本体的继承及一致性分析

明 仲[1,2],蔡树彬[1],李师贤[1],徐 晶[1]
(1. 深圳大学信息工程学院,广东深圳 518060;2. 中山大学计算机科学系,广东广州 510275)

摘 要: 本体的继承理论是目前尚未解决的重要问题.该文通过扩展继承的数学理论,系统研究了本体的继承语义、继承机制等问题.主要工作有:(1)继承系统的数学理论是描述继承系统和探讨其一致性、二义性等问题的一个重要基础理论.该理论已被广泛引用.但使用该理论描述允许例外的类/属性继承系统会出现假冲突.本文发现并定义了假冲突,扩展了原理论的单词表,重新定义了结论集、可继承等概念,使继承断言的接地扩展集不因假冲突而产生二义.从而解决了假冲突的问题.(2)证明原继承系统中的大部分重要定理在扩展后的继承系统中仍然成立,并且提出了新的定理3)将本体(D,P,U,V,H)转换为三元组(F,E,I)的表示形式,证明这种转换不削弱本体的表示能力,并且将本体映射到扩展后的继承系统,利用继承系统的定义、定理分析本体继承的一致性问题,并证明这种映射对本体的包含断言的判定是完备的.4)设计了上、下扫描算法,通过检测系统变化可能引起的冲突,保持系统的一致性.对本体继承的一致性分析为发展迅速的面向本体工程方法提供了有力支持.

关键词: 本体;继承系统;继承理论;允许例外的多继承;一致性
中图分类号: TP311 **文献标识码**: A **文章编号**: 0372-2112(2005)04-0660-07

Ontology Inheritance and Consistency Analysis

MING Zhong[1,2],LI Shi-xian[1],CAI Shu-bin[1],XU Jing[1]
(1. *Faculty of Information Engineering, Shenzhen University, Shenzhen, Guangdong* 518060, *China*;
2. *Department of Computer Science, SUN Yet-sen University, Guangzhou, Guangdong* 510275, *China*)

Abstract: Inheritance theory of ontologies is a very important unsolved problem. By extending the inheritance mathematical theory, the semantics of ontology inheritance and the mechanism of ontology inheritance are analyzed. The creative works are as followed: (1) The mathematics of inheritance systems isn't powerful enough to explain the ontology inheritance system. The "False conflict" will occur when we use the old theory of inheritance to describe the inheritance of class/property with exceptions. In order to solve this problem, the old theory is extended by adding tokens and redefining the definitions of conclusion set, inheritable and grounded expansion, etc. (2) Most theorems and corollaries prove to be correct on the new definitions. Above these new definitions, some new theorems and corollaries are got. (3) Transferring the representation of ontology (D,P,U,V,H) into triple (F,E,I). And proving the capability of the triple representation equals to the quintuple representation. The ontology is mapped into the extended inheritance system. The problem of consistency and un-ambiguity of ontology inheritance are explored with the extended theory. (4) The up-scan and down-scan algorithms are designed to detect the conflict in order to maintain the consistency of the system. The analysis of consistency of the ontology inheritance system effectively supports the ontology-oriented engineering methodologies booming now.

Key words: ontology; inheritance theory; multiply inheritance with exception; consistency

1 引言

继承具有重要的实践意义,它简化了人们对事物的认识和描述,有利于知识的表示、共享、重用和推理,是工程领域中的重要概念.在有关继承的研究中,文献[3]被频繁引用.文献[3]深入研究了基于ISA(继承)、ISNOTA(例外)和NOCONCLUSION(无结论)的继承理论,并从数学的角度讨论、证明了许多有用的结论.但该理论在表示允许例外的多继承的类/属性继承系统时,会产生假冲突问题.如图1例子中,左边的类/属性的继承图没有产生冲突,而右边使用传统继承理论表示的继承系统却产生"见习经理"包含/不包含属性"奖金"的假冲突.

假冲突是由语义超载引起的.语义超载使知识表示不清晰,并可能导致错误,文献[4]强烈批评了这种做法.在上述右边的继承系统中,ISA边语义超载了IsA(是)和Contain(包含属性)两种语义."见习员工是(IsA)员工"和"员工包含属性(Contain)工资"都使用了ISA边表示.而ISNOTA边超载了3种

收稿日期:2004-06-04;修回日期:2004-09-18
基金项目:国家自然科学基金(No.60373084);博士学科点基金(No.20030558004);广东省自然科学基金(No.04011304)

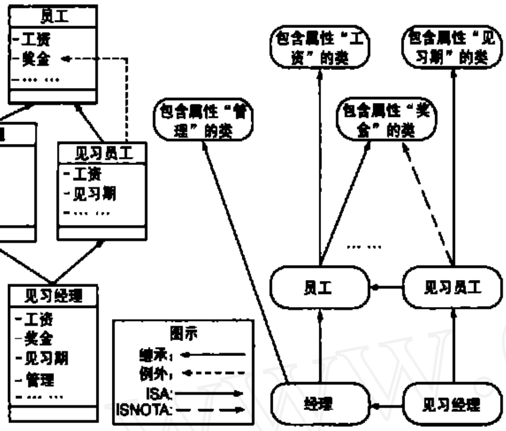


图 1 假冲突问题

语义:①IsNotA(不是祖先类),即对祖先类的例外,消除 IsA 的传递;②NotContain(不包含祖先的属性),即对继承属性的例外,不包含从祖先继承到的属性. ③IsNotCompatibleWith(不兼容),表示两个类或两个属性不兼容.

IsNotA(不是祖先类)的例子如图 2 所示,表演犬不是祖先类(IsNotA)人. 图 1 中"见习员工不包含属性奖金"则是 NotContain(不包含祖先属性)的一个例子. 注意到,"见习员工不包含属性奖金"不表示见习员工的后代不能通过其他途径包含属性奖金,而仅表示对"见习员工是员工,员工包含属性奖金,所以见习员工也包含属性奖金"这一继承推理的否定. 不包含的属性并不是一个后代可继承的属性[5,6],因此类并不传递继承"不包含的属性". 因此,NotContain 关系不沿 IsA 边传递. 上例中,见习经理多继承见习员工和经理不产生多继承冲突. 但这在传统继承理论中却产生了假冲突. 如"人不是犬类"、"邮局不是食物"等的例子则是表示 IsNotCompatibleWith(不兼容)语义的例子. 这种语义是对类及其后代类的实例施加的约束规则,表示任何一个犬类或其后代类的实例,不能同时是人类或其后代类的实例. 但仅使用 IsNot-CompatibleWith 表示约束显得表示能力不足,它不能表示如"杀人者一般不是(IsNot-CompatibleWith)好人",而"死刑执行者是(IsA)杀人者,也是(IsA)好人"这样的情况. 文献[8]详细讨论了如何处理这种情况的问题.

