特定领域本体的构造方法研究

杨明华 钱乐秋 赵文耘 彭鑫
（复旦大学计算机科学与工程系软件工程实验室 上海市智能信息处理重点实验室 上海 200433）

摘 要：构件描述是构件检索、组装的前提和基础，通过分析现有的构件描述方案，如基于刻面描述方法的局限性和缺点：没有语义信息，无法进行逻辑推理、关联，没有动态性，本文提出了一种领域本体构造方法 OntoMerge，通过该方法构造得到领域本体，再利用领域本体对构件进行本体描述。基于本体的、面向服务的构件描述，具有语义推理功能，能够面向服务的，用户需要什么样的服务，就能通过形式化语义推理功能，动态检索到提供该服务的构件，以及相关服务的构件。这样就能极大提高构件检索的查全率和查准率，并为构件入库、检索、组装等工作奠定基础。

关键词：构件描述 领域本体 OWL
文章编号： 文献标识码：A 中图分类号：TP302

Research On Merging Ontology to Generate Domain-Specific Ontology

Yang Ming-Hua QIAN Le-Qiu Zhao Wen-Yun
(*Department of Computer Science and Engineering, Shanghai Key Laboratory of Intelligent Information Processing,Fudan University, Shanghai* 200433)

Abstract： Component description is the precondition and base for Component Retrieving and Assembling.By analysing the current solutions of component description,such as facet-based method,we find that they have shortage in lacking semantic information and thus could't perform logic reasoning and association , not to mention any dynamic feature s.Domain-specific ontologies are greatly useful in knowledge acquisition, sharing and analysis. In this paper,a domain-specific ontology generation method is presented. The ontology is represented in a set of well-defined categories, and each concept is viewed as an instance of certain category. The authors also introduce some concept and algorithm to illustrate the process of merging ontologies.

Key words：Component Description, Domain-Specific Ontology,OWL

1 引言

软件构件的描述和检索是实现软件复用的关键技术，对解决软件复用、降低软件成本和提高软件质量具有重要的意义。随着构件技术的不断发展，基于构件的软件开发 (Component Based Software Development CBSD)越来越得到人们的关注，为了复用构件，必须提供有效的构件信息描述与检索，构件检索技术在基于构件的软件开发中起着举足轻重的作用；构件库的不断充实和完善，构件的数量急剧增加，对构件信息检索提出了更高的要求。开发者要在数量庞大的构件库中快速找到合适的可用的构件，就日益成为一个亟待解决的课题[1,2]。

随着面向服务软件体系结构（SOA）和语义网研究及应用的深入，可复用构件正逐步向面向服务的构件（the Service-Oriented Component）迁移[3]。相对于传统构件模型，面向服务构件具有基于 Services 的功能模块和耦合方式、更好地对语义关联推理的支持、主动柔性的接口、状态位置转移、动态规约以及运行时自动组装等特点[2]。因此，对其动态行为语义的形式化描述是问题研究的关键。

传统构件描述方法如 3C 模型和基于刻面的 REBOOT 模型[4]在支持对面向服务构件动态语义描述方面明显不够。这就需要从易于与语义网集成的、具备较强关联推理能力的角度，寻找更为合适的工具描述其行为语义，从而在更高层次上理解服务构件所提供的语义、有效地支持构件发现、适配、验证和组装。

基金项目：国家自然科学基金项目（项目号：60473062），国家 863 项目（2004AA1Z2330），上海市科委攻关项目（04D215022）
作者简介：杨明华（1979-），男，硕士研究生，主要研究方向：软件工程。钱乐秋（1942-），男，教授、博士生导师，主要研究方向：软件工程，软件复用，构件技术。赵文耘（1964-），男，教授、博士生导师，主要研究方向：软件工程；彭鑫，男，博士研究生，主要研究方向为软件工程

基于刻面的描述方法中，一个构件可以用多个刻面以及每个刻面中的多个术语来刻画，不同的刻面从不同的角度对构件进行描述。这些特征使刻面方法能够从多个角度、多个方面对构件作出更为全面的描述，在应用中取得了良好的效果[4]。但是刻面描述方法重视构件的静态特征描述而没有提供对构件动态行为和服务的描述机制，因此对于检索的查全率和查准率都会带来不可忽视的影响。

