

# 从 ER 模式到 OWL DL 本体的语义保持的翻译

许卓明<sup>1,2)</sup> 董逸生<sup>2)</sup> 陆 阳<sup>1)</sup>

<sup>1)</sup>(河海大学计算机及信息工程学院 南京 210098)

<sup>2)</sup>(东南大学计算机科学与工程学院 南京 210096)

**摘 要** 提出了一种从 ER 模式到 OWL DL 本体的语义保持的翻译方法. 该方法在形式化表示 ER 模式的基础上, 建立 ER 模式和 OWL DL 本体之间精确的概念对应, 通过一个翻译算法按照一组预定义的映射规则实现模式翻译. 理论分析表明, 该方法是语义保持的和有效的; 算法实现和案例研究进一步证实, 完全自动的机器翻译是可实现的. 该文方法是原创性的, 为 Web 本体的开发以及数据库和语义 Web 之间语义互操作的实现开辟了一条有效途径.

**关键词** 模式翻译; ER 模式; OWL; 语义保持; 本体开发; 语义 Web

中图法分类号 TP311

## Semantics-Preserving Translation from ER Schema to OWL DL Ontology

XU Zhuo-Ming<sup>1,2)</sup> DONG Yi-Sheng<sup>2)</sup> LU Yang<sup>1)</sup>

<sup>1)</sup>(College of Computer and Information Engineering, Hohai University, Nanjing 210098)

<sup>2)</sup>(School of Computer Science and Engineering, Southeast University, Nanjing 210096)

**Abstract** The Semantic Web aims to create ontology-based and machine-processable Web content. Hence its success and proliferation depends on quickly and cheaply constructing Web ontologies that meet applications' needs. An Entity-Relationship (ER) schema captures the domain knowledge of the database application. Thus the translation from ER schema to OWL ontology is helpful for learning Web ontologies from databases and for developing data-intensive Semantic Web applications such as semantic annotation of database-based Web pages. In this paper, a method for semantics-preserving translation from ER schema to OWL DL ontology is presented. Based on the formalization of ER schema, the method establishes precise conceptual correspondences between an ER schema and an OWL DL ontology and realizes the schema translation via a translation algorithm which follows a set of predefined mapping rules. Theoretic analysis shows that the method is semantics-preserving and effective. Algorithmic implementation and case study demonstrates farther that the fully automated machine translation is realizable. The proposed method is original and will doubtless facilitate the development of Web ontologies and the realization of semantic interoperations between existing database applications and the Semantic Web.

**Keywords** schema translation; ER schema; OWL; semantics-preservation; ontology development; Semantic Web

## 1 引言

当前 Web 的简单基础结构虽降低了用户使用 Web 的技术门槛,却限制了 Web 潜能的充分发挥.通过扩充当前 Web,语义 Web(Semantic Web)<sup>[1]</sup>旨在创建机器可处理的 Web 资源(内容与服务等),以便在网络计算任务中为人类用户提供更好的机器辅助.本体(ontology)<sup>[2,3]</sup>在语义 Web 中发挥着关键作用;当前,标准化的 Web 本体语言 OWL<sup>[4]</sup>已由 W3C 发布.

从本体开发的角度来说,由于借助 OWL 本体编辑器(如 Protégé)手工开发本体是一个繁重和棘手的任务,极易导致知识获取的瓶颈<sup>[5]</sup>,因此,Web 本体的可获得性早已被学术界认为是制约语义 Web 成功的巨大挑战之一<sup>[6]</sup>.从现有知识源(如文本、词典、遗留知识库或本体、数据库模式等)获取领域知识、以(半)自动方式构造或改编本体——即所谓的本体学习(ontology learning)<sup>[5]</sup>,是开发本体的有效途径.权威综述<sup>[7]</sup>指出:当前本体学习的研究主要集中在如何利用自然语言处理技术从文本源提取知识,从数据库这种结构化数据(知识)源学习 Web 本体尚无成熟的方法.

从创建机器可处理 Web 内容的角度来说,由于现有的基于本体的语义标注(semantic annotation)工具都是针对内容固定的静态网页的<sup>[8]</sup>,但动态网站响应客户请求后当场从后台数据库产生的动态网页的数量远远超过静态网页的数量(400~550倍)<sup>[9]</sup>,且动态网站后台数据库大多是关系数据库(占 77.3%)<sup>[10]</sup>,因此,如何解决基于关系数据库的动态网页的语义标注问题,已成为制约语义 Web 成功的另一巨大挑战<sup>[6]</sup>.我们认为,拥有一个能在概念上捕获动态网站底层关系数据库论域知识的本体是解决这个问题的前提.

