



中华人民共和国国家标准

GB/T 41782.3—2024/ISO/IEC 21823-3:2021

物联网 系统互操作性 第3部分：语义互操作性

Internet of things—Interoperability for systems—
Part 3: Semantic interoperability

[ISO/IEC 21823-3:2021, Internet of Things(IoT)—Interoperability for IoT
system—Part 3: Semantic Interoperability, IDT]

布发42-70-4202

2025-02-01实施

国家市场监督管理总局
国家标准化管理委员会 发布

目次

前 言..... III

引言..... M

1 范 围..... 1

2 规范性引用文件..... 1

3 术 语 和 定 义.....1

4 缩 略 语.....2

5 物联网语义互操作过程.....2

 5.1 概 述 2

 5.2 物联网语义互操作过程要 2

 5.3 物联网语义互操作模型 3

 5.4 物 联 网 语 义 互 操 作 指 南 5

6 物联网语义互操作生存周期 11

 6.1 生存周期要求..... 11

 6.2 生存周期模型 13

 6.3 生存周期实施指南 13

附录 A(资料性) 物联网语义互操作性学习的指导..... 16

附录 B(资料性)关于如何开发物联网语义互操作性的指导 19

附录C(资料性)物联网语义互操作生存周期管理指导20

附录 D(资料性) 物联网参考架构的本体规范..... 22

附录 E(资料性)相关的现有本体25

参考文献..... 32

前 言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

本文件是GB/T 41782《物联网 系统互操作性》的第3部分。GB/T 41782已经发布了以下部分：

- 第1部分：框架；
- 第2部分：网络连通性；
- 第3部分：语义互操作性；
- 第4部分：语法互操作性。

本文件等同采用ISO/IEC 21823-3:2021《物联网 物联网系统互操作性第3部分：语义互操作性》。

本文件做了下列最小限度的编辑性改动：

- 为与现有标准协调，将标准名称改为《物联网 系统互操作性第3部分：语义互操作性》；
- 删除了ISO/IEC 21823-3:2021引言中的图1~图3，因此本文件的图1对应ISO/IEC 21823-3:2021的图4，依此类推。

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专利的责任。

本文件由全国信息技术标准化技术委员会(SAC/TC 28)提出并归口。

本文件起草单位：中国电子技术标准化研究院、深圳赛西信息技术有限公司、江苏赛西科技发展有限公司、东南大学、山东省计算中心(国家超级计算济南中心)、北京邮电大学、重庆邮电大学、无锡物联网产业研究院、浙江品日科技股份有限公司、上海格麟倍科技发展有限公司、安徽电信规划设计有限责任公司、江苏中天科技股份有限公司、广东省工业边缘智能创新中心有限公司、西安航天自动化股份有限公司、北京东土科技股份有限公司、豪尔赛科技集团股份有限公司、中国南方电网有限责任公司超高压输电公司、中建科技集团有限公司、中移物联网有限公司、无锡物联网创新中心有限公司、重庆邮电大学工业互联网研究院、中国科学技术大学。

本文件主要起草人：王晓春、苏静茹、杨宏、汪品品、凌振、张弛、郭雄、卓兰、蔡廷晓、孙旭、杨明、李刚、高伟东、耿道渠、李敏、吴明娟、叶少军、孙金洋、赵鹏、刘琮、葛永新、王博、陶怡、程远、俞承志、刘妹、王宁、曾涛、王露、熊小鹏、董接莲、孔令军、孟振亚、苗付友。

引 言

GB/T 41782《物联网 系统互操作性》旨在规定物联网系统互操作框架、网络连通性、数据的语义语法和实体的行为等方面的要求，使物联网系统得到更大规模的应用。GB/T 41782拟由五个部分构成。

- 第1部分：框架。目的在于指导物联网系统及其内部各实体之间交互的框架设计。
- 第2部分：网络连通性。目的在于指导物联网系统内部网络之间和物联网系统不同网络之间的互操作及互联互通。
- 第3部分：语义互操作性。目的在于规定实现物联网系统中数据语义的互操作性要求。
- 第4部分：语法互操作性。目的在于规定实现物联网系统中数据语法的互操作性要求。
- 第5部分：行为互操作性。目的在于指导物联网互操作系统中实体的行为规范。

物联网 系统互操作性

第3部分：语义互操作性

1 范围

本文件给出了物联网系统语义互操作的基本概念，如ISO/IEC 21823-1的方面模型中所述，包括：

- 核心本体对语义互操作的要求；
- 关于如何使用本体和开发特定领域应用程序的最佳实践和指南，包括允许扩展性和连接外部本体的需要；
- 跨域规范和本体的形式化，以统一利用现有本体；
- 相关的物联网本体，以及在模块化、可扩展性、可重用性、可规模化、与上层本体的互操作等方面对特性和方法的比较研究；
- 体现语义互操作必要性和需求的用例和服务场景。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；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。

ISO/IEC 20924 物联网 词汇[Internet of Things(IoT)—Vocabulary]

3 术语和定义

ISO/IEC 20924中界定的以及下列术语和定义适用于本文件。ISO 和IEC 在以下地址维护用于标准化的术语数据库：

IEC Electropedia:<http://www.electropedia.org/>;

ISO 在线浏览平台：<http://www.iso.org/obp>。

3.1

语义互操作性 semantic interoperability

参与的系统可以理解数据模型在主题领域上下文中的含义。

[来源：ISO/IEC 19941:2017,3.1.5,有修改，删除了该术语中的“数据”]

3.2

元数据 metadata

定义和描述其他数据的数据。

[来源：ISO/IEC 11179-3:2013,3.2.74]

3.3

本体 ontology

在规定的知识领域中具体或抽象事物以及它们之间关系的说明描述。

注：这些说明描述宜为计算机可处理的。
[来源：ISO/IEC 19763-3:2010,3.1.1.1]

4 缩略语

下列缩略语适用于本文件。
ICT 信息和通信技术(Information and Communication Technology)
IoT 物联网(Internet of Things)
PKI 公钥基础设施(Public Key Infrastructure)

5 物联网语义互操作过程

5.1 概述

物联网语义互操作允许物联网系统之间使用可以被理解的数据信息模型(或语义含义)交换数据。这些模型基于本体，本体是关于物联网系统概念和关系的可处理规范。物联网语义互操作的好处是有助于满足物联网系统的互操作性、可重用性、可扩展性、可信任性。物联网语义互操作相关人员及其关注点如下：

- a) 本体工程师，专注于本体的开发；
- b) 物联网系统工程师，专注于物联网系统的开发和语义互操作能力的集成。

如图1所示，本文件为本体工程师和物联网系统工程师提供了准备和构建语义互操作的要求和方法规范。下面定义了两类要求。

- a) 物联网语义互操作要求，重点是如何创建数据信息模型，如何在物联网系统中开发和集成解释能力。
- b) 生存周期要求，重点是数据信息模型范围、数据信息模型设计，以及数据信息模型维护方面的管理。

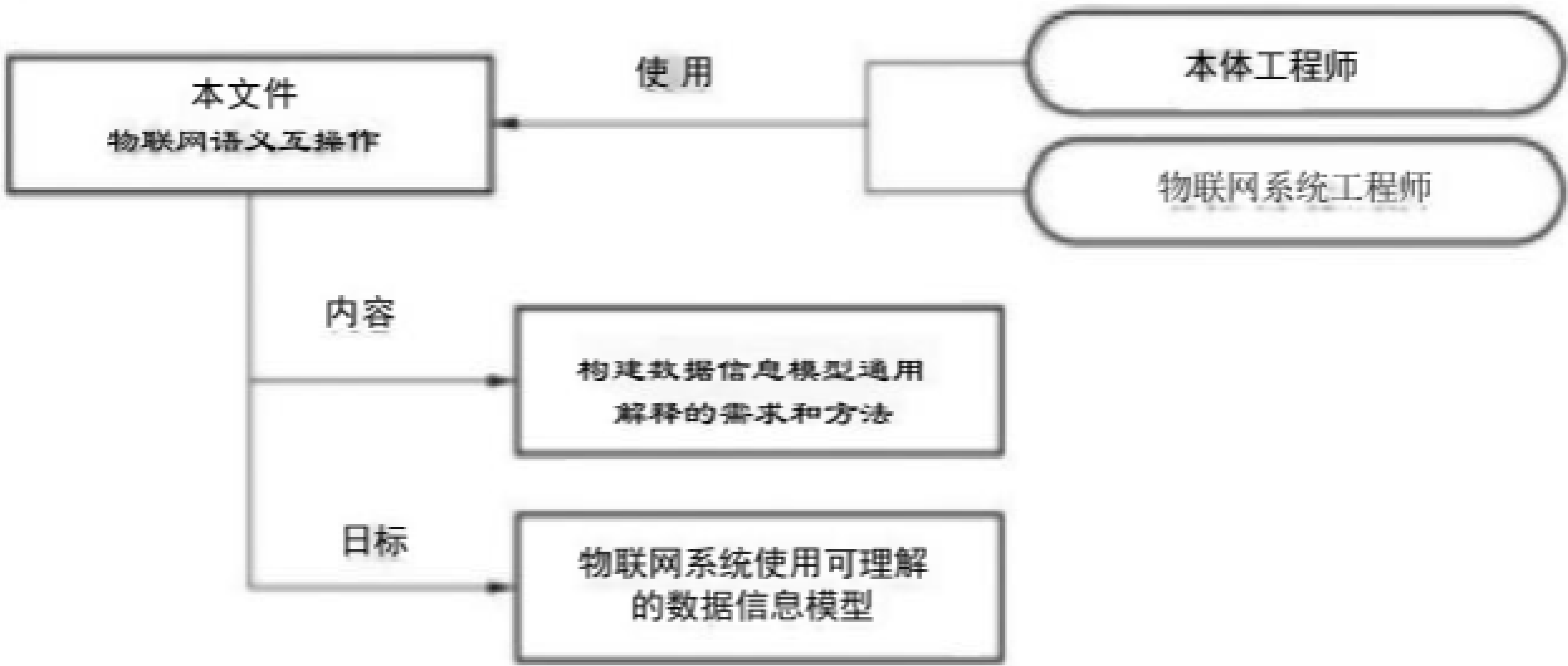


图 1 语义互操作标准的目标

5.2 物联网语义互操作过程要求

物联网系统工程师和本体开发工程师应执行以下过程：
——数据信息模型创建；
——语义互操作能力集成；
——语义互操作工程支持。
语义互操作各过程要求见表1.其中包括基本原理以及可作为合规性证明的最终工作成果。

表 1 物联网语义互操作过程要求

过程	要求	基本原理	产出
数据信息模型创建	使用通用本体获取通用知识	物联网系统依赖于通用的架构、能力和通信手段，如物联网传感器、执行器、网关、设备	基于在物联网系统中收集和记录的通用知识的数据信息模型
	使用领域本体获取特定领域的知识	物联网系统可以专注于运输、能源、健康等特定领域	物联网系统中特定领域的数据信息模型
	使用跨领域本体获取跨领域知识	物联网系统可以跨领域覆盖。例如，电动汽车物联网系统属于交通和能源领域	物联网系统中跨领域的数据信息模型
语义互操作能力集成	开发物联网系统能力以交换物联网设备互操作信息	物联网系统可以使用各种互操作参数，包括技术参数(如驱动程序、协议)和语义参数(如应用功能)。这些参数可以在设备正常运行之前使用元数据进行交换。元数据可以通过模式或子模式的形式公开，也可以通过网络本体语言(OWL)/可扩展置标语言(XML)或基于JSON表示和传输互联数据(JSON-LD)的形式公开。 当物联同设备不能直接提供这种能力时，可使用代理交换这些参数	物联同系统交换物联网设备互操作信息的能力
	开发物联网系统能力以协育互操作配置文件	物联网系统可以使用多种知识表示。需要对用于信息交换的表示达成一致。应通过协商来在关键的语义知识层面上达成一致，例如，使用参数表示并就互操作配置文件达成一致。 当物联网设备不能直接提供该能力时，可以使用代理进行协商	物联网系统协商互操作配置文件的能力
	验证和集成各种功能以实现物联网设备互操作	物联网系统集成了多种技术，以实现与不同互操作方面相对应的互操作性(如驱动程序、协议、中间件)，它们还集成了多个操作(如发现、协商)以实现互操作性	提供整体互操作能力的物联网设备
语义互操作工程支持	提供并应用工具以实现语义互操作	工程师需要发现和选择、映射、对齐、合并和集成等支持工具	可供选择的语义互操作实现工具

5.3 物联网语义互操作模型

物联网语义互操作性可以通过两种模型进行描述：过程模型和使用模型。过程模型关注的是如何创建语义互操作能力，而使用模型关注的是如何使用和交换语义信息。

物联网语义互操作性的过程模型见图2。该模型基于这样的假设：互操作性知识是以可处理的形式提供的，比如通过本体构造的知识图谱。图2中的实体将在5.4.1中进一步描述。

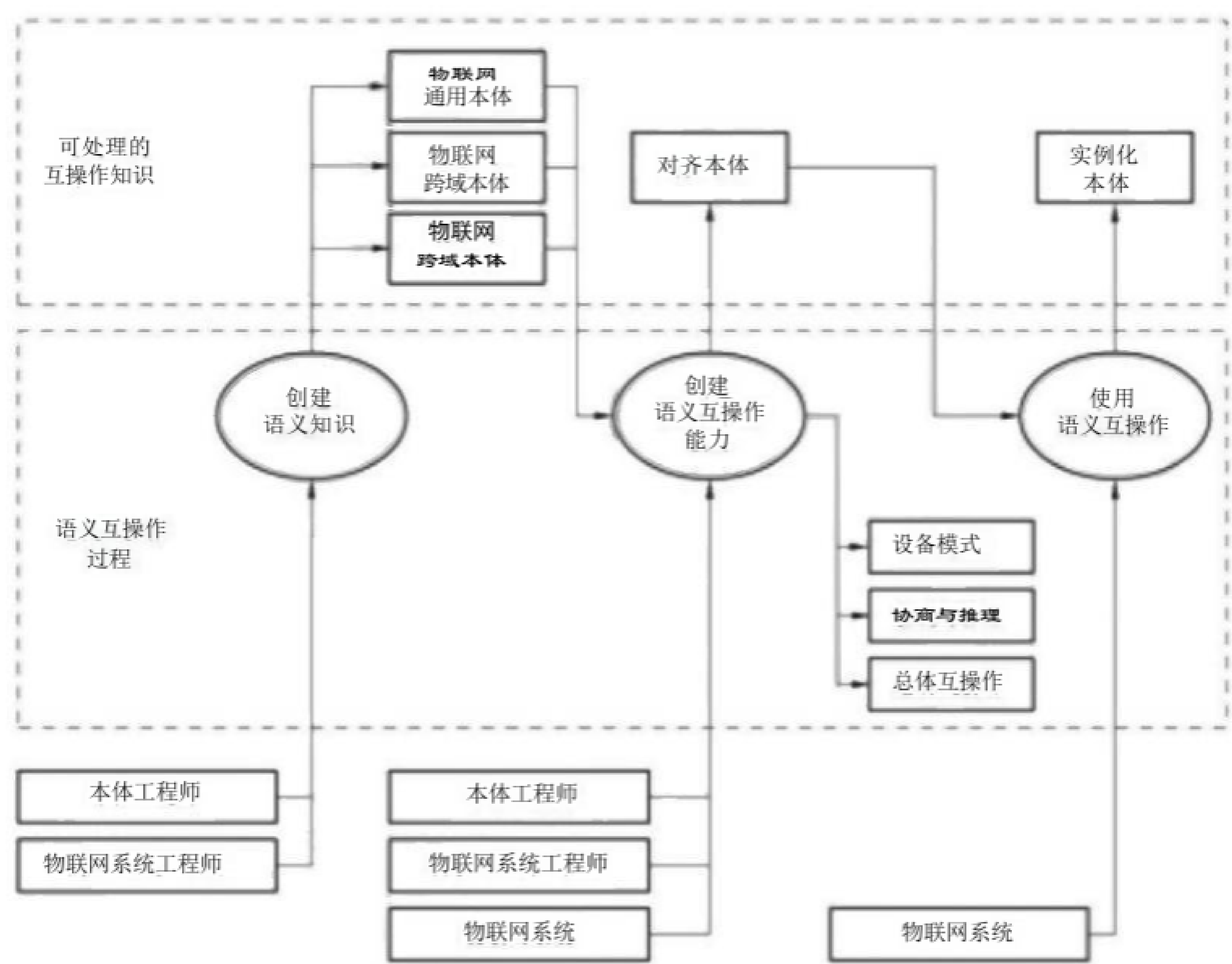


图 2 物联网语义互操作过程模型

物联网语义互操作涉及以下人员：

- 本体开发工程师，专注于本体的开发；
- 物联网系统工程师，专注于物联网系统的开发和互操作能力的集成。

物联网语义互操作包括以下过程。

- a) 创建语义知识：本过程的目的是为互操作创建语义信息，由本体工程师和物联网系统工程师共同完成。本过程的输入是工程师的知识，输出是用于互操作的语义信息的一致数字化表示，由本体表示，包括物联网通用本体、物联网跨域本体和物联网领域本体。
- b) 创建语义互操作能力：本过程的目的是创建语义互操作能力，并使语义信息适应所需要的语义信息表示，语义互操作能力包括：
 - 1)设备模式能力；
 - 2)协商与推理能力；
 - 3)总体互操作能力。

如果物联网系统具有语义信息适应处理能力，则该过程由本体工程师、物联网系统工程师以及物联网系统自身进行。本过程的输入是物联网领域本体、物联网跨域本体和物联网通用本体，输出是对齐本体。

- 本过程由本体工程支持工具以及推理能力(如基于规则的推理、基于本体的推理或机器学习)辅助。
- c) 使用语义互操作：本过程的目的是与其他物联网系统交换语义信息，由物联网系统完成。其输入是对齐的本体，输出是实例化的本体。本过程由本体工程支持工具以及推理能力(如基于规则的推理、基于本体的推理或机器学习)辅助。