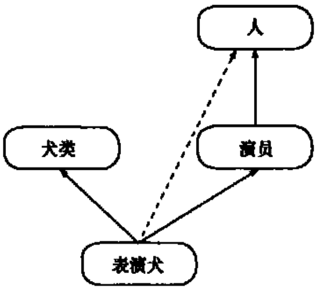


图 2 ISNOTA(不是祖先类)的例子

2　扩展的继承理论

我们对文献[3]的传统继承理论进行扩展,并称扩展后的继承理论为扩展继承理论. 在扩展继承理论中,我们使用"+"表示 IsA(是)的语义;"⊕"表示 Contain(包含属性)的语义;"-"表示 IsNotA(不是祖先类)的语义;"⊗"表示 NotContain(不包含祖先的属性)的语义,并使用约束条件集 C 表示 IsNot-CompatibleWith(不兼容)的语义. ∏表示继承系统中类名和属性名的集合,单词集 $\Theta=\{+,\oplus,-,\otimes\}\times\prod$. 并记 $\forall x\in\prod$ 的约束条件集为 $x.c$. Δ 表示继承断言的集合. 继承断言是 $\Theta\times\Theta$ 中形式良好的元素,即形式良好的单词序偶. 令 A、B 表示类名,a、b 表示属性名,如表 1,扩展的继承理论共存在 6 类形式良好的单词序偶.

表 1　形式良好的单词序偶

序偶	直观意义
$\langle +A,+B\rangle$	类 A 继承类 B
$\langle +a,+b\rangle$	属性 a 继承属性 b
$\langle +A,\oplus a\rangle$	类 A 包含属性 a
$\langle +A,-B\rangle$	类 A 不继承祖先类 B
$\langle +a,-b\rangle$	属性 a 不继承祖先属性 b
$\langle +A,\otimes a\rangle$	类 A 不包含祖先的属性 a

表 2　x 的意义

x	x 代表意义
$+a$	$\oplus a$、$-a$ 或 $\otimes a$
$-a$	$\oplus a$ 或 $+a$
$\oplus a$	$+a$、$\otimes a$ 或 $-a$
$\otimes a$	$+a$ 或 $\oplus a$

我们重新定义下述基本概念,在保持传统继承理论定理成立的情况下,修正其假冲突问题. 下面定义中,a、b 等变量是∏中的元素,x,y 等变量是 Θ 中的元素,其右上角的撇记号'表示意义如表 2 所示.

定义 1　结论集. 序列集 Φ 的结论集 $C(\Phi)$,定义为

(a) 如果 $\langle x,y\rangle\in\Phi$,则 $<x,y>\in C(\Phi)$

(b) 如果 $\langle x_1,\cdots,x_n,y\rangle\in\Phi$, $n>1$ 并且 y 的符号不是"⊗",则 $\langle x_1,y\rangle\in C(\Phi)$.

序列 $\langle x_1,\cdots,x_n,y\rangle$ 表示从 x_1 到 y 的推理过程,如 $\langle +a,+b,+c\rangle$ 表示"(是) a 是一个 b 是一个 c". 结论集中不包含"类通过继承得到的不包含的属性"的结论.

定义 2　冲突. 序列集 Φ 与序列 $\sigma=\langle x_1,\cdots,x_n\rangle$ 冲突,当且仅当有 $\langle x_1,x_i'\rangle\in C(\Phi)$, $1\le i\le n$.

定义 3　中介. y 是 Φ 中序列 $\sigma=\langle x_1,\cdots,x_n\rangle$ 的一个中介,当且仅当 ①对某个 i,有 $y=x_i$, $1<i<n$; ②或者序列 $\langle x_1\cdots x_i,y_1\cdots y_m,x_{i+1}\rangle\in\Phi$,并且对某个 j,有 $y=y_j$, $1\le j\le m$, $1<i<n$.

定义 4　排斥. 序列集 Φ 排斥序列 $\sigma=\langle x_1,\cdots,x_n\rangle$,当且仅当序列 $\langle y,x_n'\rangle\in\Phi$, y 是 Φ 中 σ 的一个中介.

中介用于表示推导距离[3],结合排斥处理继承例外的情况.

定理 1　如果序列集 Φ 冲突(排斥)序列 σ,则任一 $\Gamma\supseteq\Phi$ 也冲突(排斥)序列 σ.

证明:显然 $C(\Gamma)\supseteq C(\Phi)$,因此若 Φ 与 $\sigma=\langle x_1,\cdots,x_n\rangle$ 冲突,则存在 $\langle x_1,x_i'\rangle\in C(\Phi)$,则 $\langle x_1,x_i'\rangle\in C(\Gamma)$,那么 Γ 与 σ 冲突;若 Φ 排斥 σ,那么 Φ 包含序列 $\langle y,x_n'\rangle$, y 是 Φ 中 σ 的中介,则 Γ 也包含序列 $\langle y,x_n'\rangle$,并且 y 也是 Γ 中 σ 的中介,则 Γ 也排斥 σ.

定义 5　可继承. 序列 $\sigma=\langle x_1,\cdots,x_n\rangle$ 是序列集 Φ 可继承的,当且仅当 $n>2$, Φ 同时包含 $\langle x_1,\cdots,x_{n-1}\rangle$ 和 $\langle x_2,\cdots,x_n\rangle$, x_n 的符号不是"⊗",并且 Φ 既不与 σ 冲突,也不排斥 σ.

注意到 σ 是 Φ 可继承的,并不要求 $\sigma\in\Phi$.

我们称 Φ 是继承下封闭的,当且仅当对任何一个 Φ 的可继承序列 σ,都有 $\sigma\in\Phi$. 称 Γ 是 Φ 的扩展集,当且仅当 $\Gamma\supseteq\Phi$ 并且 Γ 是继承下封闭的. 称 Γ 在 Φ 中是接地的,当且仅当

$\Gamma - \Phi$的每个序列都是Γ中可继承的.

定义 6 Γ是Φ的接地扩展集,当且仅当Γ是Φ的一个扩展集并且Γ在Φ中是接地的.

定理 2 Γ是继承断言集Δ的接地扩展集,则$\langle x, y\rangle \in \Gamma$当且仅当$\langle x, y\rangle \in \Delta$.