从以上论述中可以看出，目前各种描述手段虽能较好地描述构件的静态特征，但对于动态行为和服务语义尚无有效的形式化描述机制[2]。而这正是影响到检索质量(查全率、查准率、性能)的重要原因之一。本文研究基于本体技术、面向服务的构件描述方法，以改进构件检索性能为出发点，提出一种不同于上述各种技术的新描述技术 即基于本体的构件描述技术。通过领域本体构建算法，构造出领域本体，然后通过实验验证该描述方案的可行性。

2 OWL对本体语义的支持

首先简单介绍本体 web 语言 OWL。 Web Ontology Language (OWL ver1.1)[7] 是逐步从 RDF、RDFS、DAML+OIL 演化来的最新版本描述语言（2004 年 2 月由 W3C.org 发布），OWL沿袭 RDF的基本事实陈述方式以及 RDFS的类和属性分层结构，并以描述逻辑为基础理论而构建语言系统。具体而言，OWL 语言提供了三种表示能力不同的子语言,来分别满足不同组织团体的语言实现者和使用者：OWL Lite, OWL DL 和 OWL Full。OWL Lite 是 OWL DL的子集，这二者都使用基于框架语言的抽象句法，同时也可映射为 RDF三元组；而其语言以及形式语义则基于描述逻辑的定义，使得本体间的推理问题能够归约为描述逻辑知识库的可满足性问题。OWL Lite, OWL DL以及 OWL FULL作为 OWL提出的三种子语言，其表达能力依次增强，且推理能力逐个向下兼容。不过前两者基于比较抽象的框架式句法，采用经典逻辑学中的解释，即：(1)要求个体、类、属性是三个不交的集合；(2)所有的个体视为资源，而将类直接视为资源的集合，属性则直接视为<资源 资源>的集合；(3)禁止递归的出现。而 OWL FULL则可视为不受限制的 RDFS的扩展，句法仍使用 RDF三元组。OWL 用类和属性描述了领域的结构，类可以用名称（URIs）、表态式（expressions）以及建立类表达式的下列构造器（constructor）组成：owl: intersectionOf, owl: -unionOf, owl: complementOf, owl: one of, owl: allValues-From, owl: hasValue, owl: someValuesFrom.其次，OWL 进行交换的语法是 RDF/XML，它具有与 RDF 和 RDF Schema 最大的兼容性。因为语义网下的服务构件，具有与生俱来的分布特性，所以采取“开放世界”（Open World）设计思想的 OWL，提供了对服务构件最好的语义描述。对服务构件动态行为正规的语法和语义描述，保证了其能被智能 Agent 无歧义地解释和处理。另外，OWL 对建立本体提供的多本体共享、演化、互操作、一致性检验、表述性与可扩展性平衡、易用、与原有大部分描述标准兼容、国际化等特性支持，也使面向服务构件动态语义描述更为完备和强大。

3领域本体及其构造的形式化表示

领域本体[8]是对特定领域内概念及概念间关系的精确描述

定义1 领域本体可以用五元组表示

$$O=(C, \leq_C, R, \leq_R)$$

其构成包括：

概念集C，它的元素被称作概念标识符(concept identifiers)；

关系集R：$C_1\times C_2\times\ldots\times C_n$，其中的元素被称作关系标识符(relation identifiers)；

概念集C的偏序集$\leq_C$，表示概念的层次（concept hierarchy 或taxonomy）；

函数$\leq_R$: $C_1\times C_2\times\ldots\times C_{n-1}\rightarrow C_n$，一类特殊的关系。

关系集R的偏序集$\leq_R$，表示关系的层次。

定义2. 对于两个概念C_i和C_j，在领域本体中，如果C_i被定义为C_j的“EquivalentClass”，则称概念C_i和概念C_j语义相等，记为$C_i\equiv C_j$；如果Cj被定义为Ci的”SubClassOf”,则称概念C_i语义包含C_j,记为$C_i\supseteq C_j$。

定义3. 对于两个概念集合SC_i和SC_j,如果对SC_j中的任一概念C_j,在SC_i中都存在一个概念C_i,满足$C_i\equiv C_j$或$C_i\supseteq C_j$,则称SC_i语义包含SC_j,记为$SC_i\supseteq SC_j$；如果有$SC_i\supseteq SC_j$且$SC_j\supseteq SC_i$,则$SC_i\equiv SC_j$,此时称SC_i和SC_j语义相等。

