

中华人民共和国国家标准

GB/T 32392.3—2015/ISO/IEC 19763-3:2007

信息技术 互操作性元模型框架(MFI) 第3部分:本体注册元模型

Information technology—Metamodel framework for interoperability(MFI)—
Part 3: Metamodel for ontology registration

(ISO/IEC 19763-3:2007, IDT)

2015-12-31 发布

2017-01-01 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	III
引言	IV
1 范围	1
2 符合性	1
2.1 概要	1
2.2 符合性的程度	1
2.3 实现符合性声明(ICS)	2
3 规范性引用文件	2
4 术语、定义和缩略语	2
4.1 术语和定义	2
4.2 缩略语	3
5 本体注册元模型的结构	3
5.1 本体注册元模型概述	3
5.2 核心模型和本体注册元模型的关系	4
5.3 本体注册元模型	5
附录 A (资料性附录) 基于本体注册元模型的本体注册示例	9
附录 B (资料性附录) 继承管理_项的所有元类	15
附录 C (资料性附录) 本体_语言列表	16
参考文献	17
图 1 本体注册元模型的范围	1
图 2 本体注册元模型	4
图 3 核心模型和本体注册元模型的关系	5
图 A.1 参考_本体 1 中的句子示例	9
图 A.2 参考_本体 1 的注册信息	10
图 A.3 参考_本体 2 中的句子示例	11
图 A.4 参考_本体 2 的注册信息	12
图 A.5 本地_本体 1 中的句子示例	13
图 A.6 本地_本体 1 的注册信息	14
图 B.1 继承管理_项的所有元类	15
表 C.1 本体_语言列表	16

前 言

GB/T 32392《信息技术 互操作性元模型框架(MFI)》包含以下几个部分:

- 第 1 部分:参考模型;
- 第 2 部分:核心模型;
- 第 3 部分:本体注册元模型;
- 第 4 部分:模型映射元模型;
- 第 5 部分:模型构件元模型框架;
- 第 6 部分:注册规程。

本部分为 GB/T 32392 的第 3 部分。

本部分按照 GB/T 1.1—2009 给出的规则起草。

本部分采用翻译法等同采用 ISO/IEC 19763-3:2007《信息技术 互操作性元模型框架(MFI) 第 3 部分:本体注册元模型》(英文版)。

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 32392.1—2015 信息技术 互操作性元模型框架(MFI) 第 1 部分:参考模型 (ISO/IEC 19763-1:2007, IDT)

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别这些专利的责任。

本部分由全国信息技术标准化技术委员会(SAC/TC 28)提出并归口。

本部分起草单位:武汉大学软件工程国家重点实验室、中国电子技术标准化研究院。

本部分主要起草人:何克清、何扬帆、王翀、王健、王静。

引 言

随着电子商务在因特网的广泛传播,跨国家和跨文化的业务贸易和其他相关信息交换已成为 IT 业内外人士主要关注的问题。

为了紧跟电子商务的发展趋势,许多行业组织已经展开领域相关业务对象的标准化工作,包括使用 UML 和 XML 等公共建模设施和交换设施的业务过程模型和软件部件。他们主要致力于规范领域相关的过程模型和标准建模构件,如数据元素、实体轮廓和值域等。

此外,自治的、基于 Web 的应用(如 Web 服务)之间的互操作问题也变得越来越重要。本体是实现此目的的关键。本体是对论域的描述。词典、分类系统、辞典和概念模型(如基于 UML 的业务过程模型)等都是本体的示例。不仅如此,本体还包括更为复杂的公理理论。

目前,业界正在制定一些与本体相关的标准。比如,OMG 即将发布的 ODM 规范定义了使用多种语言描述的本体的元模型及其之间的映射。对于 GB/T 32392 的本部分所规定和注册的本体的相关元数据,可以扩展本部分对上述本体相关标准提供支持。此外,为了促进基于本体的互操作,需要提供一个通用的框架以注册本体相关的管理信息。

本部分将提供一个基于 GB/T 32392.2—2015 的通用框架,用于注册与本体相关的管理信息。

注: UML 和 OMG 是对象管理组的商标。

信息技术 互操作性元模型框架(MFI)