证明:如果$\langle x, y\rangle \in \Delta$,则因为$\Delta \subseteq \Gamma$,有$\langle x, y\rangle \in \Gamma$;如果$\langle x, y\rangle \in \Gamma$,因为$\Gamma$是接地的,并且长度为 2 的序列在$\Gamma$中不是可继承的,而$\Gamma - \Phi$的每个序列都是可继承的,则有$\langle x, y\rangle \in \Delta$.

对序列$\sigma = \langle x_1, \cdots, x_n\rangle$,其相连子序列为$\sigma_o = \langle y_1, \cdots, y_m\rangle$, $m \le n$,并且存在一个$k < n - m$,使$y_1 = x_k$,且对从 0 到m的每个p,都有$y_p = x_{k+p}$.

定理 3 Γ是继承断言集Δ的接地扩展集,如果Γ包含序列$\langle x_1, \cdots, x_n\rangle$,则$\Gamma$也包含所有相连子序列$\langle x_i, \cdots, x_j\rangle$, $1 \le i < j \le n$.

证明:当$n = 2$时命题显然成立.假设对所有长度小于n $(n > 2)$的序列命题成立,由于Γ是Δ接地的,则序列$\langle x_1, \cdots, x_n\rangle$在$\Gamma$中是可继承的,所以$\Gamma$包含$\langle x_1, \cdots, x_{n-1}\rangle$和$\langle x_2, \cdots, x_n\rangle$,由归纳假设可得$\Gamma$包含序列$\langle x_i, \cdots, x_j\rangle$, $1 \le i < j \le n$.

定理 4 Γ是继承断言集Δ的接地扩展集,如果序列$\sigma = \langle x_1, \cdots, x_n\rangle$, $n > 2$且x_n的符号为"$\otimes$",则$\sigma \notin \Gamma$.

证明:Δ是序偶的集合, $n > 2$,显然$\sigma \notin \Delta$;Γ是Δ的接地扩展集,则$\Gamma - \Delta$的每个序列都是Γ中可继承的,而由可继承的定义,σ在任何序列集中都不是可继承的,即$\sigma \notin \Gamma - \Delta$,于是$\sigma \notin \Gamma$.

定理 5 Γ_a、Γ_b是继承断言集Δ的接地扩展集,$\Gamma_a \subseteq \Gamma_b$当且仅当$\Gamma_a = \Gamma_b$.

证明:设Γ_a、Γ_b是Δ的接地扩展集且$\Gamma_a \subseteq \Gamma_b$,假设$\Gamma_b - \Gamma_a \ne \varnothing$,序列$\sigma = \langle x_1, \cdots, x_n\rangle$是$\Gamma_b - \Gamma_a$中长度最短的序列,由定理 3, $\langle x_1, \cdots, x_{n-1}\rangle$和$\langle x_2, \cdots, x_n\rangle \in \Gamma_b$,因为$\langle x_1, \cdots, x_{n-1}\rangle$和$\langle x_2, \cdots, x_n\rangle$长度小于$n$,因此$\langle x_1, \cdots, x_{n-1}\rangle$、$\langle x_2, \cdots, x_n\rangle \in \Gamma_a$.由定义 5,则$\Gamma_b$不与$\sigma$冲突也不排斥$\sigma$,则$\Gamma_a$也不与$\sigma$冲突或不排斥$\sigma$(否则由定理 1 可得$\Gamma_b$与$\sigma$冲突或排斥$\sigma$),因此$\sigma$在$\Gamma_a$中也是可继承的,由于$\Gamma_a$是继承下封闭的,因此$\sigma \in \Gamma_a$,与假设冲突.因此$\Gamma_b - \Gamma_a = \varnothing$, $\Gamma_a = \Gamma_b$.

定理 6 Γ是继承断言集Δ的接地扩展集,如果$\langle x_1, \cdots, x_n\rangle \in \Gamma$,则从$x_1$到$x_{n-1}$都是"+"号.

证明:Δ是形式良好的序偶集合,$\langle x_1, x_2\rangle \in \Delta$中的$x_1$都是"+"号.假设命题对所有长度小于$n(n > 2)$的序列成立,如果$\langle x_1, \cdots, x_n\rangle \in \Gamma$,则$\langle x_1, \cdots, x_n\rangle$是$\Gamma$中可继承的,由定理 3, $\langle x_1, \cdots, x_{n-1}\rangle$、$\langle x_2, \cdots, x_n\rangle \in \Gamma$,由归纳假设,命题成立.

推论 1 ISA 无环的继承断言集Δ的接地扩展集Γ是有限的.

证明:因为任一$\langle x_1, \cdots, x_n\rangle \in \Gamma$,从$x_1$到$x_{n-1}$都是"+"号,若$x_i = x_j$, $1 \le i < j < n$,则$\langle x_i, \cdots, x_j\rangle$构成 ISA 环,因此没有一个"+"号单词会在一个序列中重复出现.由于单词集是有限的,则Γ也是有限的.

定义 7 一致的.序列集Φ是一致的当且仅当它不与它任何元素冲突.

定理 7 继承断言集Δ的接地扩展集Γ是一致的,当且仅当Δ是一致的.

证明:如果Δ不一致,因为$\Gamma \supseteq \Delta$,所以Γ也是不一致的;如果Γ不一致,则存在序列$\sigma_1 = \langle x_1, \cdots, x_n\rangle$、$\sigma_2 = \langle y_1, \cdots, y_m\rangle \in \Gamma$,其中$x_1 = y_1$且对某个$i$有$y_m = x_i'$, $1 \le i \le n$,由定理 3 有$\sigma_3 = \langle x_1, \cdots, x_i\rangle \in \Gamma$,由于$\Gamma$同时与$\sigma_2$和$\sigma_3$冲突,因此他们在$\Gamma$中都是不可继承的.因为$\Gamma$是$\Delta$的接地扩展集,$\Gamma - \Delta$的序列都是可继承的,于是必有$\sigma_2$、$\sigma_3 \in \Delta$,则$\Delta$也是不一致的.

定义 8 最小元.对指定集合Γ,当且仅当不存在单词y,有$\langle y, x\rangle$存在时,x是Γ的最小元.

定理 8 Γ_a、Γ_b是 ISA 无环的继承断言集Δ的接地扩展集,$\Gamma_a \ne \Gamma_b$,则序列集$\Gamma_a \cup \Gamma_b$是不一致的.

