

OWL DL 的 UML 建模方法

钟凌燕, 陈 岗

(上海财经大学信息管理与工程学院, 上海 200433)

摘要:描述了一种 OWL DL 的 UML 建模方法, 针对 OWL DL 定义了一种 UML 概要文件, 描述了各种元素及元素之间的关系。实验表明, 只要遵循该概要文件, 就可以利用任何 UML 工具来进行本体论的开发, 降低大型本体论建立与维护的复杂性。

关键词:语义网; 本体论; OWL; UML 概要文件

Modeling Method of UML Based on OWL DL

ZHONG Lingyan, CHEN Gang

(School of Information & Engineering, Shanghai University of Finance & Economics, Shanghai 200433)

【Abstract】 This paper describes a UML-based notation for OWL DL. It proposes a UML profile for OWL DL, which can be used to model the elements and relationships, such as class and relation. The experiment shows, given the profile, it uses any UML tools to develop ontology, which reduces the complexity of large-scale ontology construction and maintenance.

【Key words】 Semantic Web; Ontology; OWL; UML outline document

语义网和基于XML的语言是Web开发的主要方向。领域本体论又是语义Web应用程序的最重要的组成部分。OWL语言是W3C推荐的领域表示语言的标准, 正被越来越多的科研人员及工业团体所关注。当前用于领域创建与维护的技术建立在传统的人工智能技术的基础上, 它更适合实验人员, 而难以被广大的软件工程技术人士所使用^[1]。

UML是软件工程技术中一种标准的建模语言, 被广大的软件工程技术人士所接受与使用。更重要的是OWL是一种文本语言, 其体系是开放的。在表示大规模的领域本体论时, 通常要涉及到大量的类、属性和描述, 这些元素之间的联系很难直观地表达出来, 这给领域论的建立与维护带来了不便。

1 Web 领域语言 OWL

Web 领域语言 OWL 是一种定义和实例化“Web 本体”的语言。本体包含了类、属性和实例的描述。通过 OWL 形式语义说明, 可以通过对本体进行推理, 获取语义中蕴涵的事实。

OWL提供了3种表达能力递增的子语言, 以分别用于特定的实现者和用户^[2]: (1)OWL Lite用于提供给那些只需要一个分类层次和简单约束的用户; (2)OWL DL支持那些需要最强表达能力的推理系统的用户, 且这个推理系统能够保证计算的安全性和可判定性。它包括了OWL语言的所有成分; (3)OWL Full支持那些需要尽管没有可计算性保证, 但有最强的表达能力和完全的RDF语法的用户。

本文关于 OWL 的 UML 模型的讨论仅限于 OWL DL。

2 OWL 的 UML 表示方法

UML无法捕获XML模式中的所有丰富内容。而这一点对于OWL来说也是一样的。UML概要文件提供了一种通用的扩展机制, 用于构建特殊领域中的UML模型^[3]。UML概要文件有构造型(stereotype)、标记值(tagged value)和约束(constraint)3个关键项。

构造型是对UML的词汇的扩展, 允许创建与已有的构造

块相似而针对特定问题的构造块。新构造块有自己的具体特性(各构造型可以提供自己的标记值集)、语义(各构造型可以提供自己的约束)和表示法(各构造型可以提供自己的图标)。标记值(tagged value)是对UML元素的特性的扩展, 它允许在元素的规格说明中创建新的信息。在图形上, 把标记值表示成用花括号括起来的字符串, 并把它放在其他的元素名之下。约束(constraint)则是对UML元素的语义的扩展, 它允许增加新的规则或修改已有的规则^[4]。

本文定义了一套用于表达 OWL 中各种元素及元素间关系(如名称空间、类、个体等)的 UML 构造型和标记值。文献[5]创建了一个关于酒和食物的本体, 具有很强的代表性, 以下关于 OWL 语言的 UML 模型的示意图除个别外, 均以 Wine 本体和 Food 本体为例。

2.1 名称空间、本体头部

名称空间(namespace)是 XML 模式中一个非常重要的概念。它们表示将业务概念隔离到桶中。OWL 采用名称空间隔离不同的领域定义。在 UML 图中, 可以通过将 UML 模块保存在不同的包中, 来表示名称空间。定义构造型为“owl:namespace”的包来表示名称空间。领域论也用包来表示, 但其构造型定义为“owl:ontology”。在 OWL 中, 领域论可以引用外部的名称空间。这种引用关系可以用包与包之间的依赖关系来表示, 其构造型为 owl:refnamespace。一个领域论的定义中可以通过 owl:imports 来导入另一个领域论中的定义。这种领域论之间的导入关系也可以通过包与包之间的依赖关系来表示, 为了与前者相区分, 其构造型定义为 owl:imports。