第3部分:本体注册元模型

1 范围

GB/T 32392 的本部分规定的框架由多个部分组成,用以指导信息技术领域内元模型之间互操作的实现。

本部分规定的元模型为注册与本体相关的管理信息提供了一个基础设施,能够促进应用系统之间的互操作。

本部分不适用于规定基于特定本体描述语言的元模型及这些元模型之间的映射。这些内容在其他规范(如本体定义元模型 ODM)中已经规定。

图 1 描述了本部分的范围。

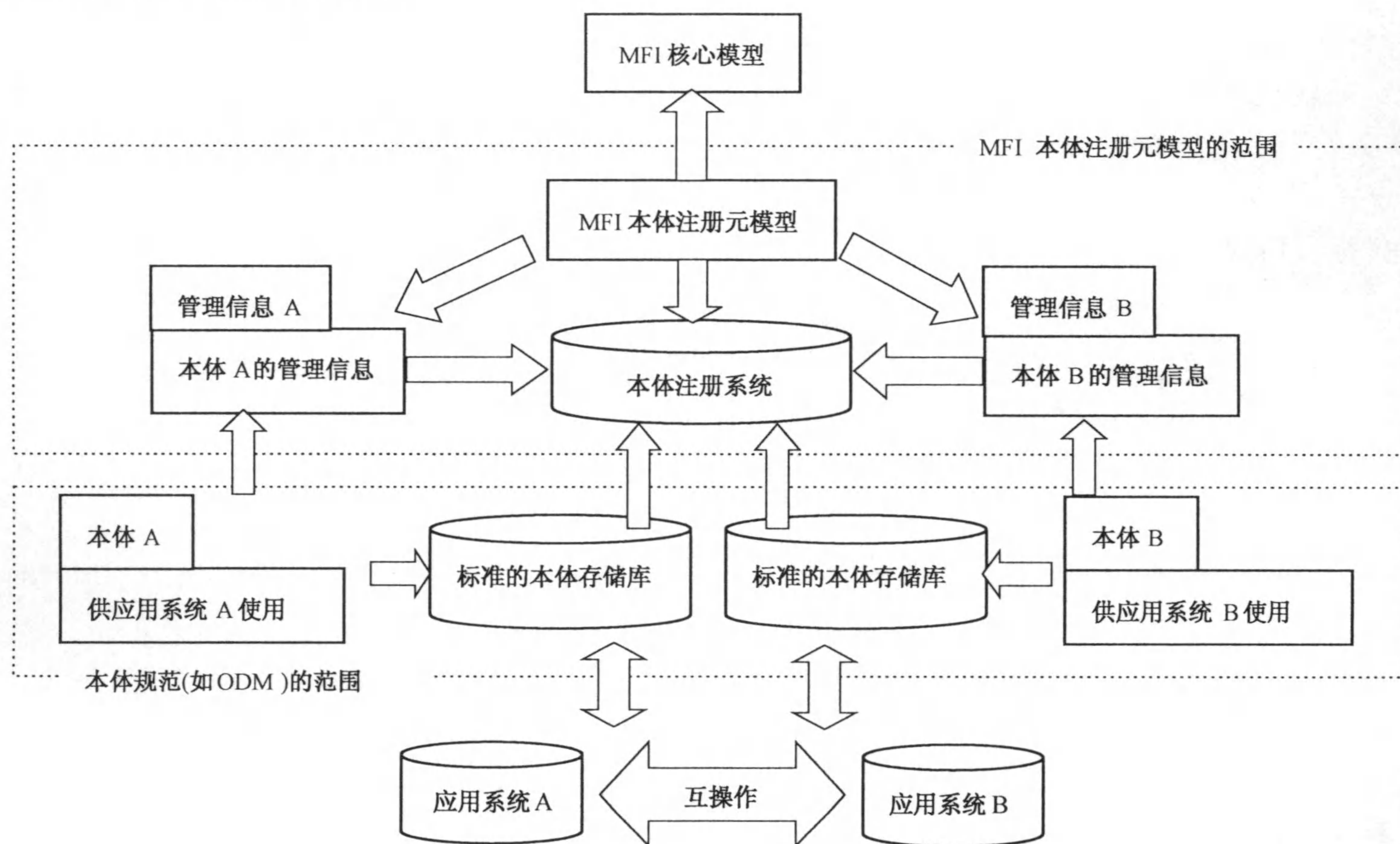


图 1 本体注册元模型的范围

2 符合性

2.1 概要

声称与本部分相符合的实现应该支持 5.3 中规定的元模型,该元模型依赖于下面描述的符合程度。

2.2 符合性的程度

2.2.1 概述

为了同时满足互操作性和可扩展性的需要,必须区别“严格符合”和“符合”实现。本部分描述了促

进互操作性的规范。扩展是根据用户、开发商、组织和行业的需要而展开的,本部分没有对其进行规定。

严格符合实现的实用性有限,但是能够最大程度地实现与本部分相关的互操作。符合实现的实用性更广,但实现与本部分相关的互操作的能力较弱。

2.2.2 严格的符合性实现

严格的符合性实现:

- a) 应该支持 5.3 中规定的元模型;
- b) 不应该支持对 5.3 中规定的元模型实施的任何扩展。

2.2.3 符合性实现

符合性的实现:

- a) 应该支持 5.3 中规定的元模型;
- b) 可以支持与 5.3 中规定的元模型相一致的扩展。

2.3 实现符合性声明(ICS)

声称符合本部分的实现应包括陈述如下的实现符合性声明:

- a) 是严格符合实现还是符合实现(2.2);
- b) 如果是符合实现,则支持哪些扩展。

3 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 18391.3:2009 信息技术 元数据注册系统(MDR) 第 3 部分:注册系统元模型与基本属性(ISO/IEC 11179-3:2003, IDT)

ISO/IEC 19763-1 信息技术 互操作性元模型框架(MFI) 第 1 部分:参考模型[Information technology—Metamodel framework for interoperability (MFI)—Part 1:Reference model]

ISO/IEC 19763-2 信息技术 互操作性元模型框架(MFI) 第 2 部分:核心模型[Information technology—Metamodel framework for interoperability (MFI)—Part 2:Core model]