证明:因为$\Gamma_a \ne \Gamma_b$,由定理 5 可知$\Gamma_b - \Gamma_a \ne \phi$,设$\sigma_1 = \langle x_1, \cdots, x_{n-1}, x_n\rangle \in \Gamma_b - \Gamma_a$,且$x_{n-1}$是$\Gamma_b - \Gamma_a$中序列准终结单词集中的一个最小元.由定理 2 可得$n > 2$,定理 4 可得x_n的符号不是"$\otimes$".如果$\sigma_2 = \langle x_1, \cdots, x_{n-1}\rangle$、$\sigma_3 = \langle x_2, \cdots, x_n\rangle$则既属于$\Gamma_b$又属于$\Gamma_a$.由于$\Gamma_a$是接地扩展,则$\sigma_1$或者与$\Gamma_a$冲突或者被$\Gamma_a$排斥.若$\sigma_1$与$\Gamma_a$冲突,那么$\Gamma_a \cup \Gamma_b$是不一致的.若$\Gamma_a$排斥$\sigma_1$,存在一个中介$z$,且$\langle z, x_n'\rangle$在$\Gamma_a$中,$\Gamma_a$必包含序列$\sigma = \langle x_1, \cdots, x_i, z_1, \cdots, z_m, x_{i+1}\rangle$.其中$z$是$z_1, \cdots, z_m$之一,且$1 \le i < n - 1$.但这就有$\langle z_m, \cdots, z_{n-1}\rangle$,与前提$x_{n-1}$是序列的最小准终结单词且这个序列只存在于一个扩展中是矛盾.则σ_1是Γ_a中可继承的.由于Γ_a是接地扩展,于是$\sigma_1 \in \Gamma_a$,这同假设矛盾,于是σ_1同Γ_a冲突.因此$\Gamma_a \cup \Gamma_b$不是一致的.

结论集包含形如$\langle +a, +b\rangle$、$\langle +a, -b\rangle$、$\langle +a, \oplus b\rangle$和$\langle +a, \otimes b\rangle$的元素.单词通过继承得到的"包含的属性"已经由结论集表示出来,但"包含的属性"本身可能具有的继承关系却尚未被表示出来.

定义 9 扩展结论集.序列集Φ的扩展结论集$E(\Phi)$,定义为①$C(\Phi) \subseteq E(\Phi)$;②如果$\langle x, \oplus a\rangle$、$\langle +a, +b\rangle \in C(\Phi)$,且$\langle x, \otimes b\rangle \notin E(\Phi)$, $\langle x, +b\rangle \notin E(\Phi)$, $\langle x, -b\rangle \notin E(\Phi)$,则$\langle x, \oplus b\rangle \in E(\Phi)$.

一般而言,类包含属性,也包含该属性的父亲属性.但属性继承的例外会对这产生影响.这可能产生下述不相容情况.

定义 10 我们称扩展结论集是相容的,当且仅当若$\langle +a, \oplus b\rangle$、$\langle +a, \oplus c\rangle$、$\langle +c, +b\rangle$、$\langle +b, +d\rangle$、$\langle +c, -d\rangle \in C(\Phi)$,则必有$\langle +a, \oplus d\rangle$或$\langle +a, \otimes d\rangle \in C(\Phi)$,否则就是不相容的.

定义 11 无二义.Φ是无二义的,当且仅当它只存在一个接地扩展集且其扩展结论集是相容的.

结合定理 8 可以知道,二义产生的一个原因是因为一对互相冲突的序列在Δ的一个扩展集中都是可继承的.两个接地扩展集是分别选择一个序列扩展得到的.

定义 12 假冲突.有序列$\sigma = \langle x_1, \cdots, x_n\rangle$, $n > 2$且$x_n = \otimes a$,序列集Φ与序列σ假冲突当且仅当$\langle x_1, x_k\rangle \in C(\Phi)$,其中$x_k = \oplus a$.

推论 2　如果序列集 Φ 与序列 σ 产生假冲突,那么 Φ 与 σ 也产生冲突.

证明:序列集 Φ 与序列 σ 产生假冲突,则对 $\sigma=\langle x_1, \cdots, x_n\rangle$, $n>2$ 且 x_n 的符号是 $\otimes$,有 $\langle x_1, x_k\rangle\in C(\Phi)$, $x_k=\oplus a$,则 Φ 与 σ 冲突.

定理 9　Γ_a、Γ_b 是 ISA 无环的继承断言集 Δ 的接地扩展集,序列集 $\Gamma_a\cup\Gamma_b$ 不与它任何元素产生假冲突.

证明:由定理 4,对任一序列 $\sigma=\langle x_1, \cdots, x_n\rangle$, $n>2$ 且 x_n 的符号为"$\otimes$",则 $\sigma\notin\Gamma_a$、$\sigma\notin\Gamma_b$. 则 $\sigma\notin\Gamma_a\cup\Gamma_b$. 于是 $\Gamma_a\cup\Gamma_b$ 不与它任何元素产生假冲突.

结合定理 8 和 9 可以发现,假冲突的两个序列 $\sigma_1=\langle x_1, \cdots, x_n, \otimes a\rangle$, $n>1$ 和 $\sigma_2=\langle x_1, \cdots, \oplus a\rangle$,在传统继承理论中,分别被表示为 $\sigma_1'=\langle x_1, \cdots, x_n, -a\rangle$, $n>1$ 和 $\sigma_2'=\langle x_1, \cdots, +a\rangle$,他们互相冲突,且使 Δ 产生多个接地扩展. 但在扩展继承理论中,假冲突不是二义性产生的原因. 如图 1,在原来继承系统中,产生两个接地扩展,是二义的;而在扩展继承系统中,仅产生一个接地扩展,是无二义的.

二义性产生的另一个原因是类同时包含一对存在继承关系的属性,并且子属性不继承他们的某个祖先,此时,我们不能简单的断定类是否包含该祖先属性. 我们使用约束条件集 C 表示不兼容(IsNotCompatibleWith)关系. 用于检测不兼容关系的约束条件组成的集合,称为存在约束条件集合,记为 C_E. 由于经典逻辑(一阶谓词逻辑)难以表示继承的例外,因此我们使用带模态词 M(M 表示逻辑一致或相容的)的模态逻辑来表示约束条件. 我们使用到的两个谓词是 ISA(X, Y)表示 X 是 Y 和 CONTAIN(X, Y)表示 X 包含 Y. 约束条件由这两个谓词构成的一阶模态谓词逻辑的子集 L^M 表示. 使用约束条件的另一个目的,是为了加速知识的重用过程. 这种用途的约束条件组成的集合称为容器约束条件集合,记为 C_C. 若元素 a 是元素 e 的组成部分,即 e 包含 a,则称元素 e 为元素 a 的容器. 若关于容器 e 及其所有组成部分的谓词不能满足元素 a 的容器约束条件集合,则 a 自动不在 e 中出现,即 e 最终不包含 a. 利用这类约束条件,可以减少知识工程师裁剪领域模型的工作量,加速知识重用过程. 与此同时,这类约束条件也可用于处理扩展结论集的不相容情况.