综上,不管从哪个角度,公共的问题是如何提取关系数据库中隐含的领域知识,并将其转换成本体.我们知道,数据库概念模式如实体-联系(Entity-Relationship, ER)模式<sup>[11]</sup>描述了数据库的论域;即使遗留系统未保留 ER 模式,也可借助数据库建模 CASE 工具通过逆向工程半自动地恢复 ER 模式<sup>[12]</sup>.因此,研究从 ER 模式到 OWL 本体的翻译方法有助于 Web 本体的开发以及动态网页语义标注的实现.

一般而言,从一个模式翻译成另一个模式可能是语义减弱的(semantics-losing),如我们熟悉的“从 ER 模式到关系模式”的转换;也可能是语义保

持的(semantics-preserving),如 Calvanese 等人提出的“从 ER 模式到 ALUNI 描述逻辑知识库”的翻译<sup>[13]</sup>.语义保持的翻译是知识保持的<sup>[12,13]</sup>,即目标模式捕获了源模式的全部知识,因而是理想的翻译.

本文第 2 节给出 ER 模式与 OWL DL 本体(OWL DL 是 OWL 的一种子语言,见下文)的形式语法与语义;第 3 节提出一种翻译算法,并形式证明翻译是语义保持的;第 4 节简要介绍算法实现及全自动的机器翻译原型工具 ER2WO,并给出一个典型验证案例;最后是总结.

## 2 翻译源与目标的语法与语义

### 2.1 ER 模式

早期用描述逻辑(Description Logics, DL)<sup>[14]</sup>表示 ER 模式并进行推理(如校验实体可满足性等)的研究已引入了一些 ER 模式形式化方法.这里,我们以 Calvanese 等人提出的 ER 模式一阶形式化定义<sup>[13]</sup>为基础,增加了实体的唯一属性的概念;限制了实体间 ISA 联系的特征——任一实体作为低层(子)实体只参与一个 ISA 联系中(即单继承的)且特殊化关系是非循环的(即无圈的),使得实体间的特殊化关系呈严格的层次结构(即“树”而非“格”结构);修改了基约束的定义——为表示实体参与联系的每个角色均定义了基约束.这种形式化不仅包含了当前大多数工业 CASE 工具(如 PowerDesigner)支持的 ER 模型不同变种中最重要的公共特性,且具有良好定义的语义,有利于在 ER 模式与 OWL 本体间建立精确的概念对应.

**定义 1.** 对有限集  $X$  和  $Y$ ,把从  $X$  的子集到  $Y$  的一个函数称为一个  $X$  标示  $Y$  元组( $X$ -labeled tuple over  $Y$ ).把这个将  $x_i \in X$  映射为  $y_i \in Y (i=1, 2, \dots, k)$  的被标示元组  $T$  表示为  $[x_1: y_1, \dots, x_k: y_k]$ .也用  $T[x_i]$  表示  $y_i$ ,称其为  $T$  的  $x_i$ -分量.

**定义 2.** 一个 ER 模式是一个五元组  $S=(L_S, isa_S, att_S, rel_S, card_S)$ ,其中:

①  $L_S = E_S \cup A_S \cup R_S \cup U_S \cup D_S$  是一个有限字母表(alphabet),其划分为两两不相交的,即一个实体(entity)符号的集合  $E_S$ ;一个属性(attribute)符号的集合  $A_S$ ;一个联系(relationship)符号的集合  $R_S$ ;一个角色(role)符号的集合  $U_S$ ;一个域(domain)符号的集合  $D_S$ .每个域符号  $D \in D_S$  有一个相关的预定义基本域(basic domain)  $B(D)$ ,且假定各基本域是互不相交的.进一步地,可假定任意两个与不同实体符号关联的属性符号子集是不相交(实际

中,总可通过属性符号改名来满足这个要求)。

②  $isa_S \subseteq E_S \times E_S$  是  $E_S$  上的一个单射 (injective)、非循环 (acyclic) 二元关系,用于表示两个实体间的 ISA 联系 (ISA relationship)。