语义信息的使用基于图3所示的模型。

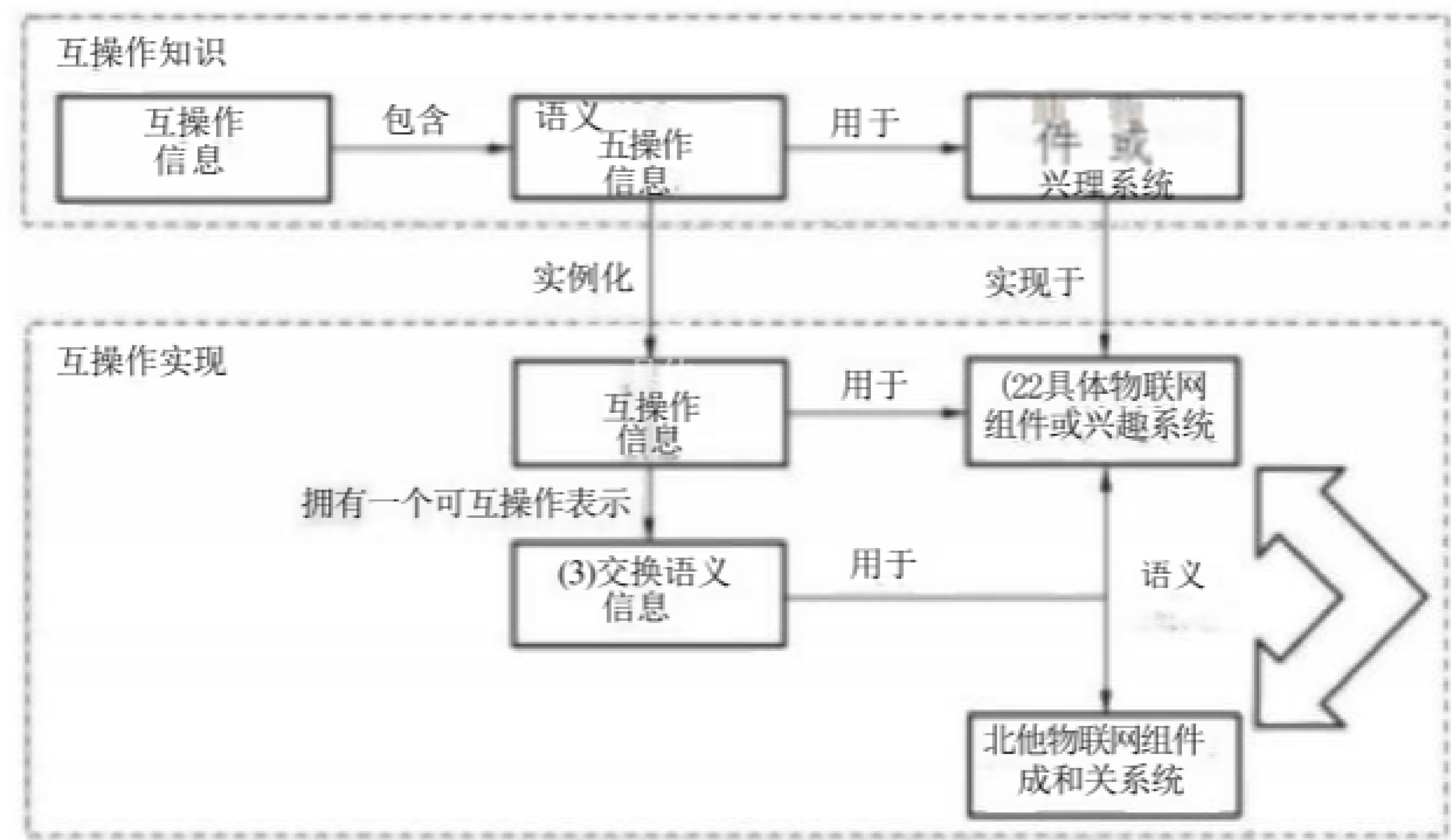


图 3 语义信息使用模型

语义信息使用模型包括以下内容。

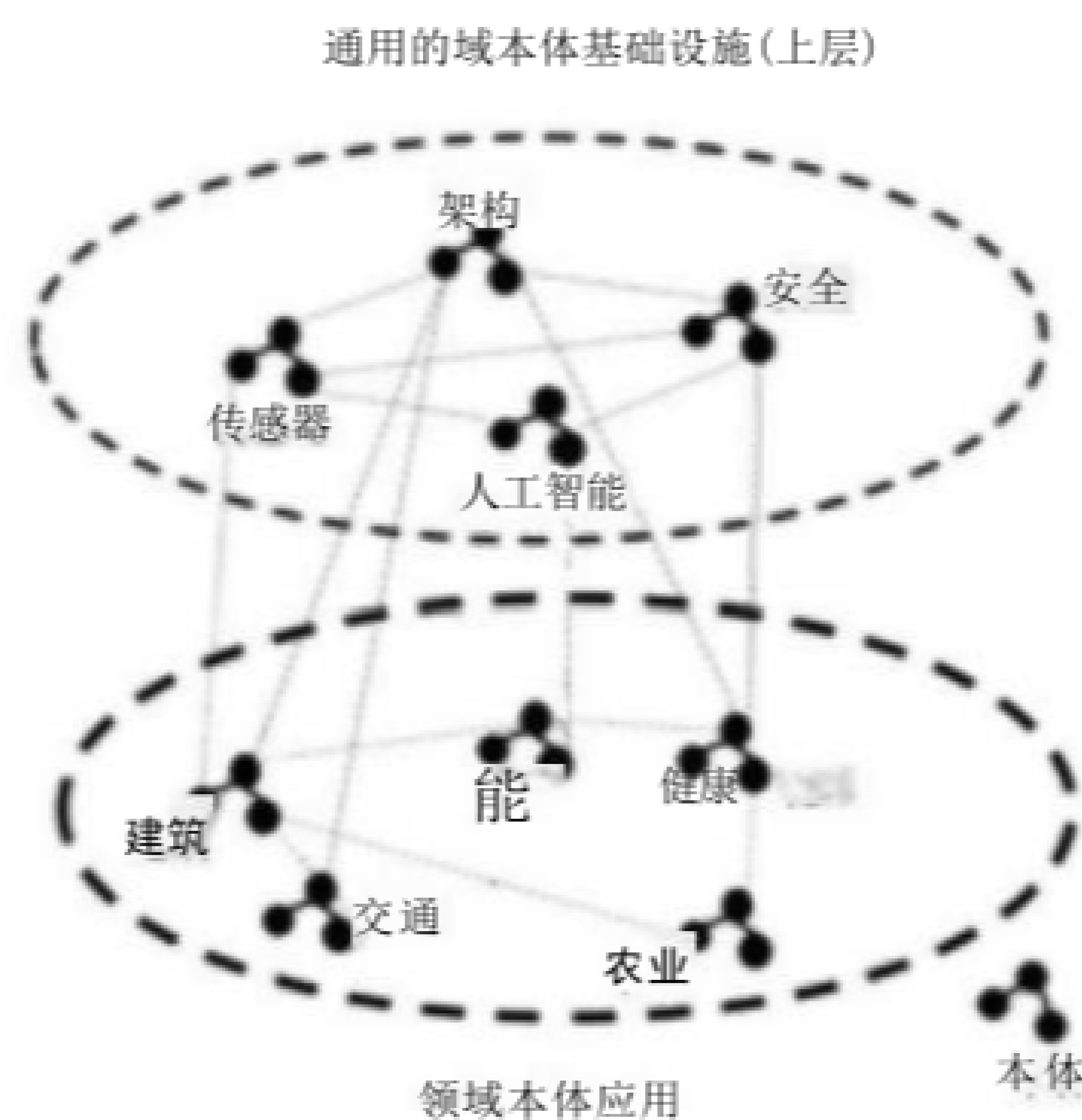
- 物联网组件或相关系统的抽象模型：互操作信息包括与领域中的抽象物联网组件或相关系统相关联的语义互操作信息，例如，与抽象物联网温度传感器相关联的温度模型。
- 物联网组件或相关系统的具体模型：具体互操作信息与操作中具体的物联网组件或相关系统相关联，例如，与物理温度传感器相关联的实例化温度模型。
- 交换的语义信息：可以供其他具体物联网组件或系统使用的语义信息的计算机可处理表示，例如，物理温度传感器提供的温度信息的互操作表示。

5.4 物联网语义互操作指南

5.4.1 获取语义含义的指南

语义含义的获取是通过图2所示的语义知识创建过程实现的。包括以下知识的使用。

- 物联网通用本体。即通用的概念，例如体系结构或框架概念。这些本体可基于物联网标准(如ISO/IEC 30141¹，ISO/IEC 20924:2018)，或映射规范(如参考文献[3])。
- 物联网跨域本体。即领域之间在应用层面(例如，结合了能源和交通知识的电动汽车本体)或在交叉关注点的共享概念。例如，隐私交叉关注点可以基于隐私标准，如ISO/IEC 29100:2011²。
- 物联网领域本体。例如，在能源领域，可使用针对能源管理的通用本体^⑤。其他例子也存在于智能城市领域^④，智能制造领域，或农业领域。



说明：本体通常用三元组表示。它们在图中用三个相连的圆直接表示。

图 4 结构化知识表示示例

构建语义互操作知识的示例见图4. 包括以下内容。

- 通用跨域本体，如架构本体、传感器本体、安全或隐私本体。
 - 与应用相关的特定领域本体，例如健康本体、交通本体、能源本体、农业本体。有许多特定领域的本体可用来构建语义互操作知识。例如物联网相关开放词汇倡议(the Linked Open Vocabulary Initiative for the Internet of Things,LOV4IoT)。
- 这些本体是使用语义网络语言如资源描述框架(RDF)[10]、资源描述框架模式(RDFS)" 和 oWL 进行构建并使用RDF 查询语言和数据获取协议(SPARQL)[13] 进行查询的。
- RDF 是一种基于三元组的基本语言。三元组由主语、谓语和宾语组成。例如，“温度是39度”。“温度“是主语，“是“是谓语，“39度“是宾语。
 - RDFS是 RDF 的扩展，它提供了额外的功能，例如描述层次结构(例如，体温计是传感器的“子类”)。
 - OWL 是一种更加丰富的RDFS, 它允许创建新的类和类之间的关系来描述特定的领域(例如，在物联网领域中添加一个新的“传感器”概念)。
 - 可以使用SPARQL 访问或查询三元组，SPARQL 是一种查询语言，用于与RDF、RDFS和 oWL 图进行交互。

在万维网联盟(W3C)中可以找到的语言层次结构通常被称为语义网络块”。

信息交换的表示被称为序列化数据集。它是语义数据集的计算机可以处理的表示。序列化是指将一个结构转换成可以存储和/或传输的一系列字节。用于这种序列化方法示例是OWL 或 JSON-LD。

5.4.2 语义互操作能力集成指南

语义互操作能力集成是通过如图2所示的语义互操作能力创建过程实现的。包括以下能力的集成。

- a) 物联网系统交换物联网设备互操作信息能力。物联网系统具有多种模式，例如操作模式、维护模式或退出服务模式。
 - 1) 设定模式的目标是定义物联网设备的操作能力。对此能力的描述是基于本体的。由此产生的描述可被设备用来告知其他物联网系统它们的能力。描述包括了技术层面的参

数(如驱动程序、协议、安全、服务质量、数据质量)以及语义层面的参数(如处理能力、访问权限、应用能力)。由此产生的设备本体可用来通知其他已连接的物联网系统将如何解释数据,如温度传感器将具有范围、精度、感知时间等属性。在公共存储库中有很多传感器本体描述,例如LOV4IoT[9]或W3C的语义传感器网络本体[16][17]。

- 2)其他模式包括了操作模式或维护模式。在操作模式下,物联网系统已经就语义互操作条件达成一致,因此设备可以进行交互,例如可以接收感应信息和发出执行信息。在维护模式下,物联网系统可以修改语义互操作参数。
- b) 物联网系统协商与推理能力。由于存在多种知识表示形式,因此需要进行协商。本体是反映了不同设计者不同观点和要求的主观表示。例如,对于一个给定的传感器配置文件可以存在不同的本体描述,这些描述可能存储在一个在线存储库中。对于本体的必要操作包括会进行对齐、映射或合并。此外,还包括推理能力,例如基于规则的推理、基于本体的推理或机器学习。
- c) 物联网设备整体互操作能力。在操作之前,物联网设备通信子系统应经过配置,从而包括所有已协商的各种组件,以实现互操作。在操作过程中,可以交换语义描述。如图3所示,语义信息使用模型存在三种表示层次结构:抽象表示、具体表示和信息交换表示。图5为基于ISO/IEC 30141实现的物联网温度传感器语义信息使用示例。
- 1) 使用抽象的物联网温度传感器表示。这种表示依赖于已达成共识的智能家居本体和基于ISO/IEC 30141的参考架构。智能家居本体包含舒适性、空间感、节能等概念。其包含一个抽象的物联网温度传感器模型,该模型包含了功能描述(例如读取温度)。其引用基于ISO/IEC 30141的架构本体,因而使用服务、网络、物联网设备和物联网网关等概念。
- 2)使用具体的物联网温度传感器表示。这个表示包含了实际信息,一个表示格式的例子是RDF图表示。RDF是W3C指定的语义网络语言。RDF图被构造为三元组,可以使用SPARQL语言对其进行访问或查询。
- 3)使用被称作序列化数据集的信息交换表示。它是语义数据集的计算机可处理表示形式。序列化是指将结构转换为可以存储和/或传输的一系列字节。使用这样的编码方法的例子有OWL/XML和JSON-LD。

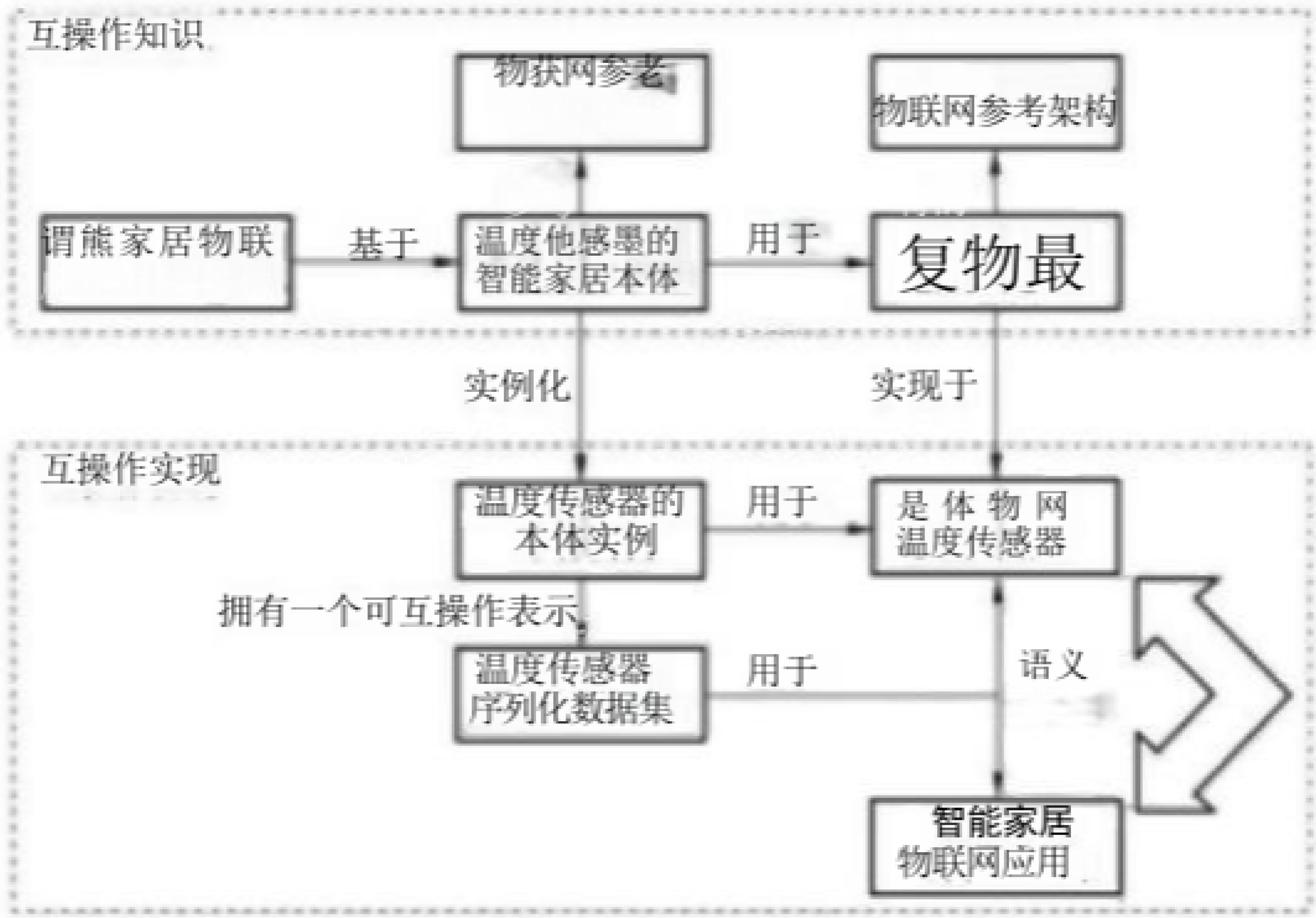


图 5 温度传感器的语义信息使用示例

5.4.3 语义互操作工程支持指南

5.4.3.1 概述

语义互操作工程需要本体匹配的专业知识和对本体操作的支持。而本体之间可能是不匹配的，其不匹配可能发生在语言层面或本体层面。因此，语义信息的创建可以包括本体协调活动，以协调不同本体之间的差异。这些协调活动可以包括对齐、映射、合并和其他所需的操作。例如，物联网应用程序可以使用这些操作来导出(或制定)发送给执行器的命令。

注1:这些操作需要本体工程支持工具和适当的管理和合并本体准则来协助完成。不同的团体可能有不同的要求。采用不同的起点，也使用不同的术语；当语义模型随着时间的推移而发展时，将会出现挑战，例如为不同版本分配标识符，或者为后续版本之间的更改提供一个模型；通过使用来自正例和反例语料库的机器学习，可以简化本体的开发和维护工作，从而允许对数据和元数据进行自动修订以跟踪本体的更改。

注2:尽管在受约束的环境中使用相同的本体是可行的，但普遍上需要使用不同的本体来解决参与方之间的互操作。这可能涉及映射数据和元数据，这些数据可能与上下文有关。从较弱的本体映射到较强的本体时如何处理丢失的细节，这也是解决上述问题面临的一个挑战。将来可以通过利用具有先验知识和过去经验的统计方法来解决这个挑战。