4 领域本体的溶合 (OntoMerge)方法及过程

领域本体的构建是一项复杂的系统工程需要众多领域专家的参与和大量的时间投入，通过考察本体间语

义信息相似度、关联程度等，构造本体溶合算法来生成领域本体。

4.1 本体溶合是指通过组合两个或两个以上的现有本体，构成新本体，参与溶合操作的可能是通用本体，或者是领域本体；溶合过程与简单的集合论中的合并运算、交集运算有本质的区别。溶合是利用两个现存本体生成新本体的过程。两个领域本体溶合的情况可以表示为图 1所示，进行溶合操作的两个初始领域本体依然被保留。

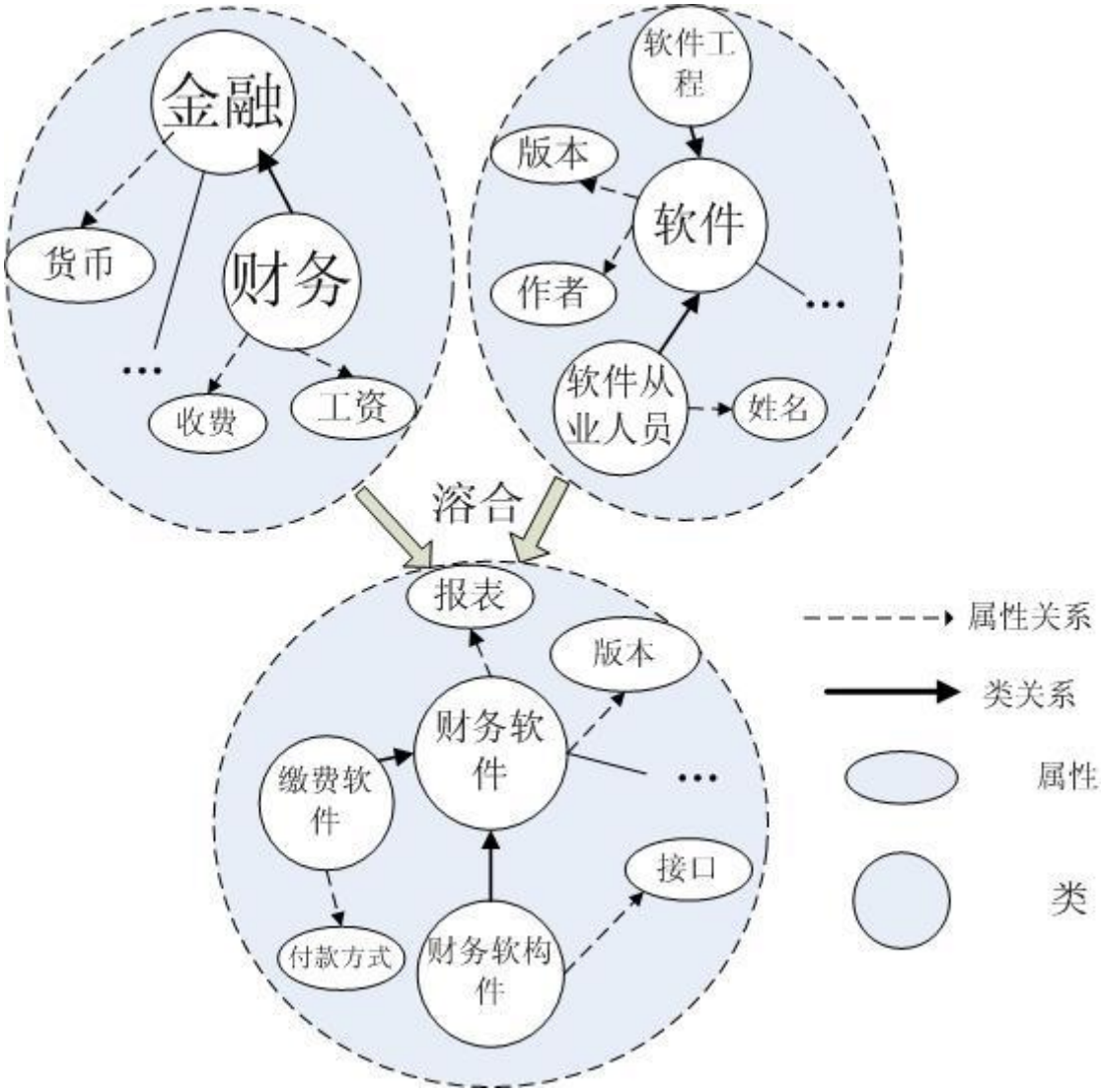

图 1 领域本体溶合运算示意图

上图是从现有的金融领域本体和软件本体溶合生成财务软件本体的过程的简单示意图。参与溶合运算的是两个领域本体：金融本体和软件本体。溶合运算生成了财务软件这个新的领域本体。新的领域本体中，财务软件这个类并不是通过多重继承金融本体中的财务类和软件本体中的软件类得到。金融本体中财务类具有的某些操作和属性，财务软件类可能没有，软件类的某些操作和属性 财务软件类也可能没有，此处我们认为财务软件类不但含有财务类的某些属性和操作，也含有软件类的某些属性和操作，但不是完全继承下来，也可能重载了 继承 下来的属性和操作；而且，财务软件类可能自己新增加一些属性和操作，也可能某个属性继承自金融本体或软件本体中的某个类，但是值域发生变化了。对于属性集也是如此，可能新产生的领域本体中，如上图，财务软件类的版本属性是直接取自参与运算的软件本体的版本属性；缴费软件类的付款方式则是从参与运算的本体中货币、收费等属性综合演化得到；财物软构件的接口属性则是新添加的，在原来的本体库中是不存在的。综上所述，本体溶合运算既不是简单地把两个本体合并成一个新本体，也不是进行类的多重继承。因此目前也无法实现机器自动生成新本体，该运算过程必须在领域专家、本体工程师的参与下进行。显然，在本体库中，类与类之间，类与属性间关系很丰富，例如 is-a,has-a,SubClassOf,ReverseOf，EquivalentClass等，类和属性的总数一般数以百

计甚至更多。所以上图仅仅是示意图。

定义 4 领域模型定义为一个加权无环图DM=(O,A,R) ,其中O是图DM的非空顶点集 ,由描述该领域知识模型的本体有限集构成 ;A∈O是图DM的根节点 ,称为领域知识模型的根本体 ;R O×O×N构成图的加权边 ,表示图中各领域本体的相关度。

定义 5 设C_1, C_2 是领域本体中的两个概念 ,$Sim(C_1, C_2)$表示这两个概念之间的语义相关度[9]，则有

