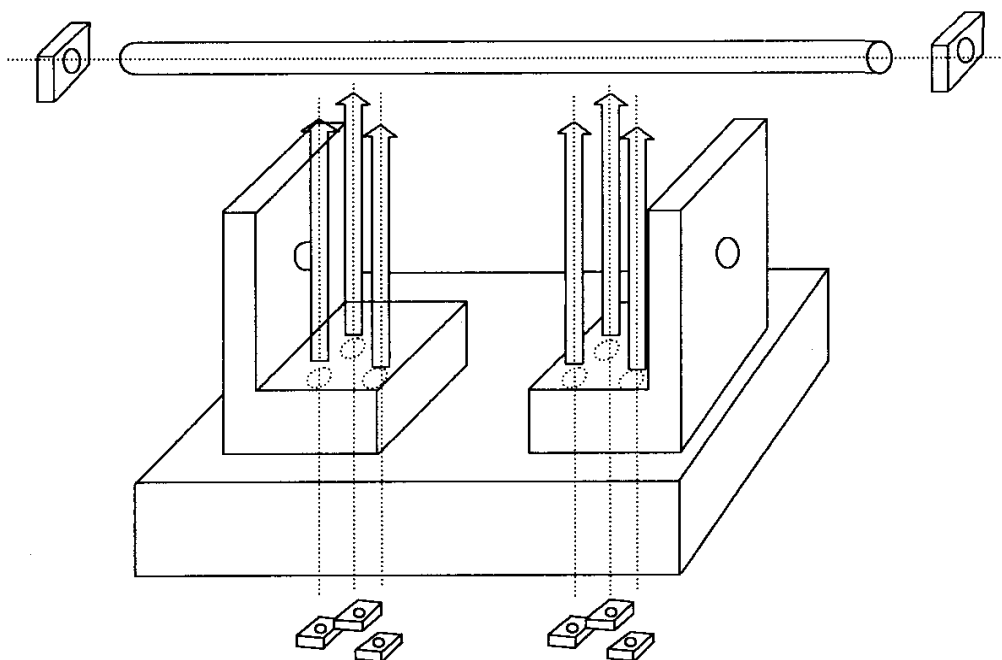


图 E.1 第 44 部分的范例产品

E. 1.2 产品结构报告例子

产品结构报告可以把一个部件分解成多层细目。这些细目包括分解程度、所使用的子部件和构件的数量、子部件与构件的位置和方向,以及用于制成其他产品的产品。细目的各层包括:

——分解程度:

- a) 单层:单个部件及其直接组成部分的子部件或构件;
- b) 多层:一个或多个部件,其中至少有一个子部件被进一步分解。

——分解类型:

- a) 扩展型定义:标识用于制造一个或多个构件的产品;
- b) 爆炸型定义:对于一个给定的分解级别,要对每种组件和部件进行描述;而同一种组件或部件的多种用法则不必进行描述;
- c) 展平型的:通过其构件来描述的一个部件;不描述其子部件;
- d) 数值型的:对于一个给定的分解层,用其数量或总数仅一次来描述第一个子部件和构件;
- e) 具体值型的定义:对于一个给定的分解层,用另一个部件或构件来描述它的每一个子部件和构件,尽管它们可能是相同的;
- f) 标记具体值型的:这是一个具体值型的分解,其中每个子部件和构件都加了标记;
- g) 定位型的:每一个组成部分都与相对于它的直接部件的坐标系的位置和方向联系起来;
- h) 分解混合型的:对于一个给定的分解级别,每一个组件和部件都要进行描述;组件或部件可通过一个分解定义或分解事件来描述。

——分解格式:每一层分解都比前一层分解缩格排印。

现在,用“第 44 部分的范例产品”给出具有混合上面列举的特点的典型报告的例子。

E. 1.2.1 例:数值、多层、缩排型产品结构报告

每一个部件、子部件和零件都放在新的一行上。可以给出指明组成部分数量或总数的数值。通过缩格排印指明的嵌套,同一层的组成部分的缩格数相同,这个报告指示“第 44 部分的范例产品”有两个 L 型支架部件,且每一个都包含三个螺母… 螺杆部件,一个杆件包含两个螺母和一个螺杆。

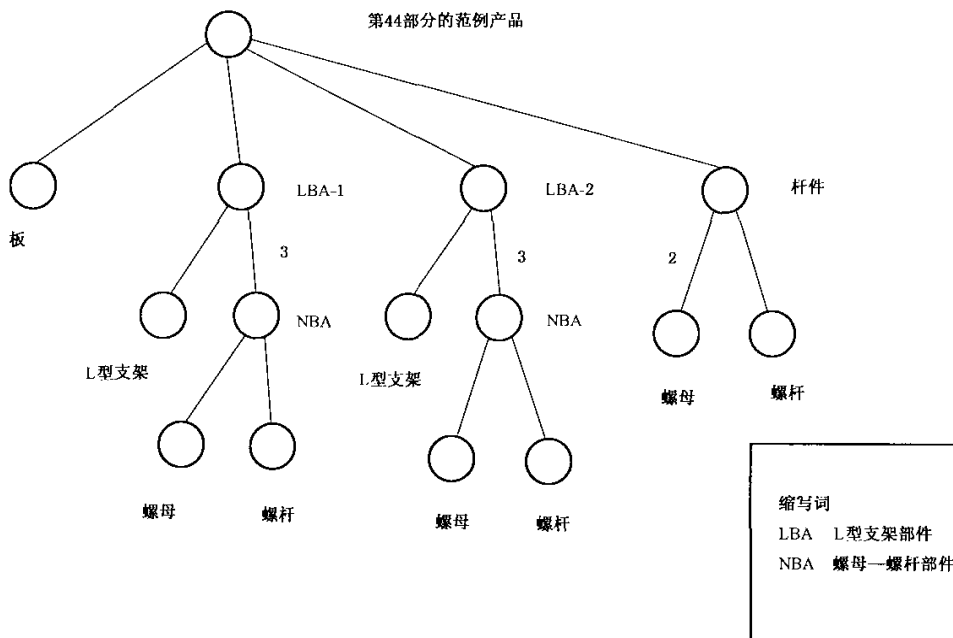


图 E.2 第 44 部分范例产品:图形产品结构

第 44 部分的范例产品

板

L 型支架部件 (2)

L 型支架

螺母 螺杆部件 (3)

螺母

螺杆

杆件

螺杆

螺母(2)

E.1.2.2 例:标记具体值、多层、缩排型产品结构报告

在这份报告中,把该产品结构分解到最底层,一个标号从上向下直到最低层地赋给每一个子部件和构件,例如螺母-1-1。利用这种形式,在完整的产品结构中能够调用螺母-2-1(第 2 个 L 型支架部件的第 1 个螺母)并且可以把它同螺母-螺杆-2(杆件的第 2 个螺母)区别开来。