后续条款介绍了以下内容：

- 本体发现(5.4.3.2)；
- 基于本体的数据集成(5.4.3.3)；
- 本体协调，例如本体映射、本体对齐、本体合并和本体整合(5.4.3.4)。

5.4.3.2 本体发现

本体发现是访问操作本体并将其用于语义互操作的过程。如图2中的语义互操作模型过程所示，本体发现可以在创建语义知识时发生，也可以在后续的设备模式设置期间发生。在本体发现过程中可以使用本体存储库，例如[9][18][19]，也可以使用不同的本体(语义网络)语言，例如RDFS[11] 或oWL[12]。

5.4.3.3 基于本体的数据集成

基于本体的数据集成使用多个不同的本体，这些本体通过组合来自多个异构源的数据或信息来提供数据的互操作性。来自多个异构源的数据具有以下几种异质性特征[21]：

- 语法异质性，即语法上的差异，例如数据的格式和结构；
- 结构异质性，即存储数据的本机模型或结构的差异；
- 语义异质性，即数据含义解释上的差异；
- 系统异构性，即源自不同操作系统和硬件平台的差异。

本体作为明确定义的概念以及这些概念之间关系的一种表示，被用于解决数据源中语义异质性的问题。所有信息都通过本体提供的通用标识方案(例如统一资源定位器(Uniform Resource Locator, URL)/ 统一资源标识符(Uniform Resource Identifier, URI))进行链接和访问。本体支持对异构信息系统中的实体进行明确的标识，并确定将这些实体联系在一起的适用命名关系。具体而言，本体发挥以下作用。

- a) 内容和/或上下文说明：本体通过显式定义本体中的术语和关系，实现从多个数据源中准确解释数据。
- b) 查询模型：在某些系统中，查询是使用本体作为一种全局查询模式来构建的。查询构建是将用户发出的原始关键字查询转换为搜索引擎使用的结构化查询表示的过程。通常使用“SPARQL” 查询信息。

- c) 验证：本体能用于验证用来集成来自多个源的数据的映射。这些映射关系既能人工指定，也能由系统(例如本体匹配工具)生成。
- 基于本体的数据集成方法主要有以下三种。
- a) 单一本体方法：单一本体在系统中被用作全局参考模型。
- b) 多本体方法：结合使用多个本体来实现数据集成，且每个本体建模一个单独的数据源。虽然此方法比单一本体方法更灵活，但它需要在多个本体之间创建映射关系。这种映射关系可能是上下文敏感的，即依赖于数据和元数据。
- c) 混合方法：使用了与通用顶级词汇表链接的多个本体。顶级词汇表定义了领域内的基本术语。因此，存在通用词汇时，混合方法更容易使用多个本体进行集成。

5.4.3.4 本体协调

5.4.3.4.1 概述

通过利用本体作为内容解释的集成模式，以及作为检测 and 解决内容不一致的工具，可以实现有效的语义互操作。由于本体是共享规范，因此本体的使用应该能够实现一定程度的语义互操作(例如，如果两个系统是由共享同一本体的两个不同实体设计的，则可以确保语义互操作)。但由于本体是反映不同设计师的不同观点和要求的主观表示，因此不能保证完全的语义互操作。本体之间的这种不匹配可能发生在语言级别或本体级别。语言层面的不匹配来自于本体语言的表达性或语义上的差异(例如，橙可以是水果，也可以是颜色)。本体层面的不匹配来自于本体语义结构的差异(例如，苹果可以被表示为一个概念，因此我们可以描述不同品种的苹果，另外苹果也可以是水果概念的一个实例)。

本体协调是一个通过执行对齐、映射、合并和其他必要操作来协调不同本体之间的差异，从而实现语义互操作的过程。本体匹配是寻找来自不同本体的资源对的过程，这些资源对在含义上是相等的。通常将[0.1]范围内的相似度值用作本体协调的不同方法的输入。

5.4.3.4.2 本体映射

本体映射是使用等价关系或顺序关系将两个或多个信息源的相似概念或关系进行关联的过程。图6展示了本体映射的示例。

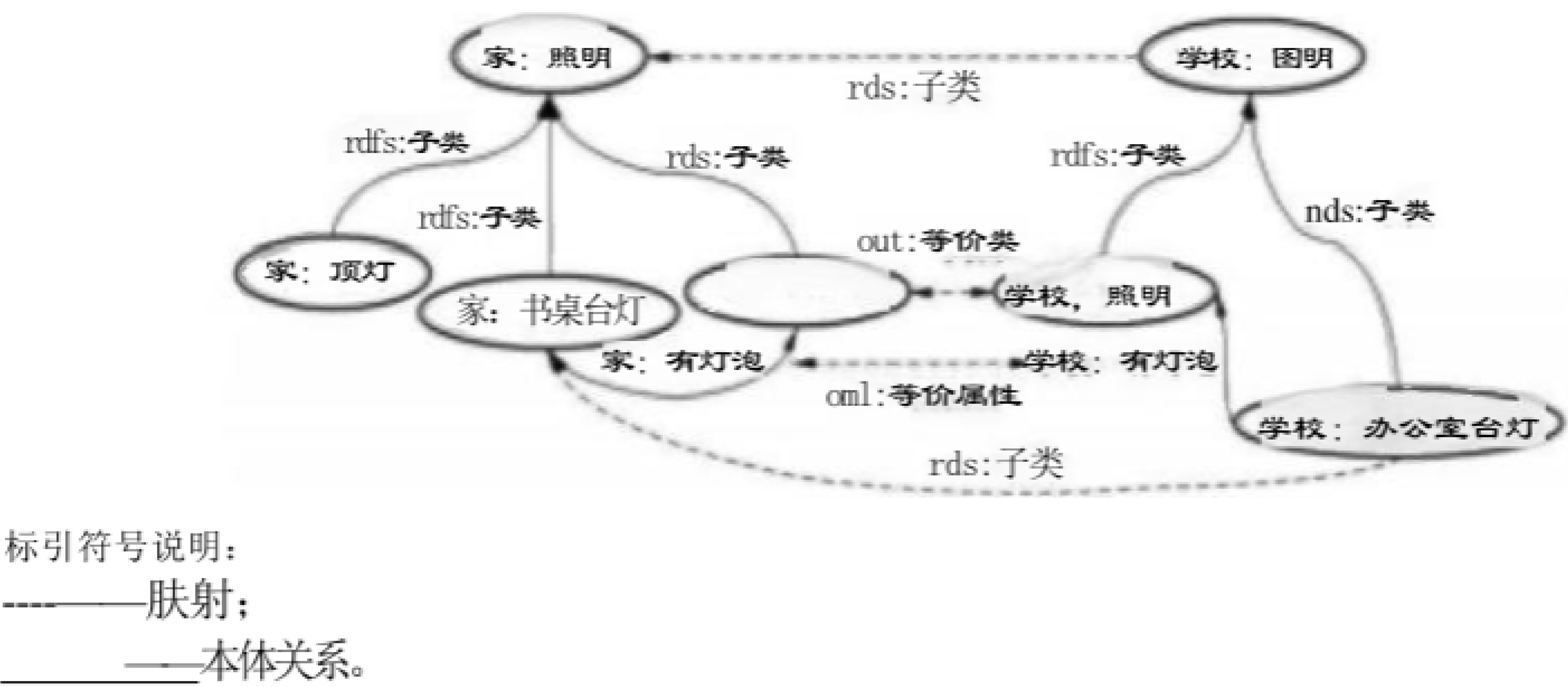
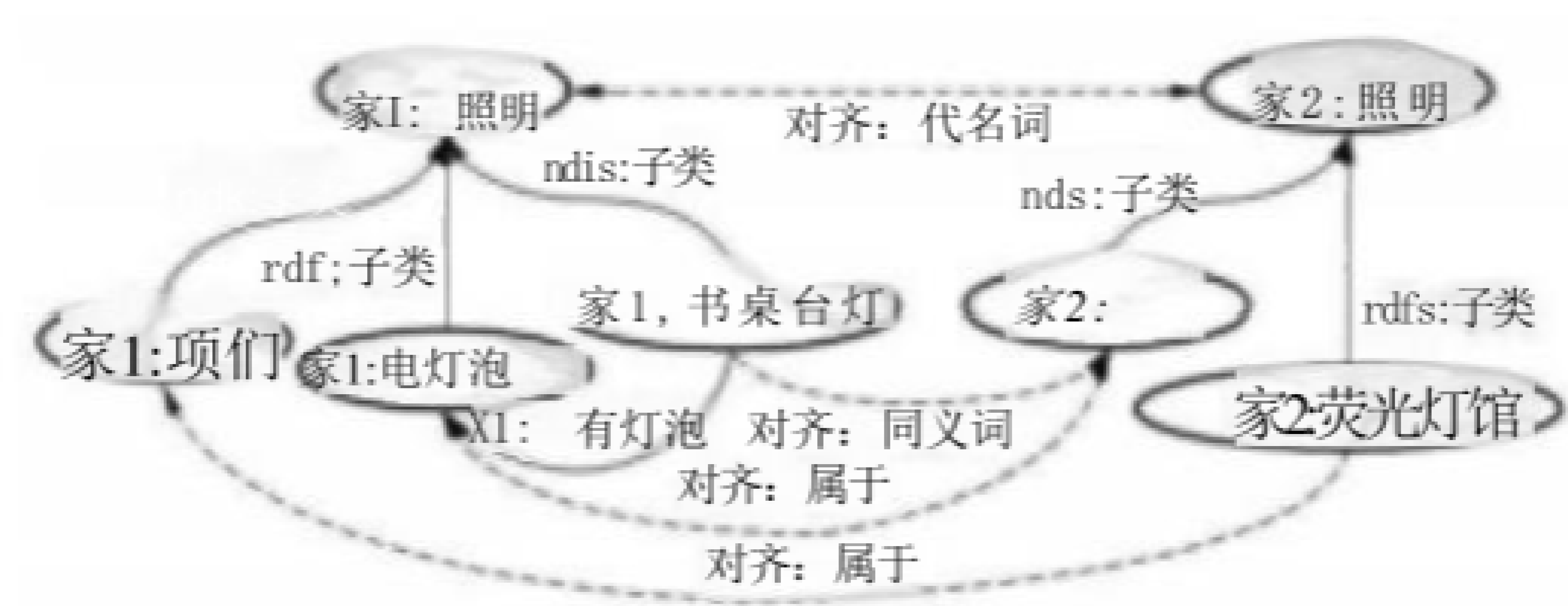


图 6 本体映射示例

5.4.3.4.3 本体对齐

本体对齐是将两个或多个本体相互达成一致，并使它们保持一致和连贯的过程。本体对齐与本体映射相似，但它利用了“属于”和“包含”等本体概念间更具表现力的关系。图7展示了本体对齐的示例。



标引符号说明：
- ——映射：
——本体关系。

图 7 本体对齐示例

5.4.3.4.4 本体合并

本体合并是从两个或多个重叠概念或定义的源本体中创建一个本体的过程。在合并过程中，从零开始创建合并的本体，统一所有源本体。图8展示了本体合并的示例。

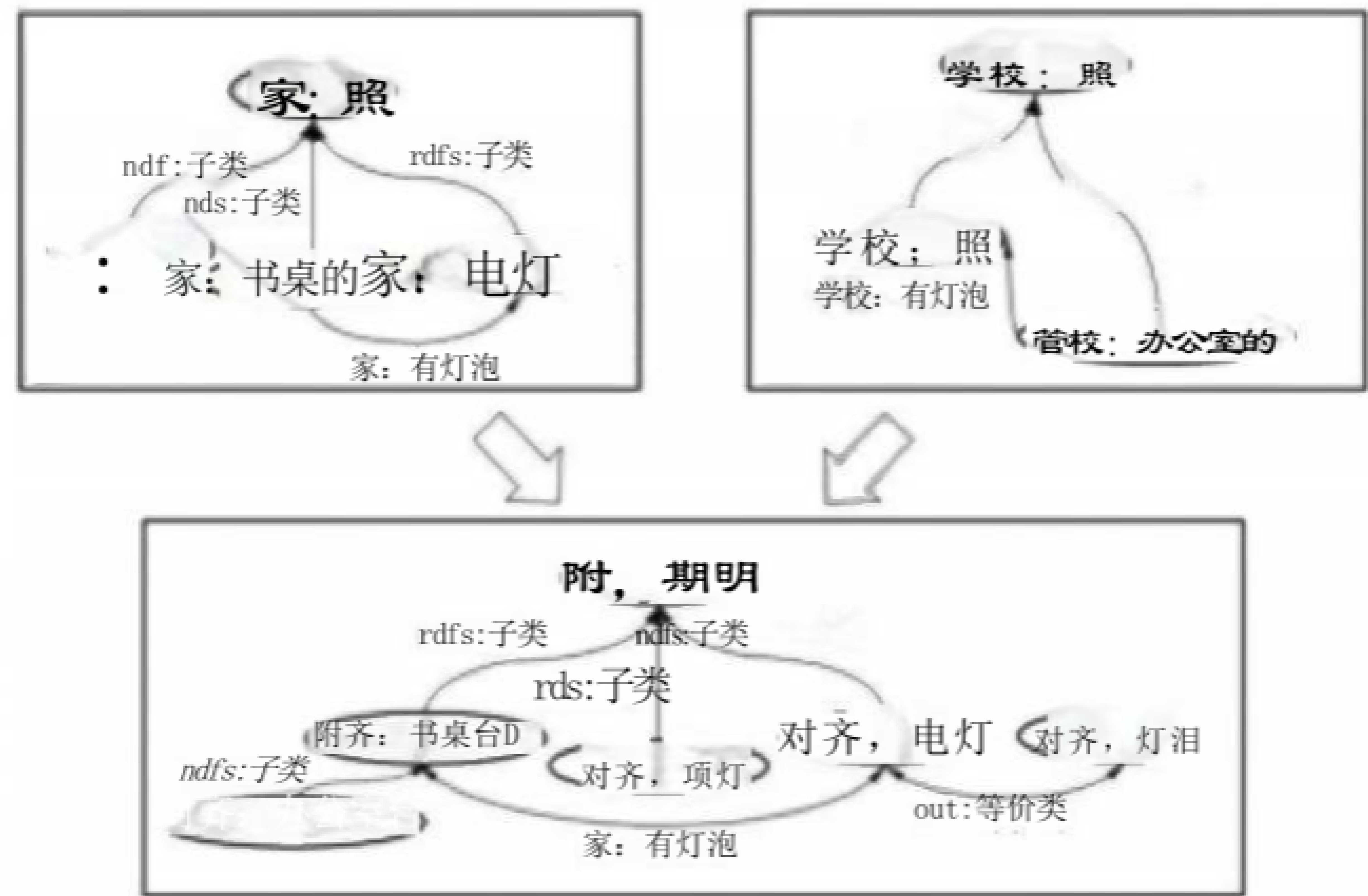


图 8 本体合并示例

5.4.3.4.5 本体集成

本体集成类似于本体合并，但是集成的本体是通过复用部分源本体来创建的。图9展示了本体集成的示例。

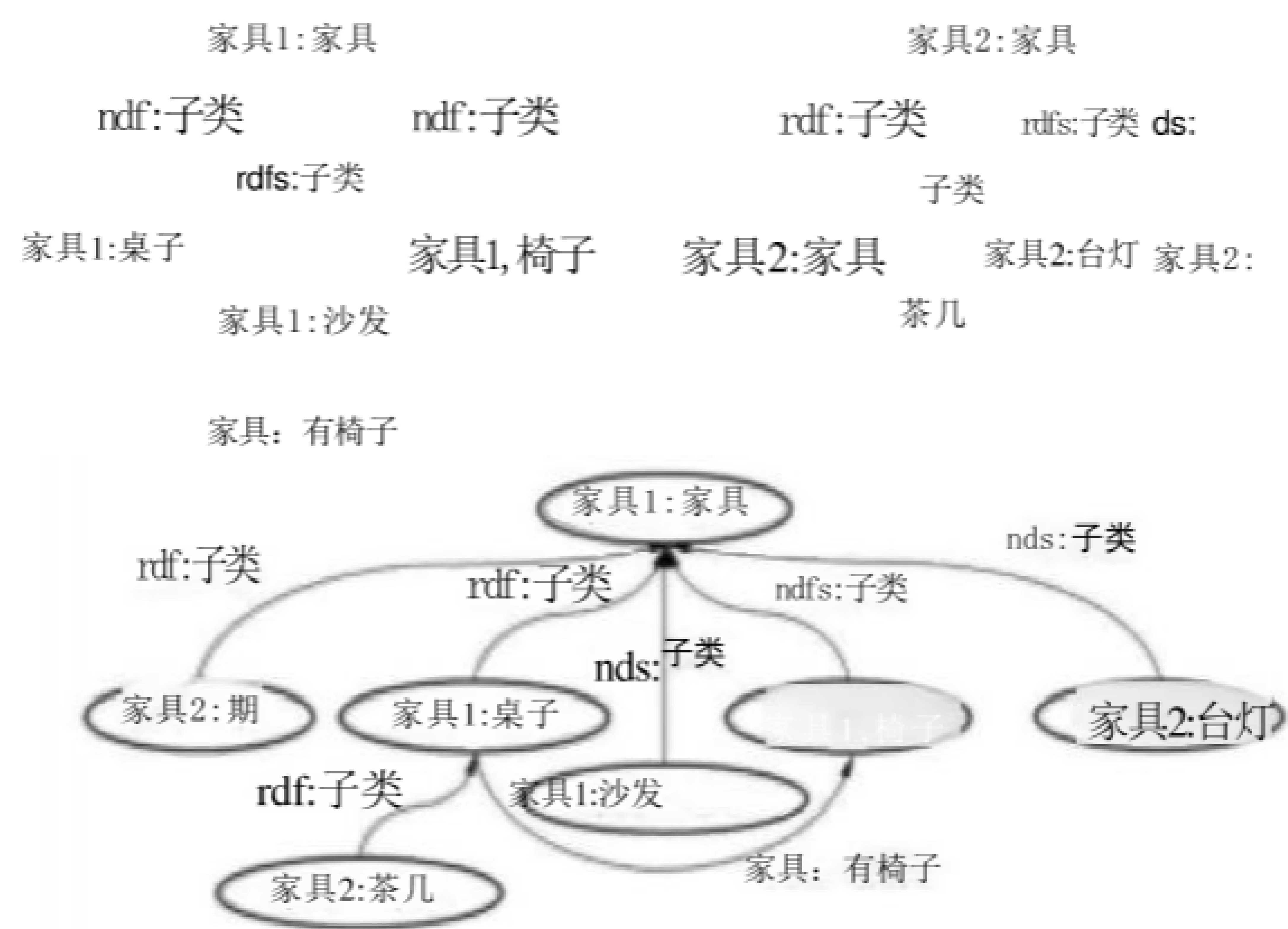


图 9 本体集成示例

6 物联网语义互操作生存周期

6.1 生存周期要求

- 开发物联网系统的组织者应开展以下活动：
- 管理语义互操作的范围，即所需的知识范围；
 - 管理语义互操作的设计；
 - 管理语义互操作的产出，即语义互操作过程的输出。
- 各活动的要求见表2,每一项需求中包括了基本原理和可以作为一致性证据的产出。