③  $att_S$  是一个将  $E_S$  上每个实体符号映射为一个  $A_S$  标示  $D_S$  元组的函数。进一步地,对一个实体  $E \in E_S$ ,且  $att_S(E) = [\dots, A; D, \dots]$ ,如果属性  $A \in A_S$  满足:对  $E$  的每个实例,  $A$  在基本域  $B(D)$  中有唯一的取值,那么称  $A$  是实体  $E$  的一个唯一属性 (unique attribute) 或单属性键 (single-attribute key)。函数  $att_S$  用于表示实体的 (唯一) 属性。为简便,假定属性均是单值和强制的。实际上,其它情形也是很容易处理的。

④  $rel_S$  是一个将  $R_S$  上每个联系符号映射为一个  $U_S$  标示  $E_S$  元组的函数。不失一般性,假定:(a) 每个角色是针对一个实体参与某个特定联系定义的;(b) 对每个角色  $U \in U_S$ ,有一个实体  $E \in E_S$  和一个联系  $R \in R_S$  满足  $rel_S(R) = [\dots, U; E, \dots]$ 。函数  $rel_S$  将一组实体与角色关联到一个联系,并隐含地决定了联系的元 (arity)。

⑤  $card_S$  是一个从  $E_S \times R_S \times U_S$  到  $N_0 \times (N_1 \cup \{\infty\})$  (其中,  $N_0$  表示非负整数,  $N_1$  表示正整数,  $\infty$  表示无穷大) 的、且满足下列条件的函数:对每个关联实体  $E \in E_S$  和联系  $R \in R_S$  的角色  $U \in U_S$ ,且  $rel_S(R) = [\dots, U; E, \dots]$ ,总是定义  $card_S(E, R, U) = (mincard_S(E, R, U), maxcard_S(E, R, U))$ ,其第一个分量  $mincard_S(E, R, U)$  表示实体  $E$  的一个实例通过角色  $U$  参与联系  $R$  的最小次数 (0 表示无约束),第二个分量  $maxcard_S(E, R, U)$  表示相应的最大次数 ( $\infty$  表示无约束)。函数  $card_S$  用于表示一个关联实体和联系的特定角色的基约束 (cardinality constraints)。

一个 ER 模式的形式语义通过数据库状态 (database state)<sup>[13]</sup> 来给定,即通过指定哪些数据库状态与这个 ER 模式所表示的信息结构相一致来给定。只有当一个数据库状态满足 ER 模式中包含的所有完整性约束时,即是合法数据库状态 (legal database state)<sup>[13]</sup> 时,其才被认为是可接受的。

**定义 3.** ER 模式  $S = (L_S, isa_S, att_S, rel_S, card_S)$  的一个数据库状态  $B = (\Lambda^B, \Theta^B)$  由一个非空有限集  $\Lambda^B$  (假定与所有基本域是不相交的) 和一个映射函数  $\Theta^B$  来指定,其中,函数  $\Theta^B$

① 将每个域符号  $D \in D_S$  映射为相应的基本域  $B(D)$ ;

② 将每个实体  $E \in E_S$  映射为集合  $E^B \subseteq \Lambda^B$ ;

③ 将每个属性  $A \in A_S$  映射为偏函数 (partial function) 的集合  $A^B \subseteq \Lambda^B \times \bigcup_{D \in D_S} B(D)$ ;

④ 将每个联系  $R \in R_S$  映射为一个  $U_S$  标识  $\Lambda^B$  元组的集合,记为  $R^B$ 。

集合  $E^B, A^B, R^B$  的元素分别称为实体  $E$ , 属性  $A$ , 联系  $R$  的实例 (instance)。

**定义 4.** 设  $B = (\Lambda^B, \Theta^B)$  是 ER 模式  $S = (L_S, isa_S, att_S, rel_S, card_S)$  的一个数据库状态。当如下完整性约束条件满足时,称  $B$  是  $S$  的一个合法数据库状态:

(1) 对每对实体  $E_1, E_2 \in E_S$  且  $E_1 isa_S E_2$ , 满足  $E_1^B \subseteq E_2^B$ ;

(2) 对每个实体  $E \in E_S$  且  $att_S(E) = [A_1; D_1, \dots, A_h; D_h]$ , 满足  $\forall e \in E^B$ , 存在唯一的  $a_i = \langle e, d_i \rangle \in A_i^B, d_i \in B(D_i), i = 1, 2, \dots, h$ ;

(3) 对每个联系  $R \in R_S$  且  $rel_S(R) = [U_1; E_1, \dots, U_k; E_k]$ , 满足:  $\forall r \in R^B$ , 有  $r = [U_1; e_1, \dots, U_k; e_k]$ ,  $e_i \in E_i^B, i = 1, 2, \dots, k$ ;