记 a 关于序列集 Φ 的有用结论集 $UC(a,\Phi)=\{\langle +a, +b\rangle \mid \langle +a, +b\rangle\in E(\Phi)\}\cup\{\langle +a, \oplus b\rangle \mid \langle +a, \oplus b\rangle\in E(\Phi)\}$. a 的存在约束条件的集合为 $a.c$. a 关于序列 $\sigma=\langle +a, \cdots, +b\rangle$ 或 $\sigma=\langle +a, \cdots, \oplus b\rangle$ 的存在约束条件集合(Constraints of Token with respect to Sequence),记为 $\mathrm{CTS}(a,\sigma)=a.c\cup\cdots\cup b.c$.

定义 13　可满足的. 序列 $\sigma=\langle +a, \cdots, +b\rangle$ 在序列集 Φ 中是可满足的,当且仅当 a 在 Φ 中的扩展结论集 $E(a,\Phi)$ 使 $\mathrm{CTS}(a,\sigma)$ 中所有约束条件都成立.

在扩展继承理论中因"IsNotCompatibleWith"而导致的冲突,在扩展继承理论中,对应的序列将是不可满足的. 更多这方面的讨论见文献[8]. 下文讨论的继承系统,我们要求它包含的所有序列都是可满足的.

3　本体的继承

文献[1]使用本体表示领域模型. 本体是一个包含本体、类、属性、方法和关系的实体,而关系包括了本体与本体、本体与对象、对象与对象三类关系. 我们使用允许例外的继承机制将本体组成继承系统,并证明这种做法不削弱原来本体的表示能力. 再将继承系统映射到扩展继承理论,提出保持本体继承系统一致性的算法. 并探讨了关系、关系参数受本体继承影响而具有的特殊性质.

3.1　本体的表示

为了更好地研究本体允许例外的多继承机制. 我们对文献[1]定义的本体表示形式进行下述转换.

定义 14　无环塞泊图的并($\cup_{ac}$)、交($\cap_{ac}$)、差($-_{ac}$)设 $C_1=(D_1, P_1, U_1, V_1, H_1)$, $C_2=(D_2, P_2, U_2, V_2, H_2)$,则 $C_1\cup_{ac}C_2=(D_1\cup D_2, P_1\cup P_2, U_1\cup U_2, V_1\cup V_2, H_1\cup H_2)$, $\cup$是集合的并运算;$C_1\cap_{ac}C_2=(D_1\cap D_2, P_1\cap P_2, U_1\cap U_2, V_1\cap V_2, H_1\cap H_2)$, $\cap$是集合的交运算;$C_1-_{ac}C_2=(D_1-D_2, P_1-P_2, U_1-U_2, V_1-V_2, H_1-H_2)$, $-$是集合的差运算.

对所有 $1\le i\le n$, $X_i\in F$,则将 $X_1\cup_{ac}X_2\cup_{ac}\cdots\cup_{ac}X_n$,记为 $\bigcup_{X_i\in F}{}_{ac}X_i$ 对所有 $1\le i\le n$, $X_i\in F$,则将 $X_1\cap_{ac}X_2\cap_{ac}\cdots\cap_{ac}X_n$ 记为 $\bigcap_{X_i\in F}{}_{ac}X_i$

我们使用 $\Psi.D$ 指代 $\Psi=(D, P, U, V, H)$ 中的 D, $\Psi.P$ 指代 P,其他类似.

现在,我们使用三元组 $O=(F, E, I)$ 表示本体. 其中,F 称为直接祖先本体集合,是指向本体 O 直接祖先本体的指针集合,即如果 $O_2\in O_1.F$,则不存在 $O_3\in O_1.F$ 使 $O_2\in O_3.F$; E 称为例外部分,是本体 O 继承直接祖先本体,但由于例外而删除的部分;I 称为增加部分,是本体 O 对比直接祖先本体新增的部分. E 和 I 都是有限嵌套的无环塞泊图,并且 $I\cap_{ac}E=\varnothing$.

定理 10　记使用本体 $\Psi=(D, P, U, V, H)$ 的表示法得到的所有本体的集合为 Ω_1;使用 $O=(F, E, I)$ 的表示法得到的所有本体的集合为 Ω_2,则 $\Omega_1=\Omega_2$.

证明:对 $\forall\Psi_1=(D_1, P_1, U_1, V_1, H_1)\in\Omega_1$, $\exists O_1=(F_1, E_2, I_2)\in\Omega_2$,其中 $F_1=E_1=\phi$, $I_1=\Psi_1$,使 Ψ_1 和 O_1 表示相同的本体,所以 $\Omega_1\subseteq\Omega_2$. 而对 $\forall O_2=(F_2, E_2, I_2)\in\Omega_2$, $\exists\Psi_2=(D_2, P_2, U_2, V_2, H_2)=(\bigcup_{X_i\in F_2}{}_{ac}X_i -_{ac} E_2)\cup_{ac}I_2$,使 Ψ_2 和 O_2 表示相同的本体,则 $\Omega_2\subseteq\Omega_1$. 所以 $\Omega_2=\Omega_1$.

定义 19　继承将 $O=(F, E, I)$ 转换为 $\Psi=(D, P, U, V, H)=(\bigcup_{X_i\in F}{}_{ac}X_i -_{ac} E)\cup_{ac}I$ 的过程称为继承.

由定理 10,我们可以使用三元组(F, E, I)来表示本体,而不削弱或增强本体的表示能力. 而且,每个 $O_1=(F_1, E_1, I_1)$ 都可以通过继承唯一表示为一个 $\Psi_1=(D_1, P_1, U_1, V_1, H_1)$ 的本体. 但对 $\Psi_1=(D_1, P_1, U_1, V_1, H_1)$,却可以表示为 $O_1=(F_1, E_1, I_1)$, $O_2=(F_2, E_2, I_2)$ 并且 $O_1\ne O_2$,使 O_1 和 O_2 继承都得到 Ψ_1. 这是五元组表示法不利于讨论本体允许例外的多继承机制的原因.

