



# 中华人民共和国国家标准

GB/T 38000.1—2019/IEC 62507-1:2010

---

## 标识系统信息交换 要求 第1部分：原则和方法

Identification systems enabling unambiguous information interchange—  
Requirements—Part 1: Principles and methods

(IEC 62507-1:2010, IDT)

2019-08-30 发布

2020-03-01 实施

国家市场监督管理总局 发布  
中国国家标准化管理委员会

中 华 人 民 共 和 国  
国 家 标 准

标识系统信息交换 要求

第 1 部分：原则和方法

GB/T 38000.1—2019/IEC 62507-1:2010

\*

中国标准出版社出版发行  
北京市朝阳区和平里西街甲 2 号(100029)  
北京市西城区三里河北街 16 号(100045)

网址: [www.spc.org.cn](http://www.spc.org.cn)

服务热线: 400-168-0010

2019 年 8 月第一版

\*

书号: 155066 • 1-63295

版权专有 侵权必究

# 目 次

前言 .....	Ⅲ
1 范围 .....	1
2 规范性引用文件 .....	1
3 术语和定义 .....	2
4 总则 .....	5
4.1 标识目的 .....	5
4.2 引用和可追溯性 .....	6
4.3 永久性 .....	6
4.4 对象种类 .....	6
4.5 标识对象的变更 .....	8
4.6 标识模式 .....	8
4.7 标识对象属性 .....	9
4.8 注册域标识 .....	9
4.9 同一个对象的多个标识 .....	11
4.10 标识代码的存储及使用 .....	12
5 标识系统文档 .....	13
6 标识代码的生成 .....	13
6.1 原理方法 .....	13
6.2 标识代码结构 .....	14
7 全球范围内标识 .....	15
8 标识代码的描述和展示 .....	16
8.1 在计算机系统中使用的表示 .....	16
8.2 可读性 .....	16
8.3 子域表示法 .....	17
8.4 域标识符的应用 .....	17
9 组织变更建议 .....	18
10 合规性 .....	18
附录 A (资料性附录) 标识系统的类型 .....	19
A.1 总则 .....	19
A.2 通过代码对管理者的标识系统进行验证 .....	19
A.3 用名称识别管理者的标识系统 .....	20
A.4 子对象的标识系统 .....	21
附录 B (规范性附录) 参考信息模型 .....	22
B.1 综述 .....	22
B.2 参考模型 .....	22

B.3 实体和属性表 ..... 22

B.4 实体描述 ..... 24

B.5 EXPRESS-G 图形模型 ..... 35

B.6 EXPRESS 源代码 ..... 36

附录 C (资料性附录) 标识系统文档的示例 ..... 43

附录 D (资料性附录) ISO/IEC 646 描述的不变式 ..... 45

参考文献 ..... 46

## 前 言

GB/T 38000 分为如下部分：

——第 1 部分：原则和方法

……

本部分为 GB/T 38000 的第 1 部分。

本部分按照 GB/T 1.1—2009 给出的规则起草。

本部分使用翻译法等同采用 IEC 62507-1:2010《标识系统信息交换要求 第 1 部分：原则和方法》。

与本部分中规范性引用的国际文件有一致性对应关系的我国文件如下：

——GB/T 2659—2000 世界各国和地区名称代码(eqv ISO 3166-1:1997)；

——GB/T 5094.2—2018 工业系统、装置与设备以及产品 结构原则和参照代号 第 2 部分：项目的分类和分类码(IEC 81346-2:2009, IDT)；

——GB/T 16656.11—2010 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(ISO 10303-11:2004, IDT)；

——GB/T 17564.1—2011 电气项目的标准数据元素类型和相关分类模式 第 1 部分：定义 原则和方法(IEC 61360-1:2009, IDT)；

——GB/T 17710—2008 数据处理 校验码系统(ISO 7064:2003, IDT)。

本部分由全国电气信息结构、文件编制和图形符号标准化技术委员会(SAC/TC 27)提出并归口。

本部分起草单位：中机生产力促进中心、北京无线电测量研究所、安徽宝龙电器有限公司、中国标准化院、安徽艺标信息科技有限公司、中国电子科技集团公司第二十九研究所、安徽省质量和标准化研究院、江门市泽信润业科技有限公司。

本部分主要起草人：高永梅、李萍、刘世伟、洪岩、崔从俊、刘旭奕、吴倩、黄贤业。



# 标识系统信息交换 要求

## 第 1 部分:原则和方法

### 1 范围

GB/T 38000 的本部分规定了对象标识系统的基本要求(如产品、“项目”、文件等,不包括人类个体)。它侧重于为对象设定标识符,以便于对对象进行引用。

不涉及对象分类的原因及验证对象是否是所需要的对象。

本部分包括人可读的及机器可读的表示标识符的建议,供构建标识符和标识代码时使用。

本部分还包括符合该系统规定的计算机可识别形式的标识符的应用及其交换要求。

不包括用于计算机间信息交换的物理文件或传输格式(句法)规范,也不包括借助物理介质[如文件、条形码、无线射频标识(RFID)等]、用于信息交换及在收录对象上标记标识信息的规范及传输格式。

### 2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件,仅注日期的版本适用于本文件。凡是不注日期的引用文件,其最新版本(包括所有的修改单)适用于本文件。

GB/T 1988—1998 信息技术 信息交换用七位编码字符集(ISO/IEC 646:1991, IDT)

IEC 61360-1 电气项目的标准数据元素类型和相关分类模式 第 1 部分:定义 原则和方法 (Standard data element types with associated classification scheme for electric components—Part 1: Definitions—Principles and methods)

IEC 81346-2 工业系统、装置与设备以及产品 结构原则和参照代号 第 2 部分:项目的分类和分类码 (Industrial systems, installations and equipment and industrial products—Structuring principles and reference designations—Part 2: Classification of objects and codes for classes)

IEC 82045-1:2001 文件管理 第 1 部分:原则和方法 (Document management—Part 1: Principles and methods)

IEC 82045-2:2004 文件管理 第 2 部分:元数据元素和信息参考模型 (Document management—Part 2: Metadata elements and information reference model)

ISO/IEC 6523-1 信息技术 组织和组织各部分标识用的结构 第 1 部分:组织标识方案的标识 (Information technology—Structure for the identification of organizations and organization parts—Part 1: Identification of organization identification schemes)

ISO/IEC 15418 信息技术 自动识别和数据采集技术 GS1 应用标识符及 ASC MH 10 数据标识符和维护 (Information technology—Automatic identification and data capture techniques—GS1 Application identifiers and ASC MH 10 data identifiers and maintenance)

ISO/IEC 15434 信息技术 高容量自动数据获取(ADC)媒体用语法 (Information technology—Automatic identification and data capture techniques—Syntax for high-capacity ADC media)

ISO/IEC 15459-1 信息技术 唯一标识符 第 1 部分:运输单元唯一标识符 (Information technology—Unique identifiers—Part 1: Unique identifiers for transport units)

ISO/IEC 15459-2 信息技术 唯一标识符 第 2 部分:登记规程 (Information technology—Unique identifiers—Part 2: Registration procedures)



ISO/IEC 15459-4 信息技术 唯一标识符 第4部分:个别项目 (Information technology—Unique identifiers—Part 4: Individual items)

ISO 3166-1 国家及下属地区名称代码 第1部分:国家代码 (Codes for the representation of names of countries and their subdivisions—Part 1: Country codes)

ISO 7064 数据处理 校验码系统 (Information technology—Security techniques—Check character systems)

ISO 10303-11 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 (Industrial automation systems and integration—Product data representation and exchange—Part 11: Description methods: The EXPRESS language reference manual)

### 3 术语和定义

下列术语和定义适用于本文件。

#### 3.1

**批次号 batch number**

**批号 lot number**

分配给某一对象的一组单体的标识代码,用于识别在特定时间内同等条件下生产的单体。

注:通常在制造对象时,分配批号。

#### 3.2

**域 domain**

某些事物所处的可辨别的抽象的或物理的空间。

注:例如一个域可以是一个组织或一个国家或其的一部分。

#### 3.3

**域代码 domain number**

**域标识 domain ID**

分配给一个域的标识代码。

注:分配的域代码可以与组织代码一致。

[IEC 82045-2 派生]

#### 3.4

**标识 [动作] identification [activity]**

将标识代码与对象联系起来的行為。

#### 3.5

**标识代码 identification number**

**标识 ID**

表达标识符的字符串值。

注1:事实上虽然术语上称作“代码”,但字符串中也可以包含其他类型的字符。

注2:本部分中作为属性的术语“标识符”和作为该属性的值的术语“标识代码”含义是不同的。

注3:通常标识代码是唯一的(即,一个对象应仅有一个代码)。这一强制要求是不必要的,在一个特定域内无歧义即可。一个对象可以具有多个标识代码。

此外,在定义中假设一个组织可能负责多个标识代码域。当组织合并时会出现这种情况。

[IEC 82045-2 派生]

#### 3.6

**标识模式 identification scheme**

标识符结构的定义和描述。



## 3.7

**标识系统 identification system**

组织内定义的或记录规则和程序的系统,旨在通过应用标识模式对任意对象进行无歧义标识和检索。

## 3.8

**标识符 identifier**

在指定域内明确标识某对象的属性。

注:在一个标识系统中,可能需要几种类型的标识符。

## 3.9

**同一性 identity**

建立的对象和标识代码之间的关系。

## 3.10

**签发组织 issuing organization**

受注册机构委托并接受注册机构管理的在特定域内分配标识代码的组织。

[ISO 6523 派生]

## 3.11

**元数据 metadata****元信息 meta information**

用于描述一个真实或抽象的对象的无限形式的信息。

[IEC 82045-1 派生]

## 3.12

**对象 object****项目**

在研发、制造、使用和处理过程中所涉及的实体。

注 1:对象可以指实在的或非实在的“物”,例如可能存在或一定存在。

注 2:对象有与之关联的信息。

[GB/T 5094.1—2018,定义 3.1]

## 3.13

**对象代码 object number****对象标识 object ID**

分配给对象的标识代码。

注 1:术语产品编号、项目编号、零件编号、货号、产品标识的编号、可追溯编号(顺序或批次)有时被用作对象代码的同义词。

注 2:通常在设计对象时给产品分配产品的标识代码。具有同一标识代码的多个对象理应具有相同的“外形、配合及功能”,因此是可互换的。

## 3.14

**对象个体 object individual**

与使用场景无关的对象类型单体。

## 3.15

**对象事件 object occurrence**

在特定环境(另一个对象或系统)中使用的对象类型,与使用哪个对象个体无关。

## 3.16

**对象类型 object type**

固有特性相同的对象类。

3.17

**组织 organization**

有独立功能和管理的公司、集团、商行、企业、机关或机构或上述某项中的一部分或其组合(无论合资或非合资、公有或私有)。

3.18

**组织代码 organization number**

**组织标识 organization ID**

分配给组织的标识代码。

注：所分配的组织代码可与域代码一致。

[ISO 6523-1 派生]

3.19

**注册机构 registration authority**

负责受理某组织希望成为签发组织的申请,并对其进行认可的机构。

[ISO 6523 派生]

3.20

**序列号 serial number**

分配给对象的单体或一个对象类型的标识代码。

注：在大多数的工业应用中,序列号用于追踪某类产品全寿命周期内产品类型的不同单体,如某具体车型中制造的一辆汽车。

在其他情况下,将序列号作为流水号以区别不同对象或特定域内的不同对象。

3.21

**可追溯性 traceability**

追踪(标识和检索)某一过程特定点信息的能力。

注：改写 GB/T 19000,定义 3.5.4。

3.22

**变体 variant**

从基本的(一般)对象类型衍生的对象类型。

注：为了变体在同一时间存在,需要对变体进行同步管理。但是,可以同时存在不同的版本,这取决于淘汰旧版本的方式。

3.23

**版本 version**

用于表示对象在全寿命周期内变化的已标识状态,已标识状态与给定的对象代码关联。

注 1：文档版本体现了文档周期内文档的开发过程中确定的状态,并对其进行标识和记录以便检索。术语文档修订通常用于表示该文档版本被正式批准,见 IEC 82045-1 和 IEC 82045-2。本部分中未采用此术语。

注 2：产品版本体现了与一系列产品全寿命周期有关的已确定类型的产品类型在开发过程中的已确定的状态。

[IEC 82045-2 派生]

3.24

**版本号 version number**

**版本标识 version ID**

分配给版本的标识代码。

注：对于版本号而言,与对象关联的对象代码可用作域代码。

[IEC 82045-2 派生]

## 4 总则

### 4.1 标识目的

标识的目的是确保无歧义精准的引用。

定位是可追溯的基本要求。

标识符是用于标识对象的一种属性。

标识代码是标识符的字符串值,可以明确地定位某一特定对象(产品、文档、信息对象等),从而使其在某一特定域(环境)中具有唯一性。

对标识代号最重要的要求是,在指定域内根据既定的规则建立的标识代号应是无歧义的。

例如,由于产品标识代码在产品本身以及在产品整个全寿命周期内相关的产品文档中有所体现,因此产品编号用于定位产品的整个全寿命周期(时间期限可达 100 年以上)。

如果通过版本管理变更标识的对象,那么对象代码将作为版本号的域代码。如果不使用版本管理,需要给相关域内变更的对象分配新的对象代码。

域的规范、并在规范中标识的对象类型以及在此域中构建标识代码的规则通常被称为标识系统。

对标识系统最重要的要求是标识系统是永久性的。

公司要了解的采用何种方法处理变更的例子见第 9 章。

由于对内和对外贸易中越来越多地使用电子信息交换,所以越来越重视标识系统的无歧义和永久性。

对于设计、工程、实践、操作、维护和处置(即,产品或系统的全寿命周期),有必要采用针对不同用途和各种对象的多个标识系统,例如:

- 用于标识各类产品的产品/零件标识系统;
- 用于标识产品单体的(产品/零件)序列标识系统;
- 用于标识相同条件下生产(因而认为所有产品是相同的)的同类型各系列产品的(产品/零件)批次号标识系统;
- 用于标识文档的文档标识系统;
- 用于标识报价/出价的报价标识系统;
- 用于标识订单/合同的订单标识系统;
- 用于资产设备管理或租赁业务标识的资产标识系统;
- 其他。

以上标识系统用于标识某组织领域内的对象,以此将所标识的对象与该组织关联起来。

第二组标识系统侧重于简化贸易和物流,通常由国际组织来负责,旨在将对象从不同的来源中标识出来,以便于进行全球追踪、查找及检索,例如:

- 交易项目(物品)标识系统;
- 资产标识系统;
- 预订标识系统;
- 银行账户标识系统;
- 系列出版物标识系统;
- 含有一个或多个交易项目的包装标识系统;
- 航空公司用包裹标识系统;
- 证书标识系统;
- 重要公共基础设施用标识系统;
- 连网设备的标识系统等。

第三组标识系统是用于将已标识的对象事件与它们所处的产品/系统/设备关联起来。

- 用于标识对象的参考命名系统；
- 用于标识文档的文档命名系统。

附录 A 描述了不同类型的标识系统及其要求。

注：标识系统的示例将在本标准的第 2 部分给出(准备中)。

4.2 引用和可追溯性

标识代码适用于一个特定对象(或一组对象)。

要满足可追溯性的要求,标识代码应指向对象:文档、文件或资料的元数据源。元数据的相关描述见图 1。

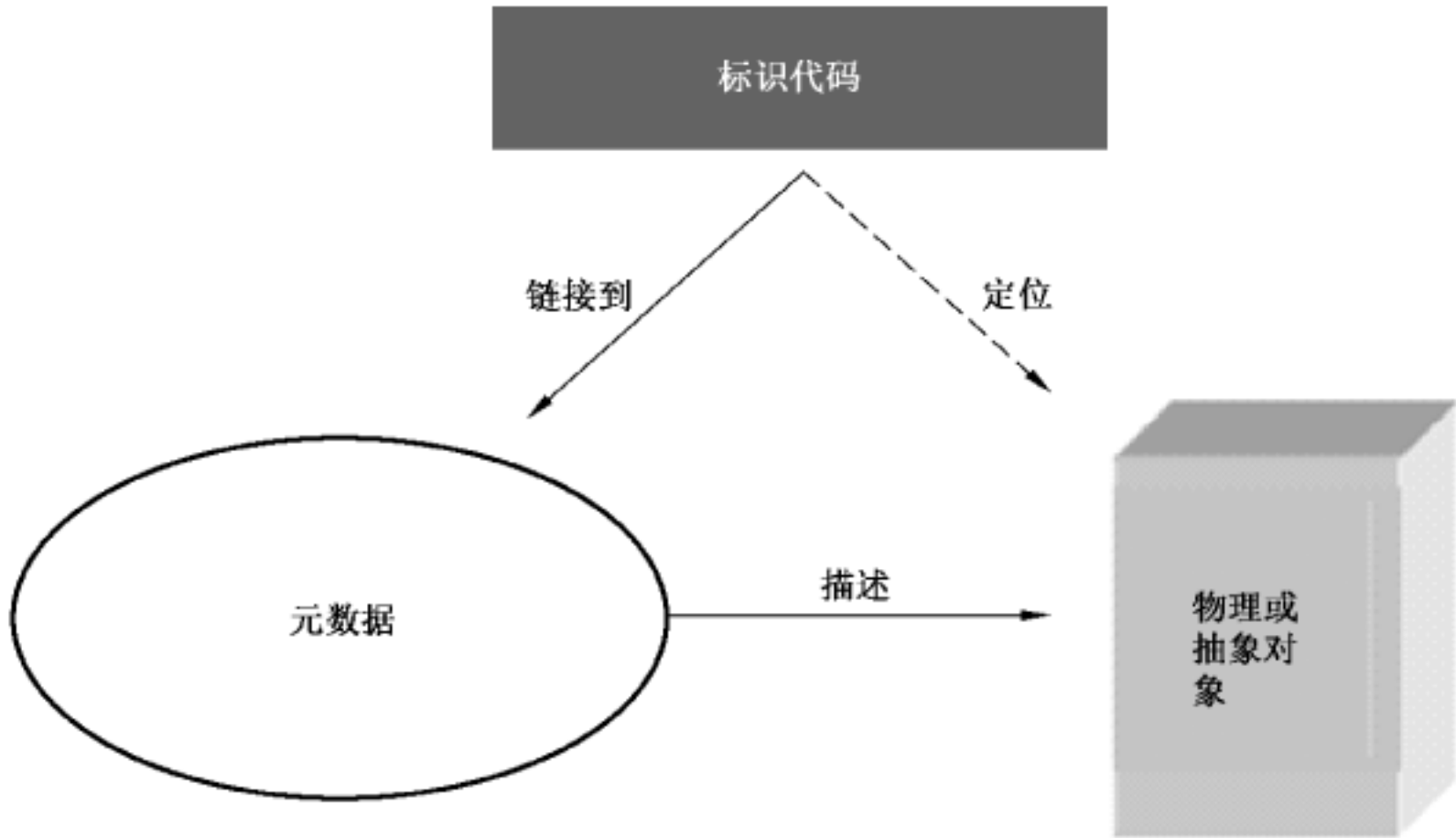


图 1 引用机制图解

当不与对象关联时,标识代码也可是指向信息本身。

4.3 永久性

标识系统的永久性要求主要包括:

- 选择永久域;
- 此域中产生标识代码的稳定规则;
- 允许检索标识代码元数据(什么时候及由谁生成代码)的寄存器。

负责标识系统的组织的内部和外部结构可能会随着时间而变化,以满足对内和对外业务的需求。使用标识符的 IT 系统环境也会随时间变化。

不过,所用标识系统应确保同一个标识代码决不会描绘两个不同的对象,并且,在同一域内一个对象不能有一个以上的标识代码。

原则上描述一个已建立并使用的标识系统的信息不得删除。

除非标识系统说明中另有规定,在确认没有人会引用标识代码前(例如,之前标识的项目已过期),不得重复使用标识代码。

注：应以现有国际、区域或国家法律(如有)为准。

因此,从时间上来看,标识系统应独立于公司或其他组织中不稳定的内部组织机构,且独立于所使用的计算机系统环境。

4.4 对象种类

一个类型是具有一组相同特征的对象类。根据共同特征的数量,类型可以是非常通用的也可以是



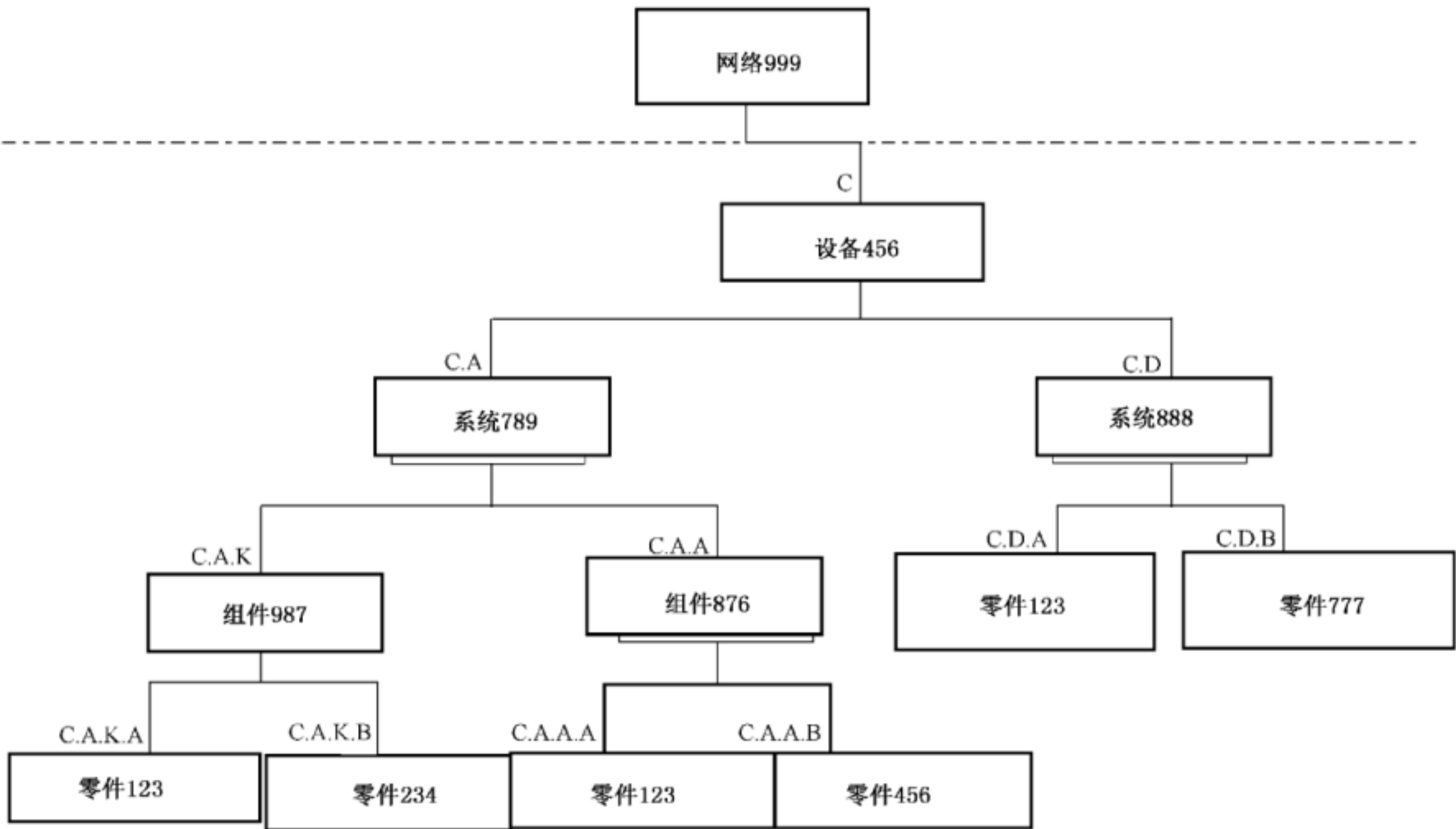
- 相当具体的。
- 通用对象类型,例如 IEC 81346-2 与 ISO/IEC 15418 中所描述的由一串字符所标识的类型;
  - 许多产品,如电动机、变压器或接触器,各自具有共同特征按规格范围进行系列化设计。在这种情况下,整体的范围标识符可以作为类型代码(类型名称),每个规格另有具体的标识符;
  - 一个产品系列的每个产品变体都有自己的标识代码;
  - 产品商业包装会导致进一步分类,例如需要不同的标识代码来区分装有 1、5 或 10 件产品的包装。

个体(对象)是产品类型一个单体,与其用于何处无关。以上提到的产品类型的每个产品单体都需要单独标记。如果不必要或实际上不可能对个体进行区分,可以使用批次代替标识。

注:在本部分中,术语个体(对象)不包括人类个体。

一个类型的事件是指设备或系统中的一类应用,不考虑它是哪个个体。

图 2 说明了类型和类型事件之间的关系。表 1 列举了各种类型的标识符、类型事件以及在不同环境下的个体的例子。



注:点划线下方的“网络 999”中标识为事件的所有对象。本示例中表示“顶部节点”的“网络 999”不能标识为事件。

图 2 类型(通过数字标识)事件(通过字母串标识)的关系树状结构图

表 1 产品标识符环境的应用

产地/主要用途	类型	事件的类型	个体
开发	类型命名 产品编号	(参照命名)	—
工程	类型命名 产品编号	参照代号	设备号码

表 1（续）

产地/主要用途	类型	事件的类型	个体
制造	类型命名 产品编号	参照代号 零件参照	序列号,批次号,批号
市场、销售、及出货	类型命名 产品编号	参照代号	序列号、批次号、批号、包 装编号、运输编号、订单号
使用和维护	类型命名 产品编号	参照代号	设备号码、序列号、资产号码

4.5 标识对象的变更

从使用角度考虑只要对象的相关性是相同的,变更对象应使用相同的对象代号标识。

此举的必要性在于在相关环境对对象引用时避免对对象代码的变更,从而避免变更带来的潜在风险。

从使用之外的角度来说,在固定标识代码的相关环境管理这种变更,完整的对象标识除包括标识代码外,还至少应包含下列中的一种:

- 版本号;
- 序列号和/或批号/批次号;
- 生产日期、包装日期或截止日期。

4.6 标识模式

附录 B 提供了参考用途的信息模型。

在下列条中,通过简化信息模型对属性进行了补充说明。

图 4、图 6 和图 8 中给出了实体-关系,信息模型见图 3。

- 从内指向外的“ENTITY”用粗体大写字母;
- 用椭圆表示相关的实体;
- 实体和相关实体之间的关系用连线表示;
- 实体和相关实体之间连线上面的文字描述他们之间的关系;
- 关系和实体组合构成了数据元素类型的属性;
- 用小数点隔开的两个数字表示属性出现次数:第一个数字表示最小的出现次数,第二个数字表示最大的出现次数;
- 关系及相应出现次数的指示位于定位关系线的同一侧;
- 在信息模型中,实体的名字用大写字母表示,相关实体的名字用小写字母表示。

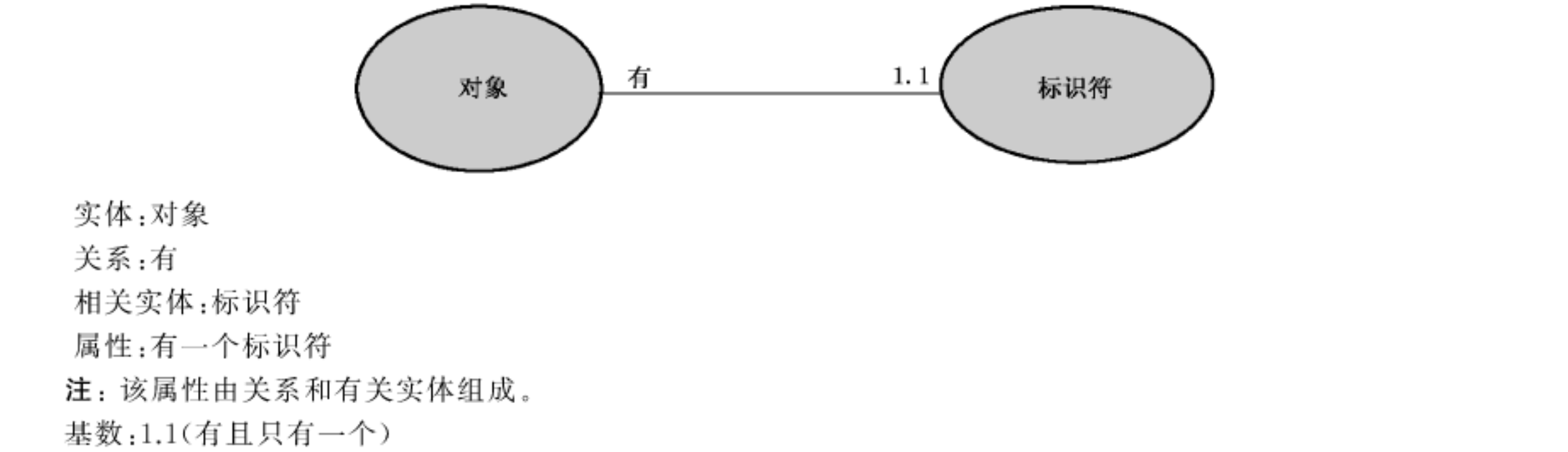


图 3 信息模型原理

## 4.7 标识对象属性

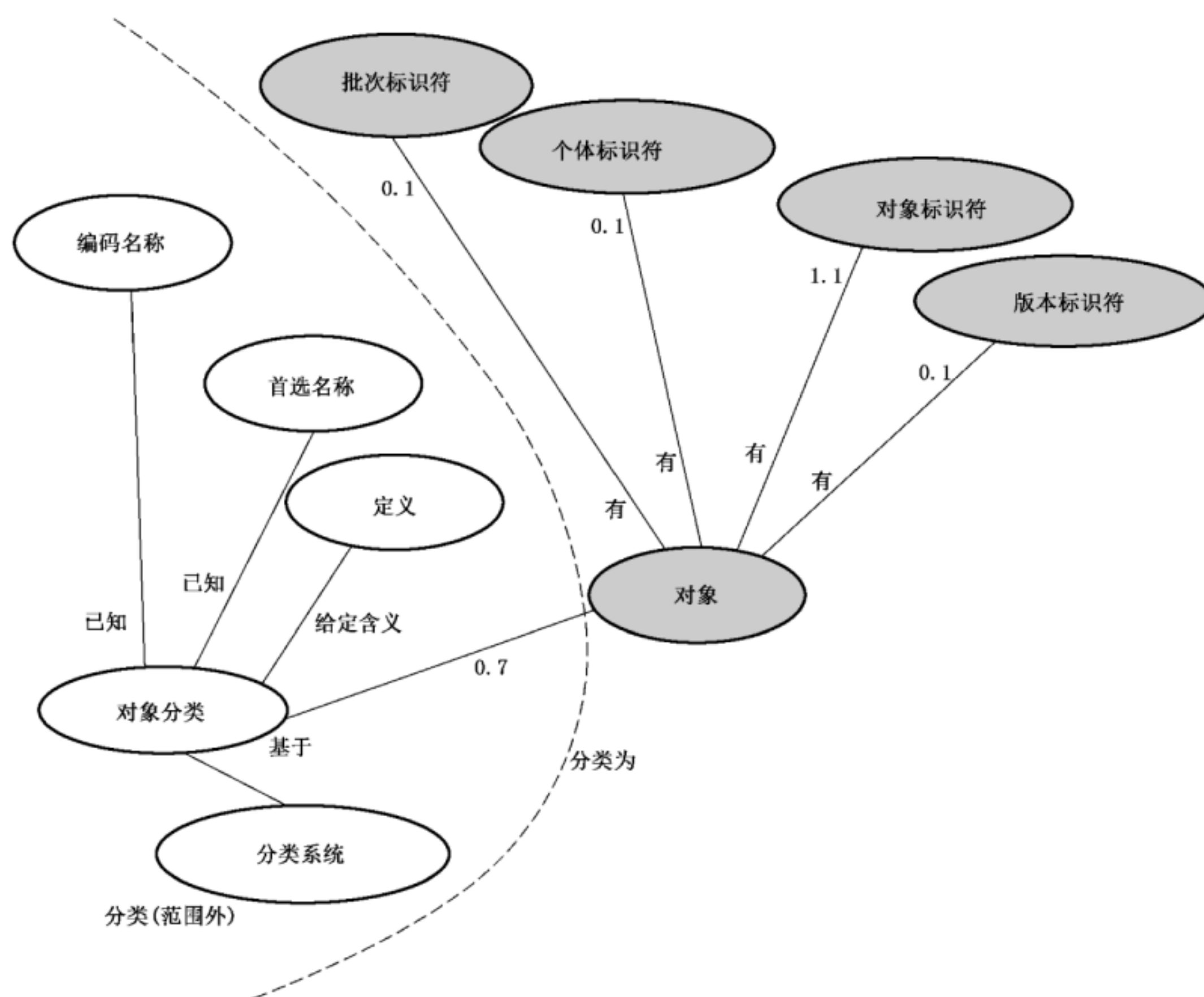


图 4 标识给定域中的对象属性的示例

对象具有标识属性,见图 4。每一个标识属性都有一个值,即相应的标识代码。标识代码属于一个已定义且标识的域。

## 4.8 注册域标识

一个域可能是另一个域的一部分,并标识。广义域内分配的标识代码应是明确的作用于部分域的域代码(域 ID),见图 5。第 6 章中所述原则也适用。

注:如果存在更大领域的注册机构,适用 6.1.2,否则适用 6.1.3。



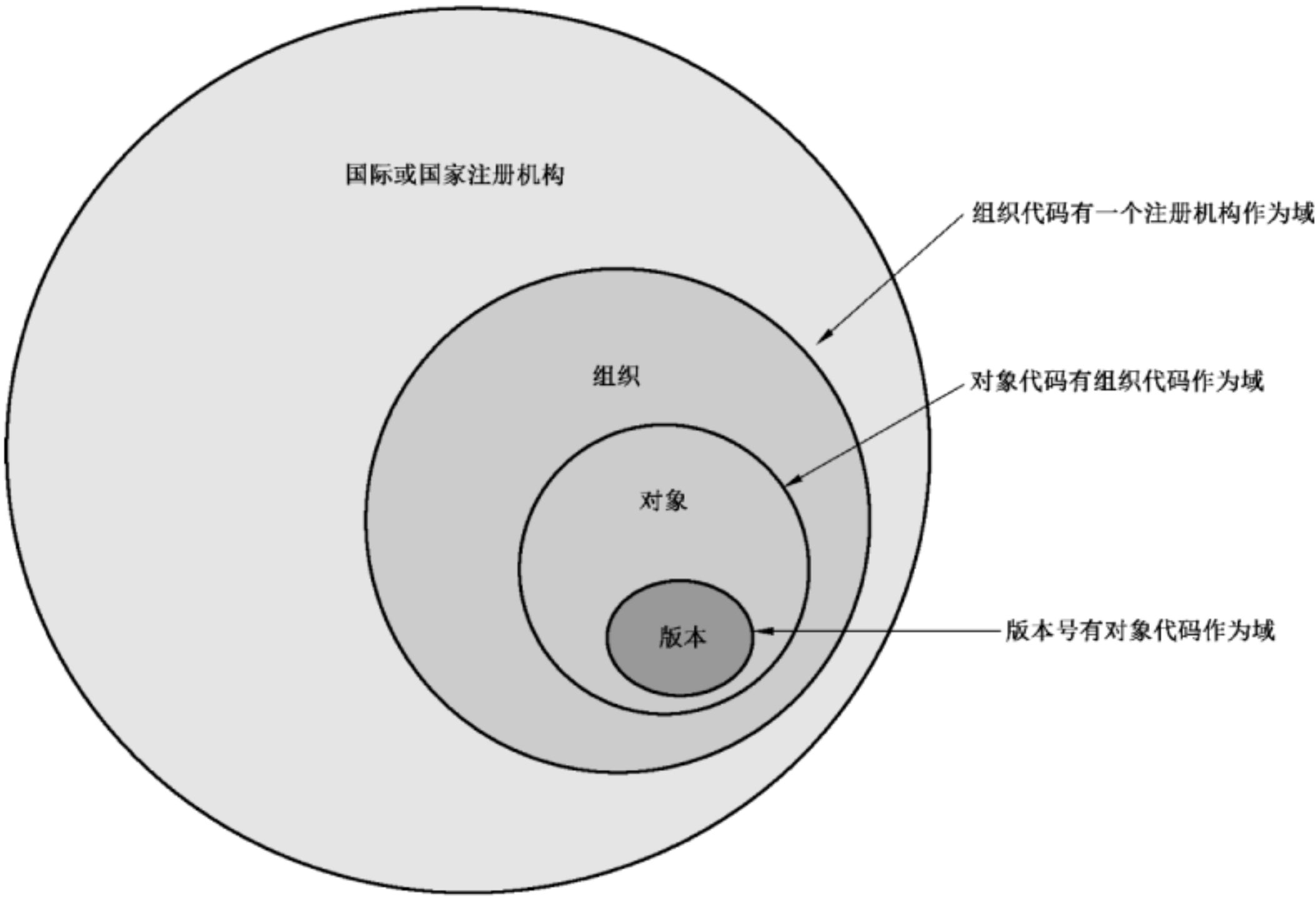


图 5 域的图解

这一原则是用来实现全球范围内标识的唯一性。签发(本地)的域内标识,应依次逐级在更大域内标识,直至全域公认的域标识[国家或国际级(区域及域)]。可以由一个国际注册机构(依据 ISO 6523-1 或 ISO/IEC 15459-2 要求)进行注册,也可以由一个国家级注册机构进行注册,然后再通过(依据 ISO 3166-1 所确定的)国家代码予以确认。