第 44 部分的范例产品

板-1:板

L 型支架部件-1:L 型支架部件

L 型支架-1-1:L 型支架

螺母-螺杆部件 1 1:螺母-螺杆部件

螺母-1-1:螺母

螺杆-1 1:螺杆

螺母-螺杆部件 1 2:螺母-螺杆部件

螺母-1-2:螺母

螺杆-1 2:螺杆

螺母-螺杆部件-1-3:螺母-螺杆部件

螺母-1-3:螺母

螺杆-1-3:螺杆

L 型支架部件-2:L 型支架部件

L 型支架-2-1:L 型支架

螺母-螺杆部件-2-1:螺母-螺杆部件

螺母-2-1:螺母

螺杆-2-1:螺杆

螺母-螺杆部件-2-2:螺母-螺杆部件

螺母-2-2:螺母

螺杆-2-2:螺杆

螺母-螺杆部件-2-3:螺母-螺杆部件

螺母-2-3:螺母

螺母-2-3:螺母

杆件-1:杆件

螺杆 1:螺杆

螺母—螺杆—1:螺母

螺母—螺杆—2:螺母

E. 1. 2. 3 例:数值、多层、缩排、扩展型产品结构报告

这份报告包括每个组成部分的个数。这份报告是扩展型的,因为它引用了制成某些构件的产品,例如,底板用金属板制成。

第 44 部分的范例产品

板

金属板

L 型支架部件 (2)

L 型支架

金属板

螺母 螺杆部件 (3)

螺母

螺杆

杆件

螺杆

棒材

螺母 (2)

E. 1. 3 BOM、零部件表以及混合数据结构

为了使一个允许的应用能生成上一节中各种产品结构报告所期望的正常信息,就需要这些信息按数据结构的表示进行物理存储。这个范例引进三种可用于表示其信息的数据结构,即在 E. 1. 3. 1、E. 1. 3. 2 以及 E. 1. 3. 3 定义的 BOM 数据结构、零件表数据结构以及混合数据结构。

E. 1. 3. 1 BOM 数据结构

BOM 数据结构适用于表示部件的产品结构的情况,BOM 结构只对用于一次装配中的不同产品作规定,进而产生一个组建一次装配所需要的产品的清单。如同结构中定义的那样,这些产品中的每一个均为标准产品,并称为一个定义。在上面的范例产品中虽然规定使用 8 个螺母,但是在表示这个结构的 BOM 数据结构中仅有表示一个螺母的一个 product_definition 实体的实例,并且这个单一实体还表示了全部 8 个螺杆(见图 E. 3)。

BOM 数据结构有助于对两个特殊定义之间的多种关系的表达。为了说明“第 44 部分的范例产品”中的三对螺母—螺杆的装配,数据结构只选取螺母—螺杆装配中的一个作为例子,但实际上有三种将其连接到父装配上的组合。

这种 BOM 数据结构有一个分歧,那就是对于每个装配中的组件,在包含该组件的所有装配中不可区分。因此,虽然 L 型支架装配中含有三组螺母—螺杆安装操作,每组都包含一个螺母和一个螺杆,但是没有办法区分 L 型支架装配中的任何一个螺杆或螺母个体,除非通过在装配中使用过程来进行区分。

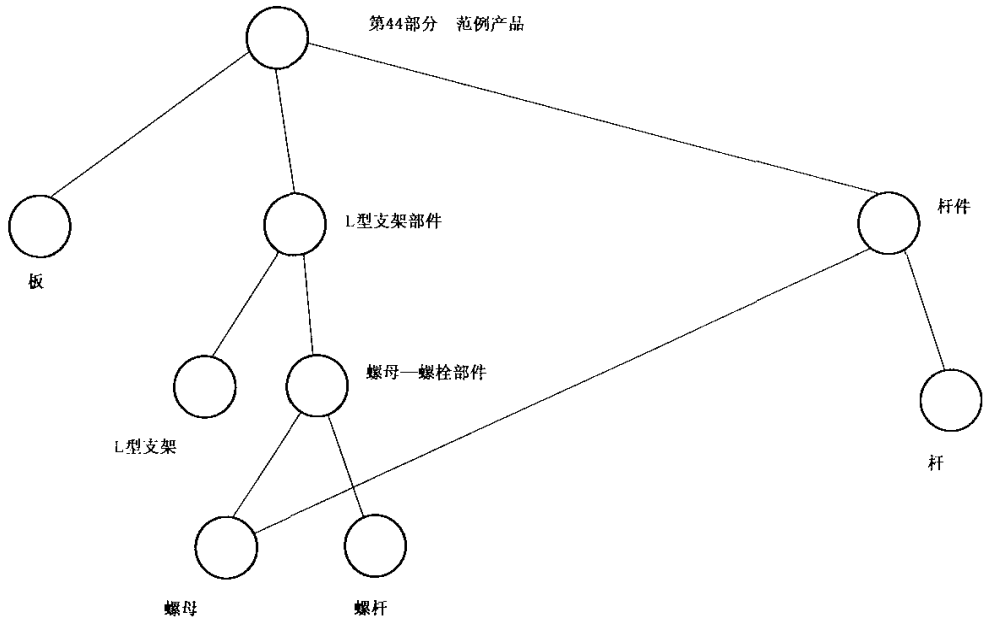


图 E.3 第 44 部分范例产品:BOM 数据结构

E. 1.3.2 零件表数据结构

零件表数据结构可表现产品结构的较低层的各零件与其所包含的较高层各部件间关系的区别。

在单次数据结构中,这种数据结构多次使用同一个定义或标准产品。每使用一个特殊定义所进行的表达就称为一次事件。每次事件不必对其配对定义作出规定。在“第 44 部分的范例产品”中如果有两个 L 型支架装配事件,部件清单的数据结构将包含两个不同的 L 型支架事件,从而允许对所有“第 44 部分的范例产品”中的 L 型支架装配中的各组件进行单独识别。每一个 L 型支架装配事件都建立在一个标准的 L 型支架装配定义的基础上,这个标准的 L 型支架装配定义包含了所有其特征化的信息。在图 E.4 中可以看到这种部件清单数据结构,其中对 L 型支架装配和螺母—螺杆装配事件都逐一进行了表达。每个单独表达出的部件和装配均为这些部件或装配的一个定义事件。这样就清楚地说明了定义的关系。

E. 1.3.3 混合数据结构

混合数据结构是一个 BOM 与部件清单数据结构的混合体。在这种混合体定义中,装配结构列出了其组件的所有使用事件。与之对应的是,一个使用事件都可以将其组件描述为具有多种组合的单个定义,从而也就描述了装配作业中组件定义的每一种特定用途。另外,构成装配作业中每个事件都可通过分解其组合定义来对其进行完全分解。

E. 1.4 使用本部分实体对产品结构的数据结构的表达

将 BOM、部件清单以及混合数据结构映射到 product_definition 与 product_definition_relationship (或子型)实体的组合中。对于纯粹的 BOM 结构而言,由于低级别组件仅作为“黑匣子”用到父装配组件上,因此,仅需使用 product_definition_usage 的 quantified_assembly_component_usage、next_assembly_usage_occurrence 或者 make_from_usage_option 的子型即可。即使“第 44 部分的范例产品”中的确有两个 L 型支架装配,但是从 BOM 数据结构看来,这些 L 型支架装配均相同,并且不能够对其装配

