

# 基于本体的 Petri 网模型转换方法

王卓<sup>1,2</sup> 冯晓宁<sup>1</sup> 徐玉如<sup>2</sup>

(哈尔滨工程大学计算机科学与技术学院 哈尔滨 150001)<sup>1</sup>

(哈尔滨工程大学船舶工程学院 哈尔滨 150001)<sup>2</sup>

**摘要** 在现有的 UML 到 Petri 网模型转化规则的基础上,引入本体将其作为各种模型到 Petri 网模型之间转化的桥梁,研究各种模型到 Petri 网的通用转化方法。为每个模型包括 Petri 网模型建立本体,再根据基于分类树的本体映射建立模型转化步骤。通过计算不同模型中概念的相似度建立不同模型中概念的对应关系,完成模型转化。

**关键词** 模型转化, Petri 网, 本体

**中图分类号** TP302.7 **文献标识码** A

## Petri Net Model Transition Method Based on Ontology

WANG Zhuo<sup>1,2</sup> FENG Xiao-ning<sup>1</sup> XU Yu-ru<sup>2</sup>

(College of Computer Science and Technology, Harbin Engineering University, Harbin 150001, China)<sup>1</sup>

(College of Shipbuilding Engineering, Harbin Engineering University, Harbin 150001, China)<sup>2</sup>

**Abstract** The method described in this paper is based on the existing transition rules between UML and Petri net. Ontology was used as a bridge to produce a general transition method between other models and Petri net. Ontology was established for every model including Petri net. Model transition process is based on ontology mapping of classified tree. The relationship between concepts of different models was established by computing their similarities to complete the transition of models.

**Keywords** Model transition, Petri net, Ontology

## 1 引言

Petri 网作为复杂并发系统的有效形式化建模方法,能够捕获其本质的特性如非决定性、同步性和并行性。目前 Petri 网作为形式化建模方法已常用于需求分析阶段,然而要在设计阶段创建形式化模型,能够验证其正确性是非常困难的。这不仅是因为技术不成熟,符号应用困难,而且同大多数形式化方法一样 Petri 网缺乏丰富的语义描述而且缺乏模块化支持。另一方面包括 UML 在内的面向对象建模方法能够充分体现软件的可重用、模块化的特性,但是其缺乏并发特性的描述和验证。因此,将两个概念结合起来使用的思路就自然产生了,所有开发人员可以从两种方法的优势中获得好处。

J. A. Saldhana 和 S. M. Shatz<sup>[1]</sup> 已提出了一个方法,将 Petri 网扩展成面向对象的 Petri 网,并与 UML 图连接在一起。他们应用状态图和协作图表达变迁机制,用状态图表示内部实体的动作与活动,用协作图将不同的通信角色连接到某一系统中。这种系统被设计成与采用面向对象的内部实施相分离的通信实体的界面,这是将 Petri 网扩充支持模块化的第一步。R. Pooley 和 P. King<sup>[2]</sup>, F. Bergenti 和 A. Poggi<sup>[3]</sup> 等专家也在这方面做了相当多的工作,然而他们的工作主要集中在一些具体的 UML 图到 Petri 网的转换。很多具有重要

意义的文献(如 K. Jensen<sup>[4]</sup>),论述了使用有色 Petri 网方法进行系统仿真、有效性和正确性验证。还有许多专家将面向对象概念融入到 Petri 理论中,提出了对象 Petri 网(OPN)<sup>[5]</sup>, VDM++<sup>[6]</sup> 等面向对象 Petri 网,这些方法虽然得到了一定的应用,但是这些方法并不完全地支持面向对象方法的主要概念。

但是上述专家的研究工作主要是增强 Petri 网本身的描述能力,或是设计一些具体的 UML 图到 Petri 网的转换规则,这些规则都是专用于某种建模方法甚至专用于某一版本,其方法适用范围有限。本文以上述研究为基础,将 Ontology 引入到 Petri 网中,旨在建立一种 UML, IDEF 等描述性模型到 Petri 网模型之间自动转化框架,提供一种通用的模型转化方法。

## 2 本体相关概念

本体论(Ontology)的概念最初起源于哲学领域,它在哲学中的定义为“对世界上客观存在物的系统描述,即存在论”,是对客观存在的一个系统的解释或说明,关心的是客观现实的抽象本质<sup>[7]</sup>。