一个对象的完整标识,需提供所有相关域的域代码。

域的相关信息如图 6 所示。如果未按上述规定为某一域分配域代码,则需要结合许多其他信息元素以实现相关环境下的唯一性。

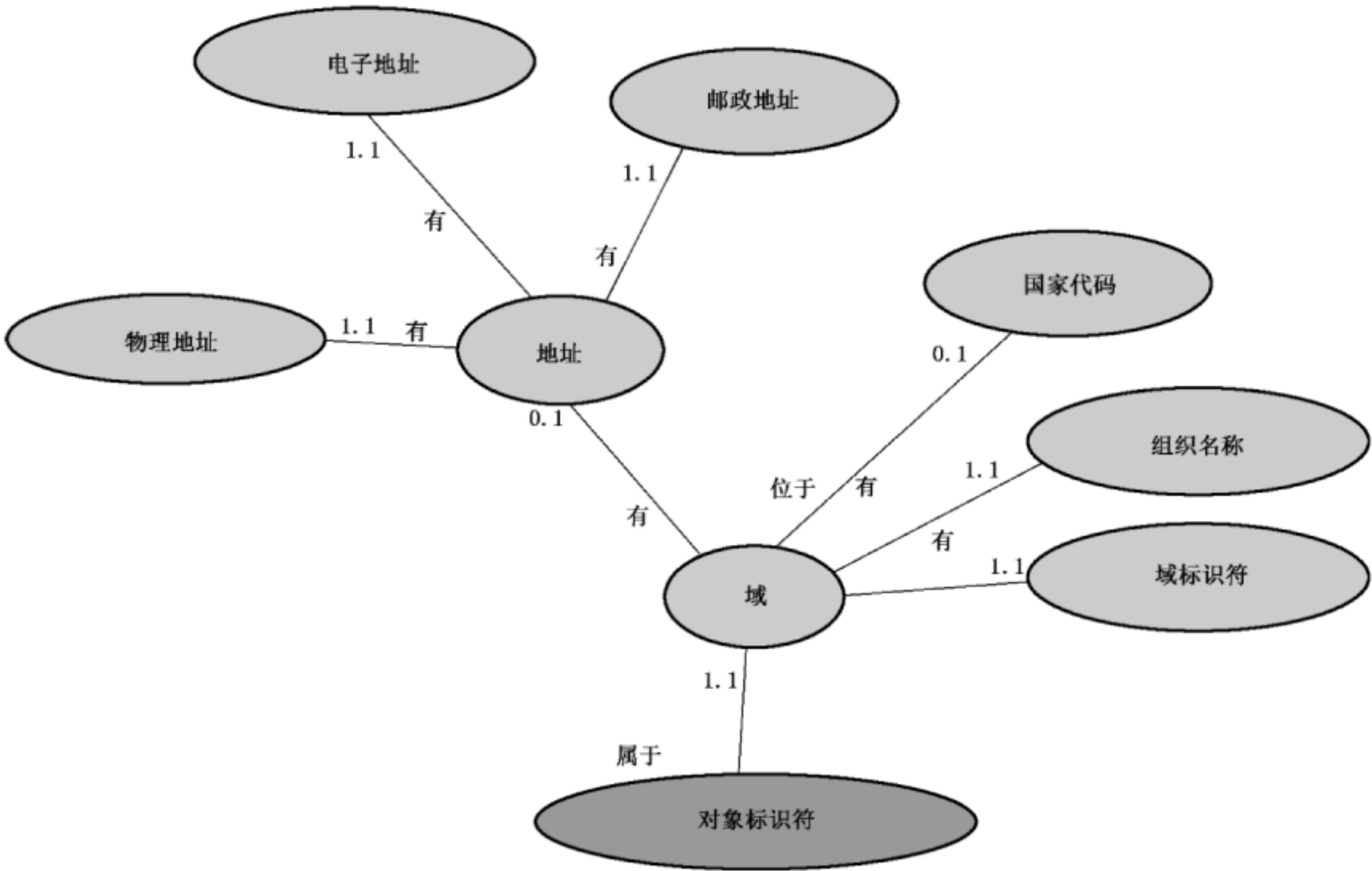


图 6 确定组织的域的标识

域、标识符和标识代码之间的关系见表 2。

表 2 域、标识符和标识代码之间的关系

签发域的标识	确定的对象	对象标识符	属性值
注册机构代码	组织	组织标识符	组织代码
组织代码	对象类型	对象标识符	对象代码
组织代码或对象代码	批次/批	批次/批标识	批次/批号
组织代码或对象代码	个体	个体标识符	序列号
组织代码或对象代码	事件	参考标识符	参照代号
对象代码	版本	版本标识符	版本号

4.9 同一个对象的多个标识

在给定域中,对象应有一个标识代码。通常标识代码将作为组织负责管理域内的对象标识,如制造领域。这取决于组织使用的标识方法,见 6.1.2 和 6.1.3。

出于不同原因,例如,用于营销或出口,宜为同一对象分配属于不同域的标识符(例如,分配一个基于国际认可的标识系统标识代码)。

强力推荐分管某对象多项标识工作的组织(责任对象所有者)运营一个注册系统,功能包括给对象分配对象代码以便交叉引用(见图 7)。例如,在引用对象版本变化的情况下,可知悉需要更新哪一个标识符。

注：在区域或全域层面上此类系统管理需要有偿服务。

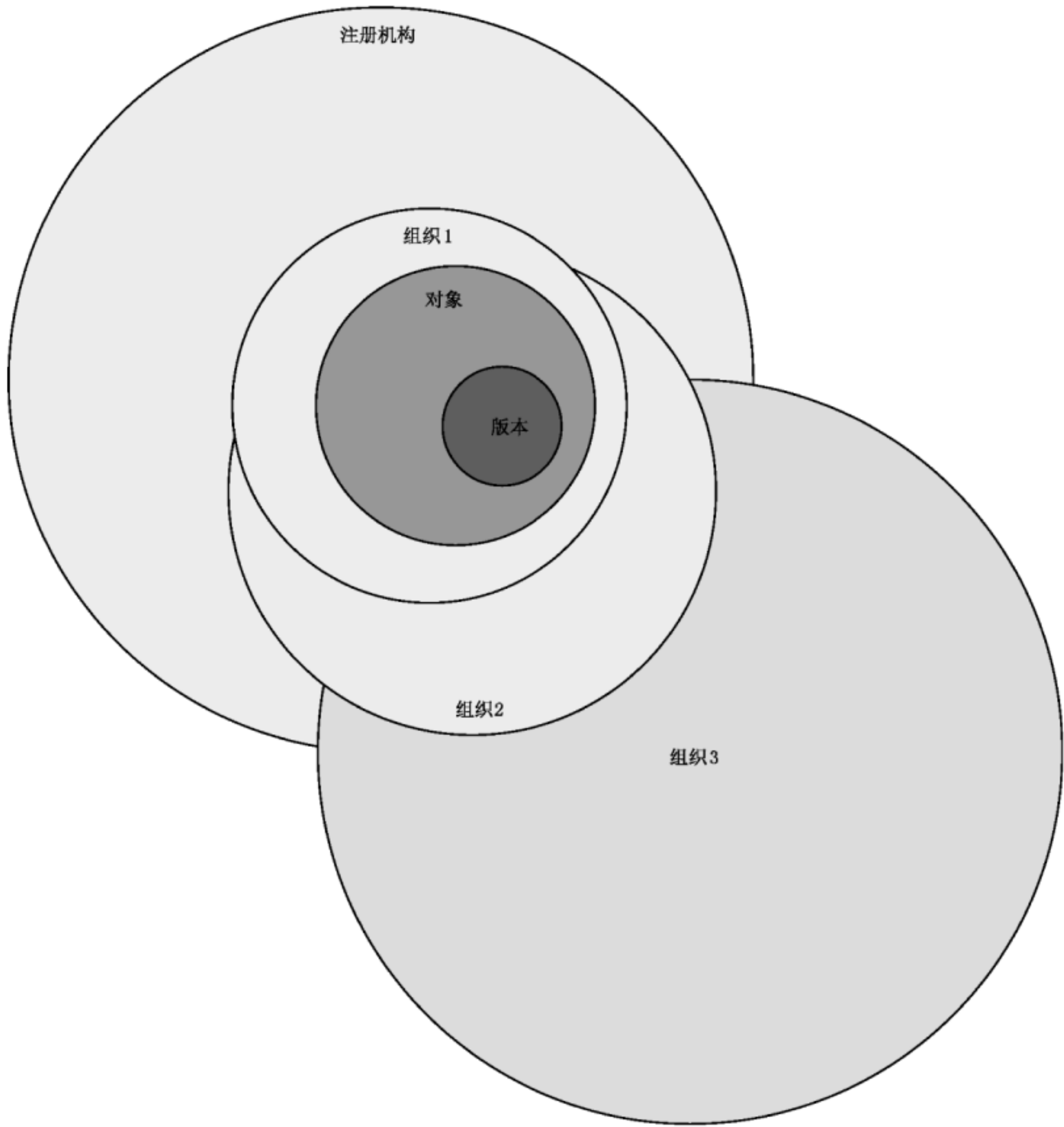


图 7 多个域的标识图解

4.10 标识代码的存储及使用

用于存储和管理不同标识系统中不同类型标识代码的系统不得限制字符串字符数量。但应满足在 6.2.1 中定义的信息交换全域的最大长度。

注：如果采用不同字符串标识,则不能限制组成串联字符串的字符数量。

## 5 标识系统文档

一种标识系统应通过以下要求进行记录：

- 其适用的域，可能包括其与应用于相同环境中的其他标识系统的关系；
- 需对此类对象进行标识验证；
- 在域中构建唯一标识代码的规则，见 6.2；
- 随着时间的推移应该如何对域进行管理的规则；
- 对系统管理相关工具的描述：至少有一个注册系统，见 6.2.2，可能附带标识代码生成器，见 6.2.3。

附录 C 给出了一个标识系统文档的示例。

应保持标识系统的完整性。为此应避免将标识系统用于预期使用以外的地方。

表 1 给出说明标识符的名称和将该域与其产生和使用内容联系起来的标识系统的通用范例。该表中的域为域的组织所负责的活动的名称。

## 6 标识代码的生成

### 6.1 原理方法

#### 6.1.1 总则

创建标识代码的方法主要有两种：

- a) 为避免重复和歧义，受管理的标识代码系统所生成的标识代码不能含有所标识对象相关的语义。标识代码表示的是标识对象所必需描述的信息；
- b) 按照定义的编码方案，标识代码将包括编码方案要求的描述信息。此类标识代码或是完整的描述或是引用元信息来提供附加信息。

#### 6.1.2 方法 1

方法 1 要求集中管理域中的标识代码注册系统。无需较长的标识代码，因此很灵活，引用的元信息数量多、可结构化且含有全部所需的信息粒度。标识代码可以轻松地随时保持稳定；同时元数据的内容可以适应当前需要（如重组、增加粒度）。

因此，该法推荐（但不限）用于对象标识符的使用。由于可追溯性、产品责任等原因，需要将对象标识符作为一个整体与一个组织联系起来，因此对象标识符的使用应受到严格的控制。

#### 6.1.3 方法 2

方法 2 不需要集中管理域中的标识代码注册系统，但所使用的编码方案应进行集中定义和管理。

因此标识代码的生成可以分散化，但为了确保唯一性，可能需要创建子域。通常，运用该方法会产生较长的标识代码，因为长度取决于代码所需属性的数目。其对信息需求的变化很敏感；当所描述对象本身不变时，增加编码属性或现有属性发生变化都会导致编码系统变更。

这种方法主要推荐（但不限于）用于标识在有限环境内发生的（信息或其他）对象事件，例如配送或库存。

方法 2 还有一个缺点就是数字不够用，因为在设置编码规则时不可能预见到未来可能出现的需求。

注：按照 RFC4122 (ISO/IEC 9834-8:2004)，关于分散化的可能性的极端例子为 UUID 系统。UUID 是基于时间标志和节点 ID 的 128 位长的标识代码，并保证空间和时间上的唯一性，无需注册。UUID 示例：b5ef6610-b746-



11da-a94d-0800200c9a66。UUID 系统通常适用于计算机系统的全域标识,但是由于标识代码长度的原因仅机读。

6.2 标识代码结构

6.2.1 总则

一个标识代码应包括一个字符串,并应符合 6.1.2 和 6.1.3 的规定。

组成标识代码的字符应从 ISO/IEC 646 国际参考版(IRV)的 G0-集中选择,不包括国家的或有固定用法的字符。

注:国家使用的或有固定用法的位组合包括 4/10、5/11-5/14、6/0 和 7/11-7/14,请参见 6.1.3 和 ISO/IEC 646 表 4 和表 5。

人工可读符号的推荐字符为:数字 0, …, 9 和大写拉丁字母 A, …, Z, 即一共 36 个字符。

对于主要用于机器读取的符号,也可以使用小写字母。

对于特定的应用,可以使用下面的标志:井号(‘#’)、和号(‘&’)、加号(‘+’)、连字号/减号(‘-’)、句号(‘.’)、斜线(‘/’)、冒号(‘:’)、分号(‘;’)、等号(‘=’)、小于号(‘<’)、大于号(‘>’)。

符合本部分的字符见附录 D。

如无特殊含义,标识代码不必很长,如表 3 中所示,为 34 个字符(不包括字母 I 和 O)。

表 3 可能的标识代码数量

字符数量( <i>n</i> )	可能的标识代码数量(34 <sup><i>n</i></sup> )
3	39 304
4	1 336 336
5	45 435 424
8	1,78 * 10 <sup>12</sup>
10	2,06 * 10 <sup>15</sup>
20	4,26 * 10 <sup>30</sup>

因字符串中的字符数量取决于其应用领域,所以并没有在此标准中对其进行限制或指定,但应考虑下列因素:

- 为了实现人工读取和管理,标识代码应尽可能短,可行即可。
- 为实现人工可读,标识代码避免使用具有类似字形的字母:大写字母 O,以免与数字 0 混淆;不要使用大写字母 I,因为其易与(小写字母 l 和)数字 1 混淆;由于并非将所有字体用于表示标识代码,所以要清楚区分这些字符。
- 对于主要通过条形码、RFID 等计算机可读的标识代码,ISO/IEC 15459-1 和 ISO/IEC 15459-4 推荐其以 20 个字符为限,EDIFACT 允许以 35 个字符为一个传送单元。在 ISO 7372 和 IEC 61360-1中给出了数据元素类型的长度信息。

需要一个接收系统进行数据交换,该系统能够接收来自发送系统的完整长度的字符。为此,推荐使用多达 256 字符的可变长度。

6.2.2 发行(注册)