表 2 生存周期要求

活动	要求	基本原理	产出
管理语义互操作的范围	定义知识范围	<p>语义互操作的实践需要对知识范围的明确定义，即在语义互操作规范中使用的一组概念和关系。</p> <p>如果选择的知识范围过大，则本体中定义的许多概念可能不会被使用。这也可能适得其反。例如，当使用跨领域本体时，重要的是不要访问整个领域本体，而是访问有助于构建跨领域本体的子集。</p> <p>如果所选知识范围太小，会造成知识概念的丢失，导致规范不完整</p>	规范知识范围的方法指述。 本体知识范围的规范

表 2 生存周期要求(续)

活动	要求	基本原理	产出
管理语义互操作的设计	使用关注点分离原则协同创造本体	<p>协同创造是指将具有不同专业知识和观点的人聚集在一起进行设计的方法。例如，领域专家和本体专家共同合作，产生相互认同的结果。</p> <p>采用关注点分离的设计原则，将要设计的项目分为不同的元素，以便每个元素处理一个单独的关注点。</p> <p>创建用于互操作本体的要求：</p> <ul style="list-style-type: none">——领域工程和语义互操作专家之间应协同创造。根据领域的不同，还可能——需要其他类别的专家，例如安全和隐私专家、以用户为中心的设计专家。——领域专家和语义互操作专家之间应侧重不同的关注点，前者侧重于功能互操作规范，而后者侧重于本体专业知识。 <p>例如，医疗保健语义互操作规范的开发可能包括健康信息和通信技术专家、语义互操作专家、安全和隐私专家以及用户(患者，护理人员)</p>	确保关注点分离和模块化设计的本体协同创造方法描述。协同创造和模块化设计中产生的本体
	应用本体的模块化设计	<p>模块化设计的实践侧重于将一个更广泛的概念构造成多个可以单独处理的更简单的子概念。在本体中，子概念由子本体描述。</p> <ul style="list-style-type: none">——使用独立的知识；——可独立设计，使用和维护；——通过定义良好的关系实现松耦合，从而保留更广泛的本体的语义丰富性；——可重复使用。 <p>模块化设计示例见图10；</p> <ul style="list-style-type: none">——能源和电动汽车本体被分为子本体(非共享和共享)；——跨域本体重用了每个域的共享本体；——最终的语义互操作规范使用了两个现有模块和一个新模块	
管理语义互操作的产出	评估本体成熟度	<p>评估规范成熟度包括两个方面：</p> <ul style="list-style-type: none">——衡量知识规范成熟度的本体完成度情况；——衡量互操作规范成熟度的规范完成度情况。 <p>规范的成熟度评估见图11。它从本体低完成度和规范低完成度开始。随着子本体的重用，本体完成度可能会迅速上升，而规范完成度可能会保持较低水平，因为规范需进行多域共识讨论</p>	用于本体成熟度方法的描述。本体成熟度演化历史
	记录和维护本体	<p>使用和维护互操作规范有更进一步的要求。</p> <ul style="list-style-type: none">——提供配置文件和服务发现。配置文件是一种服务指述，支持特定的可选功能。例如，高端设备可提供低端设备无法提供的互操作功能。服务发现是一种用于发现配置文件的互操作机制。 <p>支持版本管理。语义互操作规范可包括对规范演变或改进的考虑，以适应不同代产品(例如消费设备的新版本)的要求。语义互操作规范可包括版本管理规则(例如，向上兼容性)。</p> <p>提供本体共享和重用。语义互操作规范可以使用可公开访问的文档化本体实现更有效的开发。</p> <p>配置文件、服务发现和支持版本管理的概念可道过特定的本体进行描述</p>	维护方法文档说明。 维护本体的存储库

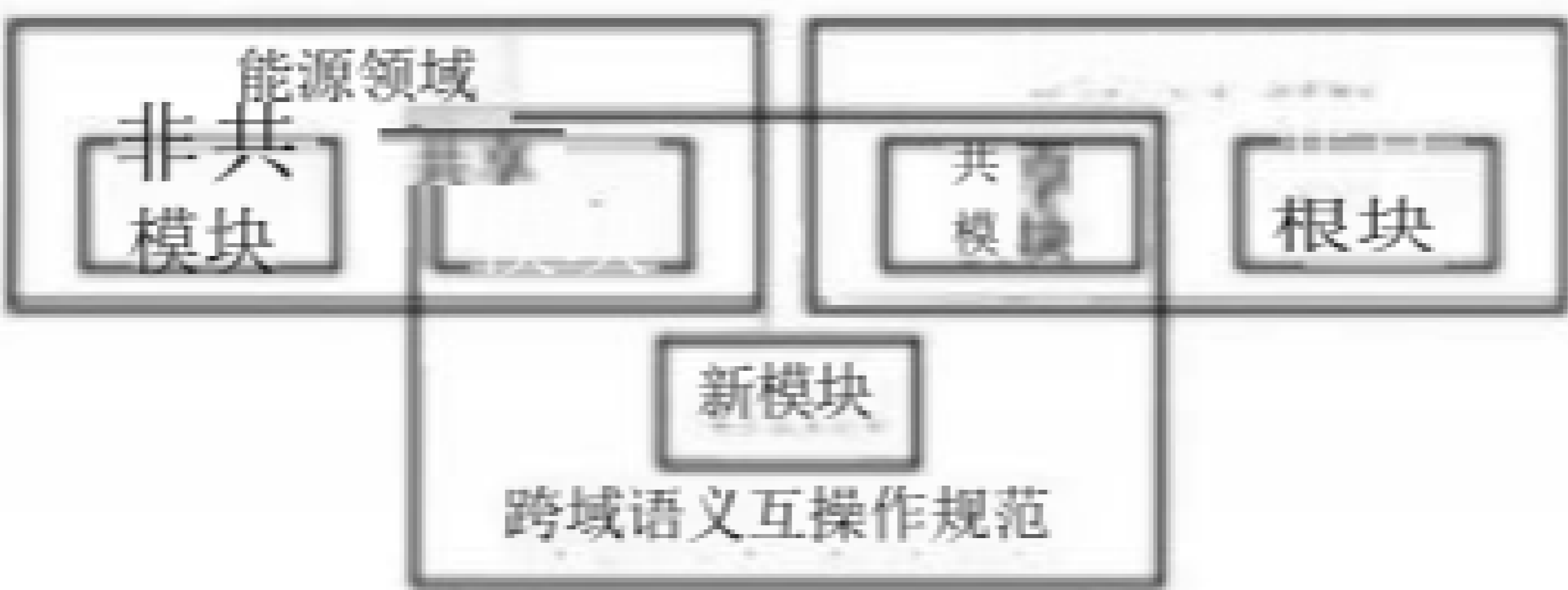


图10 模块化设计示例



图11 互操作成熟度评估结果示例

6.2 生存周期模型

集成本体生存周期所生成的生存周期模型见图12。该生存周期模型展示了各种生存周期之间的关系。下面是有关生存周期模型的一个示例：

- 本体工程师开发能源物联网架构本体、能源管理能力本体和能源信息本体；
- 物联网系统工程师使用能源信息本体来制定能源数据交换语义互操作规范，使用能源数据交换语义互操作规范来开发涉及组件之间数据交换的能源应用程序。

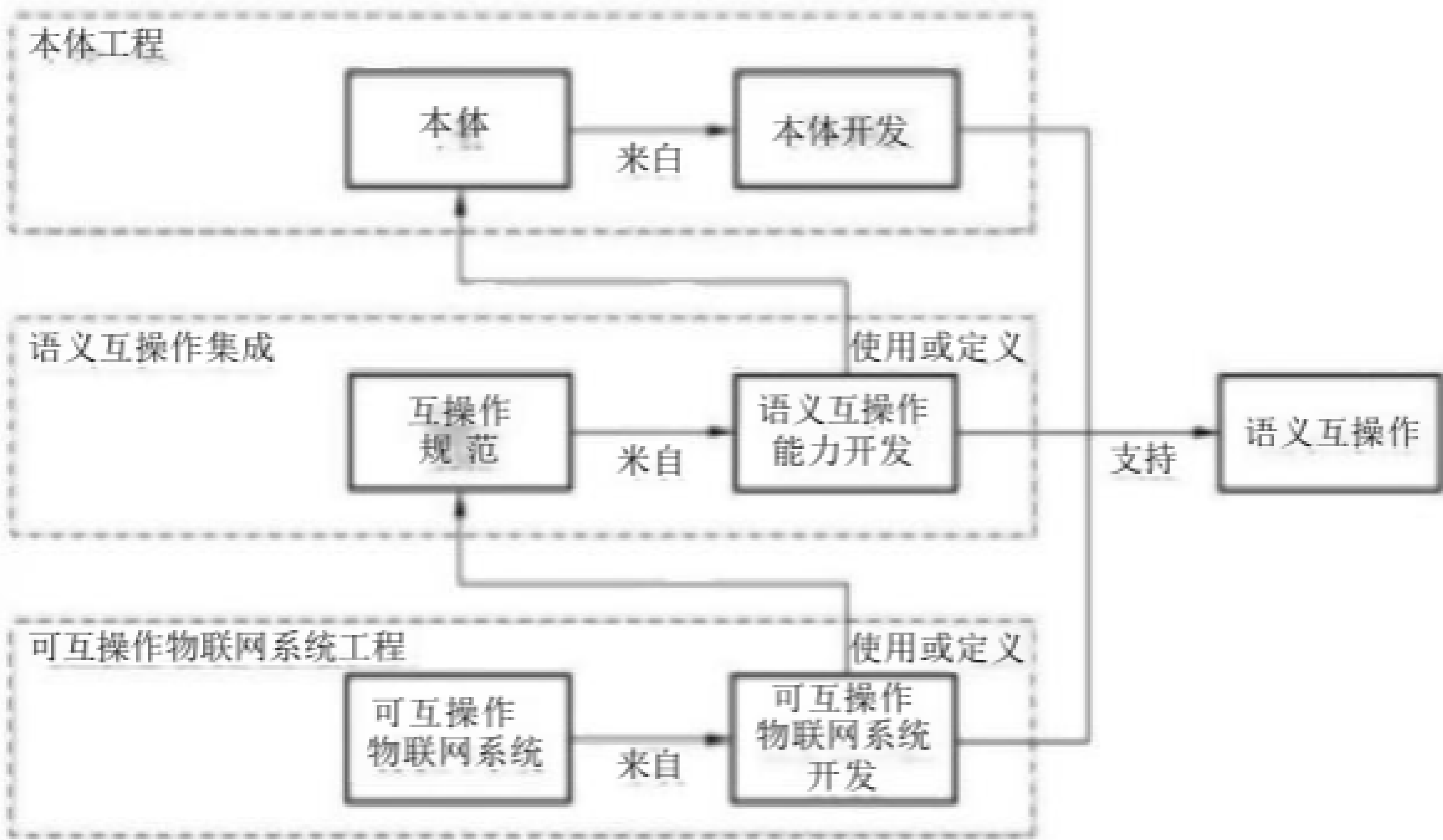


图12 语义互操作生存周期模型

6.3 生存周期实施指南

6.3.1 本体生存周期指南

本体工程学是一种拥有自己独立生存周期的实践工程。图13所示是由以下阶段组成的一个本体生存周期示例。

- 本体需求定义阶段：应定义要创建的本体需求。

- 本体协同创造阶段：应共同创建本体，过程中包括一个特定领域专家和本体专家。
- 本体一致性验证阶段：应验证本体格式正确且一致。
- 本体共识阶段：应确保领域专家同意本体捕获的概念和关系。
- 本体发布阶段：应发布、使用和维护本体。



图13 本体生存周期模型示例

6.3.2 语义互操作生存周期指导

语义互操作工程是一种在互操作规范的生存周期过程中集成语义描述的实践工程。图14所示是由以下阶段组成的互操作生存周期示例。

- 要求阶段：定义互操作规范的要求。
- 规范阶段：提供规范。
- 一致性验证：就规范达成一致。
- 发布：发布互操作规范。

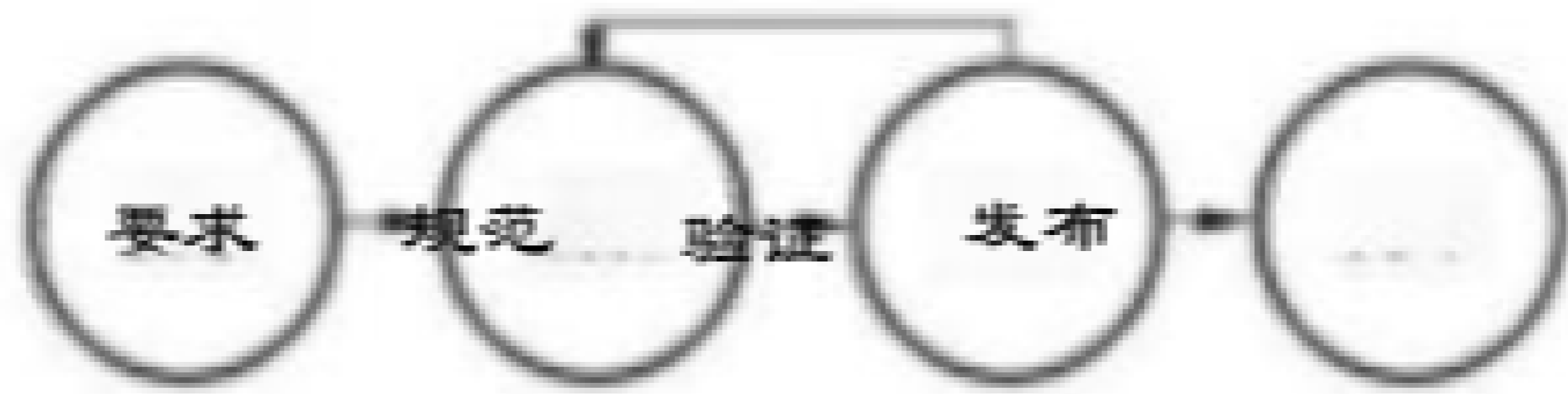


图14 互操作规范化生存周期示例

语义互操作规范中语义描述集成基于本体。

- a) 语义描述可使用现有的本体或创建新的本体。
- b) 语义描述可侧重于不同的语义项：
 - 1)信息本体(例如，协议数据单元中包含的语义信息);
 - 2)交互本体(例如，协议的语义);
 - 3)功能本体(例如，配置文件的语义描述);
 - 4)模式本体(例如，发布订阅范例的语义描述，服务发现机制的语义描述);
 - 5)体系结构本体(例如，体系结构特征的语义描述)。

本体生存周期和语义互操作生存周期之间的区别可以在下面的示例中解释：针对健康传感器制定的语义互操作的规范中，使用了体系结构、协议和健康本体。该健康本体涉及以下生存周期。

- 互操作生存周期。
- 体系结构本体生存周期。
- 协议本体生存周期。
- 健康本体生存周期。

6.3.3 物联网系统生存周期指导

可互操作的物联网系统工程是一种在物联网系统的生存周期过程中集成互操作的实践工程。GB/T 22032—2021[22]将系统生存周期定义为“抽象功能模型”，表示对系统的需求、实现、利用、演变和处置的概念化。系统生存周期被描述为一组过程，过程根据其目的、结果及其活动来描述。GB/T 22032—2021将三十个过程分为四类。

- a) 协议过程：聚焦与供应商协议有关的活动。
- b) 支持组织项目的过程：聚焦与改善组织业务或事业有关的活动。
- c) 技术管理过程：聚焦管理分配给系统工程的资源和资产。
- d) 技术过程：聚焦整个生存周期中的技术活动。

互操作的集成取决于上下文。

- a) 物联网系统可以基于现有的互操作规范或者创建新的互操作规范。
- b) 规范可以是句法层面的，也可以是语义层面的。
- c) 支持元素(如参考实现或测试工具)是否可用。

语义互操作生存周期与物联网系统生存周期之间的区别可使用以下示例进行说明：远程控制家用电器的物联网系统涉及以下生存周期：

- 远程控制物联网系统生存周期；
- 家用电器语义互操作生存周期。

附录 A
(资料性)
物联网语义互操作性学习的指导

物联网语义互操作需要特定的能力，这种能力不是可直接获取的。表A.1提供了指导人员向物联网系统工程师教授物联网语义互操作的一个教学大纲的示例。

表A.1 物联网语义互操作实践的教学案例

目标	本课程提供了对如何使用本体来实现物联网系统语义互操作的理解，涵盖内容包括本体工程(原则，本体表示，本体使用)和推理能力
学习成果	在完成该课程后，物联网系统工程师应能够： ——用本体来表示物联网系统概念(例如，传感器)； ——用本体来表示物联网系统互操作概念(例如，温度元数据)； ——利用本体开发物联网系统的互操作能力，包括推理能力(例如，当温度超过阈值时，空调被激活)； ——在物联网系统生存周期过程内整合产出(例如，生成激活空调功能的正确文件)
形式	提供两种学习形式： ——物联网工程师在本体工程师的支持下开发物联网系统； ——物联网工程师获得本体工程的能力
前提	物联网系统工程师应当熟悉物联网系统工程。例如，工程师能熟练使用ISO/IEC 20924:2018 ISO/IEC 30141[1]. GB/T 22032—2021”进行工程实践

表A.2 提供了一个基于教学大纲的课程内容范本。

表 A.2 语义互操作实际的课程内容

环节	指述	
物联网语义互操作目标	内容	解释物联网语义互操作的目标： ——捕获语义含义； ——整合语义能力； ——语义互操作工程实践
	教学材料示例	本文件(第5章规定)。 白皮书和推荐文档01.s。
选择教学用例和教学形式	内容	选择一个开发了互操作能力的物联网系统用例(例如，温度传感器触发了加热系统的激活) 选择课程教学形式： ——协同开发实践，例如，物联网系统工程师与一个本体工程师一起工作； ——开发实践，例如，物联网系统工程师开发一个本体
	教学材料示例	来自物联网系统工程师领域的材料，或者参考文献[27]
第一个本体	内容	该环节是实操的。物联网系统工程师被邀请具体介绍他/她的用例中的概念和关系
	教学材料示例	来自物联网系统工程领域的材料