名称空间的名称直接用包的名称来表示。为了在 UML 模型中能表示出名称空间所对应的 URI。在定义名称空间构

作者简介:钟凌燕(1973 -), 男, 讲师、博士, 主研方向: Agent 与软件工程, 工作流技术, 语义网与电子商务; 陈 岗, 副教授

收稿日期:2006-04-11 **E-mail:** lyzgong@shufe.edu.cn

造型 owl:defaultNameSpace(表示缺省的名称空间)和 owl:namespace 时,分别为它们定义一个 URI 标记值(即 UML 中的标记值),用以指示名称空间的 URI 值。

在定义本体论的构造型 owl:ontology 时,为它定义了以下标记值:comment、priorVersion、label,分别用来表示本体论的注释信息、版本信息、以及标签信息。

OWL 文档是基于 XML 的,遵循 XML 的语法规则,它可以通过 ENTITY 元素来定义文字的缩写,如<!DOCTYPE-rdf:RDF [<!ENTITY wine "http://www.w3.org/TR/2003/PR-owl-guide-20031209/wine#">]>。该定义表示 OWL 描述中可用"wine"来代替"http://www.w3.org/TR/2003/PR-owl-guide-20031209/wine#",这给编写 OWL 代码带来了许多方便。

由于在一个 XML 文档中,ENTITY 的数目事先是不确定的,因此采用在构造型中定义特性来表示 ENTITY 的方式就不太合适。用构造型为 owl:entity 的 UML 类来表示实体集的定义,其类名可以任意定义。在该类中,其每一个成员代表一个实体定义。成员的名字(如 wine)表示实体的名称,成员的类型固定为 string,成员的初始值表示实体对应的 URI 地址。UML 表示如图 1 所示。

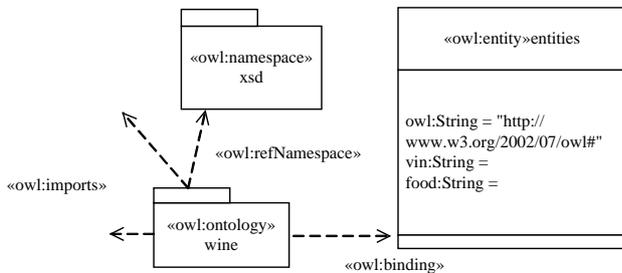


图 1 本体论、名称空间、实体、本体论导入的 UML 表示

2.2 类与个体

OWL 中的类并不是真正意义上的类,它描述了一些个体的共有特征,这与传统面向对象程序设计上的类是有本质上的不同的。其实质上是一些具有共有特征的个体的集合。尽管 UML 中的类与 OWL 中的类从理解上来说存在一定的差异,但仍然可以用 UML 中的类来表示 OWL 中的类^[6]。

OWL 中的类可以直接用 UML 中的 class 来表示,尽管这 2 个概念在理解上有一定的差异。但需要注意的是,OWL 中的类是没有操作的,而 UML 中的类是有操作的,在建模的时候,UML 类中的操作部分应该忽略掉。类的继承可以通过 UML 中类的泛化来实现。为了给 UML 的类指定一定的语义,以利于今后的代码生成。

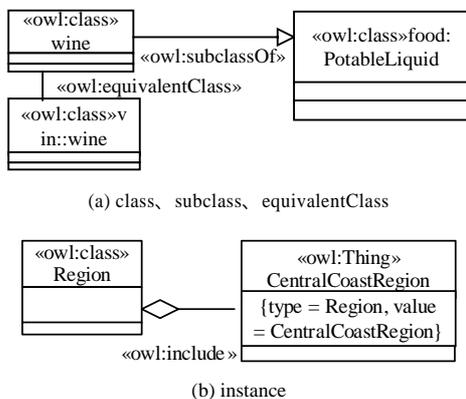


图 2 类、子类、等价类、个体的 UML 表示

构造型 owl:class 表示该类为一个 OWL 类,带构造型 owl:subclassOf 的泛化关系用来表示类与类之间的继承关系,带构造型 owl:equivalentClass 的关联用来表示类与类之间的等价关系。如图 2(a)所示,Wine 继承 food 本体论中的 PotableLiquid 类,wine 与 vin::wine 是等价的。OWL 中的类实质上是一些具有共同特征的个体的集合,因此类与个体之间的关系为一种聚合关系。

另外,OWL 中任何一个类都是继承类 Thing,所以采用构造型为 owl:Thing 的类来描述个体,并为其定义 2 个标记值:type 和 value。Type 用来指示个体所属的类名,value 用来指示个体的取值。如图 2(b),CentralCoastRegion 为 Region 的一个个体,它与 Region 类之间看成一种特定的聚合关系,其构造型为 owl:include。