4 术语、定义和缩略语

4.1 术语和定义

GB/T 18391.3—2009、ISO/IEC 19763-1 和 ISO/IEC 19763-2 中界定的以及下列术语和定义适用于本文件。

4.1.1

论域 universe of discourse

在特定语境中关注的实体,可以是具体的或者是抽象的,也可以是已经存在的或者将要出现的。

4.1.2

本体 ontology

使用计算机能够处理的语言对论域的描述。

4.1.3

参考本体 reference ontology

能够被特定组织重用和共享的本体。

4.1.4

本地本体 local ontology

针对某个已定义的应用而具体化的本体,基于至少一个参考本体。

4.1.5

句子 sentence

一段逻辑值为真或假的陈述。

4.1.6

符号 symbol

组成句子某个部分的最基本的词汇构件。

4.1.7

逻辑符号 logical symbol

其含义由指定语言明确定义的符号。

例:在 KIF 中,“非”和“或”都是逻辑符号。

4.1.8

非逻辑符号 non-logical symbol

除逻辑符号之外的符号。

4.2 缩略语

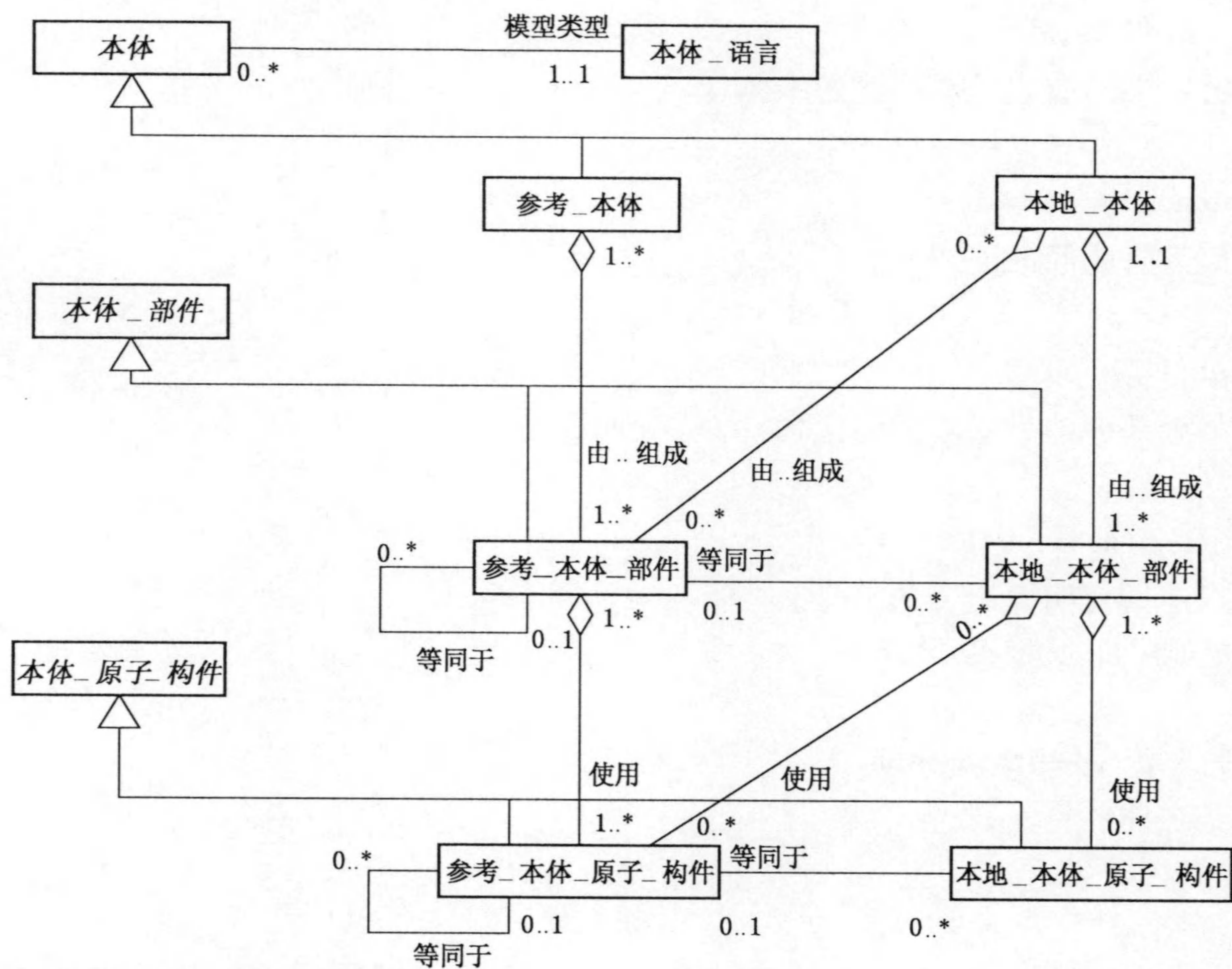
下列缩略语适用于本文件。

KIF	知识交换格式 (Knowledge Interchange Format)
MFI	互操作性元模型框架 (Metamodel framework for interoperability)
MDR	元数据注册系统 (Metadata Registry)
ODM	本体定义元模型 (Ontology Definition Metamodel)
URI	统一资源标识符 (Uniform Resource Identifiers)
OWL	本体网络语言 (Ontology Web Language)

5 本体注册元模型的结构

5.1 本体注册元模型概述

本体由句子组成,每一个句子都使用了非逻辑符号。本体注册元模型为本体、句子和所用符号的管理信息提供了一个基本的注册机制。图 2 描述了本体注册元模型的结构。



注：名称用斜体表示的元类为抽象元类。

图2 本体注册元模型

本体是一个抽象元类，指定了本体的管理信息。本体_语言表示描述本体的语言。本体_部件是一个抽象元类，指定了某个本体中包含的句子的管理信息，句子粒度的大小由用户决定。本体_原子_构件是一个抽象元类，指定了在句子中使用的某个非逻辑符号的管理信息。

参考_本体和本地_本体是本体的子类。参考_本体指定了一个标准的本体；本地_本体指定了具体应用系统使用的本体，且本地_本体应基于至少一个参考_本体。

参考_本体_部件和本地_本体_部件是本体_部件的子类。参考_本体_部件指定了组成参考_本体的句子，注册为参考_本体_部件的句子也可以包含在本地_本体中。本地_本体_部件指定了包含在本地_本体中的句子。