构件进行区分。对于部件清单结构而言,仅需使用 product_definition_usage 的 specified_higher_usage_occurrence 与 next_assembly_usage_occurrence 即可。在这种情况下,每个 L 型支架装配事件都将通过 product_definition 的单个实例来进行表达。对于混合结构而言,说明 BOM 与部件清单数据结构的所有实体都可使用。

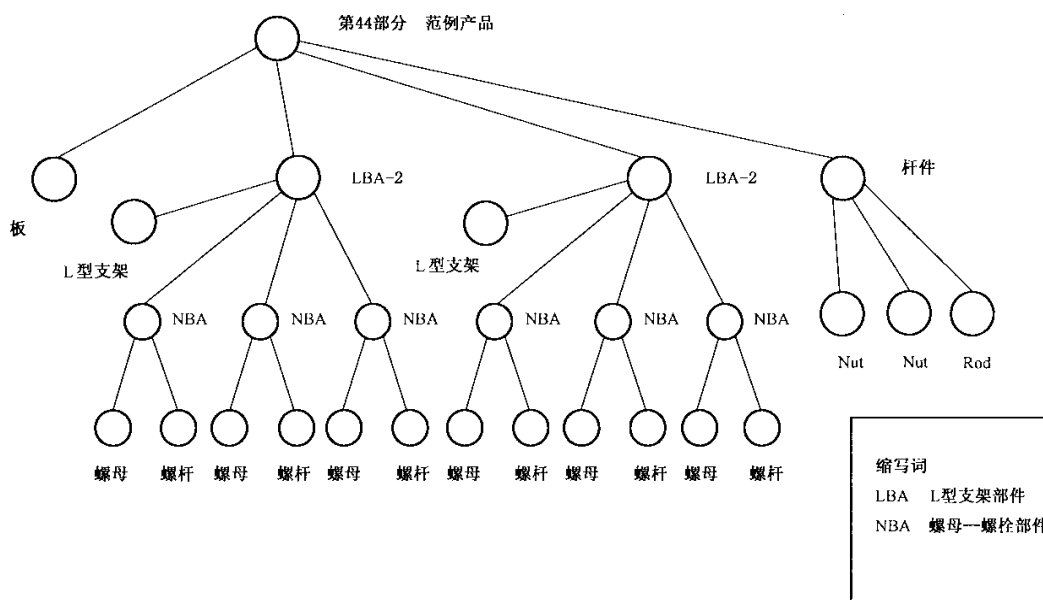


图 E. 4 第 44 部分范例产品:零件列表数据结构

图 E. 5、图 E. 7 以及图 E. 11 中都有本部分内容中实体的符号体系,这些符号体系在图 E. 5 及后面各图形中都会用到。矩形代表 product_definition 的实例,圆圈代表 product_definition_relationship 实体或其一个子型的实例。圆圈或者标上具体子型的第一个字母,或者标上“D—O”字样以表示一个说明某事件与其配对定义之间关系的 product_definition_relationship。在图 E. 5 中,列举了 product_definition_usage 的 next_assembly_usage_occurrence(标有“N”)、quantified_assembly_component_usage(标有“Q”)以及 make_from_usage_option(标有“M”)子型的实例。

E. 1. 4. 1 定义性 BOM 结构表达

为说明 BOM,可采用两种方式来表明一个装配事件中组件的数量。可用 product_definition_usage 的 quantified_assembly_component_usage 的子型来说明一次装配事件中要用到的组件的数量。图 E. 5 中就含有一个 BOM 数据结构的子集。在该图中,“第 44 部分的范例产品”与 L 型支架装配之间只有单一的关系。

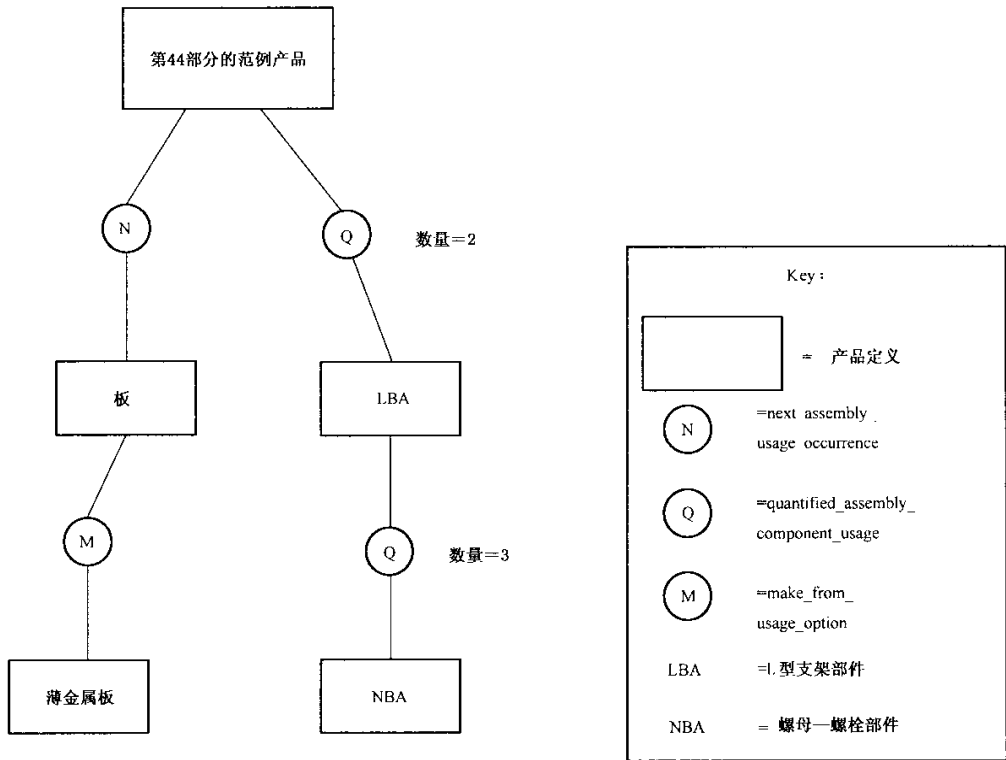


图 E.5 使用本部分实体的 BOM 数据结构

这种关系可用规定 L 型支架装配数量为 2 的 `quantified_assembly_component_usage` 来表示。类似地, L 型支架装配与螺母—螺栓安装之间也只有单一关系。 `quantified_assembly_component_usage` 为螺母—螺栓安装事件的数量规定为 3。

说明 BOM 的第二种方法包括对组件与装配事件之间的每一种关系的识别。图 E.6 通过使用本部分内容展示了 BOM 数据结构中“第 44 部分的范例产品”的完整模式。需要注意的是: `next_assembly_usage_occurrence` 实体的各个实例表示一个部件在高级装配中的各次使用。各装配事件中的组件数量可通过计算 `next_assembly_usage_occurrence` 实例的数量来计算,而不是通过用于说明数量的 `quantified_assembly_component_usage` 的单个实例来计算。