$$Sim(C_1, C_2)=\sum_{i=1}^{n}\delta_i(C_1,C_2)*\theta_i$$

其中： n是概念 C1与 C2在领域本体中所具有的最大深度；θ i是权重(可简单地取θ i =1/n); δ i (C1,C2)取值定义为：

$$\delta_i(C_1,C_2)=\begin{cases}1, & \text{当 C1 与 C2 前 i 个父类代码相同}\\ 0, & \text{当 C1 与 C2 前 i 个父类代码不全相同}\end{cases}$$

溶合运算需要使用的算法如下：

算法 1. 给定领域本体O_1,O_2以及所属领域模型DM_1和DM_2,求语义相关度为 k的新领域本体O.

(1) 若$O_1 \notin DM_1$或$O_2 \notin DM_2$,则算法终止 ,否则继续；

(2) 若 (DR(DMWDR$_1$),DR(DM$_2$))=m且(DR(DMWDR$_2$),DR(DM$_1$))=n(0<m,n),则调用算法2,否则继续;

(3) 分 3种情况：

(i) 如果(DR(DMWDR$_1$),DR(DM$_2$))=m并且(DR(DMWDR$_2$),DR(DM$_1$))=n,则调用算法3在领域模型DM_2内构造以q为基本体,p为扩展本体,语义相关度为m的领域本体O;

(ii) 如果(DR(DMWDR$_1$),DR(DM$_2$))=n并且(DR(DMWDR$_2$),DR(DM$_1$))=n,则调用算法3在领域模型DM_1内构造以p为基本体,q为扩展本体,语义相关度为n的领域本体O;

(iii) 如果(DR(DMWDR$_1$),DR(DM$_2$))=m且(DR(DMWDR$_2$),DR(DM$_1$))=n,则由用户选择领域模型DM_1(或DM_2),调用算法3在领域模型DM_1(或DM_2)内构造以p(或q)为基本体,q(或p)为扩展本体,语义相关度为m(或n)的领域本体O;否则:

(a) 若n<m,则调用算法3在领域模型DM_1内构造以p为基本体(base-ontology),q为扩展本体,语义相关度为n的领域本体O;

(b) 若n>m,则调用算法3在领域模型DM_2内构造以q为基本体,p为扩展本体,语义相关度为m的领域本体O;

(c) 若n=m,则由用户选择DM_1(或DM_2),调用算法3在DM_1(或DM_2)内构造以p(或q)为基本体,q(或p)为扩展本体,相关度为m(或n)的领域本体O;

(4) 算法结束,领域本体O为所求结果.