在人工智能界,最早给出本体定义的是 Neches 等人,他们将 Ontology 定义为“本体定义了组成主题领域的词汇表的

到稿日期:2008-07-07 返修日期:2009-03-02 本文受国家自然科学基金(40746029),哈尔滨工程大学校基金(002060260722)资助。

王卓 副教授,研究方向为系统建模、软件体系结构、软件可靠性等,E-mail:wangzhuo@hrbeu.edu.cn;冯晓宁 讲师,研究方向为系统建模、Petri 网等;徐玉如 工程院院士,研究方向为智能水下机器人等。

基本术语和关系,以及组合这些术语和关系来定义词汇表外延的规则<sup>[8]</sup>。后来在信息系统、知识系统等领域,越来越多的人研究本体,并给出了许多不同的定义。目前被大部分人公认的定义是 Tom Gruber 在 1994 年提出的<sup>[9]</sup>:“本体是关于共享概念的一致约定。共享概念包括用来对领域知识进行建模的概念框架、需要互操作的主体之间用于交互的与内容相关的协议和用于表示特定领域的理论的共同约定。在知识共享的情况下,本体的形式特化为具有代表性的词汇的定义。一种最简单的形式是一种层次结构,用来详细描述类和它们之间的包含关系。关系数据库的框架(schemata)也是一种本体,它用来描述能共享的数据库之间的关系和集成这些数据库需遵循的约束”。

尽管定义的方式不同,但从内涵上来看,不同研究者对于本体的认识是统一的,本体的概念有 4 层含义<sup>[10]</sup>:

(1)概念化(conceptualization):通过抽象出客观世界中一些现象(Phenomenon)的相关概念而得到的模型,其表示的含义独立于具体的环境状态。

(2)形式化(formal):有精确的数学描述,是计算机可读的。

(3)明确(explicit):概念与概念之间的联系及使用这些概念的约束都被明确定义。

(4)共享(share):本体中体现的是共同认可的知识,反映的是相关领域中公认的概念集,它所针对的是团体而不是个体。

总之,本体的目标是捕获相关领域的知识,提供对该领域知识的共同理解,确定该领域内共同认可的词汇,并从不同层次的形式化模式上给出这些词汇(术语)和词汇之间相互关系的明确定义。本体既包括一个领域内的知识也包括各种领域之间的知识。人类、数据库和应用软件使用本体来共享领域知识。

### 3 模型本体的构建

基于本体的模型转换方法的核心思想是,为各种建模方法建立本体,例如 Petri 网本体和 UML 本体等。在此基础上再建立各个模型本体之间的本体映射,通过映射算法得到个模型之间概念的对应关系,从而实现模型之间的自动转换。

#### 3.1 模型本体的结构

一个本体通常由一些基本的成分,即概念、概念属性、关系、关系属性、层次和实例等构成,例如 Petri 网本体就是由库所、变迁、弧及其相互关系构成的概念系统<sup>[6]</sup>。为了降低建模工作的复杂程度,实现建模工作的分工,模型本体的结构以 3 个层次构建,如图 1 所示,分别是抽象层、具体层和实例层。

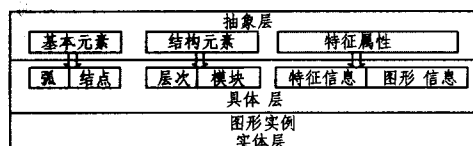


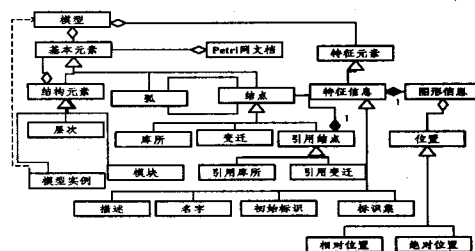
图 1 本体模型层次结构图

抽象层包含基本元素、结构元素和特征元素 3 个抽象概念,分别定义各建模方法的核心组成部分、结构和各组成部分的特征描述。基本元素是与模型中基本图元对应的概念集合。结构元素是描述模型层次结构的概念集合。特征元素是

对基本图元的实际语义进行描述的概念集合,在模型转化过程中首先根据特征元素中的概念找到描述相同语义的图元,然后再根据图元的图形描述相似度和层次结构相似度进行转化。与抽象层的概念相对应,具体层给出了不同建模方法中包含的具体概念。需要指出的是模型本体的可扩展性主要体现在特征属性的具体概念特征信息上,例如高级 Petri 网对核心概念的扩展概念定义都通过特征信息中的具体概念来体现。实体层包含具体的概念实例,这些实例构成一张用具体建模方法构建的模型图。