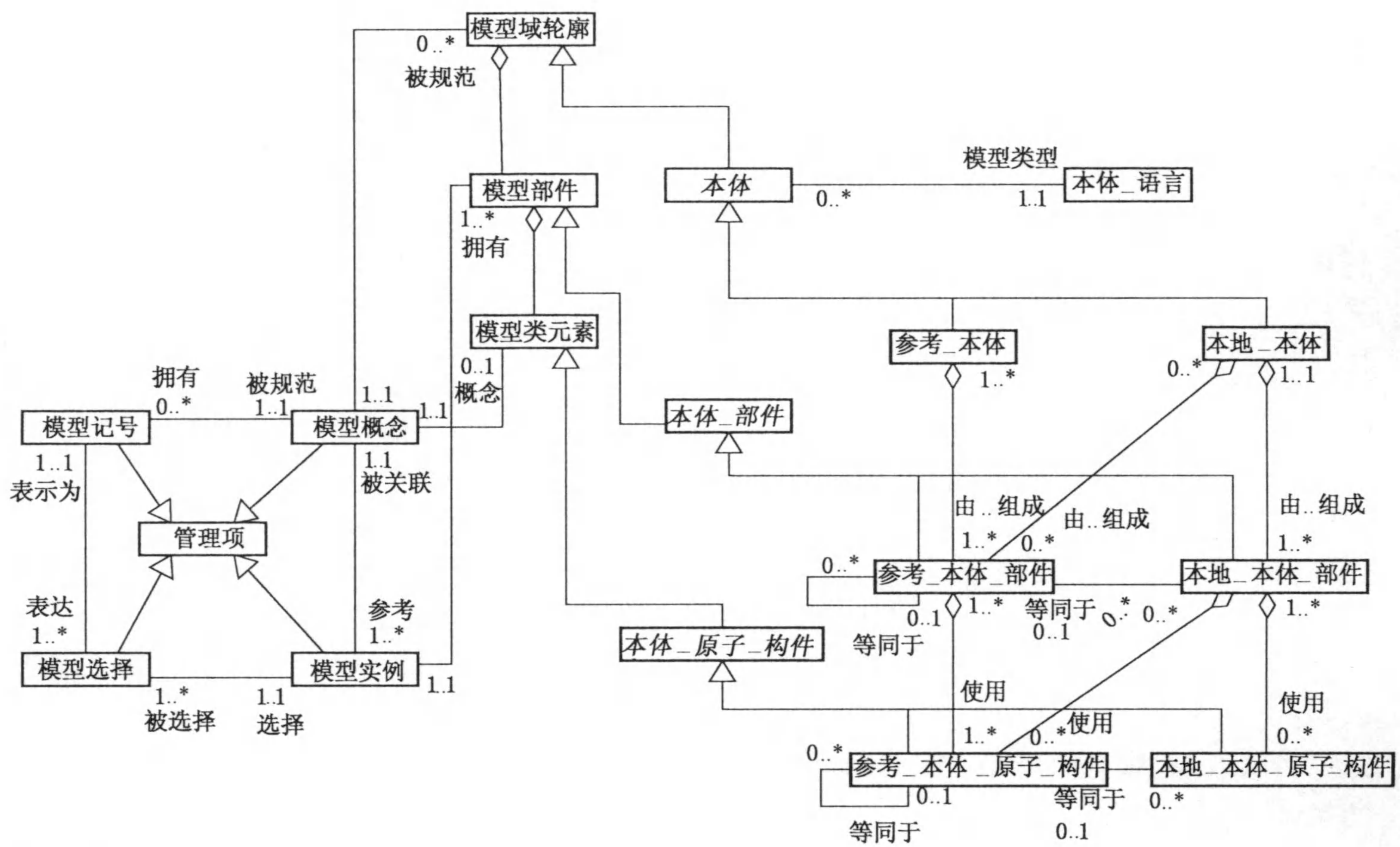
参考_本体_原子_构件和本地_本体_原子_构件是本体_原子_构件的子类。参考_本体_原子_构件指定了注册为参考_本体_部件的句子中使用的非逻辑符号，注册为参考_本体_原子_构件的非逻辑符号也可以在注册为本地_本体_部件的句子中使用。本地_本体_原子_构件指定了注册为本地_本体_部件的句子中使用的非逻辑符号；本地_本体_原子_构件指定的非逻辑符号只能够由注册为本地_本体_部件的句子使用。

5.3 给出了每个元类的详细说明。基于本体注册元模型的本体注册示例参见附录 A。

本体注册元模型中所有元类与“管理_项”的关系参见附录 B。

5.2 核心模型和本体注册元模型的关系

本体注册元模型继承了核心模型的基本结构。图 3 说明了核心模型和本体注册元模型的关系。



注：名称用斜体表示的元类为抽象元类。

图3 核心模型和本体注册元模型的关系

5.3 本体注册元模型

5.3.1 本体

本体是一个抽象元类，它是参考_本体和本地_本体的超类。

超类

管理_项

属性

URI

本体名称

参考

模型类型

数据类型

串型

串型

数据类型

本体_语言

多重度

1..1

1..1

多重度

1..1

描述

待注册本体的 URI

待注册本体的名称

描述

指定了本体的描述语言

约束

属性“URI”的值在元类中应是唯一的。

5.3.2 参考_本体

参考_本体是一个元类，指定了一个特定组织使用的标准本体。

超类

本体

参考

类

多重度

描述

由...组成	参考_本体_部件	1..*	指定了注册为参考_本体的本体所包含的句子
--------	----------	------	----------------------