算法2. 用户参与,以交互方式构造领域本体(算法描述略).

当两个领域本体所属领域模型不存在任何语义关联时,即(DR(DMWDR$_1$),DR(DM$_2$))=0且(DR(DMWDR$_2$),DR(DM$_1$))=0,则由用户使用交互方式选择或新输入本体的属性、方法及关系等.

算法3. 在领域模型DM内构造以p为基本体(p∈DM),q为扩展本体,语义相关度为m的领域本体O.

(1) 将基本体p全部复制到领域本体O中;

(2) 循环执行以下操作,直到DM中再没有新领域本体h,则结束循环.

(a) 若领域本体q∈DM,对领域本体h∈DM且(q,h)=m,则将h构成部分复制到O中;

(b) 若领域本体q∈DM,则求得q所属领域模型DM;对?Ýh∈DM;若=m,则将h构成部分复制到O中。

(3) 领域本体O即为所求。

参与本体溶合运算的本体在属性、方法或关系方面可能有语义重叠 (即语义相关度 Sim(C, C)>0)，这种情况需要用算法 1和 3来构造新本体；或者语义相关度为 0，这种情况就需要使用算法 2来构造新本体 .

所以，图1描述的实例中两个领域本体融合过程可表示为：

金融本体: O(Economy)=(C1 , ¡Ü1, R1, ¦Ò1, ¡Ø1), 其中C1={Economy, Finance ,.....},
¡Ü1={ Finance ¡ÊEconomy ,.....}, R1={Has-a,Is-a,part-of,SubClassOf,.....},
¦Ò1={ transfer (c_1,c_2),……},¡Ø1={ Is-a≤SubClassOf,.....},其中符号'≤ 表示偏序关系。
软件本体: O(SoftWare)=(C2 , ¡Ü2, R2, ¦Ò2, ¡Ø2), 其中C2={SoftWare, SoftProject,SoftWorker ,.....},
¡Ü2={SoftProject¡ÊSoftWare,.....},R2={Is-a,Has-a,Reverse-of, EquivalentClass,Descendent,....},
¦Ò2={ inherited-from (c_1,c_2),……},¡Ø2={ Is-a≤ Descendent,.....},其中符号'≤ 表示偏序关系。
运用算法 2,得到财务软件本体O(FinancialSoft)= (C3 , ¡Ü3, R3, ¦Ò3, ¡Ø3)。