E.1.4.2 定义性部件清单结构表达

如果只用定义性 `product_definition` 实例来说明部件清单,则要用到 `product_definition_usage` 的 `specified_higher_usage_occurrence` 以便明确一个特定的低级别组件在高级装配所有前后关系中的特殊用途。部件清单可清楚地识别一个低级装配的子部件,正如将其从另一个低级装配中的类似子部件区别开来一样。存在这种可能性是由于 `product_definition_usage` 的 `specified_higher_usage_occurrence` 的子型可以提供额外的前后关系信息的缘故。这种额外前后关系信息是通过规定特定直接父装配与特定顶级别装配来提供的。

在图 E.7 中,定义部件清单的额外信息由 `product_definition_usage` 的子型来表示,即 `specified_higher_usage_occurrence` 实体,其符号表示为标有“S”的圆圈。这里, `specified_higher_usage_occurrence` 实体仅可参考两个 L 型支架装配事件其中之一的一个特定螺母—螺栓安装。如同 `next_assembly_usage_occurrence` 实体表示 L 型支架装配事件一样,这种表示方法可通过将 `next_assembly_usage_occurrence` 实体与两次 L 型支架装配事件的仅仅一次联系起来即可达成。

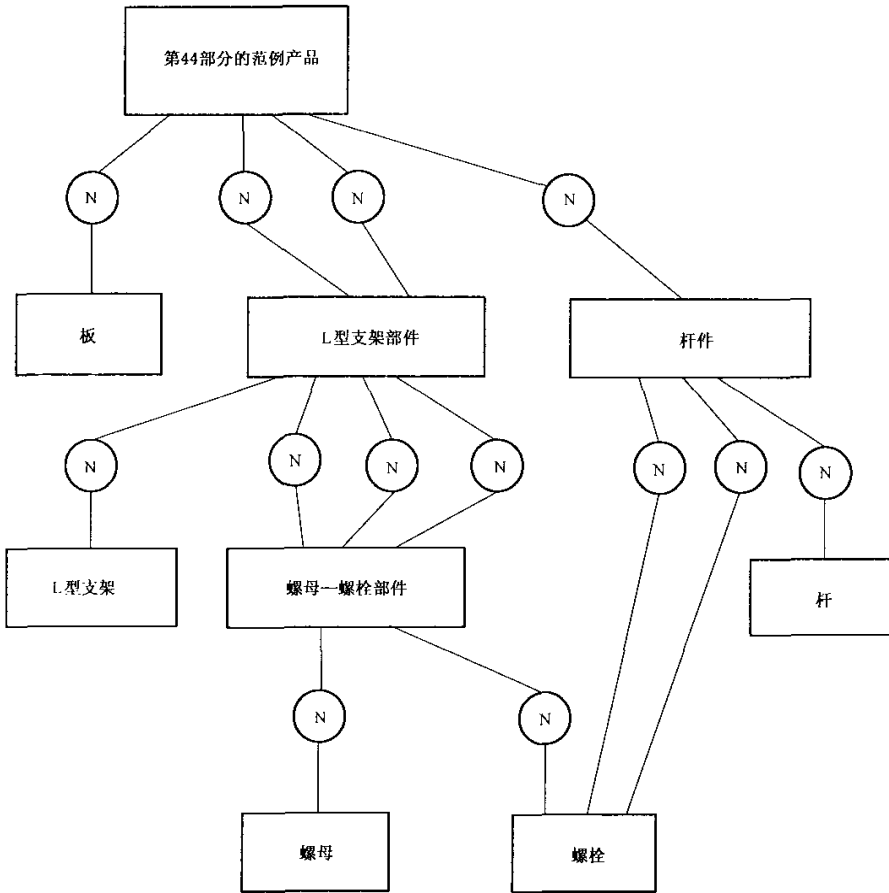


图 E.6 第 44 部分范例产品的 BOM 数据结构

注意,在本例中,即使按本部分内容的实体来完成部件清单,也不必设立低级别组件的个体实例。相反地,可通过中间的 `specified_higher_usage_occurrence` 和 `next_assembly_usage_occurrence` 实体来提供额外的识别信息。

在图 E.8、图 E.9 和图 E.10,以使用本标准实体的形式,在高层部件环境里给出了若干低层构件(螺母、螺杆、部件)的表示。在这三个图的每一图中都用一个大方框把原先的 BOM 数据结构框起来,支持这些例子的零件表方面所需的附加实体在大方框之外,并带有它们引用原先 BOM 的属性。

具体地说,图 E.8 强调 `specified_higher_usage_occurrence` 实体的应用,其表示在“第 44 部分的范例产品”中,更确切地说包括杆件实体中的杆如何使用。这是包含二层构件的一个例子,因此, `specified_higher_usage_occurrence` 实体的两个属性 `upper` 和 `next` 指的就是 BOM 结构的 `next_assembly_usage_occurrence` 构件。

图 E.9 中给出了两种不同的情况。一种情况包括两个位于上方的 `specified_higher_usage_occurrence`,其用来表示专用于“第 44 部分的范例产品”的杆件的两个螺母。注意,图中不包含到顶层“第 44 部分的范例产品”的定义和到底层螺母的定义的附加关系线,以避免图的混乱。另一种情况是用一个 `specified_higher_usage_occurrence` 实体去表示在一个 L 型支架部件的一个特定的螺母。另外,在这两种情况下都仅表示构件与部件间的一个二层关系,因此,中间的(`product_definition_usage` 实体都是 `next_assembly_usage_occurrence`。