5.3.3 本地_本体

本地_本体是一个元类,指定了具体应用系统使用的某个本体。本地_本体应该基于至少一个注册为参考_本体的本体。

超类
本体

参考	类	多重度	描述
由...组成	本体_部件	1..*	指定了注册为本地_本体的本体所包含的句子

约束

属性“由...组成”应至少有一个值是参考_本体_部件或本地_本体_部件,且本地_本体_部件的“等同于”属性的取值是某个参考_本体_部件。

5.3.4 本体_语言

本体_语言是一个元类,它指定了本体所采用的描述语言。

属性	数据类型	多重度	描述
名称	串型	1..1	本体描述语言的名称。建议该属性的取值采用附录 C 中表 C.1“名称”项中所列举的值

约束

在这个元类中,属性“名称”的取值必须是唯一的。

5.3.5 本体_部件

本体_部件是一个抽象元类,它是参考_本体_部件和本地_本体_部件的超类。

超类
模型部件

属性	数据类型	多重度	描述
名称空间	串型	1..1	可以唯一确定句子标识符的 URI
句子标识符	串型	1..1	名称空间中指定句子的标识符

约束

在这个元类中,属性“名称空间”的取值将作为属性“句子标识符”的值的前缀;属性“句子标识符”的取值应是唯一的,并且能够标识相应的句子。

5.3.6 参考_本体_部件

参考_本体_部件是一个元类,指定了注册为参考_本体的本体所包含的句子。本部分没有规定句子的粒度大小。

超类

本体_部件

参考使用	类	多重度	描述
使用	参考_本体_原子_构件	1..*	指定了注册为参考_本体_部件的句子所使用的非逻辑符号
等同于	参考_本体_部件	0..1	指定了与注册为参考_本体_部件的句子解释完全相同的句子

约束

至少存在一个参考_本体,其属性“由...组成”的值是这个已注册的参考_本体_部件。

5.3.7 本地_本体_部件

本地_本体_部件是一个元类,指定了注册为本地_本体的本体所包含的句子。本部分没有规定句子的粒度大小。

超类

本体_部件

参考使用	类	多重度	描述
使用	本地_本体_原子_构件	1..*	指定了注册为本地_本体_部件的句子所使用的非逻辑符号
等同于	本地_本体_部件	0..1	指定了与注册为本地_本体_部件的句子解释完全相同的句子

约束

一定存在一个本地_本体,其属性“由...组成”的取值是这个已注册的本地_本体_部件。

5.3.8 本体_原子_构件

本体_原子_构件是一个抽象元类,它是参考_本体_原子_构件和本地_本体_原子_构件的超类。

超类

模型类元,管理_项

属性	数据类型	多重度	描述
名称空间	串型	1..1	可以唯一确定句子标识符的 URI
非逻辑符号	串型	1..1	相应的非逻辑符号

约束

在这个元类中,属性“名称空间”的取值将作为属性“非逻辑符号”取值的前缀,且“非逻辑符号”的取值应是唯一的。

5.3.9 参考_本体_原子_构件

参考_本体_原子_构件是一个元类,指定了注册为参考_本体_部件的句子使用的非逻辑符号。

超类

本体_原子_构件

参考	类	多重度	描述
使用	参考_本体_原子_构件	0..1	指定了与注册为参考_本体_原子_构件的非逻辑符号解释完全相同的非逻辑符号

约束

存在至少一个参考_本体_部件,其属性“使用”的取值是这个参考_本体_原子_构件。

5.3.10 本地_本体_原子_构件

本地_本体_原子_构件是一个元类,指定了注册为本地_本体_部件的句子使用的非逻辑符号。

超类

本体_原子_构件

参考	类	多重度	描述
使用	参考_本体_原子_构件	0..1	指定了与注册为本地_本体_原子_构件的非逻辑符号解释完全相同的非逻辑符号

约束

存在至少一个本地_本体_部件,其属性“使用”的取值是这个本地_本体_原子_构件。

存在一个本地_本体,其属性“由...组成”的取值是这个本地_本体_部件,且这个本地_本体_部件的属性“由...组成”的取值是这个本地_本体_原子_构件。

附录 A
(资料性附录)

基于本体注册元模型的本体注册示例

A.1 参考_本体示例

假设某个组织使用 OWL 语言建立了一个关于核心单位(KernelUnits)的标准本体“参考_本体 1”。核心单位是一个不含前缀的名称,如米(metre)。“参考_本体 1”包含多个描述核心单位的句子。图 A.1 中给出了其中 3 个句子的示例,假定它们分别被命名为“参考_本体_部件 1”“参考_本体_部件 2”“参考_本体_部件 3”。

参考_本体_部件 1

```

<owl:ObjectProperty rdf:ID="dimensionality">
  <rdfs:domain rdf:resource="#Unit" />
  <rdfs:range rdf:resource="#Dimensionality" />
</owl:ObjectProperty>
```

参考_本体_部件 2

```

<owl:Class rdf:ID="KernelUnit">
  <rdfs:subClassOf rdf:resource="#Unit"/>
</owl:Class>
```

参考_本体_部件 3

```

<KernelUnit rdf:ID="metre">
  <dimensionality>
    <Dimensionality rdf:ID="length"/>
  </dimensionality>
</KernelUnit>
```

图 A.1 参考_本体 1 中的句子示例

“参考_本体_部件 1”表明度量(Dimensionality)是单位(Unit)的一个属性;“参考_本体_部件 2”表明核心单位(KernelUnit)是单位(Unit)的子类;“参考_本体_部件 3”表明米(metre)是一种核心单位(KernelUnit),它可以用于度量(dimensionality)长度(length)。

图 A.2 说明了如何基于本体注册元模型将“参考_本体 1”注册为参考本体。

<参考_本体>

对象 01

属性/参考	值/实例
管理_项_管理_记录	管理_记录 01
URI	http://refl.org/kernel-unit
本体名称	“参考_本体 1”
模型类型	“OWL”
由...组成	对象 02
	对象 03
	对象 04
	...