#### 3.2 Petri 网本体的构建

基于上述层次结构的 Petri 网本体中的部分概念及组成如图 2 所示。库所、变迁和弧是 Petri 网本体的最基本组成,属于基本元素。其中,库所和变迁二者合称为结点(node),弧的两段所连接的分别为源结点(source node)和目标结点(targetnode)。为了防止一张 Petri 网图形中描述信息过多,给出了模块和层次的概念。模块机制就是将 Petri 网图形从某些库所和变迁处断开,从而分成若干个部分,每个部分作为一个“模块”。模块与模块之间由在断开处新增添的引用库所(reference place)和引用变迁(reference transition)进行链接。两者合称为引用结点(reference node)。模块机制的引入使支撑 Petri 网结构的基本概念新增了 4 个,即引用结点、引用库所、引用变迁和模块,其中前 3 组概念属于基本元素,模块属于结构元素。其中,模块不仅包含结点和弧,还可以链接其它的模块。层次概念是为了使 Petri 网能够展现各种粒度的模块的信息而引入的,与模型配合使用。在结构上“层次”使一个完整系统的 Petri 网模型形成树状结构,上层的模块中的某些库所被扩展为下层的一张 Petri 网模型图(即一个模型),被扩展的库所称为可扩展库所。因此,层次和可扩展库所这两个概念被引入本体树,层次属于结构元素,可扩展库所属于基本元素。特征属性是高级 Petri 网为了在其库所、变迁和弧上标注信息所引入的新机制。特征元素包括特征信息和图形信息。特征信息主要指明被标注对象的性质与特征,图形信息主要提供图形信息(graphical information)规定被标注对象的位置、尺寸、线条、字体和填充等。



#### 3.3 UML 本体的构建

UML 本体的概念层次同样遵循图 1 所示的结构,分为抽象层、具体层和实例层,抽象层的基本概念仍然是基本元素、结构元素和特征元素。UML 的动态模型和静态模型中共包含 9 种图,这 9 种图中除时序图之外其他图形可以归结为由弧和结点基本元素组成,由于时序图与协作图表达的信息完全一致,只是描述的侧重点不同,因此在 UML 本体中将其时序的基本元素信息忽略不计。由于静态模型与动态模型的本质区别,因此在 UML 本体的具体层进一步将弧和节点两大

概念细分为动态节点、静态节点、动态弧和静态弧 4 个概念,其中静态节点包括类、角色和用例,动态节点包括活动、状态和对象,静态弧包括关系,动态弧包括转移。体现结构特性的概念是包和组件,它们属于结构元素。特征元素中包含特性、条件、名字和成员。UML 的概念层次结构如图 3 所示。其他的建模型方法同样可以建立三层概念层次结构。

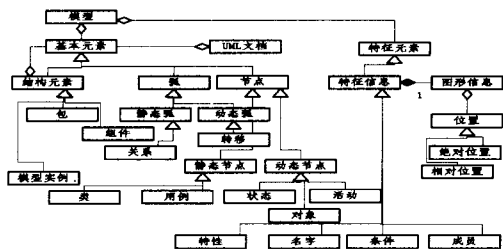


图 3 UML 本体结构图

#### 4 模型转换

为了实现 Petri 网模型与 UML 等其他模型之间的信息共享,在已经建立了包括 Petri 网本体在内的各个模型本体的基础上,进一步通过本体映射,建立各个模型本体之间的联系,从而实现模型信息的共享。本体映射方法分为:单本体、多本体和混合本体方法。鉴于模型本体本身的术语数量有限,本文采用混合本体方法实现本体映射。基于混合本体实现模型转换的步骤如下。

(1)模型本体化:为需要进行转换的模型按照图 1 的层次关系建立本体。

(2)概念相似度计算:通过特性分析自动地发现本体中相似的概念,计算相似度并产生一个相似矩阵。

(3)图元映射:根据相似矩阵生成相应的映射规则。映射规则确定了如何转换源模型的概念到目标模型中。映射关系一般有 1:1,1:n 和 m:n 三种。另外 m:n 的映射关系可以用 m 个 1:n 的映射关系来组合表示。映射规则应该包括被映射模型概念实例的条件以及相应的转换函数。