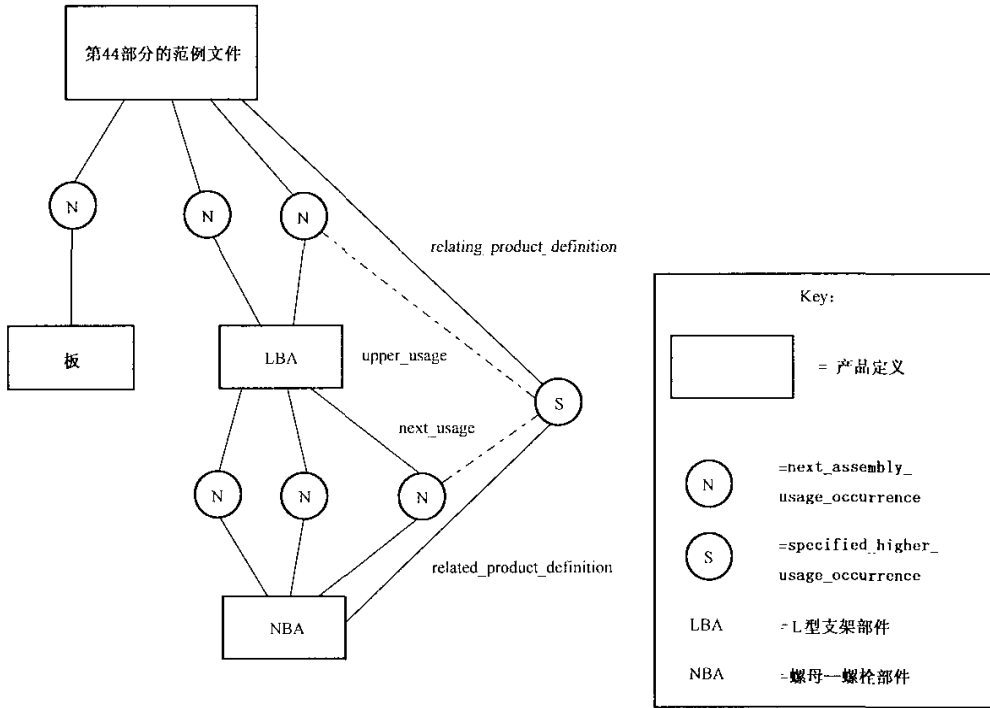


图 E. 7 使用本部分实体的零件列表结构的例子

图 E. 10 指明了零件表数据结构表达的两个附加情况：第一种情况，说明三个 specified_higher_usage_occurrence 实体与总的“第 44 部分的范例产品”中的三个不同螺母—螺栓部件相关联。第二种情况，说明一个 specified_higher_usage_occurrence 实体用于表示在总的“第 44 部分的范例产品”的一个特定的 L 型支架部件中的一个特定的螺母—螺栓部件的一个特定的螺母。后一种情形利用 specified_higher_usage_occurrence 实体建立了一个特定的螺母和总的产品间的三层结构关系模式。因此，相关联的 specified_higher_usage_occurrence 实体有其指向另一个的 specified_higher_usage_occurrence 实体，即标识一个具体的 L 型支架部件中的一个具体螺母—螺栓部件的实体的 next 属性。这就是允许我们去建立加标记的各个具体值的零件表数据结构特征模型的支持层。

部件清单表达渐增的复杂性反映了所要管理的数据结构中信息的复杂性。实际上，如图 E. 7 中见到的那样，更完整的表达模式中所包含的 specified_higher_usage_occurrence 实体的递增数量就直接与穿越部件清单数据结构模型的预计路径数量成比例。通过对 BOM 数据结构的规范就剔除了各组件的个体实例，取而代之的是单个 specified_higher_usage_occurrence 实体，以便满足在一个组件与高级别装配作业之间建立多种互连的特殊联系要求。

E. 1. 4. 3 安装部件清单结构表示方式

产品结构的一个完整部件清单的表示可用本部分内容的实体来进行说明。图 E. 4 描述了一种捕捉产品结构中特殊定义的每一个事件的数据结构的表示方式。每一个圆圈表示一个标准部件的一个事件。每一个事件都通过 product_definition 实体的一个实例来表示。代表一个事件的每一个 product_definition 反过来都与代表作为一个事件的标准部件的定义性 product_definition 有关。

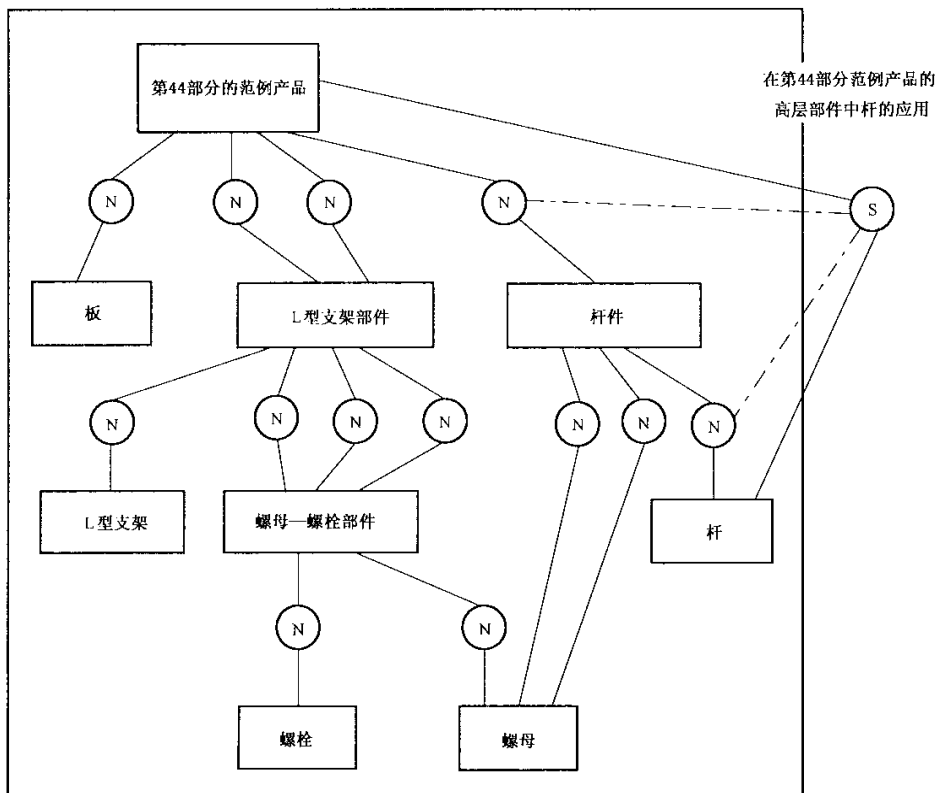


图 E.8 高层用法数值规定的例子

如果产品结构完全以这种方式表示,则它可能被横穿而到达产品结构中的任何特定事件。无 specified_higher_usage_occurrence 实例的需要,因为在任何级别,数据都可与在产品结构中表示产品特定事件的 product_definition 的单个实例联系起来。