<参考_本体_部件>

对象 02

属性/参考	值/实例
管理_项_管理_记录	管理_记录 02
名称空间	http://refl.org/kernel-unit
句子标识符	“参考_本体_部件 1”
使用	对象 05
	对象 06
	对象 07

对象 04

属性/参考	值/实例
管理_项_管理_记录	管理_记录 04
名称空间	http://refl.org/kernel-unit
句子标识符	“参考_本体_部件 3”
使用	对象 05
	对象 07
	对象 08
	对象 09
	对象 10

<参考_本体_原子_构件>

对象 05

属性/参考	值/实例
管理_项_管理_记录	管理_记录 05
名称空间	http://refl.org/kernel-unit
非逻辑符号	“度量”

对象 07

属性/参考	值/实例
管理_项_管理_记录	管理_记录 07
名称空间	http://refl.org/kernel-unit
非逻辑符号	“度量”

对象 09

属性/参考	值/实例
管理_项_管理_记录	管理_记录 09
名称空间	http://refl.org/kernel-unit
非逻辑符号	“米”

对象 03

属性/参考	值/实例
管理_项_管理_记录	管理_记录 03
名称空间	http://refl.org/kernel-unit
句子标识符	“参考_本体_部件 2”
使用	对象 06
	对象 08

对象 06

属性/参考	值/实例
管理_项_管理_记录	管理_记录 06
名称空间	http://refl.org/kernel-unit
非逻辑符号	“单位”

对象 08

属性/参考	值/实例
管理_项_管理_记录	管理_记录 08
名称空间	http://refl.org/kernel-unit
非逻辑符号	“核心单位”

对象 10

属性/参考	值/实例
管理_项_管理_记录	管理_记录 10
名称空间	http://refl.org/kernel-unit
非逻辑符号	“长度”

图 A.2 参考_本体 1 的注册信息

注 1:在参考_本体 1 中,除“参考_本体_部件 1”“参考_本体_部件 2”“参考_本体_部件 3”之外的句子都被忽略了。

注 2:除“管理_项_管理_记录”外,从其他标准继承得到的属性和参考均没有列出。

注 3:对象 n (n 从 01~10) 是对象标识符,使用它们只是为了便于描述。对这些对象实施的具体规范工作超出了本部分的范围。

A.2 另一个参考_本体示例

假设另一个组织使用 OWL 语言建立了一个关于带前缀单位(PrefixedUnit)的标准本体“参考_本体 2”。带前缀单位(PrefixedUnit)是一个带有前缀的单位,如千米(kilometre)。“参考_本体 2”包含多个描述带前缀单位的句子。图 A.3 中给出了其中两个句子的示例,假定它们分别被命名为“参考_本体_部件 4”“参考_本体_部件 5”。

参考_本体_部件 4

```
<owl:Class rdf:ID="PrefixedUnit">
  <rdfs:subClassOf rdf:resource="&ref1;Unit"/>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:about="#prefix"/>
      <owl:cardinality rdf:datatype="&xsd:int">1</owl:cardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
  <rdfs:subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:about="#kernel"/>
      <owl:cardinality rdf:datatype="&xsd:int">1</owl:cardinality>
    </owl:Restriction>
  </rdfs:subClassOf>
</owl:Class>
```

参考_本体_部件 5

```
<PrefixedUnit rdf:ID="micrometre">
  <prefix>
    <MetricPrefix rdf:ID="micro"/>
  </prefix>
  <kernel>
    <ref1:KernelUnit rdf:resource="&ref1;metre"/>
  </kernel>
</prefixedUnit >
```

图 A.3 参考_本体 2 中的句子示例

“参考_本体_部件 4”表明带前缀单位(PrefixedUnit)是单位(Unit)的子类,由一个前缀(prefix)和一个核心(kernel)组成;“参考_本体_部件 5”表明微米(micrometre)是一种带前缀单位(PrefixedUnit),它的前缀(prefix)是“微(micro)”(作为一个度量的前缀),核心(kernel)是“米(metre)”(作为一个核心单位)。

图 A.4 说明了如何基于本体注册元模型将“参考_本体 2”注册为参考本体。

<参考_本体>

对象 11

属性/参考	值/实例
管理_项_管理_记录	管理_记录 11
URI	http://ref2.org/prefixed-unit
本体名称	“参考_本体 2”
模型类型	“OWL”
由...组成	对象 12
	对象 13
	...

对象 12

属性/参考	值/实例
管理_项_管理_记录	管理_记录 12
名称空间	http://ref2.org/prefixed-unit
句子标识符	“参考_本体_部件 4”
使用	对象 06
	对象 14
	对象 15
	对象 16

<参考_本体_部件>

对象 13

属性/参考	值/实例
管理_项_管理_记录	管理_记录 13
名称空间	http://ref2.org/prefixed-unit
句子标识符	“参考_本体_部件 5”
使用	对象 08
	对象 09
	对象 14
	对象 15
	对象 16
	对象 17
	对象 18
	对象 19

<参考_本体_原子_构件>

对象 14

属性/参考	值/实例
管理_项_管理_记录	管理_记录 14
名称空间	http://ref2.org/prefixed-unit
非逻辑符号	“前缀单位”

对象 15

属性/参考	值/实例
管理_项_管理_记录	管理_记录 15
名称空间	http://ref2.org/prefixed-unit
非逻辑符号	“前缀”

对象 16

属性/参考	值/实例
管理_项_管理_记录	管理_记录 16
名称空间	http://ref2.org/prefixed-unit
非逻辑符号	“核心”

对象 17

属性/参考	值/实例
管理_项_管理_记录	管理_记录 17
名称空间	http://ref2.org/prefixed-unit
非逻辑符号	“微米”

对象 18

属性/参考	值/实例
管理_项_管理_记录	管理_记录 18
名称空间	http://ref2.org/prefixed-unit
非逻辑符号	“米制前缀”