(4)实体映射:输入源模型的相关实体,并利用已经得出的映射规则进行映射。它通过一次映射就转换源模型的所有相关实体到目标模型中,并把转换结果存在相应的知识管理系统中。

(5)映射后处理:在进行时需要考虑模型之间的图元表示。如果图元表示形式不一致,或图元位置不合理则需要根据图形信息在表示形式进行调整。

#### 5 相似度计算方法

模型转换的实质是模型本体中的概念映射,建立概念之间的对应关系,概念映射的核心是概念之间的相似度计算和图元映射。模型本体规模有限,本体层次结构较类似,而且特征元素的相似度直接影响其他概念的相似度计算,因此本文提出了基于分类的本体概念映射方法。这种映射方法的基本思想是:通过分类将模型本体重构为若干小型的本体树,以特征元素中概念作为分类树的主体,对于分类树的根节点(也称为本体库的分类节点)建立相似映射,这样所有的相似度计算均限制在建立了相似映射的分类树根节点所在分类树之间,不仅保证了元素在概念相似中的主导地位,也大大降低了概念相似度计算的复杂度。本体库中概念之间的基本关系是包含关系,通常是用树状结构描述,有分类节点和叶子概念节

点。由于模型转换过程主要是根据特征元素的相似度找到不同模型之间基本元素的对应关系,因此特征元素所起作用大于基本元素和结构元素,在本体库中优先将特征元素作为分类节点,基本元素和结构元素作为叶子节点。

针对模型本体的这些特点,在相似度计算方法中假定  $O_1$  和  $O_2$  为被映射的两个本体库,  $C_i$  和  $C_j$  为本体概念,且  $C_i \in O_1$  和  $C_j \in O_2$ 。  $T_i$  和  $T_j$  也为本体概念,且  $T_i \in O_1$  和  $T_j \in O_2$ ,但  $T_i$  和  $T_j$  不为叶子节点概念,即总有  $\exists C_i \subset T_i$  和  $\exists C_j \subset T_j$ 。将  $tree_i = \{\forall C_i | C_i \subset T_i\}$  称作以  $T_i$  为根节点的分类树  $tree_i$  ( $tree_j$  同理),  $T_i$  和  $T_j$  称为本体库的分类节点。

通常,如果两个模型本体  $O_1$  和  $O_2$  中,两个分类节点是相似的,即  $\text{sim}(T_i, T_j) > t$  ( $t$  为相似度阈值),可以认为分类树  $tree_i$  和  $tree_j$  所包含的所有节点之间是可能映射的,即  $\exists \{\text{sim}(C_i, C_j) | > t\}$ ,从而概念映射相似度的计算效率高。这一步主要由领域专家来完成。 $tree_i$  和  $tree_j$  的映射可以细化到每个分类树至少要有超过 12 个概念元素,即  $\text{number}(tree_i) \geq 12$  (包括叶子节点和分类节点)。

得到了分类节点的映射之后,对映射分类节点所引导的分类树包含的概念节点  $\{\text{sim}(C_i, C_j) | C_i \in tree_i, C_j \in tree_j\}$  进行相似度计算。计算方法是基于样本文档来计算概念间的相似度。具体计算步骤和计算公式如下:

(1)以分类树  $tree_i$  和  $tree_j$  下的所有元素  $C_i \in tree_i, C_j \in tree_j$  的笛卡尔积  $\{\langle C_i, C_j \rangle | C_i \in tree_i, C_j \in tree_j\}$  对作为关键词组  $\langle C_i, C_j \rangle$ 。对每组关键词搜索一个样本文档,组成一个样本文档集合  $\text{document} = \{D_k | D_k \in \text{document} \cap (C_i \in D_k \cap C_j \in D_k)\}$ ,样本集合的数量  $K = \text{number}(tree_i) \times \text{number}(tree_j)$ 。

(2)将  $\langle C_i, C_j \rangle$  关键词组在每个样本文档  $D_k$  中同时出现的频率记为  $f_{ij}^k$ ,概念  $C_i$  在样本文档  $D_k$  中出现的频率记为  $f_i^k$ ,  $C_j$  在样本文档  $D_k$  中出现的频率记为  $f_j^k$ ,则可能的映射对  $\langle C_i, C_j \rangle$  的相似度值计算公式如下(注意,如果  $f_i^k, f_j^k$  中有一个为 0,则  $f_{ij}^k$  一定为 0,此次计算不加在其中;另外,  $n$  表示  $f_{ij}^k$  不为 0 的次数):

$$\text{sim}(C_i, C_j) = \left\{ \sum_{D_k \in \text{document}} [2f_{ij}^k / (f_i^k + f_j^k)] \right\} / n$$