4.2 充分借鉴软件工程的思想和经验，我们提出了领域本体构造过程的原型模型，如下图所示：

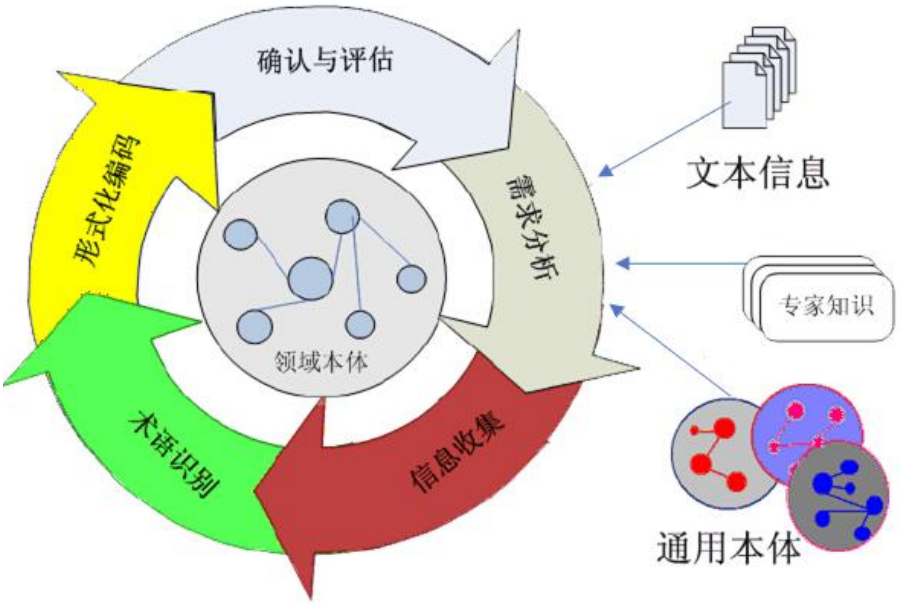

图 2 领域本体构造的原型模型

它是一种环状的结构。环形的起点是一个通用的核心本体的选择[10,11]。
任何大型的通用本体（像 Cyc、Dahlgren 的本体）、词汇-语义网（像 WordNet, GermaNet）、或者领域相关的本体（像 TOVE）都可以作为这个过程的开始。选定基础本体后，用户必须确定用于抽取领域相关实体的文本。

① 需求分析。这个阶段需要明确领域本体建设的目的、范围、用途和使用者。从表面上看,领域本体的构建是为机器服务的，要让信息变成机器可理解的。但其最终目的还是为人类提供更好的信息检索服务。因此，和软件开发过程类似，在本体建设的初期，应该首先了解其应用的具体背景和需求。一般可以通过

② 从选择的文本中获取领域相关的概念，并建立概念之间的分类关系。
除去领域无关的概念，只留下和领域相关的。这时，建立起了目标本体的概念结构。
从基础本体中会继承一些关系，其他的关系需要通过学习的方法从文本中抽取。
对得到的领域相关的本体进行评价，还可以进一步的重复上述过程。

5 相关工作

本体构建工具有很多，目前较为流行的是美国斯坦福大学开发的Protégé工具为一般的和可扩展的软件环境，它使用户可以模型化，询问和应用任何领域本体。更具体地说，Protégé提供本体 -模型化编辑器，领域专家可以利用它表示他们的知识Protégé不作任务层次的清晰表达。如果需要Protégé可以把任务 -规定知识结合到应用程序本体中，即带有具体应用系统所需要的重要知识的领域本体的扩大体。我们借助Protégé工具使用OWL语言，利用上述本体溶合算法，在领域专家的帮助下，构造了WEB服务构件本体库原模型．