3.2 本体继承系统

给定本体 $O_1=(F_1,E_1,I_1)$，对每个 $x\in F_1$，产生继承断言 $\langle +O_1,+x\rangle$，每个 $x\in E_1$，产生继承断言 $\langle +O_1,\otimes x\rangle$，每个 $x\in I_1$，产生继承断言 $\langle +O_1,\otimes x\rangle$. 使用这种方式产生本体集中所有本体的继承断言，由这些断言组成的系统，称为本体继承系统，记为 Ω.

令 $\prod_{AM}$ 表示属性、方法结点名字组成的集合，$\prod_R$ 表示关系结点名字组成的集合，$\prod_{OBJ}$ 表示对象结点名字组成的集合，$\prod_{ONT}$ 表示本体结点名字组成的集合，单词集 $\Theta=\{+,-,\oplus,\otimes\}\times(\prod_{ONT}\cup\prod_{AM}\cup\prod_{OBJ}\cup\prod_R)$，在本体继承系统中，继承断言共存在 15 种形式良好的序偶，他们的直观意义如表 3 所示. 令 ontp, ontq 表示本体的名字，objx, objy 表示对象的名字，am, an 表示属性或方法的名字，r 表示关系的名字.

表 3 本体单词序偶及其意义

序偶	直观意义	继承系统中的连接
$\langle(ontp,+ontq\rangle$	ontp 是 ontq	ontp ISA ontq
$\langle(ontp,-ontq\rangle$	Ontp 不是祖先 ontq	ontp ISNOTA ontq
$\langle(ontp,\oplus am\rangle$	ontp 包含 am	ontp Contain + am
$\langle(ontp,\oplus objx\rangle$	ontp 包含 objx	ontp Contain + objx
$\langle(ontp,\oplus ontq\rangle$	ontp 包含 ontq	ontp Contain + ontq
$\langle(ontp,\oplus r\rangle$	ontp 包含 r	ontp Contain + r
$\langle(ontp,\otimes am\rangle$	ontp 不包含祖先的 am	ontp Contain - am
$\langle(ontp,\otimes objx\rangle$	ontp 不包含祖先的 objx	ontp Contain - objx
$\langle(ontp,\otimes ontq\rangle$	ontp 不包含祖先的 ontq	ontp Contain - ontq
$\langle(ontp,\otimes r\rangle$	ontp 不包含祖先的 r	ontp Contain - r
$\langle(objx,+objy\rangle$	objx 是 objy	objx ISA objy
$\langle(objx,-objy\rangle$	Objx 不是祖先 objy	objx ISNOTA objy
$\langle(objx,\oplus am\rangle$	objx 包含 am	objx Contain + am
$\langle(objx,\otimes am\rangle$	objx 不包含祖先的 am	objx Contain - am
$\langle +am,+an\rangle$	am 是 an	am ISA an

本体是本体/类/属性的结构，同类/属性结构相似，使用扩展继承理论刻画本体，可以得到一个允许例外的多继承本体继承系统. 对本体继承系统，我们关心本体通过继承得到的本体、类、方法、属性和关系. 我们用包含断言 $CONTAIN(X,Y)$ 表示 X 包含 Y. 如果 $x=(D,P,U,V,H)$，并且 $y\in D\cup P\cup U\cup V\cup H$. 则 $CONTAIN(x,y)=true$. x 的包含断言集定义为 $\{\langle +x,\oplus y\rangle\mid CONTAIN(x,y)=true\}$.

定理 11 本体集中任一本体的包含断言集，是本体继承系统 Ω 的接地扩展集 Γ 的结论集 $C(\Gamma)$ 的子集.

证明：对 $\forall O_1=(F_1,E_1,I_1)$，若 $F_1=\varnothing$，称 F_1 的最大嵌套层数等于 0，若 $F_1\neq\varnothing$，且对每个 $f\in F_1$，f 的最大嵌套层数都等于 0，则称 F_1 的最大嵌套层数等于 1. 若对所有 $f\in F_1$，f 的最大嵌套层数的最大值等于 n，则称 F_1 的最大嵌套层数等于 $n+1$. 下面按 F_1 的最大嵌套层数进行归纳证明.

若最大嵌套层数等于 0，即 $F_1=\varnothing$，则对 $\forall Y$，若 $CONTAIN(O_1,Y)$ 成立，必有 $Y\in I_1$. 而对每个 $Y\in I_1$，继承系统产生继承断言 $\langle +O,\oplus Y\rangle$，显然 $\langle +O,\oplus Y\rangle\in C(\Gamma)$，命题成立.

若最大嵌套层数等于 1，则对 $\forall Y$，若 $CONTAIN(O_1,Y)$ 成立，必有 $Y\in I_1$ 或者存在 $f\in F_1$，使 $CONTAIN(f,Y)$ 成立并且 $Y\notin E_1$. 显然对每个 $Y\in I_1$，有 $\langle +O,\oplus Y\rangle\in C(\Gamma)$；若 $CONTAIN(f,Y)$ 成立，则必有 $\langle +f,\oplus Y\rangle\in C(\Gamma)$，而对每个 $f\in F_1$ 继承系统产生断言 $\langle +O,+f\rangle$，由于不存在 $\langle +O,\otimes Y\rangle$，序列 $\langle +O,+f,\oplus Y\rangle$ 是 Γ 中可继承的，于是 $\langle +O,+f,\oplus Y\rangle\in\Gamma$，$\langle +O,\oplus Y\rangle\in C(\Gamma)$，命题成立.

假设最大嵌套层数小于 n 时命题成立，当最大嵌套层数等于 n 时，则对 $\forall Y$，若 $CONTAIN(O_1,Y)$ 成立，必有 $Y\in I_1$ 或者存在 $f_1\in F_1$，使 $CONTAIN(f_1,Y)$ 成立并且 $Y\notin E_1$. 显然对每个 $Y\in I_1$，有 $\langle +O,\oplus Y\rangle\in C(\Gamma)$；若 $CONTAIN(f_1,Y)$ 成立，由归纳假设，若 f_1 嵌套层数小于 n，则有 $\langle +O,\oplus Y\rangle\in C(\Gamma)$. 若 f_1 嵌套层数等于 n，即 $\langle +O,+f_1,\cdots,+f_n\rangle\in\Gamma$ 并且 $\langle +f_n,\oplus Y\rangle\in\Gamma$. 于是由 $\langle +f_1,\cdots,+f_n\rangle\in\Gamma$ 和归纳假设得到 $\langle +f_1,\cdots,+f_n,\oplus Y\rangle\in\Gamma$，由于 $\langle +f_1,\oplus Y\rangle\in C(\Gamma)$ 并且不存在 $\langle +O,\otimes Y\rangle$，序列 $\langle +O,+f_1,\cdots,+f_n,\oplus Y\rangle$ 是 Γ 中可继承的，于是 $\langle +O,+f_1,\cdots,+f_n,\oplus Y\rangle\in\Gamma$，$\langle +O,\oplus Y\rangle\in C(\Gamma)$，命题成立.