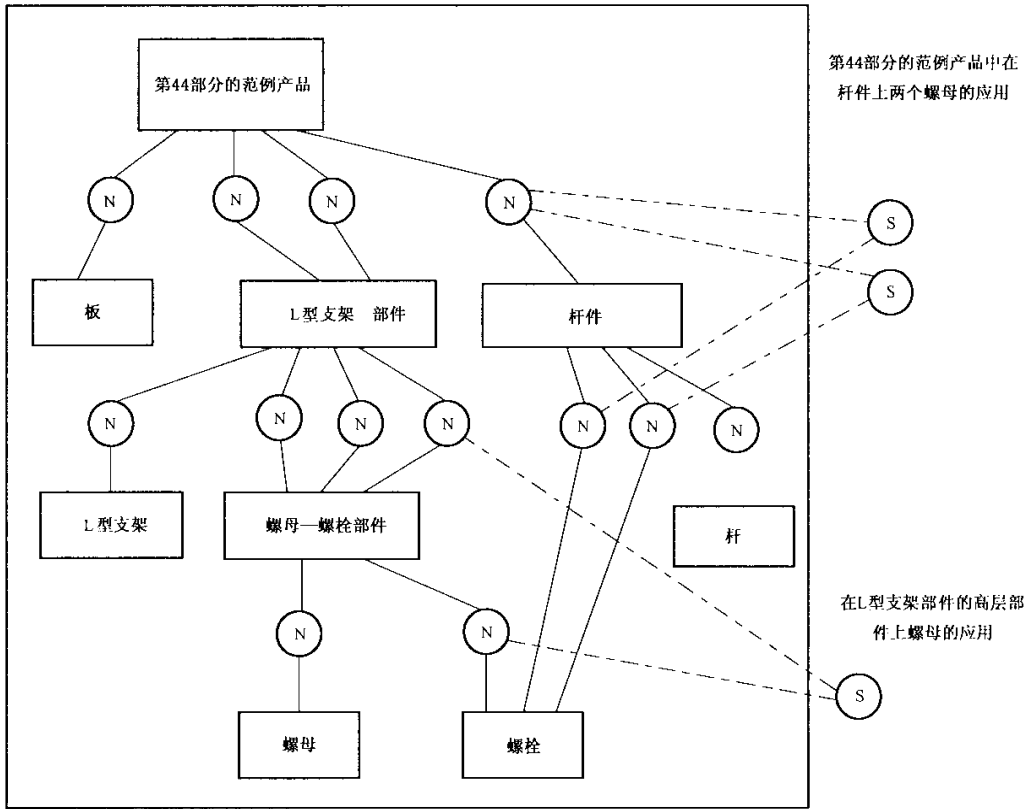


图 E.9 具备特定高层用法数值的特定用法

E. 1. 4. 4 混合结构表达

创建的混合结构中,将 BOM 与部件清单表达混合起来从而使之具有按照要求说明定义与事件的功能。图 E. 11 描述了一种采用“第 44 部分的范例产品”的产品结构的一部分的一个混合结构。每个定义与事件都由一个 product_definition 实例来说明。在图中,“第 44 部分的范例产品”的定义由 L 型支架装配的平板与两个事件的定义构成。L 型支架装配的每个事件都通过一个 product_definition_relationship 而与 L 型支架装配的单个 BOM 定义相关,其符号体系表示为标有“D—O”字样的圆圈,这种 BOM 定义将其定义为一个定义性事件。因而,L 型支架装配的定义就可以通过使用规定为三个装配事件的螺母—螺栓装配的以便部件清单结构来进行说明,并通过采用 next_assembly_usage_occurrence 而与 L 型支架装配事件相关。

因而,螺母—螺栓装配的每一个事件都逐一与此装配的定义联系起来,从而,在装配中每个事件又再次与将其规定为定义性事件的 product_definition_relationship 的实例一起联系起来。必要的情况下,要通过 next_assembly_usage_occurrence 的多个实例采用纯 BOM 表达方式对每个装配事件进行定义。

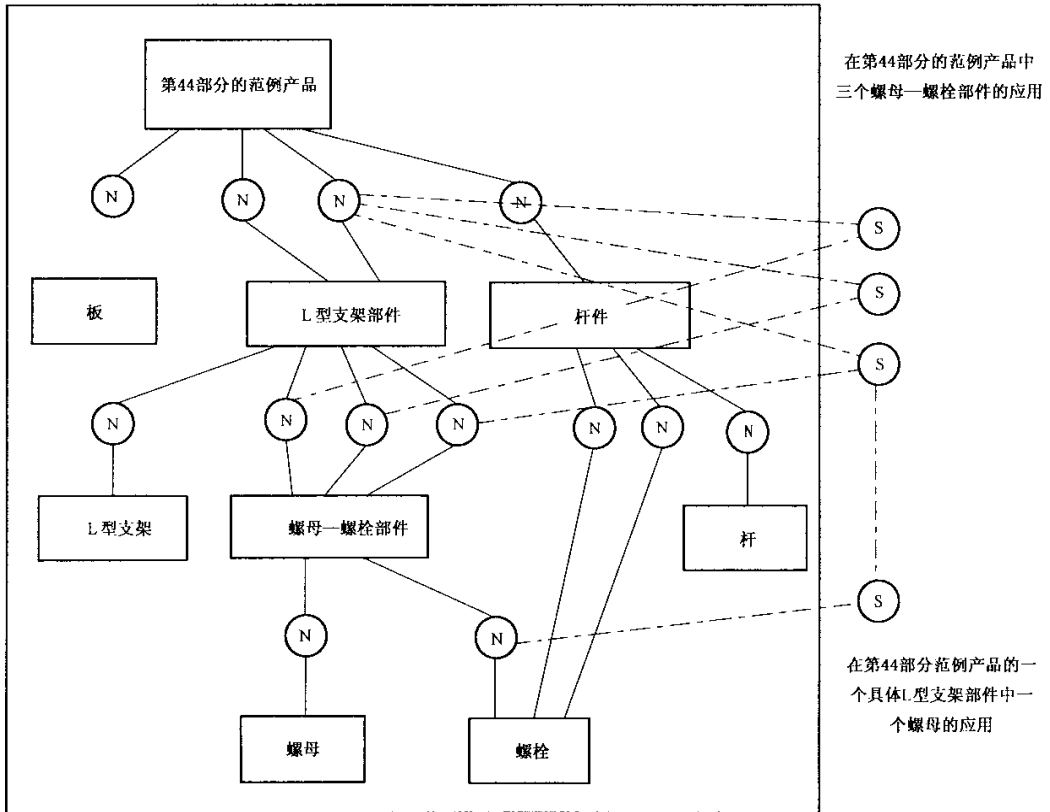


图 E.10 特定高层用法数值的复杂性

在“第 44 部分的范例产品”规定的前后关系中,对于螺母—螺栓配件的一个特殊用途而言,不能对混合结构中的特殊事件 product_definition 加以简单地说明。这里,specified_higher_usage_occurrence 需要作为产品结构中间级别定义的插入事件而出现。为了说明“第 44 部分的范例产品”中的第二个 L 型支架装配事件中的第三对螺母—螺栓安装,specified_higher_usage_occurrence 要指明在每一中间级别的装配中说明使用组件的两个 next_assembly_usage_occurrence。由于在混合结构情况下,定义与事件并不直接相关,因此,specified_higher_usage_occurrence 要在定义的产品结构中说明各个使用事件。

E.1.5 产品结构报告的生成

一旦使用本标准的实体表示了实际的产品结构,则生成本附录前面所述的各种产品结构报告便是可能的。该算法的要点是标识要生成报告的顶层产品,然而反复地通过它的组成分子部件经 quantified_assembly_component_usage, next_assembly_usage_occurrence 或 make_from_usage_option 实体,并根据遍历这些实体时可利用的信息,调用数值及标志,实施适当的索格排印。众所周知,有用于向下通过该结构的若干层的树的遍历算法,其可提供必要的多层报告。此外,根据遍历的发生,通过 specified_higher_usage_occurrence 实体可利用的附加结构信息可被存取,并按需要可应用于所产生的报告。

这样,报告的生成就变成了使用由每一个这样的实体类所唯一提供的附加信息。在 product_definition 实体间的遍历。这个遍历经过 product_definition_usage 实体类型的各种子类型,即 make_from_usage_option、quantified_assembly_component_usage、next_assembly_usage_occurrence 以及 specified_higher_usage_occurrence。