表 A.2 语义互操作实际的课程内容(续)

环节	描述	
第一个被描述的本体	内容	解释本体描述(例如, 三元组)。 在协同开发实践中, 指导人员介绍本体明确规定的环境和开发本体的描述。 在开发实践中, 解释本体规范环境, 物联网系统工程师在指导人员的支持下熟悉了它的使用和开发本体的描述
	教学材料示例	描述方案, 例如RDF, RDF S ² , owl 101本体开发 网络软件WebProtege的使用
第一条规则	内容	该环节是实操的。物联网系统工程师被邀请具体说明他/她想要实现的规则(例如, 如果温度低于10℃, 则激活加热系统)
	教学材料示例	来自物联网系统工程领域的材料
第一条被描述的规则	内容	解释规则的描述(例如, 规则, 推理API)。 在协同开发实践中, 指导人员介绍基于规则的开发环境并且实现规则的描述。 在开发实践中, 解释基于规则的开发环境, 物联网系统工程师在指导人员的支持下熟悉它的使用和开发规则的描述
	教学材料示例	开发框架, 例如Apache Jena
第一个语义处理	内容	该环节是实操的。物联网系统工程师使用其标准的开发实践实现规则(即通过硬连接代码实现规则)
	教学材料示例	物联网系统工程师使用的开发环境的信息
第一个被实现的语义处理	内容	解释推理能力, 例如, 查询本体表示(例如, 使用SPARQL)。 在协同开发实践中, 指导人员介绍推理能力开发环境并基于推理能力实现规则。 在开发实践中, 解释推理能力开发环境, 物联网系统工程师在指导人员的支持下熟悉它的使用并实现与规则相关的语义处理
	教学材料示例	Apache Jena推理引擎
第一个用于互操作的语义标注	内容	解释为确保互操作性而进行语义标注的需要。 该环节是实操的: 物联网系统工程师确定由一个具体的物联网设备提供的与物理数据相关的元数据(例如, 外部气温, 单位: 摄氏度)
	教学材料示例	物联网系统工程师使用的典型物联网设备的信息
第一次实现互操作的语义标注	内容	解释在异构设备产生的本体和数据之间创建动态映射能力的需要。 在协同开发实践中, 指导人员介绍在推理能力开发环境中的映射程序接口, 并进行语义标注。 在开发实践中, 解释映射接口, 物联网系统工程在指导人员的支持下熟悉它的使用并实现语义标注
	教学材料示例	在开发框架中使用的语义标注工作成果的文档(例如Jena)。 关于语义标注的图例

表 A.2 语义互操作实际的课程内容(续)

环节	指述	
融入物联网系统生存周期过程	内容	解释本体开发的方法，以及它们在物联网系统生存周期过程的整合。 解释本体生存周期、互操作生存周期和物联网系统生存周期之间的联系。 介绍物联网语义互操作实践的工作成果(例如，本体、互操作规范、互操作测试)
	教学材料示例	101本体开发方法，NeOn。参见[34]中关于本体方法论的思维导图。 关于生存周期的问题，参见本文件(第6章、附录B、附录C)和ISO/IEC/IEEE 15288

附录 B
(资料性)
关于如何开发物联网语义互操作性的指导

B.1 开发语义互操作性能力

如果某个物联网系统不具备必要的语义互操作能力，那就需要进行开发。这一开发是由希望实现互操作的物联网系统工程师驱动的，通常与本体工程师合作完成，并且可使用本体工程支持工具。

在开发互操作能力时，物联网系统工程师可以处于不同的位置，这取决于他/她是为物联网设备供应商那里工作(作为物联网设备开发者),还是作为一个希望通过使用物联网设备成品构建和部署物联网系统的组织中的物联网系统集成人员。二者都有意愿创建可互操作的系统。

作为物联网设备开发者的物联网系统工程师有更多的自主权来影响设备的工作方式。因此，有可能通过改变设备的属性以匹配公共本体，或者开发一个开发者特有的本体(或本体中的缺失部分)以匹配设备的能力。

作为系统集成人员的物联网系统工程师最关注的是为其构建的系统选择合适的设备。这些设备往往来自多个供应商，物联网工程师依赖于语义互操作能力来提供互操作性。

如果物联网设备供应商提供语义互操作能力(例如，本体),那它们可以被所有使用这些设备的不同物联网系统集成人员(客户)使用。然而，如果某个物联网设备的语义互操作能力不能从物联网设备供应商那里获得，物联网系统集成人员仍然希望通过与本体工程师合作去开发缺失的本体(或者缺失的本体条目),以使用本体方法去实现语义互操作。在这一场景下，除非开发的本体条目被贡献给一个公共本体，否则开发工作将由每个物联网系统集成人员单独进行。

B.2 创建步骤

表B.1 提供了一个例子，可供主要利益相关者(物联网系统工程师)用以构建语义互操作。它也适用于次要利益相关者：物联网系统集成方，物联网设备开发者，系统集成方。

表B.1 中的步骤可以被看作是在开发物联网语义互操作时如何使用本文件的指南。

表B.1 物联网语义互操作性的构建步骤

步骤	描述
步骤1	这种互操作的目的和要求是什么
步骤2	涉及的使用方是谁(例如，组织) 涉及哪些系统(例如，IT端点，物联网端)。 哪些机制(例如，证书和公钥基础架构)被用来识别并认证数字使用方
步骤3	哪些信息将被传输(例如，格式、频率，质量，标识)
步骤4	是否应用语义互操作。 对方可用的语义信息处于什么水平
步骤5	如果步骤4的答案是“是”，那么找出语义互操作和连接参数(例如，基于OneM2M的本体)。然后开发可被添加到发送方的功能，以测试与对方的连接
步骤6	开始测试。阶段一检查双方的合法身份(例如，使用公钥基础架构)
步骤7	测试的第二个阶段检查双方能否交换信息。例如，发送内容差异较大的测试信息以便双方进行分析和质量检测。结果需由双方进行验证，以确保互操作性
步骤8	当测试程序成功运行后，双方认定语义互操作得到保证并使用本体进行交互

附录 C
(资料性)
物联同语义互操作生存周期管理指导

C.1 支持本体的互操作规范化生存周期

表C.1 描述了一个基于本体的互操作规范化生存周期。

表C.1 互操作规范化生存周期示例

互操作规范化生存周期	描述	
要求	识别要创建或使用的本体	——如果需要，使用诸如[28]、[29]等教程熟悉本体的创建和使用； ——识别在本体中有必要获取的知识类型(领域，跨领域和横向，例如，健康，交通和安全)； ——确定现有的本体能否被复用或者扩展。可以使用诸如1.0V4IoT之类的目录； ——定义可能会使用本体的语义描述：信息，交互，功能，模式，体系结构等等； ——定义操作要求(例如，兼容性)； ——确立本体版本管理方案
规范	协同创造本体	——建立领域和本体专家的合作方案以构建本体的结构和内容。在[25]中提供了指南。在[35]中提供了最佳实践； ——验证本体是否语义一致； ——验证模块化水平； ——在领域和本体专家之间建立共识
	明确互操作规范和本体的关系	——列出本体需要支持的概念和关系
	在本体的帮助下检证互操作规范	——列出可被验证的依赖关系； ——确立本体表达的符合性约束； ——检查互操作规范中的约束
符合性验证	语义互操作调试以及部署整合相关本体	——必要时更新本体和规范； ——根据生态系统评估其成熟度(例如，使用本体的组织)； ——集成到本体版本管理中
发布	语义互操作本体的维护	——在现有目录中发布创建或扩展的本体； ——更新和增强暴露的本体； ——验证更新的本体
	语义互操作本体的关闭	更新本体版本管理

C.2 支持互操作的物联网系统生存周期

表C.2 描述了一个支持互操作的物联网系统生存周期。

表C.2 物联网系统生存周期示例

物联网系统生存周期	互操作整合活动
利益相关方的需求和需求定义	确定将要使用或创建的互操作规范
系统需求定义过程	将互操作规范的要求集成到系统要求中
体系结构定义过程	无具体活动
设计定义过程	设计的功能需满足互操作规范的要求
系统分析过程	无具体活动
实现过程	实现的功能需满足互操作规范的要求，必要时开发互操作测试功能
集成过程	无具体活动
核实过程	测试物联网系统的互操作性(内部)
过渡过程	测试物联网系统的互操作性(第三方，插拔测试)
验证过程	部署验证
操作过程	无具体活动
维护过程	整合互操作规范的修改
清除过程	无具体活动

附录 D
(资料性)
物联网参考架构的
本体规范

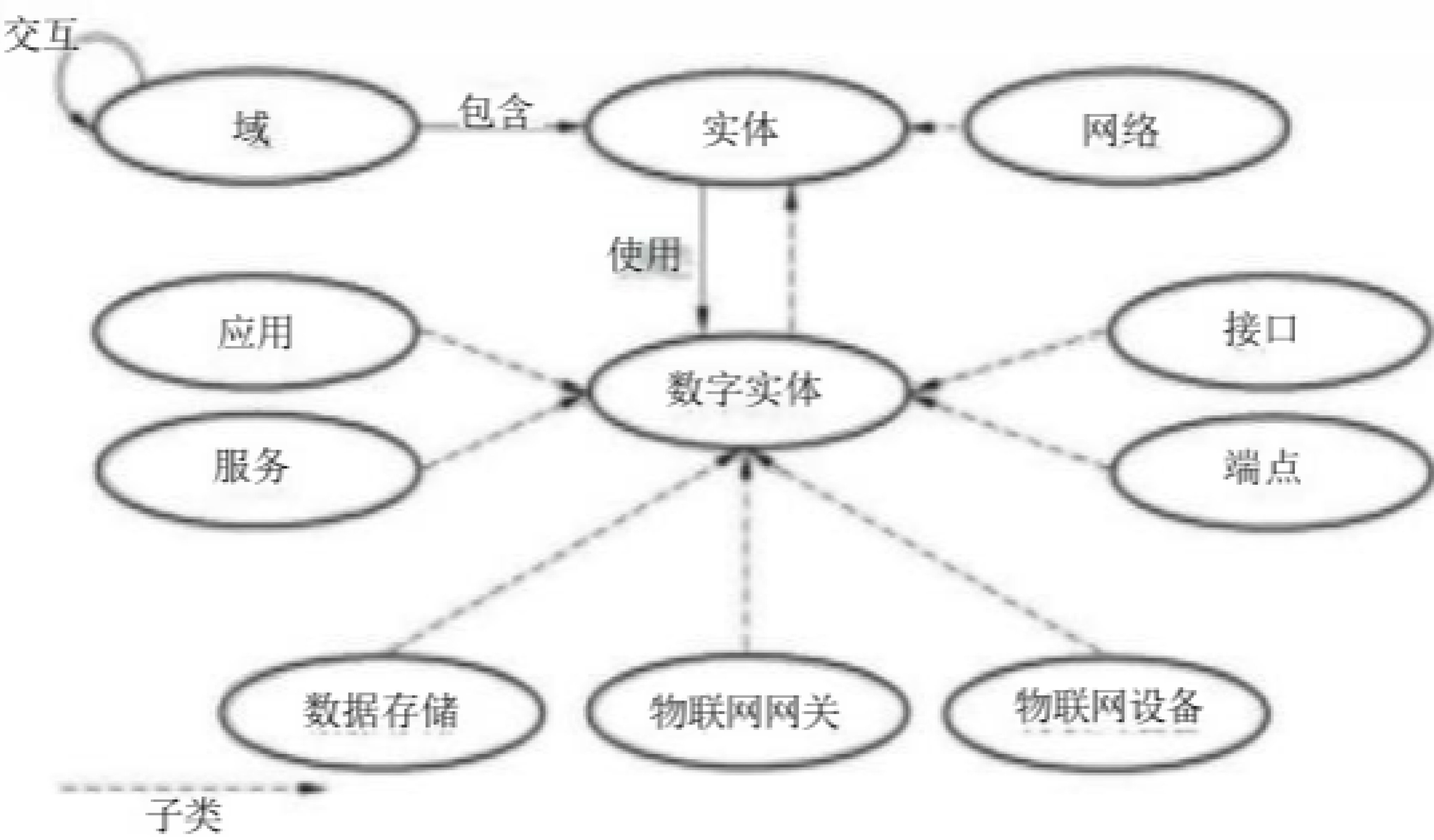
D.1 通则

ISO/IEC 30141标准[1]以定义物联网的系统特性及架构视图的方式指定了通用的物联网参考架构。附录D 提供了参考架构的本体规范。本体规范用Web 本体语言(OWL) 编写[12]. 并在OWL 本体(VOWL) 的可视化表示法中进行了描述[36]。

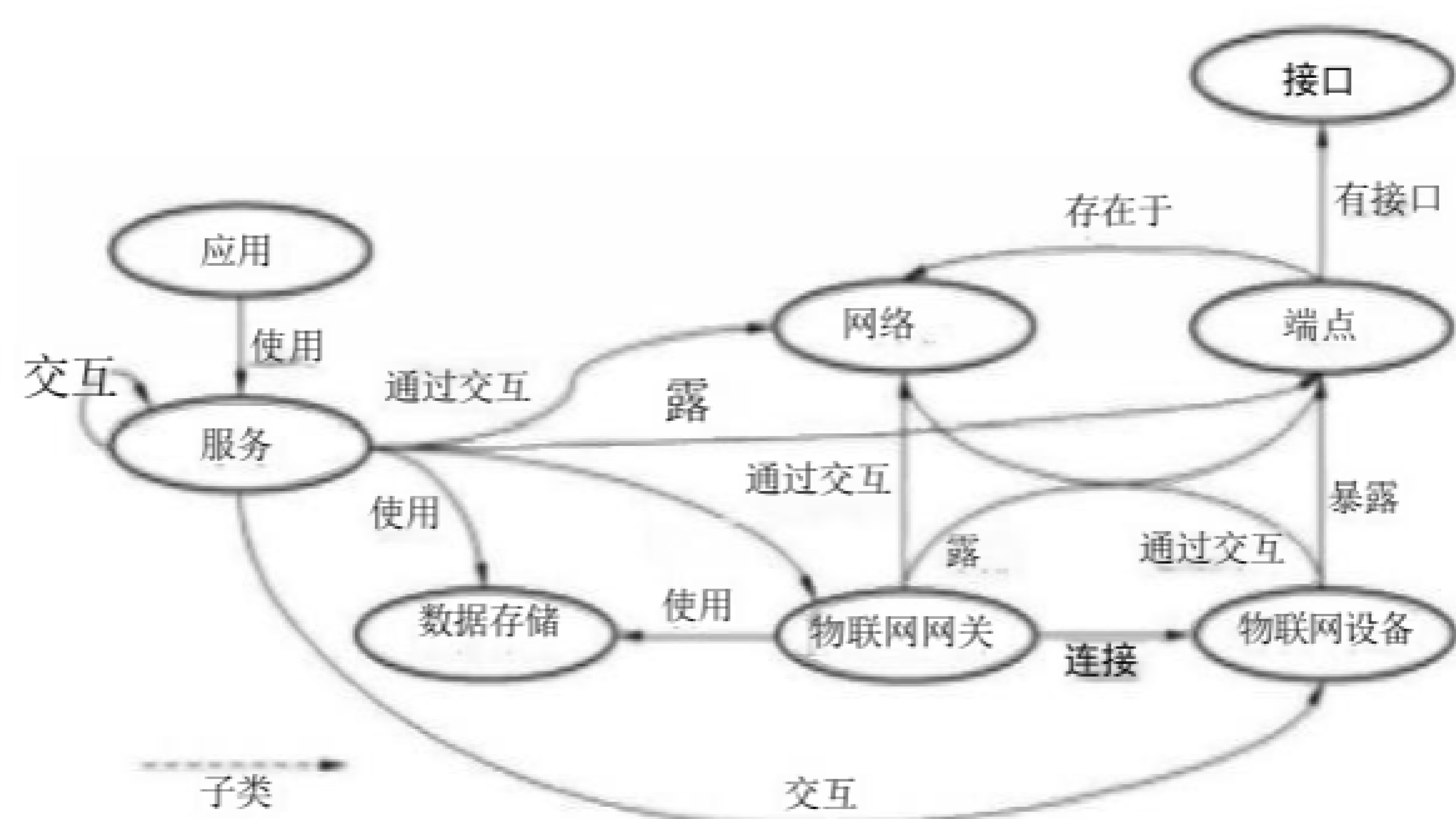
物联网系统交换包括物联网架构构件参考的信息时需要ISO/IEC 30141本体，例如在传感器和应用程序发现阶段。ISO/IEC 30141标准确保了本体使用标准化的概念。这一点很重要，因为不同的本体可以使用同名的概念。例如，语义传感器网络(SSN) 本体论[16]定义了“设备”的概念。同样，OWL-S 本体论[20]定义了“服务”的概念。如果这些本体与物联网参考架构(IoT RA)本体同时使用，那么应该确定这些概念是否具有相同的含义。

D.2 服务、网络、物联网设备和物联网网关

服务是一个抽象的概念。服务由一个或多个组件实现。同一个服务可以有多个备选实现。通过网络进行交互的实体通过在网络上公开一个或多个端点来实现。网络连接端点。服务公开一个或多个端点，通过这些端点可以调用服务。端点具有一个或多个网络接口。端点可以通过通信网络上的网络接口访问位于远程的服务。端点存在于一个或多个网络上。与服务、物联网设备和物联网网关相关联的数据可以保存在由一个或多个实体使用的数据存储中。图D.1和图D.2 描述了与服务、网络、物联网设备和物联网网关相关的整体物联网实体。