推论 3 本体集中任一本体的包含断言集，是本体继承系统 Ω 的接地扩展集 Γ 的扩展结论集 $E(\Gamma)$ 的子集.

由于 $E(\Gamma)\supseteq C(\Gamma)$，推论显然成立. 因为包含断言集并没有考虑本体包含的各个部分受他们自身继承的影响，所以，包含断言集断言的数量，比扩展结论集的少. 包含断言集是扩展结论集的子集，是继承影响本体表示内容的一个重要表现.

3.3 扩展结论集及一致性、无二义性检测

为了解决假冲突问题，建立一致无二义的本体继承系统. 我们在继承系统断言集发生变化时，利用下述上、下扫描算法，扫描并验证系统的变化，则不但可获得变化相关结点的结论集、扩展结论集，还可保持继承系统的一致性和无二义性. 下面的算法中，结点 x 指 $\prod_{ONT}\cup\prod_{AM}\cup\prod_{OBJ}\cup\prod_R$ 中的任意元素，Cgen 是用于检测结点，产生结论集的队列，而 Egen 则是用于产生扩展结论集的集合.

上扫描算法：

输入：$x\in\prod_{ONT}\cup\prod_{AM}\cup\prod_{OBJ}\cup\prod_R$.

输出：结论集 $C_x=\{\langle x,y\rangle\mid\langle x,y\rangle(C(\Gamma)\}$；

扩展结论集 $E_x=\{\langle x,y\rangle\mid\langle x,y\rangle(E(\Gamma)\}$.

```
Upscan(node x){
    对每个跟 x 通过 Contain-边连接起来的结点 y{
   置 y 的标记为‘⊗’;
   将〈+x,⊗y〉添加到继承结论集 Ix 中;}
 x 进 Igen 队列;
 while(Igen 队列非空){
   取队头元素到 z;
   对每个跟 z 通过 ISA 边连接起来的结点 y  {
     if(y 的标志为空){
       y 进 Igen 队列;
       置 y 的标记为‘+’;
     将〈+x,+y〉添加到继承结论集 Ix 中;
       将 z 添加到 y 的标记来源集中;
     }else if (y 的标记是‘+’)
       将 z 添加到 y 的标记来源集中;
     else if (y 的标记是‘-’)  {
```

```
      if(x 继承 z 而排除了 y){
      置 y 的标记为'+';
      从继承结论集 I_x 中删除〈+x,-y〉;
      将〈+x,+y〉添加到继承结论集 I_x 中;
    y 的标记来源集等于{z};
      y 进 Igen 队列;
    }else 检测到冲突
  }else 检测到冲突}
 对每个跟 z 通过 ISNOTA 边连接起来的结点 y{
  if (y 的标志为空){
    置 y 的标记为'-';
    将〈+x,-y〉添加到继承结论集 I_x 中;
    将 z 添加到 y 的标记来源集中;
 }else if (y 的标记是'-')
    将 z 添加到 y 的标记来源集中;、 else if (y 的标记是
'+'){
    if (x 继承 z 而排除了 y){
      置 y 的标记为'-';
      从继承结论集 I_x 中删除〈+x,+y〉;
      将〈+x,-y〉添加到继承结论集 I_x 中;
      y 的标记来源集等于{z};
    }else 检测到冲突
  }else 检测到冲突}
 对每个跟 z 通过 Contain+边连接起来的结点 y{
  if (y 的标记是'+'或'-')
    检测到冲突;
  else if(y 的标记是'⊗'并且 z=x)
    检测到冲突;
  else {
    置 y 的标记为'⊕';
    将〈+x,⊕y〉添加到继承结论集 I_x 中;
    将 y 添加到 Cgen 集合中.
    }    }    }
结论集 C_x = I_x;
对 Cgen 集合中每个元素 y{
 将递归调用 Upscan(y)得到的结论集 C_y;
 对每个〈+y,+z〉或〈+y,⊕z〉∈C_y{
  if (z 的标记是空或'⊕'){
    置 z 的标记为'⊕';
    将〈+x,⊕z〉添加到结论集 C_x 中;
  }else 检测到冲突
 }    }
返回结论集 C_x;}
```

算法中"x 继承 z 而排斥了 y"的判断,就是对 y 标记来源集中每一元素 z',判断 z 是否都是序列〈$+x,\cdots,z',\pm y$〉的中介,若是,则〈$+x,\cdots+z,+y$〉排斥原继承序列〈$+x,\cdots,+z',-y$〉(或〈$+x,\cdots+z,-y$〉排斥序列〈$+x,\cdots,+z',+y$〉). 若 z 是序列〈$+x,\cdots,+z',+y$〉的中介,则存在一条从 z 到 z 的 ISA 链. 判断有向图中两结点间是否连通的问题可用标记传播算法实现.

分析算法的时间复杂度,需要注意的两个部分是上述的"x 继承 z 而排斥了 y"的判断和对每个 Egen 中元素的递归调用. 我们假设继承系统的结点数为 n,最长的 ISA 链长度为 p,最长的 Contain+链长度为 q. ISA 链由形如〈$+a,+b,+c\cdots$〉的序列表示,Contain+链则是由〈$+a,\oplus b$〉,〈$+b,\oplus c$〉…的序列组合成的连接表示. 由此可知,"x 继承 z 而排斥了 y"的判断,复杂度为 $O(n^p)$. 注意到,如果没有 p 的限制,则判断图中两结点是否连通的时间复杂度是 $O(n^n)$,而 $n \gg p$. 一般 p 是一个很小的数,远小于 100. Contain+链的长度,则决定了递归调用的次数. 也就是说,算法的最坏时间复杂度是 $O((n^p)^q) = O(n^{pq})$. 对本体继承系统,存在的包含关系有类包含属性方法和本体包含本体、类、属性方法和关系. 本体的嵌套包含,是 q 可能变大的唯一原因. 在现有知识表示体系中,领域的嵌套包含层数并不大,q 也远小于 100. 于是 pq 的数量级最多是 10^2. 同时,q、p 并不随 n 的增大而变化,他们依赖的是现有的知识体系结构. 因此,上扫描算法的最坏时间复杂度是 $O(n^k)$,其中 k 是一个数量级为 10^2 的常数. 也就是说,上扫描算法是一个多项式时间的算法.