每个被分配的标识代码应由发行者在相关域内进行注册,以确保未重复签发。

出于实际原因,组织中的此项职责可能会委派到不同的组织单位中。

注 1: 职责分派的方式可以是正式创建标识代码起始位置是字符的子域,也可以是为组织单位分配有限的标识代码

子系列。使用的字符位置不能用于解释标识对象的“含义”，因为指定系列的责任会随时间变化。

注2：对标识对象信息的管理不在本部分的范围内。更多信息请参阅，IEC 与 ISO 共同制定的关于文档管理的国际标准 IEC 82045。

不得在一个域内重复使用已注册的标识代码。

### 6.2.3 标识代码生成器

标识代码生成器通常为(但不限于)一个依据既定规则创建标识代码的软件程序，可确保不会发出重复标识代码并可记录结果。

标识代码生成器通常集成到计算机系统中，用于创建系统所需的标识代码。

如果此类标识代码要与外部系统交互，例如，用作产品标识代码，则标识代码的创建规则应与该标识代码所属域内的标识代码的长期规则一致，因为相对于(例如)投资产品的全生命周期，计算机系统的全生命周期可能较短。

如果在某一机构内使用多个此类标识代码生成器，可按 6.2.2 的规定划分标识代码产生的职责，但这种情况下不是针对组织单元，而是针对标识代码生成器。

注：设置标识代码生成器时，宜添加子域或部分系列，以便用于培训、教育、软件测试、故障探测和类似用途。另外，人们使用旧代码或重新界定旧事物，因为不允许他们创建新数字，而不触发集成环境中的其他机制。

### 6.2.4 验证

由于标识代码可能会在介质之间多次传输，例如，人工键入或扫描(手动或自动)在进一步处理前检查代码的完整性非常有用。在比较简单的情况下，可以通过格式校验或更安全的校验码完成。

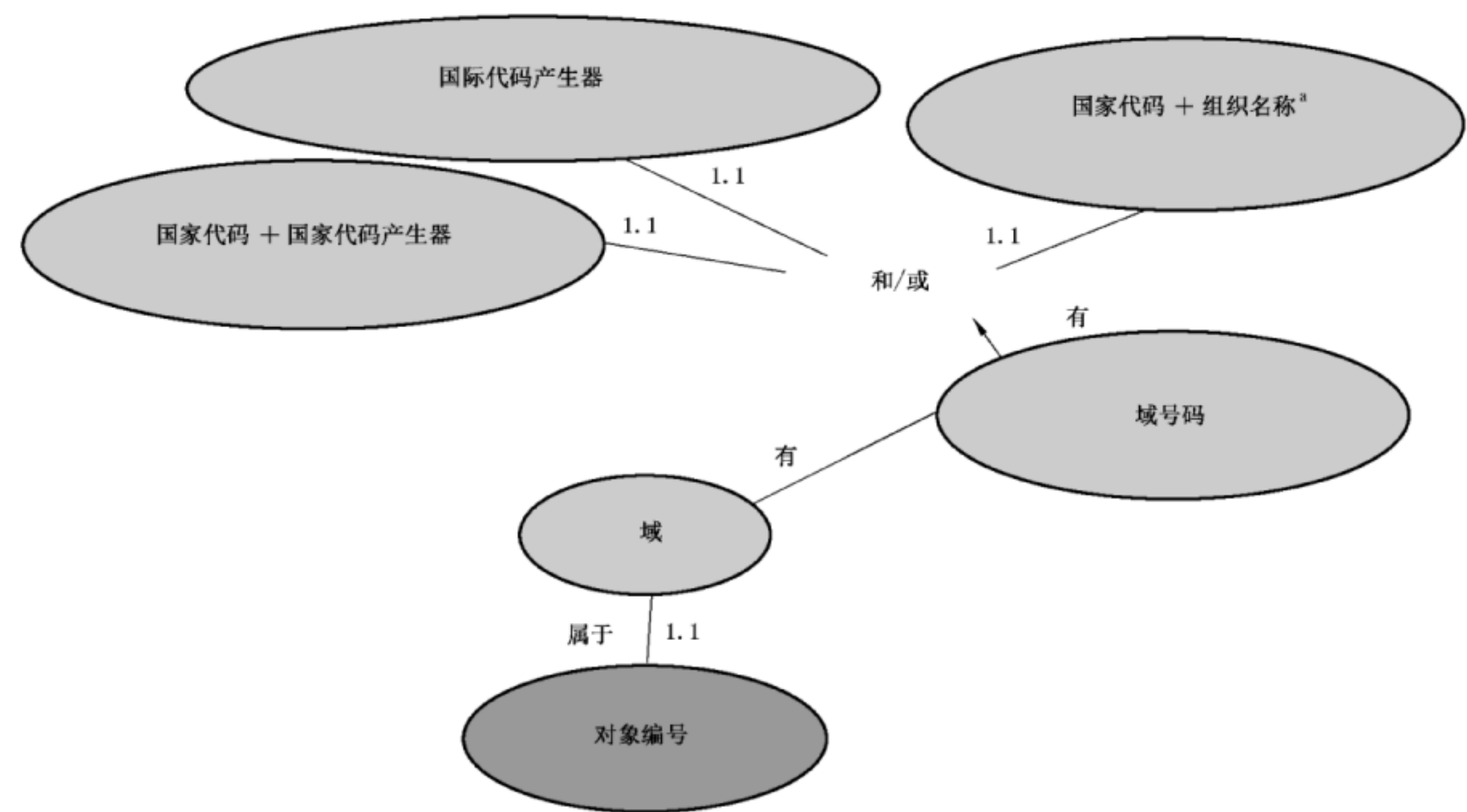
通过格式检查或校验码进行验证适用于单个标识代码或多个标识代码串联。

ISO 7064 提供了有关校验码应用程序的信息。如果使用校验系统，应可以公开访问标识系统中使用的校验系统。建议提供允许分散检查的工具。

## 7 全球范围内标识

要实现全球范围内的明确标识，可以使用以下两种不同的方法：

- 用域标识补充组织内有效的注册(本地)域也可能用国家标识(代码)进一步补充。
- 通过一个国际认可的注册机关注册确定并标识为全球的标识系统的域。



\* 如在一个国家出现相同名称的公司,则需要更多信息。

图 8 区域标识图解

8 标识代码的描述和展示

8.1 在计算机系统中使用的表示

对于用交换介质的信息,如计算机系统、条码、无线射频标识等等,各不同的标识符应与其关联的标识代码一同传输。

表示是由物理电子传输格式定义的:

- 交换文档中适用的不同标识代码间的分隔字符种类和;
- 标识代码中不同标识元素间的分隔字符种类。

作为分隔字符的替代性解决方案,字符 GROUP SEPARATOR SIGN (GS)(ASCII 字符 029)可根据 ISO/IEC 15434 进行使用。

这允许 IT 系统按照 8.2.2 及 8.2.3 的规定将接收到的信息向读者展示。

注: IT-系统宜保留对象标识的不同标识符的结构,以减少电子交易流程并简化后期处理步骤,而非为了展示将该结构缩减为一个字符串。如果要打印对象的标识,例如,用条形码,数据结构被展开为一个字符串。条码扫描软件读取字符串,并再次将字符串转换成之前的结构。

8.2 可读性

8.2.1 综述

标识代码应由单个字符串表示。

如果遇到标识代码类型理解歧义的问题,所用标识代码对应的标识符应以可视方式表示,可以是人可读的文本也可以是代码相关标识代码的前面应加一个分隔符空格(“ ”)。

如果完整的对象标识要求体现关联域的标识代码,则应将其与它们的标识符一同明确列出或按以



下规则将其串联成单个字符串。

- 如果完整的标识符被串联成单个字符串,则应从最高层域的标识符开始,从左至右表示。
- 不同的标识符可以按发行者的定义用一个分隔符分开,分隔字符应无语义。

注:分隔符仅用作处理系统中的控制字符。

### 8.2.2 人可读的串联标识符的表示

由于可以将几个标识符串联来表示,所以可以用发行者定义的分隔符对相关的标识代码进行视觉分隔。此类分隔字符不具有任何意义。按照从顶层(最大域)到较小元素(最小域)的顺序依次表示。

为了增强其对于读者的可读性,以下字符可以作为分隔符:空格符(“ ”)或下划线(“\_”)。其他分隔字符需要共同商定。

示例:

根据 ISO 13616 国际银行账号的结构由以下序列组成:

- 编码标识符 IBAN,其后是字符空格(“ ”);
- 根据 ISO 3166-1 的双字母国家代码;
- 完整标识代码的两位校验码;
- 基本账户号码最多为 30 位,包括所标识银行的银行标识代码和账户标识代码,按发行者的定义用空格进行分隔(“ ”)

德国国际银行帐号表示示例:

IBAN DE21 7005 1995 0000 0072 29

### 8.2.3 人可读的多个标识符的表示

如果同一对象由多个标识代码来表示,每个标识代码应位于标识代码类型的标识符名称(可能是缩写或编码)之后,以免产生歧义。

示例:

ABC	12345678
DEF	9012345678

## 8.3 子域表示法

如果在同一标识系统内使用不同的表示法,例如,使用不同数量的字符来表示系统内的子域,则应通过连字符(-)加字符串(例如字符的数量)的方式,来扩展该标识系统已编码的标识符,以表示相应子域。

编码域标识符后为空格,其后为标识代码。

示例(域标识符的值为 ABC):

ABC-8	40804330
ABC-13	400267801721

## 8.4 域标识符的应用

通常标识代码本身就足以充当标识符,因为在特定的环境中域是已知且预设的。

- 产品铭牌上的标识代码体现的是属于铭牌上的注明了名称或商标的组织所对应的定义域。
- 用作文档号的标识代码体现的是属于文档标题栏或信头中注明了名称或商标的签发组织对应的域。
- 文档(零部件列表、文档、文档相关的参考文献等)内容中列出的标识代码体现的是属于文档的页眉或页脚中注明的名称或商标的签发组织对应的域。

在提供或使用标识代码时,如果域所处环境不明确,应确切说明有关域标识符的信息。

域代码在不同的环境中可以用不同的方式表示：

- 作为屏幕上显示的字段名,表格和文本中的标题等,域可在字段名、列标题等中显示；
- 在定期处理多个不同域标识代码的应用软件中,将域代码(或域名)及标识代码被定义为成对事件的相关联的字符段；
- 作为包含域代码的标识字符串的串联部分,标识代码可通过分隔符来可视化,见 6.2。

在各方通信中,域标识符和标识代码应作为两个独立但成对的标识符进行传送,除非域不相同(在这种情况下,该域可以预设)。

## 9 组织变更建议

永久性要求很容易满足,但前提是管理域的签发机构在一定时间内也保持稳定状态。然而,组织有时会分立或与其他组织合并。合并时,新组织将面临两个或多个域的情况,在不理想的情况下,这些标识代码可能不能合并,因为有些标识代码可能已经不再唯一。为避免在同一公用 IT 系统环境中出现冲突,有必要对原组织中的标识代码进行处理。

在这种情况下重新标识对象,既不实际也不是最佳的解决方案。大多数情况下这样做代价极高(因为需要对现有文档进行全面变更),例如,任何情况下产品号码都被打印在长期以来所销售及分销产品的标牌上。

通过域标识符创建全球内唯一的标识的方法也是出于避免冲突的目的。在这种情况下,域代码描述了前签发机构,即曾生成标识代码的编码方案的所有者和管理者。该域成为新用户域内的一个子域。

注意,多数情况下过去的组织将不复存在,因此也没有可用的国际或国家组织代码。因此,在新的组织中域代码可能需要当作一个完全独立的实体来处理。

## 10 合规性

声明符合本部分的标识系统应遵守第 5 章到第 9 章(包括引用第 4 章的部分内容)的规定。

附录 A 给出了如何定义和界定此类标识系统的示例,附录 C 给出了怎样记录标识系统。

如果全球或地区使用的标识系统中的数据元素符合 A.2.2、A.2.3 或 A.3.2 中规定的至少一个数据元素子集,就可以声明符合 GB/T 38000 标准。

如果全球或地区使用的标识系统中的数据元素符合 A.2.4 或 A.3.3 中规定的至少一个数据元素子集,就可以声明符合 GB/T 38000 标准。

如果对象-边界域中的标识中的数据元素符合 A.4.2 或 A.4.3 中规定的至少一个数据元素子集,就可以声明符合 GB/T 38000 标准。

## 附录 A

### (资料性附录)

### 标识系统的类型

#### A.1 总则

以下各条列出了标识系统的非详尽清单。

A.2 和 A.3 描述了用于将对象与组织关联起来的标识系统。

A.4 描述了用于将对象与其他对象(系统)关联起来的标识系统。

EXPRESS-模型中使用表达式用括号括起。

#### A.2 通过代码对管理者的标识系统进行验证

##### A.2.1 综述

通过代码对管理者进行验证,需要使用注册机构(或代表该机构运作的发行机构)给管理者分配和注册的组织代码(coded\_organization\_id)。此类注册机构可以是国际性的也可以是国家性的。

##### A.2.2 国际注册标识系统

在国际层面使用的注册标识系统要求向国际层面作为域管理者的机构进行注册,注册过程中,管理者获得相应的域代码[domain\_id]作为该域内对象标识的属性。

对象的全球标识需要以下相关属性:

- 组织代码;
- 域代码;
- 对象代码(由标识系统的管理者分配);

版本标识在国际层面的注册的标识系统中并不常见,但如果使用,应添加以下属性:

- 版本号(由负责对象的组织分配)。

如果是串联的,那么串联的字符串应具有以下结构:组织代码+域代码+对象代码+版本号。

参见附录 B 中的 EXPRESS 模型:[coded\_organization\_id+domain\_id+object\_id+version\_id]。

示例:企业标识系统在国际注册。标识包括序列号和批次号/批号,以便这些编号与组织域直接关联(对照 A.4.2 例 2 给出的其他方法)。

##### A.2.3 国家层面的注册标识系统

在国家层面使用的注册标识系统要求向国家层面域管理者的机构注册进行注册,注册过程中,管理者获得组织代码[coded\_organization\_id]以及域代码[domain\_id]作为该域内对象标识的属性。为了使域代码在全球范围内无歧义,标识还应增加国家代码属性[country\_code]。

对象的全球标识需要以下相关属性:

- 国家代码(符合 ISO/IEC 3166-1);
- 组织代码;
- 域代码(由注册机关分配);
- 对象代码(由标识系统的管理者分配);
- 版本号码(由负责该对象的组织分配)。

如果是串联的,那么串联的字符串应具有以下结构:国家代码+组织代码+域代码+对象代码+版本号。

参见附录 B 中的 EXPRESS 模型:[country\_code+coded\_organization\_id+domain\_id+object\_id+version\_id]。

示例:机构在国家层面注册的企业标识系统。标识包括产品号、序列号和批次号/批号,以便这些编号与组织域直接关联(对照 A.4.2 例 2 给出的其他方法)。

#### A.2.4 子域的注册标识系统

可以依据国际或国家注册系统(较高级别的域)对子域的标识系统进行定义,域代码可以用于这个域的所有子域。

在子域中使用的注册标识系统要求向较高级别的作为域管理者的机构注册进行,注册过程中,子域管理者获得组织代码[coded\_organization\_id]以及域代码[domain\_id]作为该子域内对象标识的属性。

子域中使用的对象的全球标识要求包括注册域标识符以外,还包括在本地注册的域标识符:

——...+域代码(由较高级别域的管理者分配);

——对象代码(由标识系统的管理者分配);

——版本号(由负责该对象的组织分配)。

如果是串联的,串联的字符串应从国际或国家级注册域的标识符开始,其后为以下结构:

...+域代码(较高级别的域)+域代码(子域)+对象代码+版本号。

参见附录 B 中的 EXPRESS 模型:[...+domain\_id(较高级别域)+domain\_id(子域)+object\_id+version\_id]。

示例:子域示例:ABC-8,ABC-13,DEF-A,DEF-B,DEF-C。

### A.3 用名称识别管理者的标识系统

#### A.3.1 综述

通过名称识别管理者的标识系统要求提供确保无歧义识别组织的充足信息。

#### A.3.2 全球注册标识系统

在全球进行识别,需要指定标识系统的管理人:

——国家名称或国家代码;

——组织名称。

参见附件 B 中的 EXPRESS 模型:[country\_name|country\_code+organization\_name]。

在某些国家并未要求国内机构名称是唯一的。在这种情况下,至少应将物理地址信息补充到上述信息中,如附录 B 所示。

除此之外域内对象的全球标识还包括:

——...+对象代码(由标识系统的管理者分配);

——版本号码(由负责该对象的组织分配)。

如果是串联的:对象代码+版本号码。

参见附录 B 中的 EXPRESS 模型:[...+object\_id+version\_id]。

示例:没有注册的企业标识系统。其中包括产品号码、序列号和批次/批号等系统,以便这些编号与组织域直接关联(对照 A.4.2 例 2 给出的其他方法)。

#### A.3.3 子域的标识系统

可以依据国际或国家注册系统(较高级别的域)对子域的标识系统进行定义,域代码可以用于这个



域的所有子域。

为了识别与较高级别域之间的关系,使用子域注册系统时需要向作为较高级别域或此类域的子域的管理者机构注册进行注册。

此情况与上述 A.2.4 基本一致,不同之处在于较高级别域应按照 A.3.2 处理。

示例:没有注册的企业标识系统和企业层面建立和注册的子域。

## A.4 子对象的标识系统

### A.4.1 综述

子对象的标识系统用于将子对象从一个给定的方面与较高级别对象关联起来,标识方法见 A.2 和 A.3 中的规定。

### A.4.2 单级标识

在实际环境中高级别对象作为其子对象标识系统中的域。

其他环境中的对象标识需要:

- 对象代码(对实际环境中最高层对象的,用作较低层对象的域);
- 对象代码(子对象的,此域内)。

参见附录 B 中的 EXPRESS 模型:  $[\text{object\_id}(\text{for the domain} = \text{domain\_id}) + \text{object\_id}]$ 。

示例 1:单级参照代号((IEC 81346-1 中的顶级节点标识),本地电话号码。

示例 2:单个对象类型单体的序列号用作序列号码的域标识。

### A.4.3 多级标识

A.4.2 中定义的对象代码  $[\text{object\_id}]$  可以依次用作其子对象的域代码。通过这种方法,可以逐层细分。

其他环境中的对象标识要求:

- 对象代码(实际情况中“最高级别的对象”,用作其次级对象的域);
- 对象代码(子对象的,在上面定义的域中唯一且用作其次级对象的域);
- 其他。

参见附件 B 中的 EXPRESS 模型:  $[\text{object\_id}(\text{用作域} = \text{domain\_id}) + \text{object\_id}(= \text{domain\_id}) + \text{Etc}]$ 。

示例:多级参照代号((IEC 81346-1 中的顶级节点标识)、国际电话号码、IP 地址。

附录 B  
(规范性附录)  
参考信息模型

B.1 综述

本附录中的参考信息模型是根据本部分的概念和方法建立的正式模型。  
它是数据交换的规范,即数据传输和交换时应符合这一参考模型。

B.2 参考模型

B.5 中的参考模型是用图形方式表示,B.6 中定义的应用对象的结构和约束条件中参考模型的图形使用 EXPRESS-G 表示。参考模型是独立于任何实现方法。

参考模型描述了需求设置,尽可能使用 ISO 10303 系列标准的应用参考模型的子集。

本模型无论是在整体资源模型框架内还是 ISO 10303 系列标准开发的应用参考模型框架内都是不完整的。它只是根据本部分的要求建立的完成模型。

注 1: EXPRESS-G 的介绍,见 <http://tc3.iec.ch/txt/xpress.pdf>。

注 2: 本附录 B.3 实体和属性名仅提供英文版。

B.3 实体和属性表

本条提供了一个按字母顺序编排的实体和属性清单,参考信息模型描述见 B.4。

(INV) has_version S[0:?]	25
(INV) identified_by S[1:?]	24, 25, 28
addition	27, 29
Address	34
affecting	27, 30
associated_with	24
classified_as S[0:?]	26
coded_organization_id	33
Coded_organization_identifier	33
copy	27, 30
country_code	33, 34
country_name	34
custodian	31, 31
customer	31
decomposition	27, 30
derivation	27, 30
description S[0:?]	26
Domain	24
domain_id	25

Domain_identifier	25
Domain_relationship	25
domain_relationship_type	25
Electronic_address	34
employee	33
employer	32
id	26
identifier	33
identifies	25, 26, 28
is_role_of	31
license_holder	31
local_representative	31
maintained_by	24
manufacturer	31
Named_organization_identifier	34
Object	25
Object_class	26
Object_id	26
Object_identifier	26
Object_relationship	26
Object_relationship_type	27
Object_version	28
Object_version_identifier	28
Object_version_relationship	29
Object_version_relationship_type	29
Operator	31
Organization	33
Organization_identifier	33
Organization_name	34
owner	31
Party	30
Party_relationship	32
party_relationship_type	32
Party_role	30
Party_to_address_relationship	32
peer	27, 30
Person	33
Physical_address	34
Postal_address	35
referencing	28, 30
related_address	32
related_domain	25
related_object	27



related_object_version .....	29
related_party .....	32
related_to .....	26, 29
relating_domain .....	25
relating_object .....	27
relating_object_version .....	29
relating_party .....	32, 32
responsible_party .....	32
Role_type .....	31
sequence .....	28, 30
substitution .....	28, 30
superseding .....	28, 30
supplier .....	32
translation .....	28, 30
uses_classification_system .....	26
valid_domain .....	33
variant .....	28, 30
vendor .....	32
version_id .....	29
version_of .....	28
withdrawal .....	28, 30

## B.4 实体描述

### B.4.1 Domain

Domain 是建立 Domain\_identifier 与作为域管理者的组织之间关系的属性集合。如果是对象作为域,则需给该对象创建 Object\_identifier。

与域相关的数据如下:

- (INV) identified\_by S[1:?];
- maintained\_by;
- associated\_with。

#### B.4.1.1 (INV) identified\_by S[1:?]

表明 Domain 和 Domain\_identifier 之间的关系。一个域至少有一个标识符,可以有不止一个。

#### B.4.1.2 maintained\_by

表明域和管理者之间的关系。

#### B.4.1.3 associated\_with

属于一个组织的域可以属于与域相关的其他组织。

域可以任意地与其中标识的对象关联,此类对象的对象标识符可以作为这个对象的子对象的 domain\_id。

注:这种关联描述是根据 IEC 81346 的基于树状结构的参照代号,分层的参照代号是由此类域的标识代码串联

而来。

#### B.4.2 Domain\_identifier

Domain\_identifier 是域的标识。

与 Domain\_identifier 相关的属性有：

- identifies;
- domain\_id。

##### B.4.2.1 identifies

表明与已标识的域的关系。

##### B.4.2.2 domain\_id

表明域标识代码的字符串。

#### B.4.3 Domain\_relationship

此实体描述包含了域之间的关系。

相关属性有：

- relating\_domain;
- related\_domain;
- domain\_relationship\_type。

##### B.4.3.1 relating\_domain

指定与 Domain\_relationship 相关的两个域中的第一个。

##### B.4.3.2 related\_domain

指定与 Domain\_relationship 相关的两个域中的第二个。

##### B.4.3.3 domain\_relationship\_type

domain relationship\_type 指定的关系的含义。

常见的 domain\_relationship\_types 是：

- decomposition;
- substitution。

#### B.4.4 Object

Object 实体是建立 Object\_identifier, Object\_version and Object\_class 之间关系的属性集合。

与 Object 相关的数据如下：

- (INV) identified\_by S[1:?];
- (INV) has\_version S[0:?],
- classified\_as S[0:?]

##### B.4.4.1 (INV) identified\_by S[1:?]

表明 Object 和 Object\_identifier 之间的关系,一个对象至少有一个标识符,可以有很多个。

#### B.4.4.2 (INV) has\_version S[0:?]

表明 Object 和 Object\_version 之间的关系,对象不必有版本,也可以有很多版本。

#### B.4.4.3 classified\_as S[0:?]

表明从实体 Object 到实体 Object\_class 的关系,一个对象可以属于很多类。

### B.4.5 Object\_class

Object\_class 是定义一个对象多重分类的属性集合。

与 Object\_class 相关的数据如下:

- id;
- description S[0:?];
- uses\_classification\_system。

#### B.4.5.1 id

基于已定义的分类系统的、与特定对象种类相关的分类代码。

#### B.4.5.2 uses\_classification\_system

定义应用的分类系统的信息。

#### B.4.5.3 description S[0:?]

基于给定分类系统的、特定对象内相关分类代码的文本描述。

### B.4.6 Object\_identifier

Object\_identifier 表示的是由 domain\_id 标识的域内的对象的标识。

Object\_identifier 的相关属性有:

- identifies;
- related\_to;
- object\_id。

#### B.4.6.1 identifies

表明与已标识对象之间的关系。

#### B.4.6.2 related\_to

表明与 Domain\_identifer 之间的关系。

#### B.4.6.3 object\_id

表示对象代码的字符串。

### B.4.7 Object\_relationship

此实体描述对象之间的关系。

相关属性有:

- relating\_object;

- related\_object;
- object\_relationship\_type。

#### **B.4.7.1 relating\_object**

由 Object\_relationship 联系起来的两个对象中的第一个。

#### **B.4.7.2 related\_object**

由 Object\_relationship 联系起来的两个对象中的第二个。

#### **B.4.7.3 object\_relationship\_type**

object\_relationship\_type 定义了关系的含义。

常见的 object\_relationship\_type 有：

- addition;
- affecting;
- copy;
- decomposition;
- derivation
- peer;
- referencing;
- sequence;
- substitution;
- superseding;
- translation;
- variant;
- withdrawal。

##### **B.4.7.3.1 addition**

间接项目为直接项目提供补充信息。

##### **B.4.7.3.2 affecting**

间接项目影响直接项目。

##### **B.4.7.3.3 copy**

间接项目是直接项目的复制品。

##### **B.4.7.3.4 decomposition**

间接项目由直接项目分解而来。

##### **B.4.7.3.5 derivation**

间接项目由直接项目派生而来。

注：“基于”常常被用作派生的同义词。

##### **B.4.7.3.6 peer**

间接项目提供关于直接项目所要求的信息。等同项目对于完整理解项目含义十分必要。

#### **B.4.7.3.7 referencing**

间接项目引用直接项目。

#### **B.4.7.3.8 sequence**

间接项目依次排在直接项目之后。

#### **B.4.7.3.9 substitution**

间接项目代替直接项目。

#### **B.4.7.3.10 superseding**

间接项目接替直接项目。

#### **B.4.7.3.11 translation**

间接项目是由直接项目转化而来。

#### **B.4.7.3.12 variant**

间接项目是由直接项目的变体。

#### **B.4.7.3.13 withdrawal**

间接项目被取消,没有代替品。

### **B.4.8 Object\_version**

Object\_version 是建立 Object 和 Object\_version\_identifier 之间关系的属性集合。

Object\_version 相关的数据如下:

- version of;
- (INV) identified\_by S[1:?].