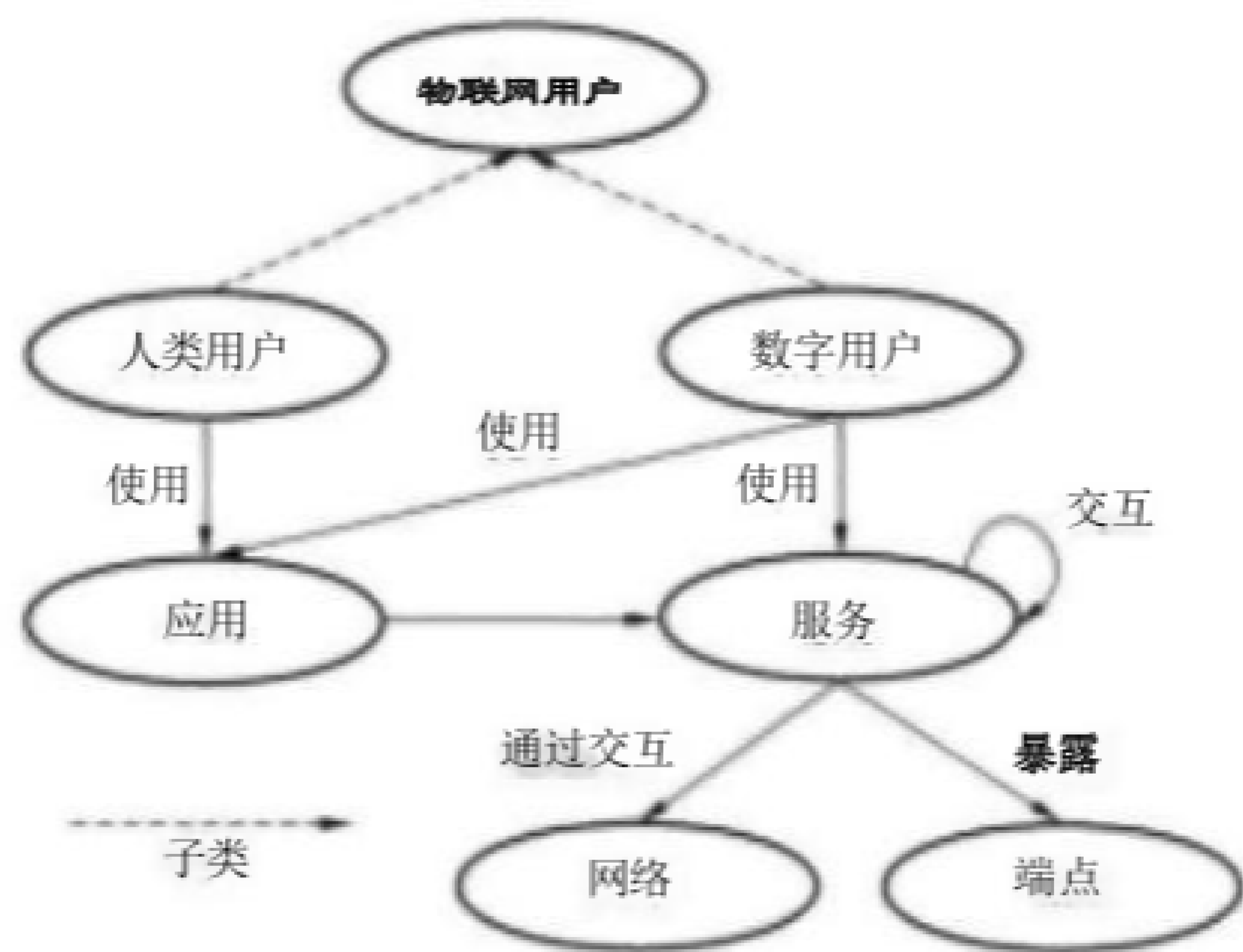
图D.1 物联网实体



图D.2 服务、网络、物联网设备和物联网网关

D.3 物联网用户

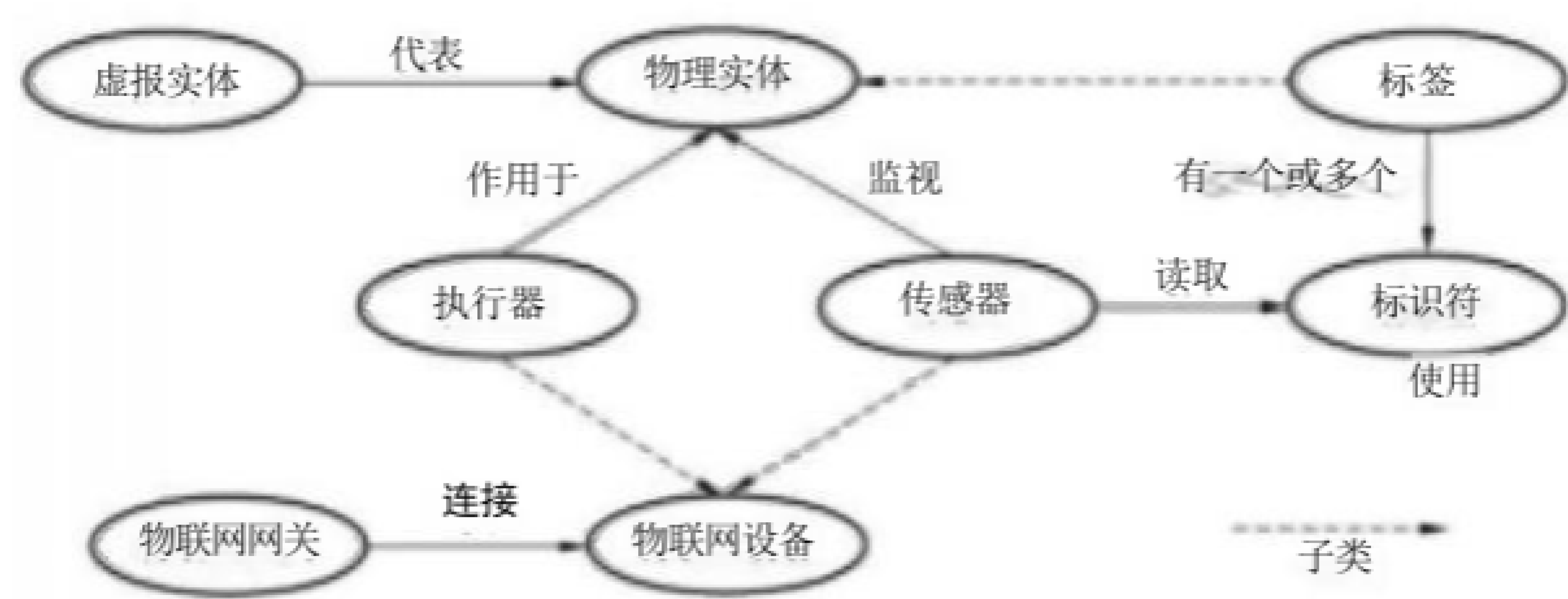
物联网用户可以是人类(人类用户)或数字(数字用户)。数字用户包括代表人类用户的自动化服务，例如机器对机器交互。数字用户通过服务端点直接或间接地与一个或多个服务交互。人类用户通过一个或多个应用程序进行交互。图D.3 将物联网用户描述为人类用户和数字用户的超类。



图D.3 物联网用户

D.4 虚拟实体、物理实体和物联网设备

执行器和传感器是与物理实体有直接或间接接触的物联网装置。执行器对接收到的数字信息进行操作以作用于(改变)物理实体的某些属性。传感器感知物理实体的某些特性，并将其转换为可通信的数字表示。一个物理实体可以有一个或多个标签，传感器可以监视标签，而不是物理实体本身。执行器和传感器是两种物联网设备，它们将一个物理量的变化定量地转换为另一个物理量的变化。单个物联网设备可以在单个设备中容纳多个传感器。图D.4 说明了虚拟实体、物理实体和物联网设备之间的关系。



图D.4 虚拟实体、物理实体和物联网设备

D.5 基于域的参考模型

通过允许逻辑(有时是物理)细分，域帮助设计人员专注于需要执行的各种任务。域主要用于对责任领域中的功能进行排序，这些领域(功能领域)通常被部署在单独的子系统中。确定的域是：用户域、运营和管理领域、应用和服务域、资源访问和交换域、感知和控制域以及物理实体域。每个确定的域与所有其他域互斥。图D.5 给出了基于实体的参考模型和基于域的参考模型之间的关系。从不同的视角来看，每个领域都有不同种类的实体。

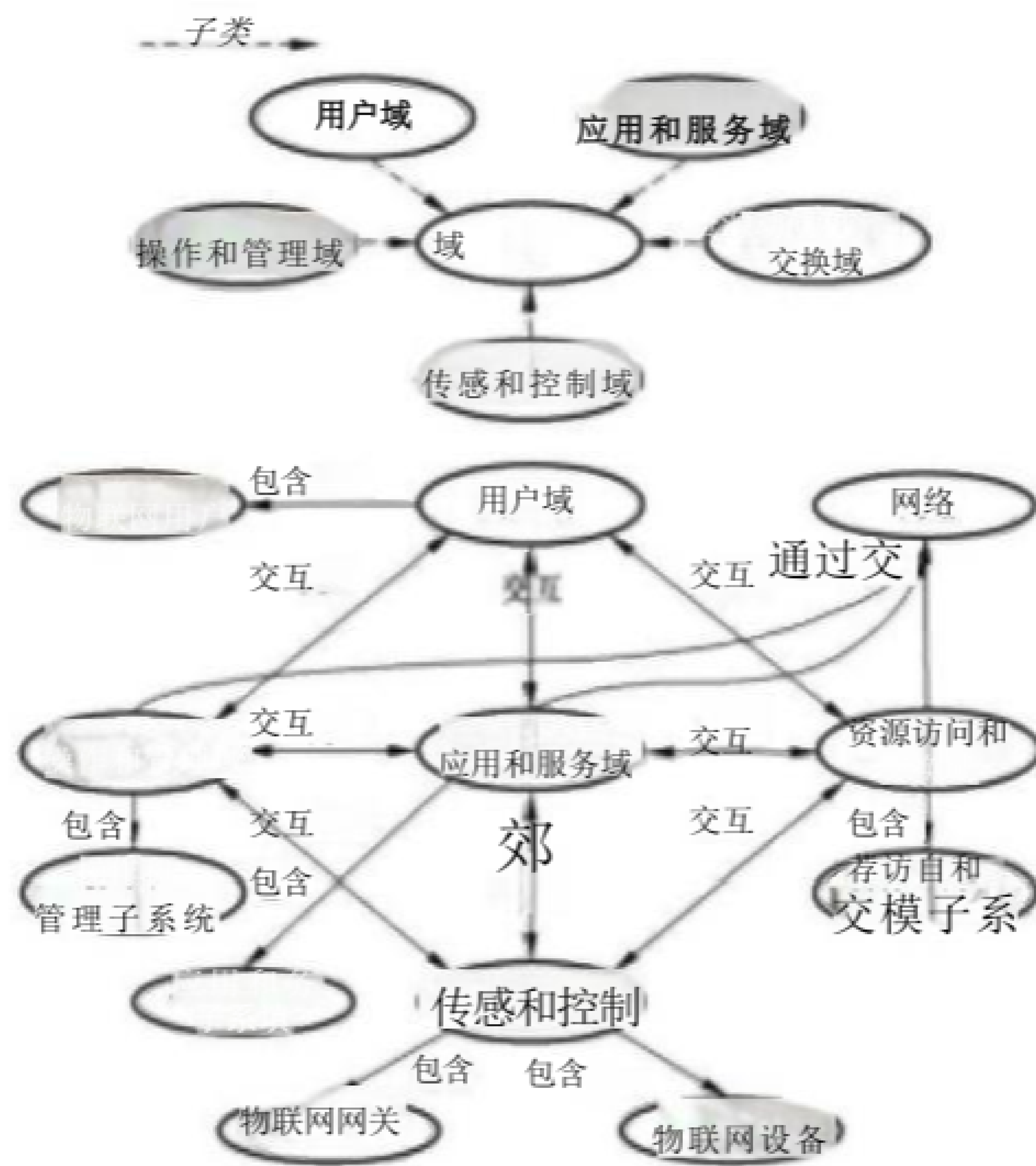


图 D5 基于域的参考模型

附 录 E
(资料性)
相关的现有本体

E.1 W3C语义传感器网络本体论

万维网联盟(W3C) 网络空间数据工作组(WG) 开发了SSN 本体论[16], 它为描述传感器及其观测结果、所涉及的程序、所研究的感兴趣的特征、涉及的样本和所观测的属性, 以及执行器提供了高级模式。SSN 通过为其基本类和属性包括一个轻量级但独立的被称为SOSA (传感器、观测、样本和执行器) 的核心本体遵循水平和垂直的模块化架构。

SSN 本体论围绕系统、过程和观测的概念。它支持对传感器的物理和处理结构的描述。传感器并不局限于物理传感设备: 更准确地说, 传感器可以是任何可以估计或计算一个现象的值的对象。因此一个设备或计算过程或其组合亦可以扮演传感器的角色。传感器在本体中的表示方法将其测量的对象(环境域现象)、物理传感器(设备)和其功能与处理过程(模型)连接在一起。

SSN 本体论围绕着“刺激—传感器—观测”模式。在这里, 刺激是物理世界中可探测的变化。传感器是执行观测操作的物理对象。此外, 观测操作作为传入的刺激物、传感器和传感器的输出之间的纽带。

SSN 本体有多个根据上述模式建立的以覆盖关键的传感器概念的概念模块。

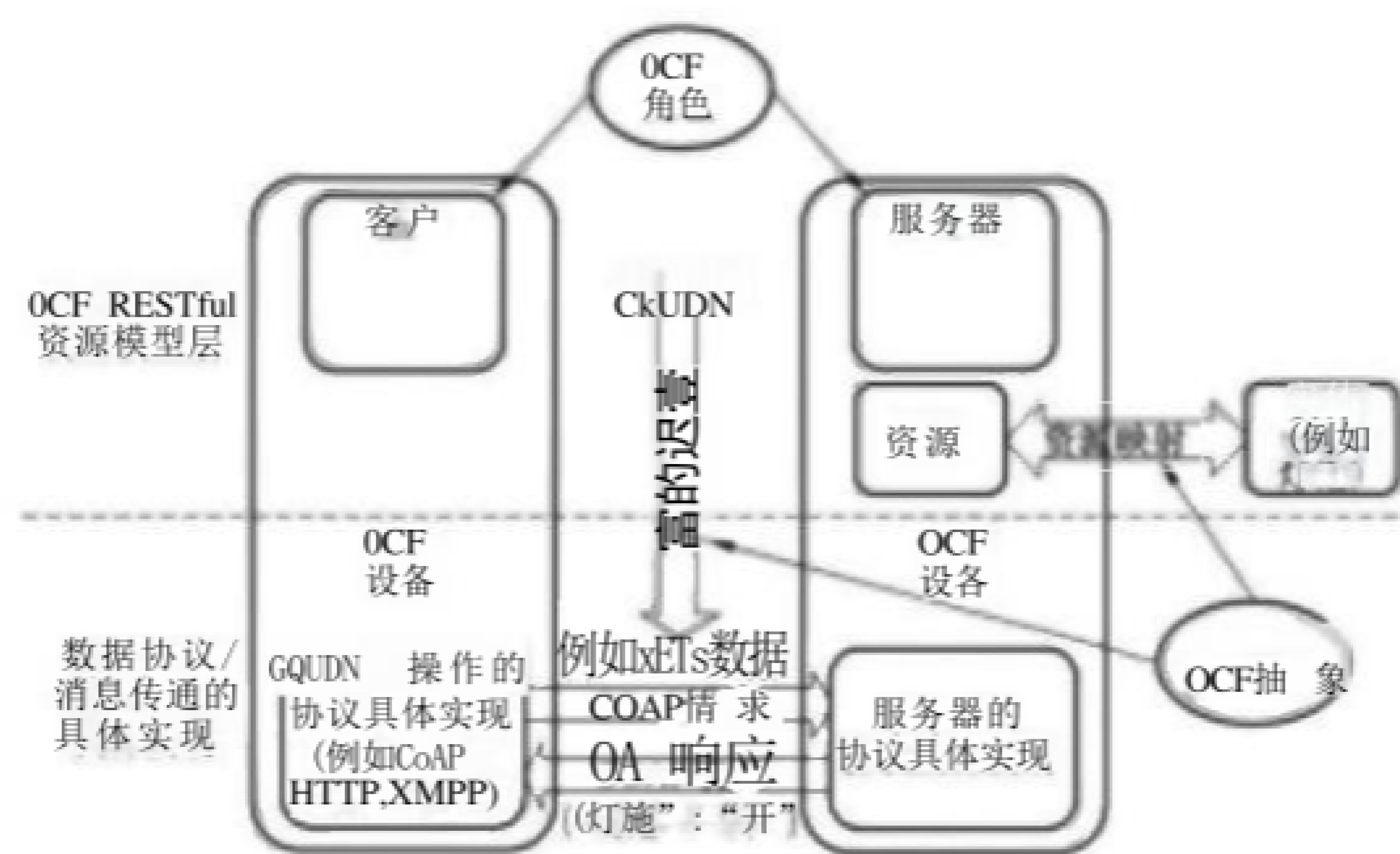
E.2 轻量级物联网(IoT-Lite)

IoT-Lite本体论[37]可以代表物联网(IoT)资源、实体和服务。此外, IoT-Lite本体, 作为SSN 本体的一个轻量级的实例, 可以描述关键的物联网概念, 使异构的物联网平台能够实现传感数据的互操作和发现。

IoT-Lite 本体论将物联网概念描述为三类: 系统、资源和服务。物联网设备可分为三类: 传感设备、执行设备和标签设备, 虽然不仅限于此。IoT-Lite 本体论主要集中在传感方面, 尽管它有一个关于执行的高级概念, 允许未来在这一领域进行任何扩展。服务被描述为一个覆盖范围。这个覆盖范围表示物联网设备所覆盖的二维空间[37]。