下扫描算法:

```
  输入:x ∈ ∏_ONT ∪ ∏_AM ∪ ∏_OBJ ∪ ∏_R.
  输出:后代集 CxdSet = { y|〈+y,+x〉∈E(Γ)}.
  说明:CxdGen 队列是用于产生 CxdSet 的队列
  downscan(node x){
  x 进 CxdGen 队列;
  while (CxdGen 队列非空){
     取队头元素到 z;
  对每个通过 ISA 边连接到 z 的结点 y{
  if(y 的标志为空){
  调用 Upscan(y)得到扩展结论集 Ey;
     if〈+y,+x〉in Ey {
  置 y 的标志为"+";
  y 进 CxdGen 队列;
  将 y 添加进后代集 CxdSet 中;
          }else 置 y 的标志为"-";
  }    }    }    }
```

对于一致的原继承系统 Ω,增加或删除继承断言〈x,y〉后的得到继承系统 Ω',对其运行上扫描算法得到结点 x 的结论集 C_x、扩展结论集 E_x,检查是否一致无二义的;对 x 再运行下扫描算法,产生 x 后代结点的结论集 Cx_dSet,其间对 x 每个后代 y,产生相应的扩展结论集合 $E(y)$,并检查是否是一致无二义的. 由于产生并检查了扩展结论集 $E(\Gamma)$ 中与结点 x、x 的后代结点相关的所有序列,若 $E(\Gamma)$ 没有产生冲突,即其不与任何属于它的序列冲突,于是继承系统 Ω 也不产生冲突,保持了一致.

3.4 关系的继承

关系是本体的重要组成部分,它必然受到本体继承的影响. 在本体继承系统中,关系使用描述关系模式的关系结点表示,并且通过关系边指向关联的参数. 将一个关系中的名词用

参数(即位置信息)代替,可得到关系模式.如关系“经理管理员工”,其关系模式为“(1)管理(2)”.关系边则是一条带权值的有向边,边的权值表示关系边所指向的结点是关系模式中的第几个参数.关系参数沿继承传递的性质,可分为如下4类:

(1)参数固定,不沿继承传播.关系不受继承影响.我们给参数加标记 fixed 表示.

(2)参数可沿继承向上传递,即该参数对应关系边所指结点及其祖先结点都可具有该关系.我们给这类参数加标记 upTransferable 表示.

(3)参数可沿继承向下传递,即该参数对应关系边所指结点及其后代结点都可具有该关系.我们给这类参数加标记 downTransferable.这种性质是关系参数的缺省性质.

(4)参数根据条件沿继承传递,即只有该参数对应关系边所指结点及其祖先、后代结点满足某种条件时,他们才可以具有该关系.我们给这类参数加标记 onCondition:Condition.第二个 Condition 表示具体的条件.

对一个二元关系,除了其关系参数可以具备上述性质,关系本身还可以具备下列关系性质:自反的、反自反的、对称的、反对称的、传递的、泛函的.泛函的是指:若存在关系 $R(x,y)$、$R(x,z)$,则必有 $y=z$.其他性质表示的意义显然,这里不展开说明.

从扩展结论集得到的结果,利用关系参数和关系的性质,可进一步扩展产生所有符合条件的关系并显式表示.这里也可使用约束条件对产生的关系进行裁剪.

4 总结

本文结合允许例外的本体多继承机制,通过对传统继承理论的深入考察,发现并分析传统继承理论存在的假冲突问题,扩展该继承理论,证明传统继承理论的定理在扩展继承理论中仍然成立,也证明了一些新的定理和推论,解决了假冲突问题.然后使用允许例外的继承机制将本体组成继承系统,并证明这种做法不削弱原来本体的表示能力.再将继承系统映射到扩展继承理论,提出保持本体继承系统一致性的算法.并探讨了关系、关系参数受本体继承影响而具有的特殊性质.

允许例外的本体多继承机制的确立和一致性问题的解决,将能极大地促进领域本体的发展,从而推进软件工程向自动化发展的进程.而对文献[7]介绍的基于本体的知识库,一致性问题的解决则为知识推理提供了更坚实的基础.

参考文献:

[1] Lu R,Jin Z. Domain Modeling-Based Software Engineering[M]. Boston:Kluwer Academic Publishers,2000.

[2] Lu Ruqian,Jin Zhi. Formal ontology:foundation of domain knowledge sharing and reusing[J]. Journal of computer science and technology 2002,17(5):535 - 548.

[3] David S. Touretzky. The Mathematics of Inheritance Systems[M]. Pitman,London: Morgan Kaufmann publishers,1986.

[4] W A Woods. What's in a link:Foundations for semantic networks[A]. D G Bobrow and A Collins(Eds.):Representation and Understanding[C]. New York,Academic Press,1975. 35 - 82.

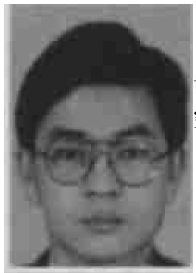
[5] Yair Wand,Veda C. Storey,Ron Weber. An ontological analysis of the relationship construct in conceptual modeling[J]. ACM Transactions on Database Systems,1999,12,24(4):494 - 528.

[6] Bunge M. Ontology II:A World of Systems. Treatise on Basic Philosophy[M]. 1977,4,New York,NY,D. Reidel Publishing Co. Inc.

[7] CAO Cungen,FENG Qiangze,GAO Ying,GU Fand,et al. Progress in the development of national knowledge infrastructure[J]. Journal of computer science and technology 2002,17(5):523 - 534.

[8] 明仲,蔡树彬,李师贤.使用约束条件支持领域本体的重用[J].计算机科学,2004,31(4):104 - 108. Ming Zhong,Cai Shubin,Li Shixian. Supporting reuse of domain ontology by using constraint[J]. Computer Science,2004,31(4):104 - 108(in Chinese).

作者简介:



明 仲 男,1967年出生于江西宁都,教授,博士,目前主要从事软件工程和本体论方面的研究. Email:mingz@szu.edu.cn

蔡树彬 男,1979年出生于广东汕头,博士研究生,主要研究方向为本体论和软件工程.

李师贤 男,1944年出生于江西,教授,博士生导师,主要研究方向为形式语义学和软件工程.

徐 晶 男,1981年出生于贵州,硕士研究生,主要研究方向为数据库和数据挖掘.