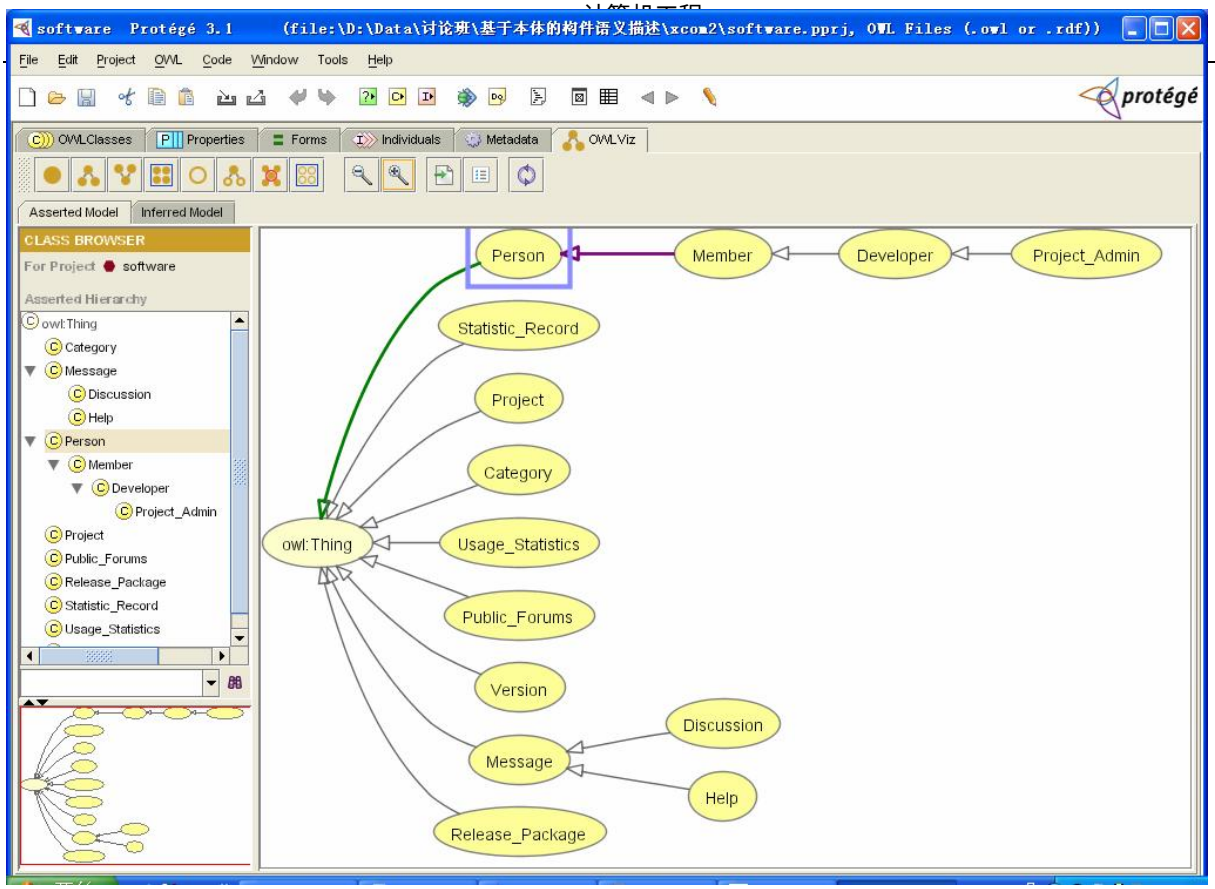

图 3 WEB服务构件本体描述在Protégé中的图形化显示

6结束语

领域本体需要知识专家与领域专家共同来构建，这原本就是一件非常复杂的事情，首先得建一些初始的原型，经过实践检验后，如果真的有用而且好用，才可能得到公认，这是一个渐进的过程，现在每个从事领域本体研究的人正在各个领域努力着，机会与挑战并存我们也看到，构建领域本体对软件构件进行描述及应用还处于初步阶段，其理论和方法还有待于进一步完善。其主要困难体现在以下几个方面：

(1)目前的Ontology很多都是人工开发的，这样需要耗费很多的人力、物力和财力，时间周期也很长，这在一定程度上影响了Ontology 的应用。

(2)Ontology构建的原则、方法及其表示等许多方面都没有形成一个统一的标准，这也使得Ontology 只是作为某一个单独的团体或组织内的共享，真正意义上的共享和重利用仍然没有实现。

(3) 合理界定领域边界是开发领域本体的首要前提，否则在实际开发过程中，概念、关系会无限蔓延，最终还是突破领域，进入一般知识层面，这样也就背离了开发领域本体的初衷。

(4)在Ontology 的理论基础方面，Ontology 的评价方法以及形式化方法还需要进一步研究与探讨。今后Ontology的研究重点也将围绕着这些困难继续进行，在理论与应用两个方面不断地深入下去。

参考文献

[1] 朱三元，钱乐秋，宿为民.软件工程技术概论,北京：科学出版社

[2]《基于XML的软件构件查询匹配算法研究》 徐如志，钱乐秋，程建平，王渊峰，朱三元 软件学报 2003年 0期

[3] Burbeck, S. Evolution of Web Applications into the Service-Oriented Components with Web Services, Online Document, October 2000.

[4] Getting Started With Protégé： http://protege.stanford.edu/doc/tutorial/get_started/index.html

[5] OWL Web Ontology Language Overview http://www.w3.org/TR/owl-features/

[6] A Practical Guide To Building OWL Ontologies Using The Protégé-OWL Plugin and CO-ODE Tools Edition 1.0: Matthew

Horridge,Holger Knublauch, Alan Rector1, Robert Stevens1, Chris Wroe The University Of Manchester Stanford University

[7] Ivan Herman, 步向语义网 http://www.w3.org/2003/Talks/1112-BeijingSW-IH/Chinese/1.html

[8] E.Bozsak, Kaon-towards a large scale semantic web, In Proceedings of the Third International Conference on E-Commerce and Web Technologies(EC-Web 2002), Springer Lecture Notes in Computer Science, 2002

[9] 朱礼军 陶兰 刘慧: 领域本体中的概念相似度计算. 华南理工大学学报 Vol32,November 2004

[10] Jie Yang,Lei Wang,Song Zhang,Xin Sui,Ning Zhang,Zhuoqun Xu: Building Domain Ontology Based on Web Data and Generic Ontology.Proceeding of the IEEE/WIC/ACM International Conference on Web Intelligence(WI'04)

[11] 何海芸，袁春风. 基于Ontology的领域知识构建技术综述. 计算机应用研究 2005年第3期