对象 19

属性/参考	值/实例
管理_项_管理_记录	管理_记录 19
名称空间	http://ref2.org/prefixed-unit
非逻辑符号	“微”

图 A.4 参考_本体 2 的注册信息

注 1: 在参考_本体 2 中,除“参考_本体_部件 4”“参考_本体_部件 5”之外的句子都被忽略了。

注 2: 除“管理_项_管理_记录”外,从其他标准继承得到的属性和参考均没有列出。

注 3: 对象 n (n 从 06~19) 是对象标识符,使用它们只是为了便于描述。对这些对象实施的具体规范工作超出了本

部分的范围。

A.3 本地_本体示例

假设某个应用系统基于“参考_本体 1”“参考_本体 2”建立了满足其需要的、关于单位(Unit)的本体“本地_本体 1”。“本地_本体 1”是用 KIF 而不是 OWL 描述的,包含多个描述该应用系统中“单位”的句子。图 A.5 中给出了其中 3 个句子的示例,假定它们分别被命名为“本地_本体_部件 1”“本地_本体_部件 2”“本地_本体_部件 3”。

本地_本体_部件 1

```
(and (KernelUnit metre) (Dimentionality length) (dimentionality metre length))
```

本地_本体_部件 2

```
(and (PrefixedUnit micron) (MetricPrefix micro) (KernelUnit metre)
(prefix micron micro) (kernel micron metre))
```

本地_本体_部件 3

```
((forall ?Unit1 ?Unit2 ? Dimentionality 1 ? Dimentionality 2) (implies (and (PrefixedUnit ?Unit1)
(kernel ?Unit1 ?Unit2) (dimentionality ?Unit1 ?Ddimensionality1)
(dimentionality ?Unit2 ? Dimentionality 2)) (equal ? Dimentionality 1 ? Dimentionality 2))))
```

图 A.5 本地_本体 1 中的句子示例

“本地_本体_部件 1”与“参考_本体 1”中的“参考_本体_部件 3”具有相同的含义,并且使用了相同的非逻辑符号。

“本地_本体_部件 2”与“参考_本体 2”中的“参考_本体_部件 5”具有相同的含义;除了用“微(micro)”代替“微米(micrometre)”外,这个应用系统所使用的其他非逻辑符号均与“参考_本体 2”中“参考_本体_部件 5”所使用的非逻辑符号相同。

“本地_本体_部件 3”是一个新增加的句子,既不在“参考_本体 1”中,也不在“参考_本体 2”中。“本地_本体_部件 3”表示某个核心单位(KernelUnit)的度量标准与带前缀单位(PrefixedUnit)的度量标准是相同的,而这个带前缀单位的核心(kernel)是该核心单位。

图 A.6 说明了如何基于本体注册元模型将“本地_本体 1”注册为本地本体。

<本地_本体>

对象 20

属性/参考	值/实例
管理_项_管理_记录	管理_记录 20
URI	http://local1.org/unit
本体名称	“本地_本体 1”
模型类型	“KIF”
由...组成	对象 21
	对象 22
	对象 23
	...

<本地_本体_部件>

对象 21

属性/参考	值/实例
管理_项_管理_记录	管理_记录 21
名称空间	http://local1.org/unit
句子标识符	“本地_本体_部件 1”
使用	对象 05
	对象 07
	对象 08
	对象 09
	对象 10
等同于	对象 t04

对象 23

属性/参考	值/实例
管理_项_管理_记录	管理_记录 23
名称空间	http://local1.org/unit
句子标识符	“本地_本体_部件 3”
使用	对象 05
	对象 14
	对象 16

<本地_本体_原子_构件>

对象 24

属性/参考	值/实例
管理_项_管理_记录	管理_记录 24
名称空间	http://local1.org/unit
非逻辑符号	“微米”
等同于	对象 04

对象 22

属性/参考	值/实例
管理_项_管理_记录	管理_记录 22
名称空间	http://local1.org/unit
句子标识符	“本地_本体_部件 2”
使用	对象 08
	对象 09
	对象 14
	对象 15
	对象 18
	对象 19
	对象 24
等同于	对象 13