(4) 对每个关联实体  $E \in E_S$  和联系  $R \in R_S$  的角色  $U \in U_S$  且  $rel_S(R) = [\dots, U; E, \dots]$ , 满足  $\forall e \in E^B, mincard_S(E, R, U) \leq \# \{r \in R^B \mid r[U] = e\} \leq maxcard_S(E, R, U)$  (其中  $\# \{\dots\}$  表示集合  $\{\dots\}$  的基数)。

进一步地,当且仅当一个 ER 模式存在合法数据库状态时,称这个 ER 模式是可满足的 (satisfiable)。

事实上,定义 4 给出了 ER 模式的一个合法数据库状态应满足的语义约束。条件 (1)~(4) 依次为构造子 ISA 联系、实体的属性、联系、角色及基约束的语义约束。值得强调的是,数据库状态的定义反映了数据库中的一个惯常假设——数据库状态是有限结构。事实上,尽管各基本域  $B(D), D \in D_S$  不必是有限的,但对一个 ER 模式  $S$  的每个合法数据库状态  $B$ , 在实际中仅对来自于各基本域的一个有限值集 (作为属性的取值) 是真正感兴趣的。我们可将这个有限值集定义为数据库状态的活动域 (active domain)<sup>[13]</sup>  $\Delta_{act}^B$ , 即  $\Delta_{act}^B = \{d \in B(D) \mid D \in D_S \wedge \exists A \in A_S, e \in \Lambda^B. \langle e, d \rangle \in A^B\}$ 。据此,我们可约定:在合法数据库状态下,各基本域是由有限值所组成的,即  $B(D) \subseteq \Delta_{act}^B, D \in D_S$ 。

## 2.2 OWL DL 本体

OWL 具有三个表达力渐增的子语言: Lite, DL 和 Full, 它们为不同的用户应用需求而设计<sup>[4]</sup>。OWL DL 具有确保计算完备性和可判定性的最大表达力,它有两种语法:① 交换语法 (exchange syn-

tax), 即 RDF/XML 语法<sup>[15]</sup>, 以一组 RDF 三元组 (triples) 的 XML 序列化格式来表示一个本体, 以便在 Web 上发布和共享本体; ② 抽象语法 (abstract syntax)<sup>[16]</sup>, 即框架风格的语法, 其中, 关于一个类或属性的一组信息用一个尽可能大的、类似于自然语言语法的构造子来表示, 便于用户理解和评价一个本体. 同一本体使用不同语法时, 具有相同的形式语义——语义由底层相同的 RDF 图所决定<sup>[4]</sup>.

形式上, 一个用 OWL DL (抽象语法) 表示的本体简单地包含一个可选的本体标识符、一组可选的标注 (annotation) 以及一组公理 (axiom). 标注用于记录本体的作者身份以及对其它本体的输入引用等信息; 本体的主要内容由公理提供, 包括类公理、属性公理以及个体公理 (即事实). 定义 5 给出了 OWL DL (抽象语法) 本体的简要形式定义 (其中省略了与本文问题无关的标注、个体标识符与公理、RDF 字面量以及部分 OWL 构造子).

**定义 5.** 一个 OWL DL (抽象语法) 本体是一个二元组  $O = (ID_O, axiom_O)$ , 其中:

①  $ID_O = CID_O \cup DRID_O \cup OPID_O \cup DPID_O$  是一个有限的标识符 (identifier) 集, 其划分为两两不

相交的: 一个类 (class) 标识符的集合  $CID_O$ , 其中除了用户定义的类标识符外还可包括 owl: Thing 和 owl: Nothing 两个预定义类标识符; 一个数据值域 (data range) 标识符的集合  $DRID_O$ , 每个数据值域标识符有一个预定义的 XML Schema 数据类型标识符 (如 xsd: integer); 一个对象属性 (object property) 标识符的集合  $OPID_O$ ; 一个数据类型属性 (datatype property) 标识符的集合  $DPID_O$ . 所有标识符的完整语法格式是一个 URI 引用 (URI reference), 它由一个表示名空间 (namespace) 的绝对 URI 或前缀 (prefix) 和一个片断标识符 (fragment identifier) 所组成 (为简便, 本文省略名空间部分).

②  $axiom_O$  是一个有限的公理序列, 其中包含类公理和属性公理. 每个公理是由作用于标识符或描述 (description) 的若干构造子 (如 SubClassOf, ObjectProperty 等) 而实现. 描述要么是直接的类标识符, 要么是通过匿名类构造子 restriction 由标识符构造而成.