OWL 也能表达类的运算关系,类与类之间可以通过交(intersection)、并(union)、补(complement)这 3 种运算来约束其他类。其 UML 表示方法如图 3 所示,WhiteBurgundy 是 Burgundy 和 WhiteWiner 的交集,用构造型 owl:intersection 来表示;WineDescriptor 是 Winetaste 和 WineColor 的并集,用构造型 owl:union 来表示;而 NonConsumableThing 则与 ConsumableThing 成互补关系,在二者之间用带构造型 owl:complement 的依赖关系来表示。

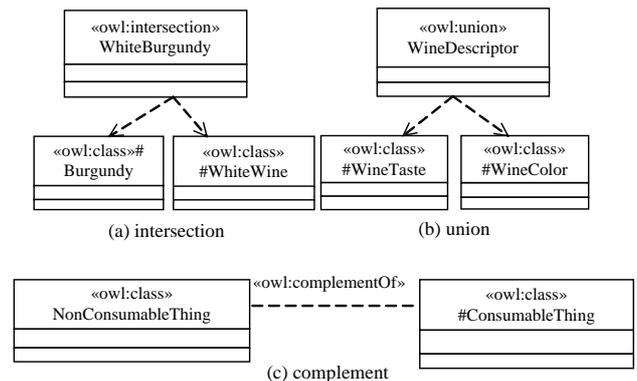
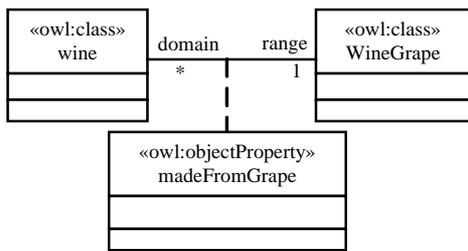


图 3 类运算的 UML 表示

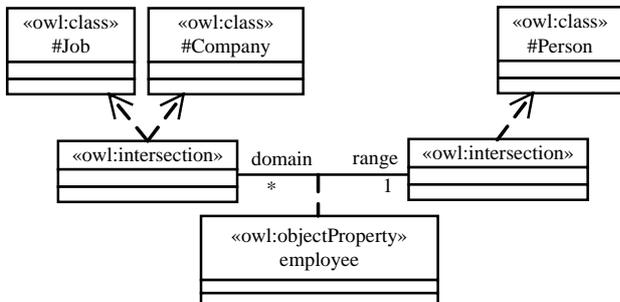
2.3 属性

OWL 中的属性分为数据类型属性与对象属性 2 种,分别用构造型 owl:datatypeProperty 与构造型 owl:objectProperty 来表示。OWL 使用 XML Schema 内嵌数据类型中的大部分,如 xsd:string, xsd:decimal 等。在 UML 模型中,这些数据类型可以引用。而自定义的数据类型属性的 UML 表示则与个体的 UML 表示方法相同,为构造型 owl:datatypeProperty 定义标记值 domain 及 range,分别表示数据类型属性的值域与定义域。对象属性的 UML 如图 4 表示。

对象属性定义了从定义域到值域的一种二元关系,在 UML 中可被理解成一种关联关系,其关联于某个对象属性上。为此用 UML 中的关联类来表示 OWL 中的对象属性。另外,在 OWL 中不含显式操作符的元素序列代表一个隐式的合取(conjunction),为同一属性声明多个定义域(domain)表明该属性的定义域是所有这些类的交集(多个值域声明也类似这样)。为了能够在 UML 模型中表达出这层含义,不妨将定义域看成是一或多个类通过交运算所产生的一个匿名类。值域可以采用同样的方法来处理。图 4(a)表示一个单定义域单值域的对象属性的 UML 表示方法,图 4(b)表示了一个多定义域多值域的对象属性的 UML 表示方法。



(a) single range and single domain



(b) multiple ranges or multiple domains

图 4 对象属性的 UML 表示

此外,OWL 提供了一种强有力的机制以增强对一个属性的推理,这是通过为属性定义一些特性来实现的。OWL 中的属性有以下几种特性:传递性,对称性,函数性,反函数性。前 3 种特性可分别用构造型 owl:transitiveProperty、owl:symmetricProperty、owl:functionalProperty 来表示,可与其他构造型(如 owl:ObjectProperty)同时作用于一个对象属性。而反函数性表示的是 2 个对象属性之间的一种二元关系,可用带构造型 owl:inverseOf 的双向关联来表示。