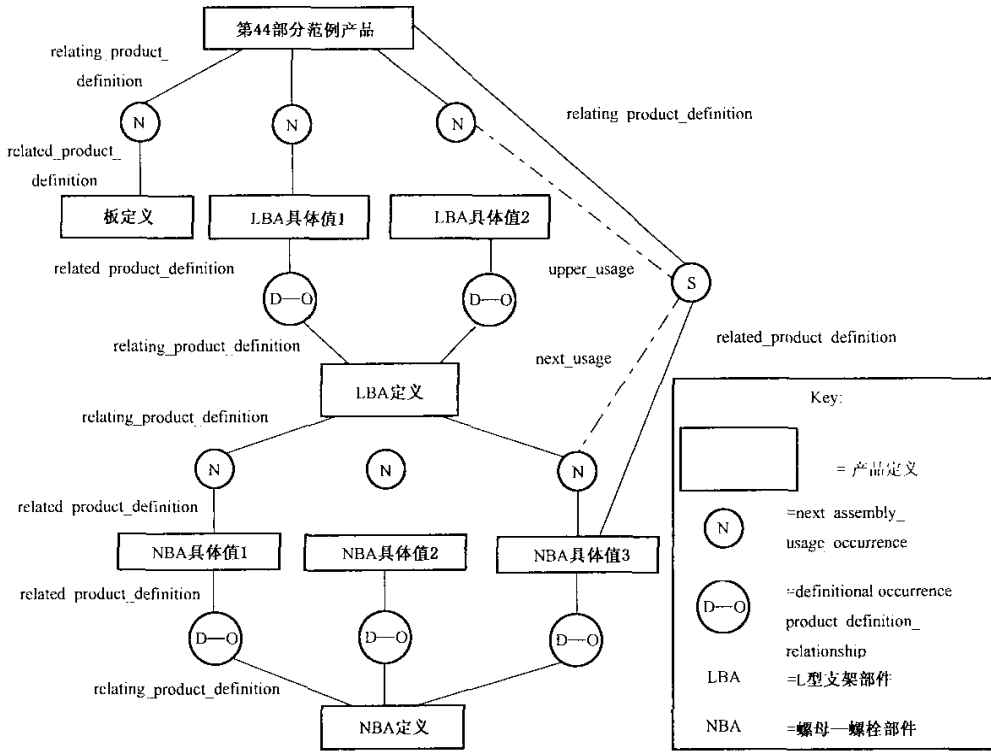


图 E. 11 第 44 部分范例产品的混合结构

E. 1. 6 结束语

本附录给出了支持 BOM 数据结构及零件表数据结构的“第 44 部分的范例产品”的几种表达方法。恰当地应用这些表达法的每一种都能生成本附录其前面小节中所提供的各种产品结构报告。从几个例子来看,本标准所定义的实体的主要特点是它允许数据建模者用同样的原实体去建立各种类型或产品结构数据结构(例如 BOM、零件表等),这样就简化了用户间有意义的信息交换,而不需要借助代价很高的数据模型映射工作。作为该特定特点的例子,它说明了仅增加 specified_higher_usage_occurrence 实体的实例,不需要改变任何已有的 BOM 数据结构的实现,本标准的实体就能够支持从纯 BOM 数据结构的实现到零件表数据结构的实现递增的演变。

E. 2 产品特征与产品概念

根据为识别产品概念的不同特征而规定的依赖性与条件,允许一个产品概念具有数个有效的产品变化。在这个附录中,列举了不同类型的关于产品特征间的依赖性的例子,并采用本部分内容来表达这些依赖性。

由汽车公司制造的特殊模型可通过 product_concept 来表达。在制成的模型上,要安装上柴油发动机(DE)或者火花点火发动机(SI)。每一个这种发动机都可通过 product_concept_feature 的实例来表达。可用全球性条件“DE XOR SI”表达以保证在每一辆成品汽车中恰好只有这些发动机的一种。这种条件通过 conditional_concept_feature 引用一个带有 condition_operator 的 concept_feature_relation-

ship_with_condition 来表达,其中 conditional_operator 为一种具有“XOR”名称的 concept_feature_operator。对 concept_feature_relationship_with_condition 的说明将引用一个 product_concept_feature,并将 id “DE”作为其 relating_product_concept_feature 的 product_concept_feature,同时将 id “SI”作为其 related_product_concept_feature。

一个特定的汽车模型的特征包括空调系统(AC)和电动摇窗机(EW)。如果这些设备都被订购,则还需要一个电力更大的电池组(SB)。这些汽车特征中的每一种都可采用 product_concept_feature 的实例来表达。对于因空调和电动摇窗机都作订购而需要安装的电力更大的电池组这个条件,则可通过 concept_feature_feature 的实例来表达。可采用 concept_feature_operator 来定义不同的逻辑形态以便对这种条件进行说明。

图 E. 12 表明了通过采用二进制操作符:与(AND)、或(OR)和否(NOT)来表达这种条件的一种方式。其布尔逻辑表达式为“(((AC AND EW) AND SB) OR (NOT (AC AND EW)))”。一个 concept_feature_relationship_with_condition 与一个具有“AND”名称的 concept_feature_operator 一起建立了这种对空调系统和电动摇窗机都作要求的关系。这种关系就是 concept_feature_feature 的条件。将 concept_feature_feature 与更大电力的电池组以及一个 AND 操作符联系起来,则可以说所有三个特征都具备了。按照惯例,concept_feature_feature 也用于同它自己的关系之中,以便确立否定条件。最终的 concept_feature_feature 与一个 OR 操作符联系起来,可以说要么具备了三个特征,要么空调系统与电动摇窗机这两个特征都不具备。