#### **B.4.8.1 version\_of**

表明 Object\_version 是对象的哪一个版本。

#### **B.4.8.2 (INV) identified\_by S[1:?]**

表明 Object\_version 和 Object\_version\_identifier 之间的关系。Object\_version(如果有)至少有一个标识符。

### **B.4.9 Object\_version\_identifier**

Object\_version\_identifier 是在由 object\_id 标识的域内为相关对象的 Object\_version 提供标识。

Object\_version\_identifier 相关属性有:

- identifies;
- related\_to。

#### **B.4.9.1 identifies**

表明与已标识的 Object\_version 的关系。

**B.4.9.2 related\_to**

表明与 Object\_identifer 的关系。Object\_identifer 的 object\_id 用作 version\_id 的 domain\_id。

**B.4.9.3 version\_id**

表示版本号的字符串。

**B.4.10 Object\_version\_relationship**

此实体描述的是 Object\_version 之间的关系。

相关属性有：

- relating\_object\_version;
- related\_object\_version;
- object\_version\_relationship\_type。

**B.4.10.1 relating\_object\_version**

由 Object\_version\_relationship 联系起来的两个 Object\_versions 中的第一个。

**B.4.10.2 related\_object\_version**

由 Object\_version\_relationship 联系起来的两个 Object\_versions 中的第二个。

**B.4.10.3 Object\_version\_relationship\_type**

object\_version\_relationship\_type 定义了关系的含义。

常见的 object\_version\_relationship\_type 有：

- addition;
- affecting;
- copy;
- decomposition;
- derivation;
- peer;
- referencing;
- sequence;
- substitution;
- superseding;
- translation;
- variant;
- withdrawal。

**B.4.10.3.1 addition**

间接项目为直接项目提供补充信息。

**B.4.10.3.2 affecting**

间接项目影响直接项目。

**B.4.10.3.3 copy**

间接项目是直接项目的复制品。

**B.4.10.3.4 decomposition**

间接项目由直接项目分解而来。

**B.4.10.3.5 derivation**

间接项目由直接项目派生而来。

注：“基于”常常被用作派生的同义词。

**B.4.10.3.6 peer**

间接项目提供关于直接项目所要求的信息。等同项目对于完整理解项目含义十分必要。

**B.4.10.3.7 referencing**

间接项目引用直接项目。

**B.4.10.3.8 sequence**

间接项目依次排在直接项目之后。

**B.4.10.3.9 substitution**

间接项目代替直接项目。

**B.4.10.3.10 superseding**

间接项目接替直接项目。

**B.4.10.3.11 translation**

间接项目是由直接项目转化而来。

**B.4.10.3.12 variant**

间接项目来自直接项目变体。

**B.4.10.3.13 withdrawal**

间接项目被取消,没有代替品。

**B.4.11 Party**

Party 是个人与组织的统称。

**B.4.12 Party\_role**

Party\_role 指定了 Party 中的角色。

在本模型中,Party\_role 是 Party 作为域管理者的角色。

属性有:

——is\_role\_of;



——custodian。

#### **B.4.12.1 is\_role\_of**

指定主管角色的 Party。

#### **B.4.12.2 custodian**

角色的指定类型,参见 Role\_type 清单。

#### **B.4.12.3 Role\_type**

指定个人或组织对其应用的项目的责任。其值可以由用户定义也可以是预设的。

预设的 Role\_type 值有:

——custodian;  
 ——customer;  
 ——license\_holder;  
 ——local\_representative;  
 ——manufacturer;  
 ——operator;  
 ——owner;  
 ——responsible\_party;  
 ——supplier;  
 ——vendor。

##### **B.4.12.3.1 custodian**

对项目的存在和完整性负责的个人或组织。

##### **B.4.12.3.2 customer**

作为项目购买者或消费者的个人或组织。

##### **B.4.12.3.3 license\_holder**

领有项目生产执照的个人或组织。

##### **B.4.12.3.4 local\_representative**

作为项目在当地的联系点的个人或组织。

示例:工地上的现场管理者员可以作为公司的当地代表。

##### **B.4.12.3.5 manufacturer**

生产项目的个人或组织。

##### **B.4.12.3.6 operator**

运行项目的个人或组织。

##### **B.4.12.3.7 owner**

拥有项目的个人或组织。

**B.4.12.3.8 responsible\_party**

负责管理项目的个人或组织。

**B.4.12.3.9 supplier**

提供项目的个人或组织。

**B.4.12.3.10 vendor**

销售项目的个人或组织。

**B.4.13 Party\_to\_address\_relationship**

此实体将团体与有效地址关联在一起。

属性有：

——related\_address；

——related\_party。

**B.4.13.1 related\_address**

此属性定义与相关地址的关系。

**B.4.13.2 related\_party**

此属性定义团体。

**B.4.14 Party\_relationship**

本实体包含描述各方之间关系。

相关的属性有：

——relating\_party；

——related\_party；

——party\_relation\_type。

**B.4.14.1 relating\_party**

由 relating\_party 联系起来的两个团体中的第一个。

**B.4.14.2 related\_party**

由 relating\_party 联系起来的两个团体中的第二个。

**B.4.14.3 party\_relationship\_type**

party relationship\_type 定义的是关系的含义。

常见的 party relationship\_type 有：

——employer；

——employee。

**B.4.14.3.1 employer**

被其他人雇用的团体。

**B.4.14.3.2 employee**

雇用其他人的团体。

**B.4.15 Person**

Person 是 Party 的子类。

属性有：

- id;
- last\_name;
- first\_name;
- middle\_name;
- prefix\_title;
- suffix\_title;
- digital\_signature。

**B.4.16 Organization**

Organization 是 Party 的子类。

属性有：

- identifier。

**B.4.16.1 identifier**

描述 Organization 和 (ABS) Organization\_identifier 之间的关系。一个组织至少有一个且可以有多个 Organization\_identifiers。

**B.4.17 Organization\_identifier**

Organization\_identifier 是名 Named\_organization\_identifier 和 Coded\_organization\_identifier 的总称。

**B.4.18 Coded\_organization\_identifier**

Coded\_organization\_identifier 是 Organization\_identifier 的子类,标识符可以是国际的也可以是国家的。

- coded\_organization\_id;
- valid\_domain; and optionally
- country\_code。

**B.4.18.1 coded\_organization\_id**

coded\_organization\_id 是组织的代码。

**B.4.18.2 valid\_domain**

有效域定义的是与组织是管理者的域之间的关系。

**B.4.18.3 country\_code**

ISO 3166-1 中定义的双字母国家编码。

#### **B.4.19 Named\_organization\_identifier**

Named\_organization\_identifier 是 Organization\_identifier 的子类。

属性如下：

- country\_name;
- organization\_name; and optionally
- country\_code。

##### **B.4.19.1 country\_name**

ISO 3166-1 中定义的国家名称。

##### **B.4.19.2 organization\_name**

组织的官方名称。

##### **B.4.19.3 country\_code**

ISO3166-1 中定义的双字母国家编码。

#### **B.4.20 Address**

Postal\_address, Electronic\_address 和 Physical address 的统称。

##### **B.4.21 Electronic\_address**

Electronic\_address 是地址的子类。它是用来定位团体电子地址的属性集合。

属性有：

- electronic\_mail\_address;
- telephone\_number;
- mobile\_number;
- facsimile\_numkber;
- telex\_number;
- url。

##### **B.4.22 Physical\_adress**

Physical\_address 是地址的子类。它是用来定位团体物理地址的属性集合。

属性有：

- postal\_code;
- country;
- region;
- town;
- street
- street\_number;
- internal location;
- property\_name。

注：除上述列出的属性之外如果还需要其他属性，则需要取得涉及到的团体的同意。

#### B.4.23 Postal\_address

Postal\_address 是地址的子类。它是专属于邮政用途的属性集合。注意：很多情况下物理地址也被用作邮政地址。

属性有：

- postal\_name;
- postal\_box;
- postal\_code;
- country。

#### B.5 EXPRESS-G 图形模型

EXPRESS-G 是 ISO 10303-11 中定义的图例模型。为了表示方便,完整的模型分别画在 two 页纸上,见图 B.1 和图 B.2。

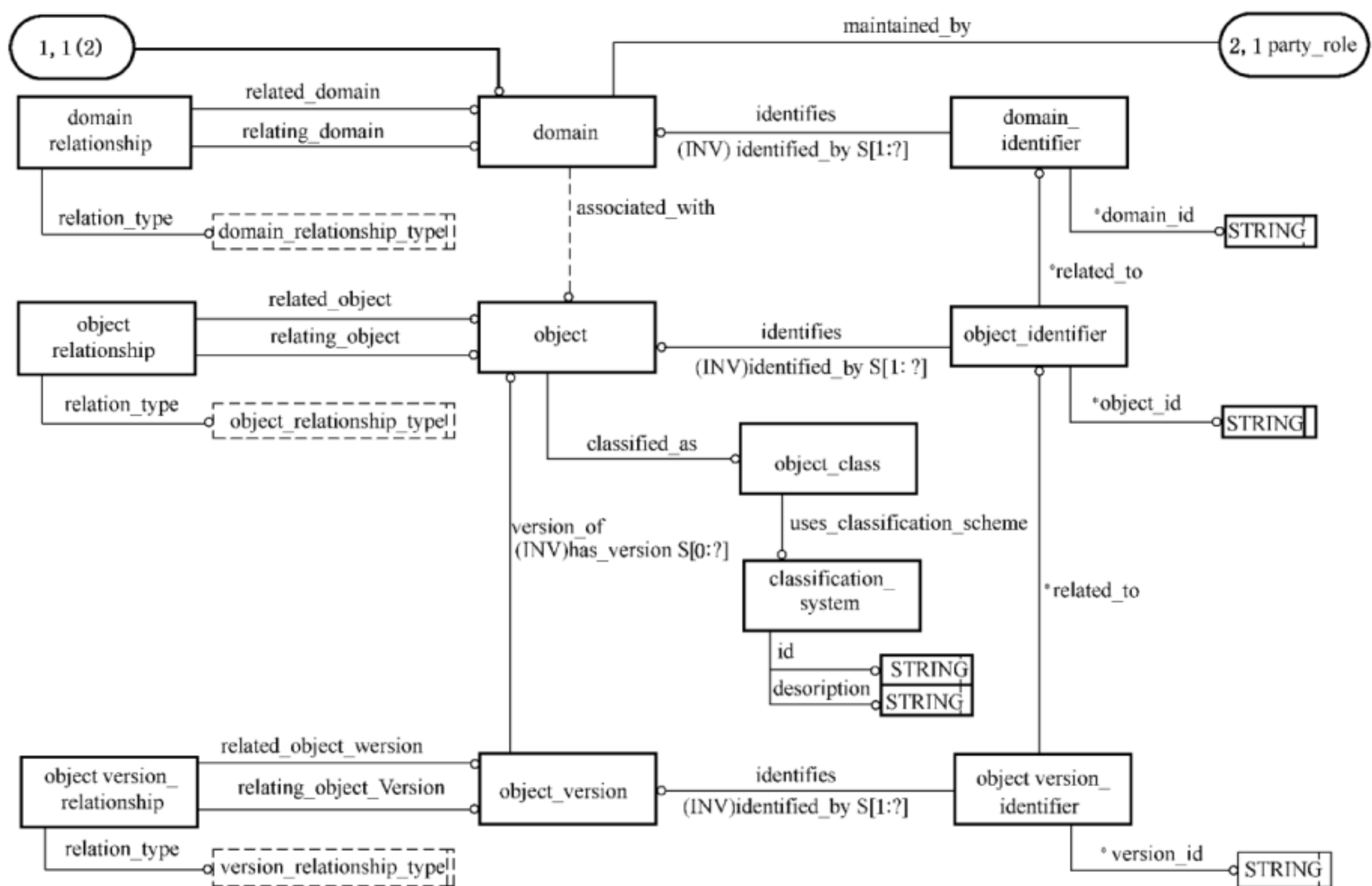


图 B.1 完整的模型(一)