E.3 开放连接基础(OCF) 本体论

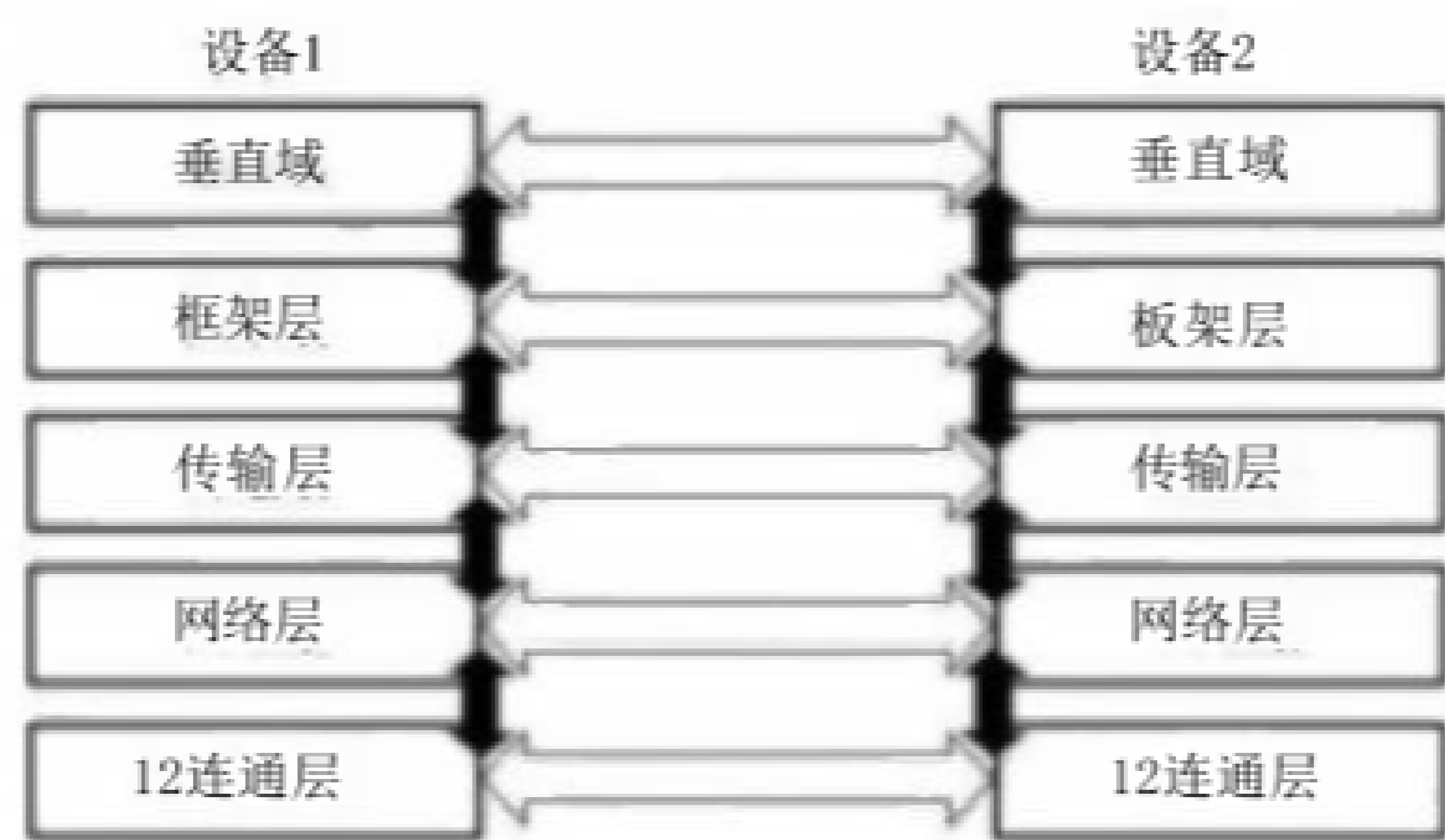
OCF 架构[38]描述了一个客户端—服务器架构, 如图E.1所述。架构的服务器端由物理传感器和执行器组成。OCF 建立在RESTful 概念之上, 即其使用以下一组通用操作来驱动: 新增(CREATE)、检索(RETRIEVE)、更新(UPDATE)、删除(DELETE) 以及通知(NOTIFY), 即CRUDN。



来源: ISO/IEC 30118-1。

图E.1 架构-概念

该架构概念上被分为三个主要方面对关键点进行了整体分类: 资源模型、RESTful 操作, 以及抽象。当两个设备相互通信时, 设备中的每个功能模块都与对方设备中对应的功能块进行交互, 如图E. 2 所示。



来源: ISO/IEC 30118-1。

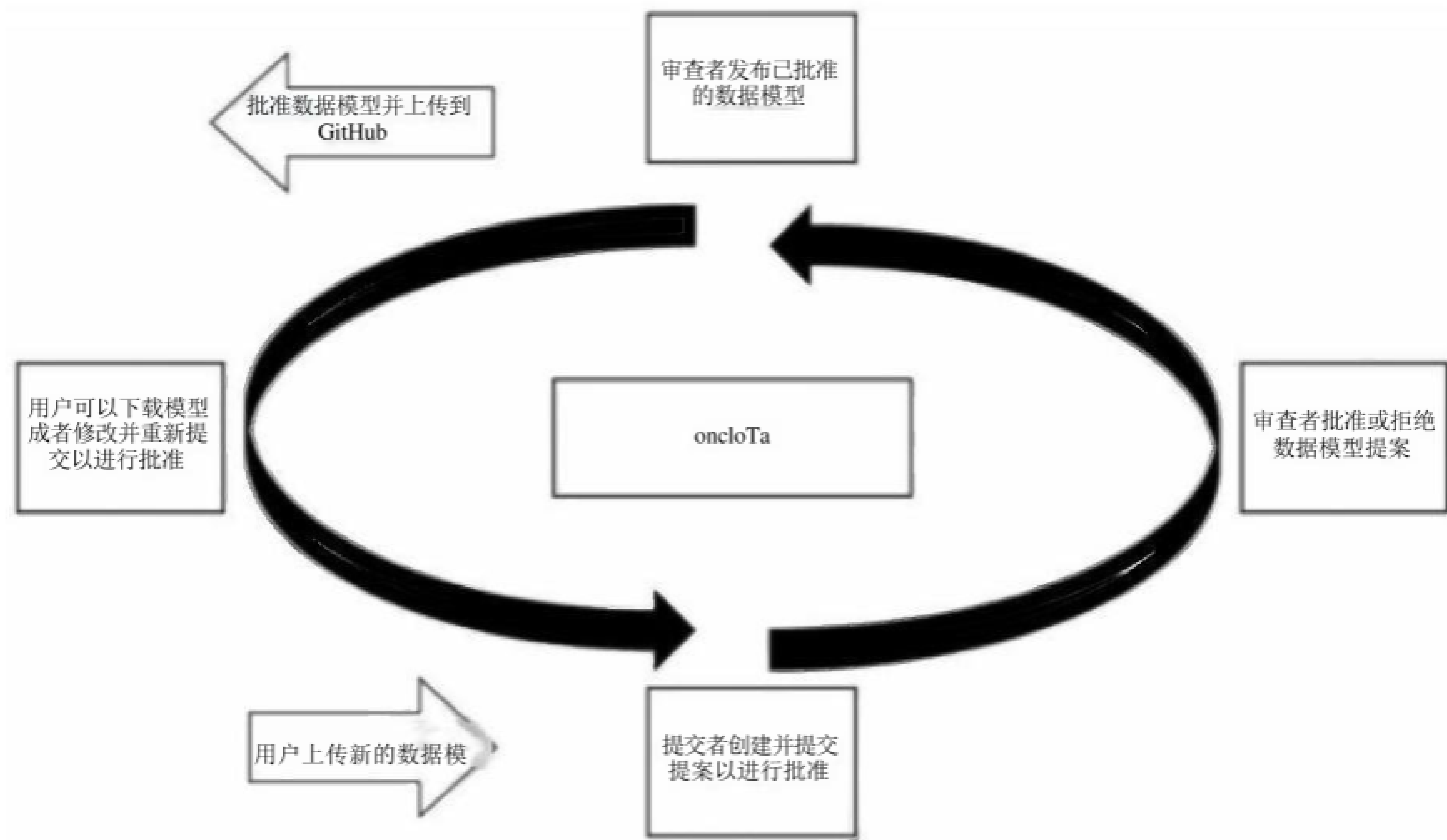
图E.2 通信层模型

垂直域由资源模型描述。资源模型由RESTful API设计语言OpenAPI2.0(swagger) 描述。资源建模建立在一个OCF 设备承载资源这一架构的概念上。每个OCF 设备都有一个设备类型(例如, 冰箱, oic.d.冰箱)。与每个设备类型都关联一个由该设备类型来实现的最小的强制资源集。额外的资源可以有选择地被实现。以下是一些已有的不同类型的资源:

- 描述OCF 架构所需的资源, 包括安全所需的资源;
- 描述传感器的资源;
- 描述执行器的资源;
- 描述传感器和执行器配置相关的资源;
- 描述服务使能概念的资源, 如集合(Collection)、场景(Scenes)和规则(Rules)。

传递值的资源描述的表述方式是与资源是否被建模为传感器或执行器无关的。资源是对设备上的物理实现的虚拟描述。在此意义上, 该架构遵循图E.1。

资源建模和OCF 框架的分离也意味着资源模型的开发与架构的解耦。为了帮助数据模型的开发, 开放互联基金会创建了一个名为oneloTa(www.oneloTa.org) 的工具, 如图E.3 所示。oneloTa 工具是一个集成开发环境(IDE), 支持数据模型的开发, 并有一个数据模型的接受和发布的集成过程。

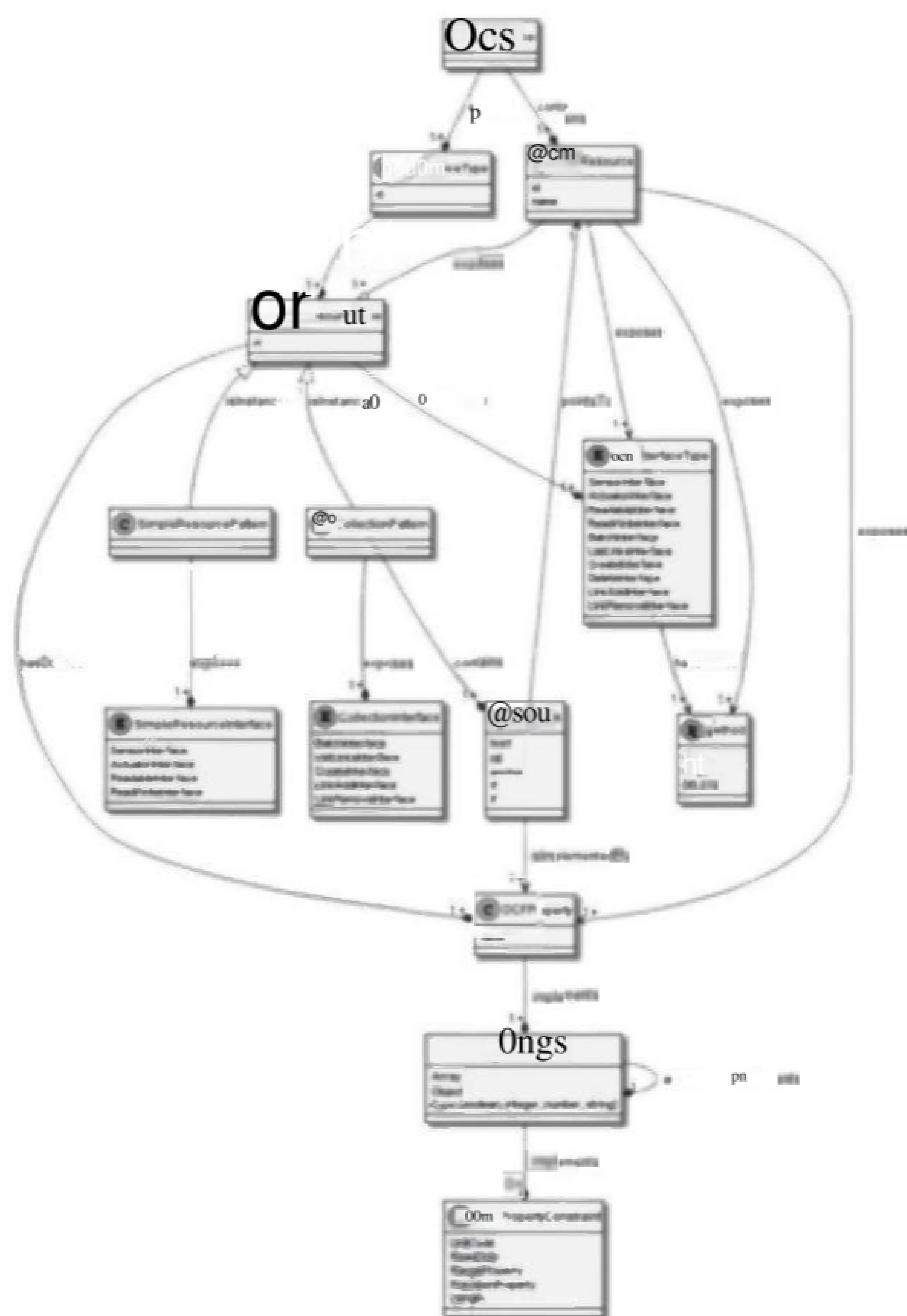


来源: oneloTa用户指南[39]。经授权再版。

图 E.3 oneloTa

oneloTa 工具亦支持衍生模型。衍生模型是一种描述OCF 模型和其他生态系统的模型之间的转译的机制。这种衍生模型的一个例子是OCF 和Zigbee 模型之间的转译。

图E.4 描述了OCF 本体论。所有的OCF 服务器和客户端都是OCFDevic 的实例。本体还描述了简单和复杂的资源，即那些包括CollectionInterface和OCFLink 的实例。CRUDN 操作在方法定义中表示，由OCFResource 暴露，并由OCFInterfaceType 定义。设备的表示由OCFProperty 值提供。



标引序号说明:

C——一个UML类, 代表由资源实现的实体;

E—一个UML枚举，代表一个类型或方法可能采用的一组值。

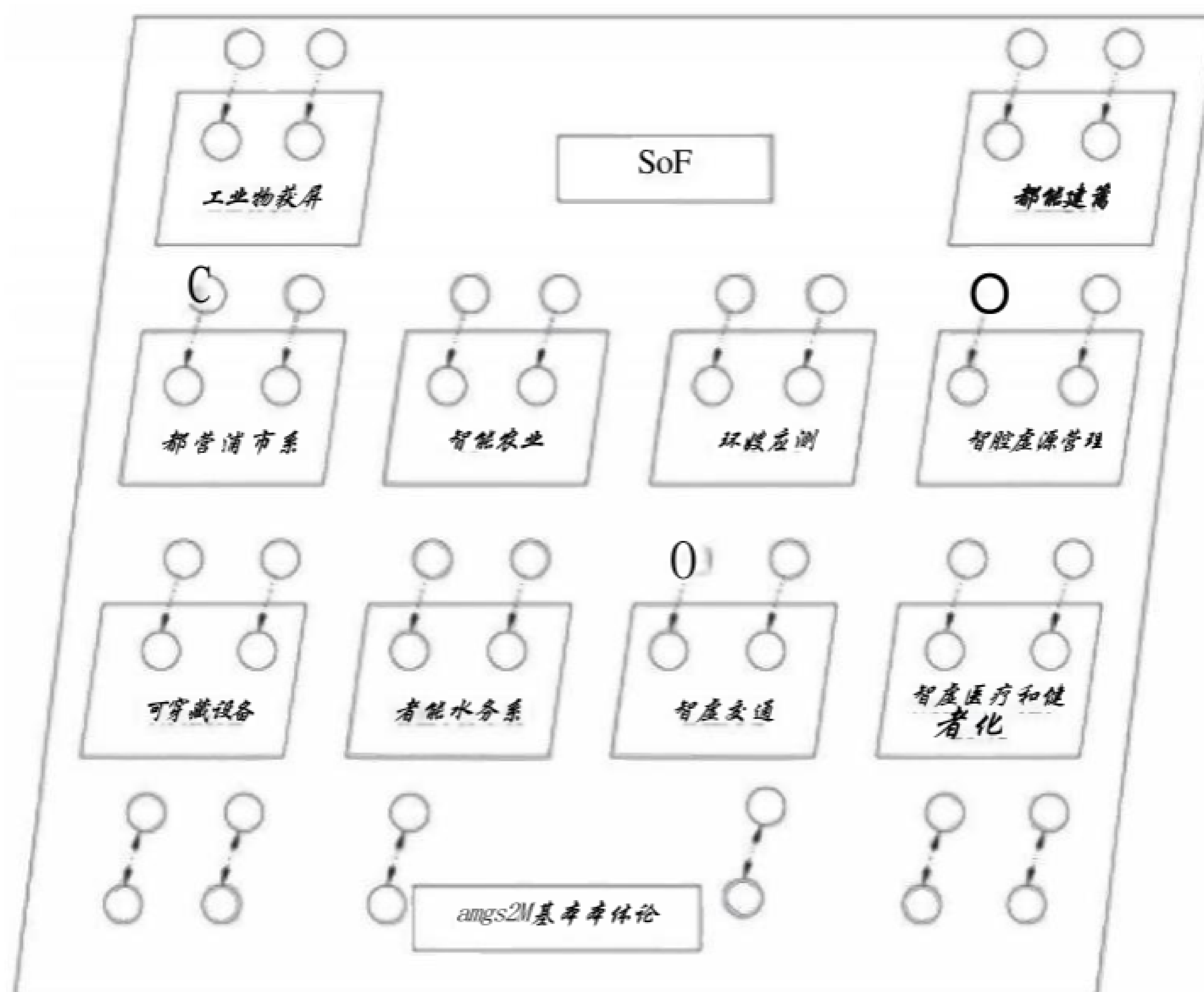
图E.4 OCF本体论

E.4 ETSI智能应用参考本体

智能应用参考本体(Smart Application REference ontology, SAREF) [40]是由 ETSI SmartM2M TC 开发的,旨在实现物联网(IoT)不同提供商和各种活动领域解决方案之间的互操作性,从而促进全球数字市场的发展。

SAREF 是智能应用的参考本体，包含在多个领域中重复使用的概念。为了能够处理这些附加概念并为不同领域提供反映该领域特定需求的适当本体，SAREF 的扩展被创建。图E.5 显示了SAREF 核心模型为基础，在不同领域创建的扩展。包括：能源、环境、建筑、农业、工业与制造、智慧城市、汽车、电子健康和老年人健康、可穿戴设备和水域。

SAREF 本体托管在SAREF 本体门户网站[41]中, 这是一个公共门户网站, 使用户可以浏览SAREF 本体和其扩展的不同版本, 并与之交互, 为本体的持续发展做出贡献。



来源: ETSI TS 103264(V3.1.1)[40],经授权再版.