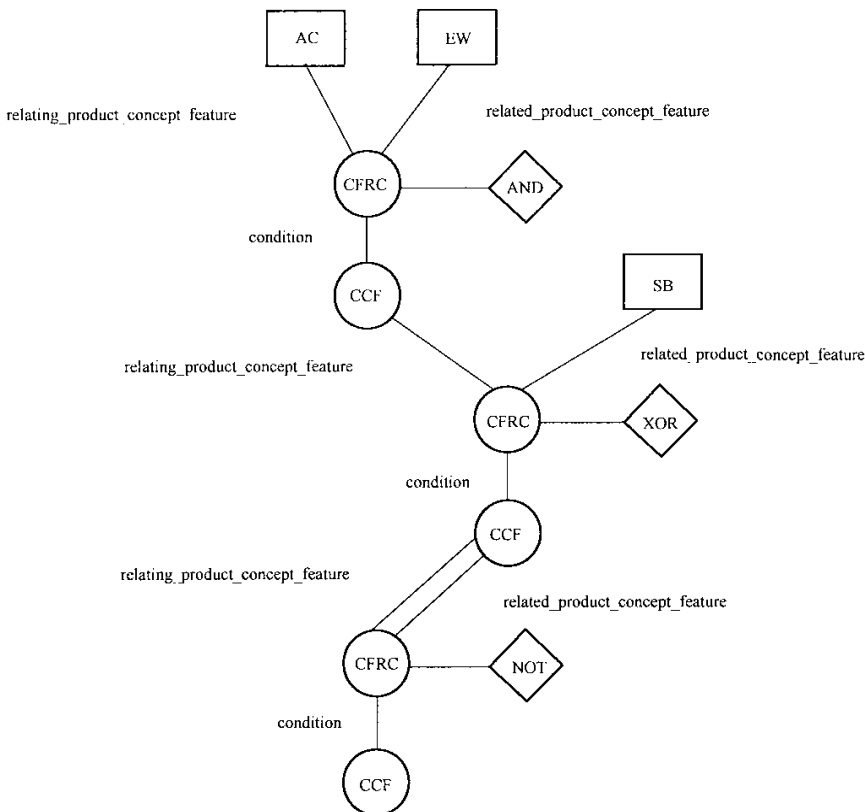


图 E. 12 condition_concept_feature 的交替二元运算符

图 E.13 表明了 在空调系统与电动摇窗机这两个特征都具备的条件下,将汽车中安装更大电力电池组与否作为额外约束的类似条件的一种表达方式。这种方式也采用二进制操作符:与(AND)、或(OR)和否(NOT)。其布尔逻辑表达式为“NOT((AC AND EW)XOR SB)”。一个 concept_feature_relationship_with_condition 与一个具有“AND”名称的 concept_feature_operator 一起建立了这种对空调系统和电动摇窗机都作要求的 关系。这种关系就是 concept_feature_feature 的条件,将 concept_feature_feature 与更大电力的电池组以及一个 XOR 操作符联系起来,则可以说要么空调系统与电动摇窗机的特征都具备,要么具备更大电力的电池组特征。按照惯例,concept_feature_feature 也用于同它自己的关系之中,以便确立否定条件,进而确立所需要的特征组合。

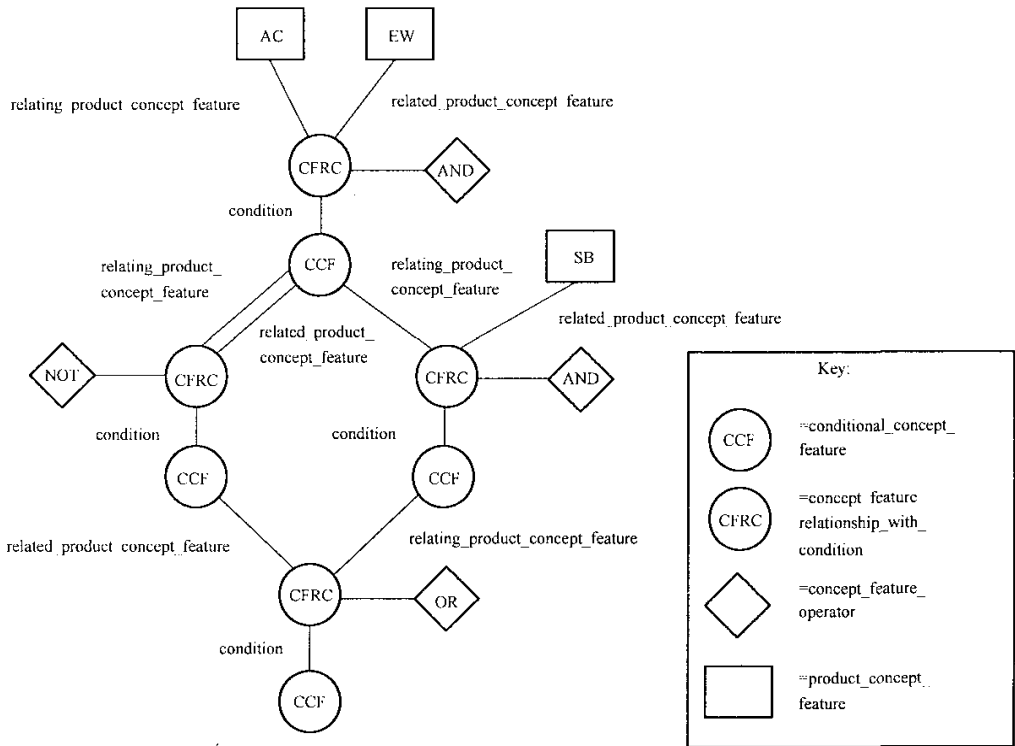


图 E.13 condition_concept_feature 的二元运算符

图 E.14 表明了通过使用二进制操作符 AND 并引入隐含操作的一个操作符来表达同样条件的一种方式。其表达式为“(AC AND EW)IMPLIES SB”。一个 concept_feature_relationship_with_condition 与一个具有“AND”名称的 concept_feature_operator 一起建立了这种对空调系统和电动摇窗机都作要求的 关系。这种关系就是一个与更大电力电池组以及新引入的 IMPLIES 操作符联系在一起的 concept_feature_feature 的条件。这种关系确立了上述条件:要么三个特征都具备,要么空调系统与电动摇窗机的特征都不具备。

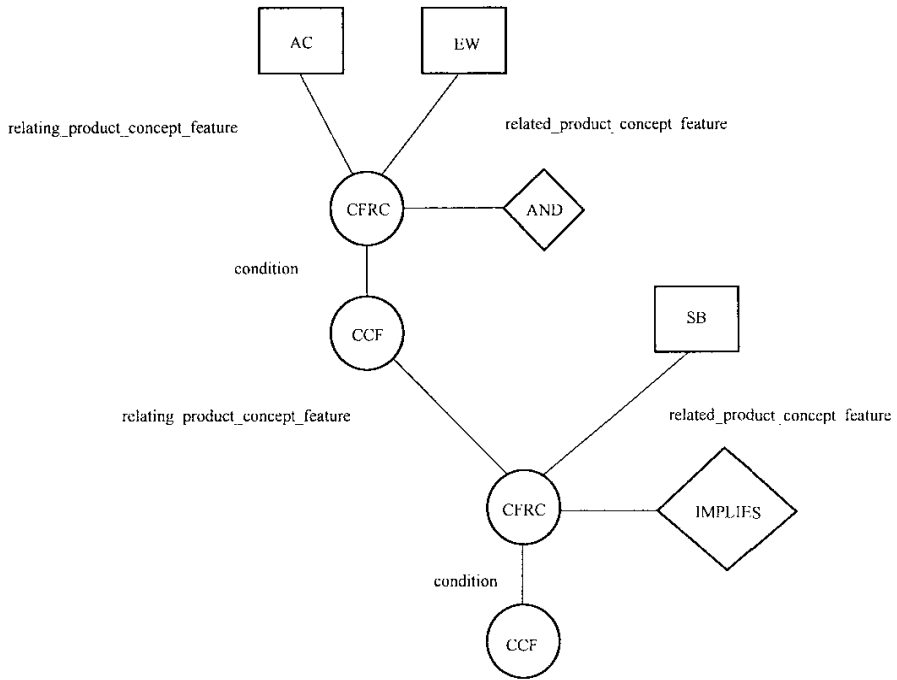


图 E. 14 `condition_concept_feature` 的定义运算符

参 考 文 献

- [1] Hartsfield N and Ringel G. Pearls in Graph Theory; A Comprehensive Introduction. Academic Press, Inc. San Diego, 1990
- [2] GB/T 16656.42—1998 工业自动化系统和集成 产品数据表达与交换 第42部分:集成通用资源:几何与拓扑表达 (idt ISO 10303-42:1994)
- [3] GB/T 16656.43—2008,工业自动化系统与集成 产品数据表达与交换 第43部分:集成通用资源:表达结构(ISO 10303-43:2000, IDT)
-