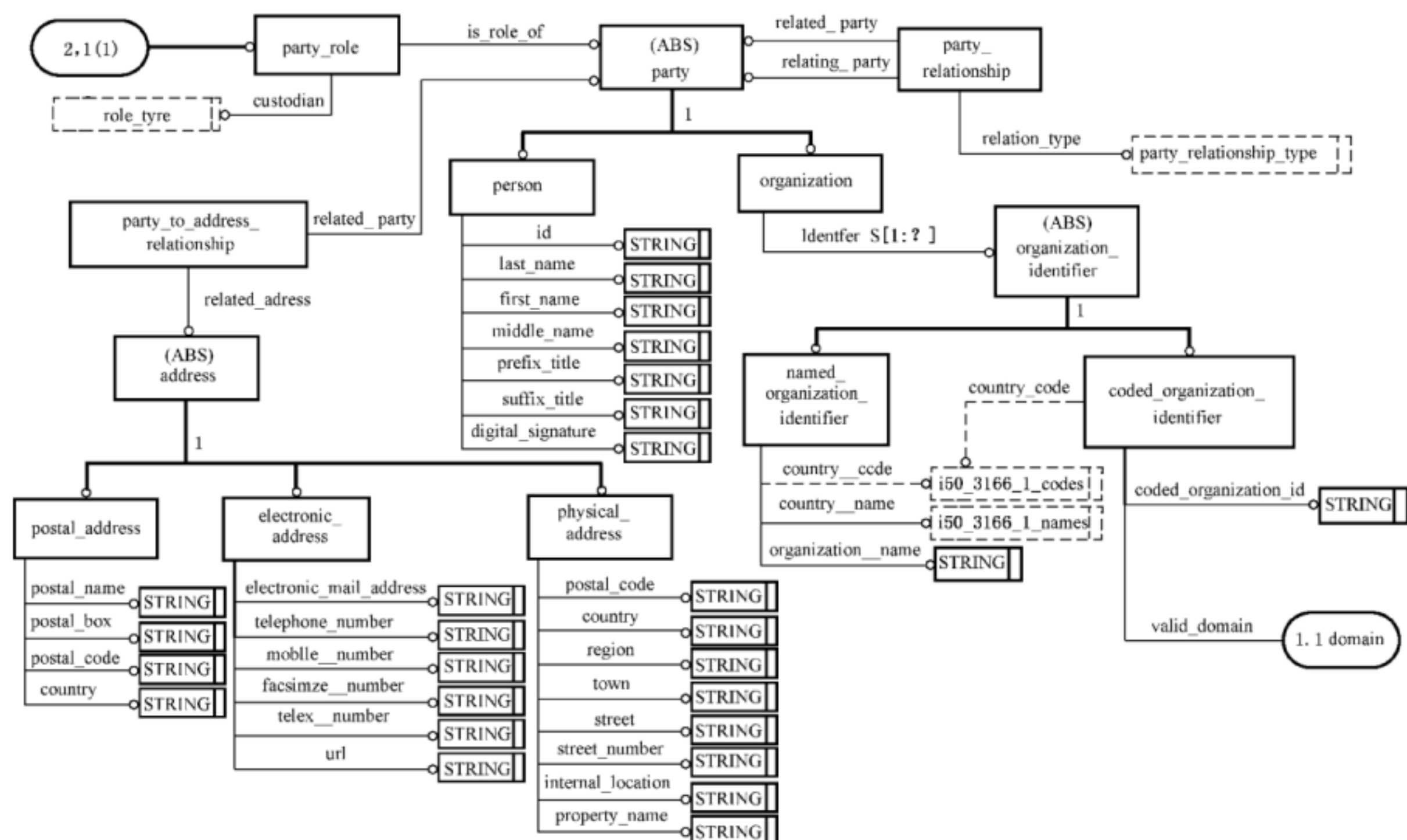


图 B.2 完整的模型(二)

## B.6 EXPRESS 源代码

本条提供了 EXPRESS 源代码。其电子文本见 [http://tc3.iec.ch/stp/IEC62507-1 EXPRESS V10 SCHEMA.txt](http://tc3.iec.ch/stp/IEC62507-1_EXPRESS_V10_SCHEMA.txt), 本文将随标准出版提供电子版本。

注: 列举出来的 ISO3166-1 代码只包含了一个子集, 完整集合参见 ISO 3166-1。

SCHEMA IEC\_62507-1\_V00;

```
TYPE iso_3166_1_codes = ENUMERATION OF
```

(BE,

DE,

DK,

ES,

FI,

FR,

IT,

NL,

NO,

PT,

SE,

UK,

US) ;



```
END_TYPE;  
TYPE iso_3166_1_names = ENUMERATION OF  
    (BELGIUM,  
    DENMARK,  
    FINLAND,  
    FRANCE,  
    GERMANY,  
    GREAT_BRITAIN,  
    ITALY,  
    NORWAY,  
    PORTUGAL,  
    SPAIN,  
    SWEDEN,  
    THE_NETHERLANDS,  
    USA);  
END_TYPE;
```

```
TYPE role_type = ENUMERATION OF  
    (CUSTODIAN,  
    CUSTOMER,  
    LICENSE_HOLDER,  
    LOCAL_REPRESENTATIVE,  
    MANUFACTURER,  
    OPERATOR,  
    OWNER,  
    RESPONSIBLE_PARTY,  
    SUPPLIER,  
    VENDOR);  
END_TYPE;
```

```
TYPE domain_relationship_type = ENUMERATION OF  
    (DECOMPOSITION,  
    SUBSTITUTION);  
END_TYPE;
```

```
TYPE object_relationship_type = ENUMERATION OF  
    (ADDITION,  
    AFFECTING,  
    COPY,  
    DECOMPOSITION,  
    DERIVATION,  
    PEER,  
    REFERENCING,
```

SEQUENCE,  
SUBSTITUTION,  
SUPERSEDING,  
SUPPLIED,  
TRANSLATION,  
VARIANT,  
WITHDRAWAL);  
END\_TYPE;

TYPE version\_relationship\_type = ENUMERATION OF  
(ADDITION,  
AFFECTING,  
COPY,  
DECOMPOSITION,  
DERIVATION,  
PEER,  
REFERENCING,  
SEQUENCE,  
SUBSTITUTION,  
SUPERSEDING,  
SUPPLIED,  
TRANSLATION,  
VARIANT,  
WITHDRAWAL);  
END\_TYPE;

TYPE party\_relationship\_type = ENUMERATION OF  
(EMPLOYER,  
EMPLOYEE);  
END\_TYPE;

ENTITY classification\_system;  
    id : STRING;  
    description: STRING;  
END\_ENTITY;

ENTITY object\_class;  
    uses\_classification\_scheme: classification\_system;  
END\_ENTITY;

ENTITY domain\_relationship;  
    relating\_domain: domain;  
    related\_domain : domain;

```

    relation_type : domain_relationship_type;
END_ENTITY;

```

```

ENTITY domain;
    associated_with: OPTIONAL object;
    maintained_by : party_role;
    INVERSE
        identified_by : SET [1:?] OF domain_identifier FOR identifies;
END_ENTITY;

```

```

ENTITY object;
    classified_as: object_class;
    INVERSE
        has_version : SET OF object_version FOR version_of;
        identified_by: SET [1:?] OF object_identifier FOR identifies;
END_ENTITY;

```

```

ENTITY object_version;
    version_of : object;
    INVERSE
        identified_by: SET [1:?] OF object_version_identifier FOR identifies;
END_ENTITY;

```

```

ENTITY object_version_identifier;
    identifies: object_version;
    related_to: object_identifier;
    version_id: STRING;
    UNIQUE
        identifier: version_id, related_to;
END_ENTITY;

```

```

ENTITY object_identifier;
    identifies: object;
    related_to: domain_identifier;
    object_id : STRING;
    UNIQUE
        identifier: object_id, related_to;
END_ENTITY;

```

```

ENTITY domain_identifier;
    identifies: domain;
    domain_id : STRING;
    UNIQUE

```

```
    identifier: domain_id;  
END_ENTITY;
```

```
ENTITY party_role;  
    is_role_of: party;  
    custodian : role_type;  
END_ENTITY;
```

```
ENTITY party  
    ABSTRACT SUPERTYPE OF (ONEOF(person, organization));  
END_ENTITY;
```

```
ENTITY person  
    SUBTYPE OF(party);  
    id                : STRING;  
    last_name         : STRING;  
    first_name        : STRING;  
    middle_name       : STRING;  
    prefix_title      : STRING;  
    suffix_title       : STRING;  
    digital_signature : STRING;  
END_ENTITY;
```

```
ENTITY organization  
    SUBTYPE OF(party);  
    identifier: SET [1:?] OF organization_identifier;  
END_ENTITY;
```

```
ENTITY organization_identifier  
    ABSTRACT SUPERTYPE OF (ONEOF(named_organization_identifier, coded_organization_  
identifier));  
END_ENTITY;
```

```
ENTITY named_organization_identifier  
    SUBTYPE OF(organization_identifier);  
    country_code      : OPTIONAL iso_3166_1_codes;  
    organization_name : STRING;  
    country_name      : iso_3166_1_names;  
END_ENTITY;
```

```
ENTITY coded_organization_identifier  
    SUBTYPE OF(organization_identifier);  
    coded_organization_id: STRING;
```

```

country_code      : OPTIONAL iso_3166_1_codes;
valid_domain      : domain;
END_ENTITY;

```

```

ENTITY object_relationship;
    relation_type   : object_relationship_type;
    relating_object : object;
    related_object  : object;
END_ENTITY;

```

```

ENTITY object_version_relationship;
    relation_type       : version_relationship_type;
    relating_object_Version : object_version;
    related_object_version : object_version;
END_ENTITY;

```

```

ENTITY party_to_address_relationship;
    related_address : address;
    related_party   : party;
END_ENTITY;

```

```

ENTITY address
    ABSTRACT SUPERTYPE OF (ONEOF(postal_address, electronic_address, physical_address));
END_ENTITY;

```

```

ENTITY postal_address
    SUBTYPE OF(address);
    postal_name      : STRING;
    postal_box       : STRING;
    postal_code      : STRING;
    country          : STRING;
END_ENTITY;

```

```

ENTITY electronic_address
    SUBTYPE OF(address);
    electronic_mail_address : STRING;
    telephone_number       : STRING;
    mobile_number          : STRING;
    facsimile_number       : STRING;
    telex_number           : STRING;
    url                    : STRING;
END_ENTITY;

```



```
ENTITY physical_address
  SUBTYPE OF(address);
    postal_code      : STRING;
    country          : STRING;
    region           : STRING;
    town             : STRING;
    street           : STRING;
    street_number    : STRING;
    internal_location : STRING;
    property_name    : STRING;
END_ENTITY;

ENTITY party_relationship;
  relation_type : party_relationship_type;
  relating_party : party;
  related_party : party;
END_ENTITY;

END_SCHEMA;
```

附 录 C  
(资料性附录)  
标识系统文档的示例

域 ID			
签发机构	组织	名称 组织 ID 邮政地址 电子地址 物理地址	
维护组织	组织	名称 组织 ID(如果有) 邮政地址 电子地址 物理地址	
	管理者	姓名 人员 ID 邮政地址 电子地址 物理地址	
签发子域(如果有)	域 ID	负责机构	引用文件
	域 ID	负责机构	引用文件
	域 ID	负责机构	引用文件
标识系统描述	对象种类		
	<input type="checkbox"/> 允许同一对象多重标识		
	<input type="checkbox"/> 禁止同一对象多重标识		
	标识代码创建规则		引用文件
	创建方法		<input type="checkbox"/> 方法 1
	注：参见 6.1		<input type="checkbox"/> 方法 2
	标识代码注册	<input type="checkbox"/> 手动注册 ID 和位置 <input type="checkbox"/> 在线注册网页地址：	
	注册的元数据	<input type="checkbox"/> 标识代码注册的日期	
		<input type="checkbox"/> 生成标识代码的人员姓名	
		号码类型：	
		<input type="checkbox"/> 标识代码	
		<input type="checkbox"/> 序列号	

	<input type="checkbox"/> 批次号/批号	
	<input type="checkbox"/> 包装日期	
	<input type="checkbox"/> 生产日期	
	<input type="checkbox"/> 截止日期	
<input type="checkbox"/> 标识代码重复使用	<input type="checkbox"/> 不允许	<input type="checkbox"/> ……年后允许
<input type="checkbox"/> 版本	<input type="checkbox"/> 未被使用	<input type="checkbox"/> 被使用
<input type="checkbox"/> 标识代码		标识：
<input type="checkbox"/> 使用的分隔符(如果有)		

附 录 D  
(资料性附录)  
ISO/IEC 646 描述的不变式

	0	1	2	3	4	5	6	7
0			SP	0		P		p
1			!	1	A	Q	a	q
2			“	2	B	R	b	r
3			#	3	C	S	c	s
4			\$	4	D	T	d	t
5			%	5	E	U	e	u
6			&	6	F	V	f	v
7			‘	7	G	W	g	w
8			(	8	H	X	h	x
9			)	9	I	Y	i	y
10			*	:	J	Z	j	z
11			+	;	K		k	
12			,	<	L		l	
13			—	=	M		m	
14			.	>	N		n	
15			/	?	O	—	o	

### 参 考 文 献

- [1] GB/T 5094.1—2018 工业系统、装置与设备以及工业产品 结构原则与参照代号 第1部分:基本规则(IEC 81346-1, IDT)
  - [2] GB/T 17969.8—2010 信息技术 开放系统互连 OSI 登记机构操作规程 第8部分:通用唯一标识符(UUID)的生成和登记及其用作 ASN.1 客体标识符部件(ISO/IEC 9834-8, IDT)
  - [3] GB/T 19000—2008 质量管理体系 基础和术语(ISO 9000:2005, IDT)
  - [4] ISO 7372 Trade data interchange—Trade data elements dictionary
  - [5] ISO 10303 (all parts) Industrial automation systems and integration—Product data representation and exchange
  - [6] ISO 13616 Banking and related financial services—International bank account number (IBAN)
- 