2.4 属性限制

OWL 除了可以指定属性外,还能够通过“属性限制”来进一步限制属性的值域。属性限制其本质是通过给某个属性的定义域再加上一个约束条件(如 hasMaker = “#Winery”)。OWL 中的属性限制有以下几种: someValuesFrom、allValuesFrom、基数限制、haveValue 限制。用 UML 中的受限关联来表示属性限制。图 5(a)表示了一种 someValuesFrom 类型的属性限制,至少有一个 Wine 类个体的 madeFromGrape 属性是指向 ChardonnayGrape 的。allValuesFrom 可采用类似的方法。图 5(b)表示的是一种基数限制,Vintage 恰好有一个 hasVintageYear。图 5(c)表示的是一种 hasValue 限制,Burgundy 至少有一个 hasSugar 等于 Dry。

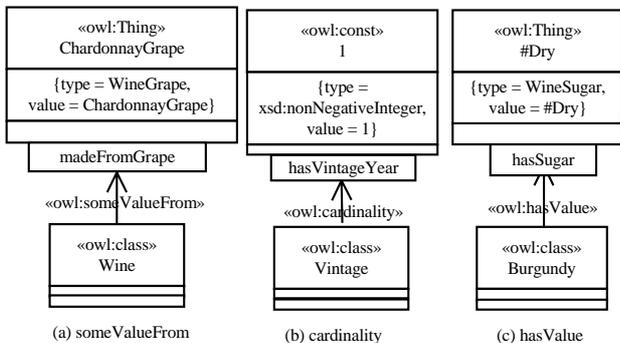
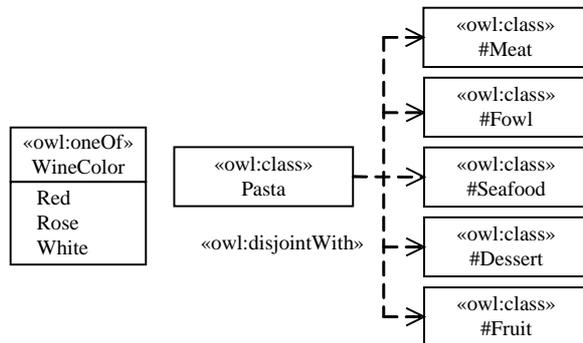


图 5 属性限制的 UML 表示

2.5 其他

OWL 提供了一种通过直接枚举类的成员的方法,来描述类。这是通过 oneOf 结构来完成的,采用构造型为 owl:oneOf 的 UML 枚举类型来表示。图 6(a)表示 WineColor 是一个枚举类,它的成员是 White、Rose 和 Red 这 3 个个体。



(a) oneOf

(b) disjointWith

图 6 枚举、不相交类的 UML 表示

OWL 中,owl:disjointWith 构造可以表达一组类是不相交的,它保证了属于某一个类的个体不能同时又是另一个指定类的实例。图 6(b)表示 Pasta 与其他几个类是不相交的(Meat、Fowl、Seafood、Dessert、Fruit),但并不保证那几个类之间是不相交的。

3 结论

领域本体论是语义 Web 程序不可或缺的一部分,从数据交换程序到语义 Web 服务,直观而简洁的本体论建模方法已变得必不可少。UML 是一种经过试验和测试的用于面向对象系统的标准建模语言,正被广大的开发人员、业务分析人员所使用。本文定义的用于表达 OWL 语言的 UML 概要文件,能够准确直观地建立表示元素之间关系的 UML 模型。然而,目前对 OWL 的 UML 表示模型的完整性、正确性缺乏一定的校验手段,只能依靠建模人员的经验进行检查。另外,从 OWL 的 UML 模型到 OWL 语言的自动生成以及从 OWL 语言到 OWL 的 UML 模型的反向工程也是 OWL 建模研究的重要内容。

参考文献

- 1 Djuric D, Gasevic D, Viadan D. Ontology Modeling and MDA[J]. Journal of Object Technology, 2005, 4(1).
- 2 McGuinness D L, Harmelen F V. OWL Web Ontology Language Overview Technical Report[EB/OL]. <http://www.w3.org/TR/owl-features/>, 2003.
- 3 Malik A. 使用 UML 设计 XML 模式[EB/OL]. <http://www-128.ibm.com/developerworks/cn/xml/x-umlschem/index.html>, 2003.
- 4 Booch G, Rumbaugh J, Jacobson J. 邵维忠译. UML 用户指南[M]. 北京: 机械工业出版社, 2001-06.
- 5 W3C. OWL Web Ontology Language Guide Recommendation. World Wide Web Consortium[EB/OL]. <http://www.w3.org/TR/2004/REC-owl-guide-20040210/>, 2004.
- 6 Brockmans S, Volz R, Eberhart A, et al. Visual Modeling of OWL DL Ontologies Using UML[EB/OL]. <http://www.aifb.uni-ka-rlsruhe.de/WBS/sbr/publications/iswc04%20sbr.pdf>, 2004.