图 A.6 本地_本体 1 的注册信息

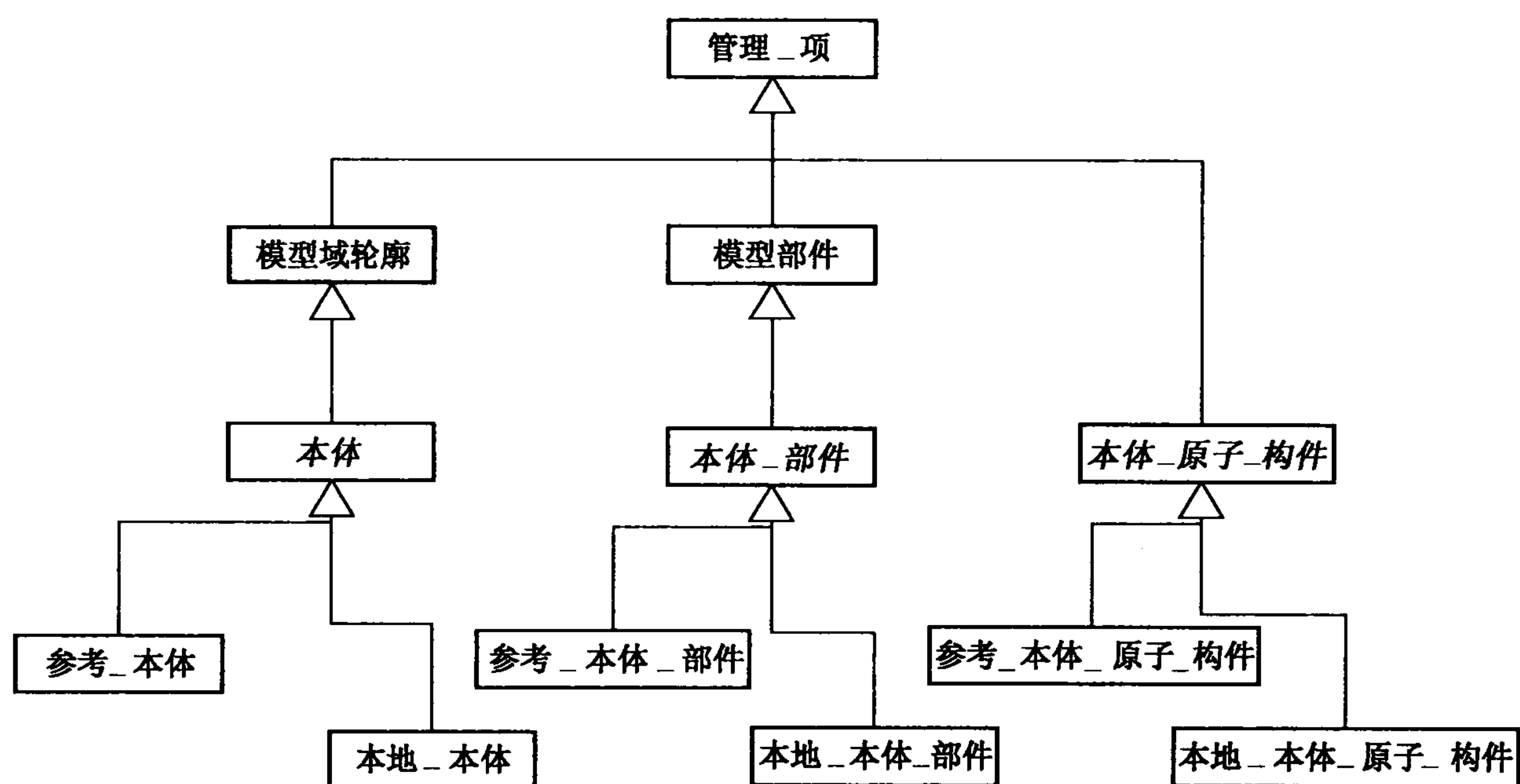
注 1: 在本地_本体 2 中,除“本地_本体_部件 1”“本地_本体_部件 2”“本地_本体_部件 3”之外的句子都被忽略了。

注 2: 除“管理_项_管理_记录”外,从其他标准继承得到的属性和参考均没有列出。

注 3: 对象 n (n 从 04~24) 是对象标识符,使用它们只是为了便于描述。对这些对象实施的具体规范工作超出了本部分的范围。

附录 B
 (资料性附录)
 继承管理_项的所有元类

图 B.1 描述了继承“管理_项”的所有元类。



注：名称用斜体表示的元类为抽象元类。

图 B.1 继承管理_项的所有元类

附 录 C
(资料性附录)
本体_语言列表

建议“本体_语言”中属性“名称”的取值应该是表 C.1 中“名称”列中的取值。

表 C.1 本体_语言列表

名称	描 述
概念图交换格式(CGIF)	公共逻辑的一个分支,在 ISO/IEC 24707《信息技术 公共逻辑(CL):基于逻辑语言族的框架》的附录 B 中规定
公共逻辑(CL)	一种逻辑语言,符合 ISO/IEC 24707《信息技术 公共逻辑(CL):基于逻辑语言族的框架》
公共逻辑交换格式(CLIF)	公共逻辑的一个分支,在 ISO/IEC 24707《信息技术 公共逻辑(CL):一个基于逻辑语言族的框架》的附录 A 中规定
描述逻辑(DL)	一种逻辑语言,是一阶逻辑的子集
实体-关系模型(E/R)	一种面向对象描述语言
知识交换格式(KIF)	一种逻辑语言
网络本体语言(OWL)	符合《OWL 网络本体语言的语义和抽象句法》的所有子语言
资源描述框架模式(RDFS)	符合《资源描述框架:概念和抽象句法》和《RDF 词汇描述语言 1.0: RDF 模式》的语言
主题映射(TM)	符合 ISO/IEC 13250《主题映射 信息技术 文件描述和处理语言》
统一建模语言(UML)	符合 ISO/IEC 19501《信息技术 开放分布式处理 统一建模语言》
可扩展公共逻辑置标语言(XCL)	公共逻辑的一个分支,在 ISO/IEC 24707《信息技术 公共逻辑(CL):基于逻辑语言族的框架》的附录 C 中规定
其他	
注: UML 是对象管理组的商标。	

参 考 文 献

- [1] Ontology Definition Metamodel, Fifth Revised Submission to OMG/RFP ad/2003-03-40, 2006-04-12.
- [2] Knowledge Interchange Format, Draft proposed American National Standard (dpANS), NCITS.T2/98-004, 1998.
- [3] BERNERS-LEE, et. al.. RFC 2396. Uniform Resource Identifiers (URI): Generic Syntax, 1998.
-

中 华 人 民 共 和 国
国 家 标 准
信息技术 互操作性元模型框架(MFI)
第 3 部分:本体注册元模型
GB/T 32392.3—2015/ISO/IEC 19763-3:2007

*

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.net.cn
总编室:(010)68533533 发行中心:(010)51780238
读者服务部:(010)68523946
中国标准出版社秦皇岛印刷厂印刷
各地新华书店经销

*

开本 880×1230 1/16 印张 1.5 字数 36 千字
2016 年 2 月第一版 2016 年 2 月第一次印刷

*

书号: 155066·1-53006 定价 24.00 元

如有印装差错 由本社发行中心调换
版权专有 侵权必究
举报电话:(010)68510107



GB/T 32392.3-2015