阈值设为 0.6 (可根据具体应用领域调整)。该公式的基本思想为:可能的映射对  $\text{sim}(C_i, C_j) \geq 0.6$  在样本文档中成对出现的频率要远远大于非映射对  $\text{sim}(C_i, C_j) < 0.6$  在样本文档中成对出现的频率。

(3)对于无法建立映射的分类节点,其分类树包含的概念节点按照传统相似度计算方法进行计算。

**结束语** 本文在研究了现有的 UML 到 Petri 网转化规则的基础上,将本体引入到模型转化方法中,作为其他模型到 Petri 网之间的桥梁。为每种模型包括 Petri 网建立本体,利用基于分类树的本体映射方法建立不同模型中各概念之间的对应关系。这种模型转化方法具有两个特点,一是一种通用的模型转化方法,UML, IDEF 和 E-R 图等模型都可以通过建立本体完成与 Petri 网之间的自动转化。二是通过将特征元素作为分类树的根节点保证了特征元素在映射过程中的主导地位,符合模型转化的特点而且降低了计算复杂度。

#### 参考文献

- [1] Saldhana J A, Shatz S M. UML diagrams to object petri net models: an approach for modeling and analysis[C]// Proceeding of Twelfth International Conference on Software Engineering and Knowledge Engineering. 2000: 103-110

示噪声点。

(1) 对  $W_n[x_{ij}]$  中的像素排序, 得到

$$m_{ij} = \min(W_n[x_{ij}])$$

$$M_{ij} = \max(W_n[x_{ij}]) \quad (1)$$

$$g_{ij} = \begin{cases} 1, & (x_{ij} = m_{ij}) \text{ or } (x_{ij} = M_{ij}) \\ 0, & \text{others} \end{cases} \quad (2)$$

(2) 对所有满足  $g_{ij} = 1$  的候选噪声点, 将  $W_n[x_{ij}]$  中与  $m_{ij}$  和  $M_{ij}$  都不相等的  $L$  个像素所对应的灰度值组成一个集合  $T_{ij}$ :

$$T_{ij} = \{t_{ij} \mid t_{ij} \neq m_{ij} \text{ and } t_{ij} \neq M_{ij}, t_{ij} \in W_n[x_{ij}]\} \quad (3)$$

计算集合  $T_{ij}$  的灰度均值  $u_{ij}$ :

$$u_{ij} = \frac{\sum t_{ij}}{L} \quad (4)$$

(3) 对满足  $g_{ij} = 1$  的候选噪声点的灰度值  $x_{ij}$  与  $u_{ij}$  比较, 若差值大于阈值  $th$ , 则认为真正的噪声点, 否则为信号点。

$$z_{ij} = \begin{cases} 1, & (|x_{ij} - u_{ij}| > th) \text{ and } (g_{ij} = 1) \\ 0, & \text{else} \end{cases} \quad (5)$$

$$SNR = 10 \lg \left[ \frac{\sum_{i,j} s_{ij}^2}{\sum_{i,j} (s_{ij} - y_{ij})^2} \right] \quad (7)$$

阈值  $th$  由实验确定。

$$y_{ij} = \begin{cases} \text{med}(D_{ij}), & \text{if } (z_{ij} = 1), nb > 0 \\ x_{ij}, & \text{else} \end{cases} \quad (8)$$

## 2 实验结果与分析

本文以一幅  $308 \times 242$  的图像为例, 对  $5 \times 5$ 、 $7 \times 7$  的标准中值滤波、 $5 \times 5$  EM 滤波、 $5 \times 5$  IEM 滤波进行比较 (取  $th = 15$ )。  $[x_{ij}]$  表示输入图像,  $[y_{ij}]$  表示输出图像,  $[s_{ij}]$  表示未加噪声的图像, 则输出图像信噪比, 如图 1 所示。