图E.5 SAREF及其扩展

SAREF 聚焦设备的概念和设备所进行的测量。这些测量可以与关注的不同属性或特征相关联。本体提供了一个可以视为构建块的基本功能列表，以便在单个设备中组合更复杂的功能。当设备连接到网络时，将提供一个用于表示一个或多个功能的网络服务，使它们能够被网络中的其他设备发现、注册和远程控制。图E.6 概述了SAREF 的主要类及其关系。

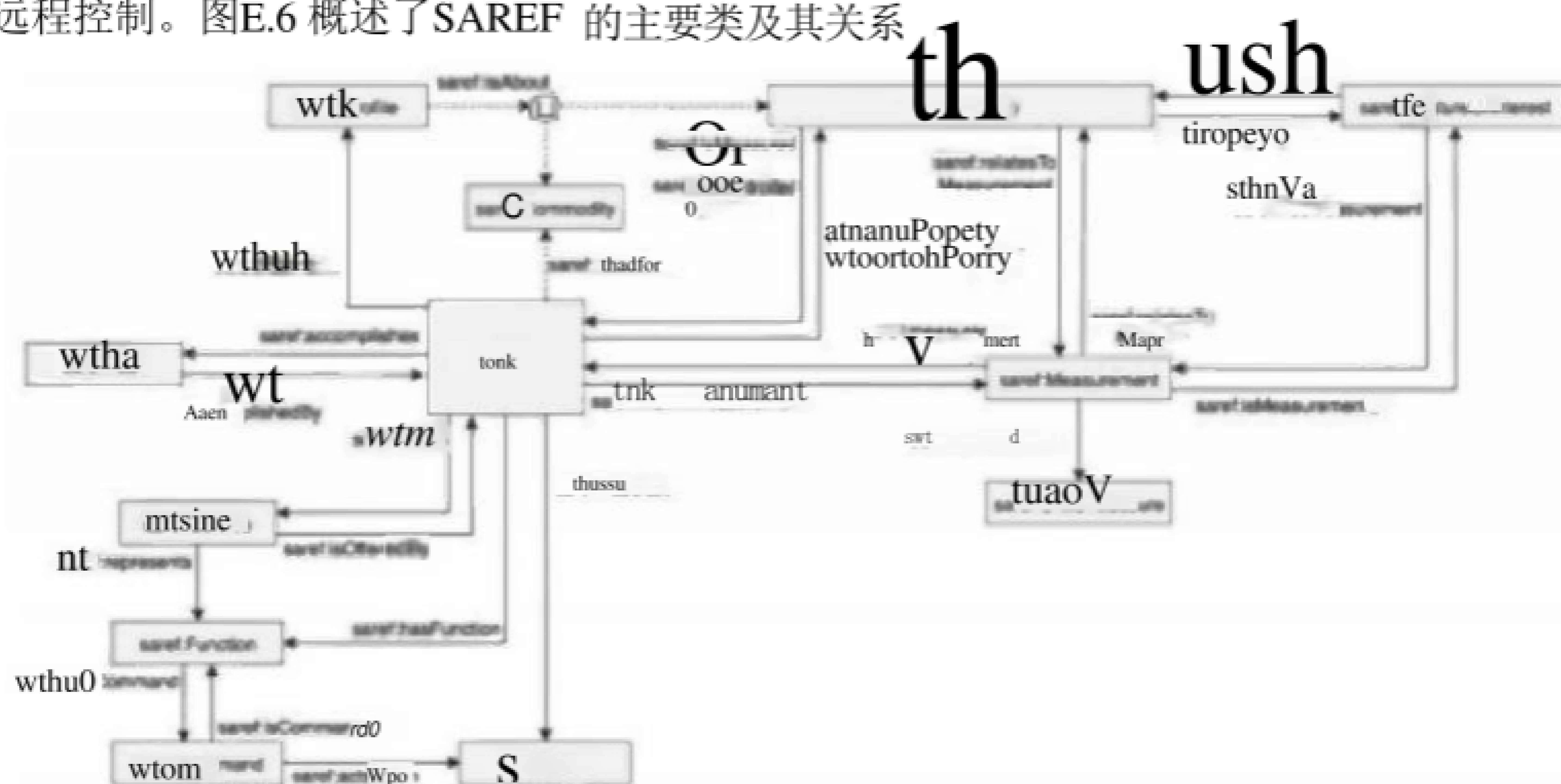


图 E.6 SAREF 库体论总览

E.5 oneM2M基础本体

在oneM2M中,本体及其OWL表示用于提供oneM2M系统与外部系统的语法和语义互操作性。这些外部系统将由本体进行描述。如[42]所述,oneM2M中指定的唯一本体是oneM2M基础本体。oneM2M基础本体是最小的本体(即要求的约定数量最少),以便其他本体可以映射到oneM2M中。

语法互操作性允许与非oneM2M设备进行交互,例如在区域网络中。在这种情况下,一个包含区域网络特定类型的通信参数(操作名称、输入/输出参数、类型和结构等)的本体(表示为OWL文件)被用于将互通代理实体配置到oneM2M的解决方案中。

外部组织 and 公司将贡献自己的本体,这些本体可以映射到oneM2M基础本体(例如通过子类、等价等方式)。然后,他们使用其数据的含义或目的信息来补充oneM2M数据。

可以通过将来自外部组织的本体映射到oneM2M基础本体(可能在协调本体的帮助下),可以实现语义互操作。

E.6 传感器模型语言(SensorML)

传感器模型语言(Sensor Model Language, SensorML) [43]的主要关注点是提供一种强大且语义相关的方法,用于定义与观测值的测量和后处理转换相关的过程和处理组件。这包括传感器和执行器以及应用于测量前和测量后的计算过程。主要目标是实现互操作性,首先是在语法层面,然后是在语义层面(通过使用本体和语义协调),这样传感器和过程可以被机器更好地理解,在复杂的工作流程中自动利用,并在智能传感器网络节点之间轻松共享。传感器语言模型标准是在OGC的传感器网络启用(Sensor Web Enablement,SWE)活动下产生的几个实施标准之一。

E.7 IoT-0

IoT-O[44]是一个核心领域的模块化物联网本体,提供了描述连接设备及其与环境的关系的词汇表。IoT-O可以语义化地描述设备和数据,以使系统了解其环境、演变以及可能给环境带来的变化。这样的描述允许智能代理通过连接的传感器感知其环境,并基于此通过连接的执行器转换其环境。IoT-O由多个模块组成。

为了在广泛的领域中重复使用,IoT-O包含一组关键概念,以便于合并不同领域收集的数据进行水平应用,并允许本体成为可扩展的核心领域本体。IoT-0的关键概念如下。

- 感知模块描述输入数据。该模块的主要类来自SSN:ssn:Sensor和ssn:Observation。ssn:Device及其特征(ssn:OperatingRange,ssn:Deployment等)提供了通用的设备描述。
- 执行模块描述系统如何与物理世界交互。其主要类来自SAN:san:Actuator和san:Actuation。它还重用了不仅限于感知的SSN类,例如ssn:Device。
- 生存周期模块对用来指定系统生存周期和设备使用的状态机进行了建模。该模块的主要类是lifecycle:State和lifecycle:Transition。
- 服务模块表示Web服务接口。其主要类来自MSM:msm:Service和msm:Operation。服务生成和消耗msm:Messages,并且可以用hRest描述RESTful服务。
- 能源模块:IoT-O的能源模块由PowerOnt定义。它提供了poweront:PowerConsumption和一组属性,以表示电器的功耗特性。
- 作为核心领域本体,IoT-O旨在根据特定的应用需求和实际设备和服务进行扩展。受SSN启发,这种设计使IoT-O独立于应用程序。IoT-O既是本体的名称,也是顶级模块的名称。它独立于应用程序提供了物联网领域的概念化,这提供了类和关系以连接底层模块。由于许多概念已在模块中定义,因此IoT-0的核心本体仅限于14个类、18个对象属性和4个数据

属性。IoT-O 的关键类是 `iot-o:IoT.Thing`。它可以是 `ssn:Device` 或 `iot-o:SoftwareAgent`。`Ssn:Devices` 的功耗与 `lifecycle:State` 和 `poweront:PowerConsumption` 相关联。`iot-o:IoT.Thing` 是 `msm:Service` 的提供者，`msm:Operation` 可以在 `ssn:Property` 上具有 `opt-o:ImpactOnProperty`，通过设备将抽象服务与物理世界联系起来。

E.8 IoT本体的统一方法

FIESTA-IoT 本体[45],[46]是一个物联网本体网络，提供了一个遵循最佳实践的统一语义模型。FIESTA-IoT 本体合并了来自各种本体的概念，例如IoT-Lite、M3-lite分类法、W3C SSN。

——机器间测量(machine-to-machine measurement,M3)本体[47]已经在M3-lite(<http://llpurl.org/gliot/vocab/m3-lite#>)内得到完善，并与其他物联网本体(例如W3C SSN)集成。M3 本体引用了30多个传感器、测量值、单位和约10个领域。M3 本体聚焦于W3C SSN本体中描述传感器和观测值以及相关概念的 `ssn:ObservationValue` 概念。M3-lite 分类法是M3 本体的精简版本。M3-lite 包含各种数量种类(通常称为物理和环境现象)的分类、测量单位、不同类型的传感器和不同类型的兴趣领域。

——IoT-Lite 在 E.2 中介绍。

——W3C SSN也在E.1 中进行了介绍。

FIESTA-IoT 本体被分为两个部分：资源描述和产生的观测。所有概念都与资源描述和观测相关。使用这些信息，测试平台可以将其数据注释为符合FIESTA-IoT 的要求，而开发的注释器可以实现FIESTA-IoT 认证。文档提供了资源图和观测图的注释示例作为教程。

M3-lite已经形成五个试验平台，这些试验台希望成为FIESTA IoT提供的联盟的一部分。有关上述测试平台以及已加入FIESTA-IoT 的测试平台的更多信息，请访问：http://fiesta-iot.eu/index.php/fiesta_testbeds。

此外，以上的本体需尽可能遵循语义网络的最佳实践(例如，在本体目录上被引用，复用现有本体)。

参 考 文 献

- [1]ISO/IEC 30141 Internet of Things(IoT)—Reference Architecture
- [2]Martin Bauer et al.Towards semantic interoperability standards based on ontologiesAloTI white paper.June 2019
- [3]ETSITS 103264V2.1.1,SmartM2M;Smart appliances;Reference ontology andOneM2M mapping.2017.https://www.etsi.org/deliver/etsi_ts/103200103299/103264/02.01.0160/ts_103264v020101p.pdf
- [4]ISO/IEC 29100:2011 Information technology—Security techniques—Privacy framework
- [5]Matthias Weise,Maria Poveda—Villalon,Mari Carmen Suarez—Figueroa,Raul GarciaCastro,Jerome Euzenat,Luz Priego,Bruno Fies,Andrea Cavalaro,jan Peters—Anders<leopatra Tsagkari.Ontologies and datasets for energy management systeminteroperability(2014).Research gate
- [6] ETSI TS 103410-4 V1.1.1 SmartM2M:Extension to SAREF:Part 4:Smart CitiesDomain.2019.https://www.etsi.org/deliver/etsi_ts/103400103499/10341004/01.01.0160/ts_10341004v010101p.pdf
- [7]ETSI TS 103410-5 V1.1.1 SmartM2M:Extension to SAREF:Part 5:Industry andManufacturing Domains.2019.https://www.etsi.org/deliver/etsi_ts/103400103499/10341005/01.01.0160/ts_10341005v010101p.pdf
- [8]ETSI TS 103410-6 V1.1.1 SmartM2M;Extension to SAREF;Part 6:Smart Agricultureand Food Chain Domain.2019.https://www.etsi.org/deliver/etsi_ts/103400103499/10341006/01.01.0160/ts_10341006v010101p.pdf
- [9]Linked Open Vocabularies forInternet of Things(LOV4IoT).<http://lov4iot.appspot.com/>
- [10]RDF Primer.W3C Recommendation.https://www.w3.org/TR/rdf_primer/
- [11]RDF Schema.W3C Recommendation.https://www.w3.org/TR/rdf_schemal
- [12]Web Ontology Language(OWL).W3C.<https://www.w3.org/owl/>
- [13] SPARQL 1.1 Query Language.W3CRecommendation<https://www.w3.org/TR/spargl11-query/>
- [14]Semantic Web-XML2000,slide 10.W3C.<https://www.w3.org/2000/Talks/1206-xml2k.tbl/slide10-0.html>
- [15]JSON-LD 11.A JSON-based Serialization for Linked Data.W3C Recommendation16 July 2020.<https://www.w3.org/TR/ison-ld/>
- [16]Semantic Sensor Network Ontology.W3C Recommendation 19 October 2017https://www.w3.org/TR/vocab_ssn/
- [17]Review of Sensor and Observations Ontologies.W3C[https://www.w3.org/2005/Incubator/ssn/wiki/Review of Sensor and Observations Ontologies](https://www.w3.org/2005/Incubator/ssn/wiki/Review_of_Sensor_and_Observations_Ontologies)
- [18]Linked Open Vocabularies(LOV).<https://lov.linkeddata.es/dataset/lov/>
- [19]Friend of a Friend(FOAF):an experimental linked information system.<http://www.foaf-project.org/>
- [20]oWL-S:Semantic Markup for Web Services.W3C Member Submission 22 November2004.<https://www.w3.org/Submission/OWL-S/>
- [21]Amit P.Sheth.Changing Focus on Interoperability in Information Systems:in SystemSyn-

tax, Structure to Semantics, Interoperating Geographic Information Systems, M F Goodchild, M J Egenhofer, R Fegeas and C A Kottman (eds). Kluwer, pp 5-29, 1999

[22] GB/T 22032—2021 系统与软件工程系统生存周期过程 (ISO/IEC/IEEE 15288:2015, IDT)

[23] ISO/IEC 20924:2018, Internet of Things (IoT)—Vocabulary

[24] Paul Murdock et al. Semantic Interoperability for the Web of Things, ResearchGate August 2016. https://www.researchgate.net/publication/307122744Semantic_Interoperability_for_the_Web_of_Things

[25] Martin Bauer et al. Semantic IoT Solutions: A Developer Perspective. June 2019

[26] Internet of Things IoT Semantic Interoperability: Research Challenges, Best Practices, Recommendations and Next Steps. European Research Cluster on the Internet of Things. 2015. http://www.internet-of-things-research.eu/pdf/IERC_Position_Paper_IoT_Semantic_Interoperability_Final.pdf

[27] ISO/IEC TR 22417:2017, Information technology—Internet of Things (IoT)—IoT use cases

[28] Natalya F. Noy, Deborah McGuinness. Ontology development 101: A guide to creating your first ontology, https://protege.stanford.edu/publications/ontology_development/ontology101.pdf

[29] WebProtege user guide. <https://protegewiki.stanford.edu/wiki/WebProtegeUsersGuide>

[30] Jena Framework to build semantic web and linked data applications <https://jena.apache.org/>

[31] Jena inference support. <https://jena.apache.org/documentation/inferencel>

[32] 3rd tutorial on 'semantic web meets internet of things and web of things'. 2017 http://semantic-web-of-things.appspot.com/tutorial_ISWC2017/05_HandsOnSession_%201C2017Tutorial_Amelie_Gyrard_V1.pdf

[33] Mari Carmen Suarez, Asuncion Gomez-Perez, Mariano Fernandez-Lopez, The Neon-Methodology for Ontology Engineering, in Ontology Engineering in a Networked World 2012. Springer Com.

[34] FAQ for Ontologies: Exploring the state of the art with interactive mind maps http://perfect-semanticweb.appspot.com/?p=ontology_sota#div_ontology_methodology_mindmap

[35] Amelie Gyrard, Martin Serrano, Ghislain Ateazing. Semantic Web Methodologies, Best Practices and Ontology Engineering Applied to Internet of Things. IEEE World Forum on Internet of Things (WF-IoT), Milan, Italy, December 14-16, 2015. https://test.insight.centre.org/sites/default/files/publications/paperwfiot2015_bestpracticessemanticwebiot.pdf

[36] vOWL: Visual Notation for OWL Ontologies, <http://purl.org/vowl/spec/>

[37] IoT_Lite Ontology, 2015 W3C Member Submission 26 November <https://www.w3.org/Submission/2015/SUBM-iot-lite-20151126>

[38] ISO/IEC 30118-1: Information technology—Open Connectivity Foundation (OCF) Specification—Part 1: Core specification

[39] OneLoTa User Guide. http://openconnectivity.org/oneiota/OneloTa_User_Guide.pdf

[40] ETSI TS 103264 V3.1.1, Smart M2M; Smart Applications; Reference Ontology and oneM2M Mapping. 2020 https://www.etsi.org/deliver/etsi_ts/103200/103299/103264/03.01.0160/ts103264v030101p.pdf

[41] SAREF ontology portal, <https://saref.etsi.org/>

[42] oneM2M Technical Specifications, TS-0012 oneM2M Base Ontology, version 2.2.2 Release 2, 2018-03-12, <http://www.onem2m.org/technical/published-drafts>

[43] OGC SensorML: Model and XML Encoding Standard, <http://docs.ogc.org/is/12.000r2/12->

000r2.html

[44]Seydoux,Khalil Drira,Nathalie Hemandez,and Thierry Monteil,IoT-O,aCore—DomainIoT Ontology to Represent Connected Devices Networks,20th InternationalConference19-23EKAW2016.Bologna,2016Italy.Novemper<https://www.researchgate.net/publication/309695394> oT-O a Core—Domain IoT Ontology to RepresentConnected Devices Networks

[45]FIESTA-IoT:Federated interoperableSemantic IoT/Cloud Testbeds and Applications.H2020 EUproject,2015-2018.<https://cordis.europa.eu/project/id/643943>

[46] Agarwal et al.Unified IoT Ontology to Enable Interoperability and Federation of TestbedsThird IEEE World Forum on Internet of Things(WF-IoT),2016.

[47]Amelie Gyrard.Designing Cross—Domain Semantic Web of Things Applications.Ph.DThesis,Telecom ParisTech,2015.http://www.eurecom.fr/fr/publication/4553/download/cm_publi-4553.pdf

[48]ISO/IEC 11179-3:2013 Information technology-Metadata registries (MDR)—Part 3:Registry metamodel and basic attributes

[49]ISO/IEC 19763-3:2010 Information technology—Metamodel framework for interoperability (MFD—Part 3:Metamodel for ontology registration

[50]ISO/IEC19941:2017 Informationtechnology-Cloudcomputing Interoperabilityandportability

[51]ISO/IEC 21823-1:2019 Internet of Things(IoT)—Interoperability for IoT systems—Part 1: Framework