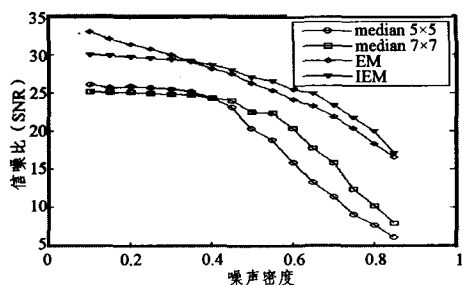


图 1 各种输出图像信噪比与噪声密度的关系

由图 1 可见, 在各种输入噪声密度下, EM 和 IEM 都比标准的  $5 \times 5$  和  $7 \times 7$  中值滤波效果好。当噪声密度较大时 ( $D > 0.35$ ), IEM 方法明显好于 EM 方法; 而当噪声密度  $D <$

$0.35$  时, EM 方法要好于 IEM 方法, 产生这种现象的原因是当噪声密度较小时, 由于阈值 ( $th$ ) 相对较大, 因此应相应地减小  $th$  的值, 或者在噪声密度较小时, 直接使用 EM 方法; 而在噪声密度  $D > 0.35$  时, 使用 IEM 方法。

图 2 是噪声密度  $D = 0.6$  条件下各滤波器的滤波效果。可见, IEM 方法在  $D = 0.6$  条件下比 EM 方法, 不仅噪声滤波效果好, 而且更好地保护了图像细节。

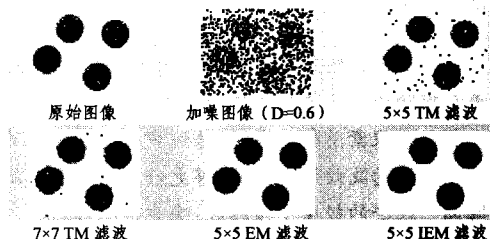


图 2 各种滤波算法比较

**结束语** 本文提出的 IEM 方法与 EM 方法相比较, 主要有以下两个方面的改进:

- 1) 滤波前对噪声点的判断更加合理, 只对真正的噪声点进行滤波, 对信号点不做处理;
- 2) 噪声检测算法一方面解决了当噪声点个数大于窗口像素点总数一半时, 传统方法无法解决这一问题, 另一方面也防止了噪声的集结与传播, 改善了图像质量, 尤其当噪声密度较大时优越性更加明显。

## 参考文献

- [1] Brownning D R K. The weighted median filter[J]. Communication of the Association for Computing Machinery, 1984, 27(8):807-818
- [2] Sun T, Neuvo Y. Detail-preserving median based filters in image processing[J]. Pattern Recognition Letters, 1994, 15(4):341-347
- [3] Wang Junhua, Lin Lianda. Improved median filter using min-max algorithm for image processing[J]. Electronics Letters, 1997, 33(16):1362-1363
- [4] Wang Z, Zhang D. Processing switching median filter for the removal of impulse noise from highly corrupted images[J]. IEEE Trans. on Circuits and Systems, Analog and Digital Signal Processing, 1999, 46(1):78-80
- [5] 刑藏菊, 王守觉, 邓浩江, 等. 一种基于极值中值的滤波算法[J]. 中国图象图形学, 2001, 6(6):533-536

(上接第 149 页)

- [2] Pooley R, King P. Derivation of petri net performance models from UML specifications of communications software[C]//Proceedings of the fifteenth annual UK Performance Engineering Workshop. 1999:45-56
- [3] Bergenti F, Poggi A. Exploiting UML in the design of multi-agent systems[C]//A. Omicini, R. Tolksdorf, F. Zambonelli, eds. Engineering Societies in the Agents World. 2000:106-113
- [4] Jensen K. An introduction to the theoretical aspects of coloured Petri nets, a decade of concurrency. 1994:230-234
- [5] 夏世芬. 基于 Petri 网模型的归结自动推理研究[M]. 西安: 西

南交通大学, 2006:12-13

- [6] 黄映辉, 李冠宇. Petri 网 Ontology: 定义与类型[J]. 系统仿真学报, 2007, 19(Suppl. 1. 8):34-37
- [7] 邓志鸿, 唐世渭, 张铭, 等. Ontology 研究综述[J]. 北京大学学报: 自然科学版, 2002, 38(5):730-738
- [8] Neches R, Fikes R E, Gruber T R, et al. Enabling Technology for Knowledge Sharing[J]. AI Magazine, 1991, 12(3):36-56
- [9] Gruber T. <http://ksl-web.stanford.edu/people/gruber/>
- [10] Fensel D. The semantic web and its languages[J]. IEEE Computer Society, 2000, 15(6):67-73