以上标识符、描述、类公理和属性公理的 OWL DL 抽象语法及对应的描述逻辑语法格式由表 1 给出.

表 1 OWL DL 标识符与 (部分) 构造子的语法与语义

OWL DL 抽象语法 OWL DL Abstract Syntax	描述逻辑语法 Description Logics Syntax	模型论语义 Model-Theoretic Semantics
描述 (用 $C$ 表示, 可以带下标)		
$A$ (类标识符——URI 引用)	$A$	$A^I \subseteq \Delta^I$ (个体解释域)
owl: Thing (预定义)	$\top$	$\top^I = \Delta^I$
owl: Nothing (预定义)	$\perp$	$\perp^I = \emptyset$
restriction( $R$ allValuesFrom( $C$ ))	$\forall R.C$	$(\forall R.C)^I = \{x \mid \forall y. \langle x, y \rangle \in R^I \rightarrow y \in C^I\}$
restriction( $R$ minCardinality( $n$ ))	$\geq nR$	$(\geq nR)^I = \{x \mid \#\{y \mid \langle x, y \rangle \in R^I\} \geq n\}$
restriction( $R$ maxCardinality( $n$ ))	$\leq nR$	$(\leq nR)^I = \{x \mid \#\{y \mid \langle x, y \rangle \in R^I\} \leq n\}$
restriction( $R$ cardinality( $n$ ))	$= nR$	$(= nR)^I = \{x \mid \#\{y \mid \langle x, y \rangle \in R^I\} = n\}$
restriction( $U$ allValuesFrom( $D$ ))	$\forall U.D$	$(\forall U.D)^I = \{x \mid \forall y. \langle x, y \rangle \in U^I \rightarrow y \in D^I\}$
restriction( $U$ cardinality( $n$ ))	$= nU$	$(= nU)^I = \{x \mid \#\{y \mid \langle x, y \rangle \in U^I\} = n\}$
数据值域 (用 $D$ 表示, 可以带下标)		
$D$ (预定义 XML Schema 数据类型标识符——URI 引用)	$D$	$D^I \subseteq \Delta_D^I$ (数据值解释域)
对象属性 (用 $R$ 表示, 可以带下标)		
$R$ (对象属性标识符——URI 引用)	$R$	$R^I \subseteq \Delta^I \times \Delta^I$
数据类型属性 (用 $U$ 表示)		
$U$ (数据类型属性标识符——URI 引用)	$U$	$U^I \subseteq \Delta^I \times \Delta_D^I$
类公理		
Class( $A$ partial $C_1 \dots C_n$ )	$A \subseteq C_1 \cap \dots \cap C_n$	$A^I \subseteq C_1^I \cap \dots \cap C_n^I$
SubClassOf( $C_1 C_2$ )	$C_1 \subseteq C_2$	$C_1^I \subseteq C_2^I$
DisjointClasses( $C_1 \dots C_n$ )	$C_i \cap C_j = \perp, i \neq j, i, j = 1, 2, \dots, n$	$C_i^I \cap C_j^I = \emptyset, i \neq j, i, j = 1, 2, \dots, n$
属性公理		
ObjectProperty( $R$ )		
domain( $C_1 \dots C_m$ )	$\geq 1R \subseteq C_i, i = 1, 2, \dots, m$	$R^I \subseteq C_i^I \times \Delta^I, i = 1, 2, \dots, m$
range( $C_1 \dots C_l$ )	$\top \subseteq \forall R.C_i, i = 1, 2, \dots, l$	$R^I \subseteq \Delta^I \times C_i^I, i = 1, 2, \dots, l$
[inverseOf( $R_0$ )]	$R = (\neg R_0)$	$R^I = (R_0^I)^-$
DatatypeProperty( $U$ )		
domain( $C_1 \dots C_m$ )	$\geq 1U \subseteq C_i, i = 1, 2, \dots, m$	$U^I \subseteq C_i^I \times \Delta_D^I, i = 1, 2, \dots, m$
range( $D_1 \dots D_l$ )	$\top \subseteq \forall U.D_i, i = 1, 2, \dots, l$	$U^I \subseteq \Delta^I \times D_i^I, i = 1, 2, \dots, l$
[Functional]]	$\top \subseteq \leq 1U$	$U^I$ 具有函数性

OWL 用描述逻辑风格的模型论(model theory)来定义 OWL 本体的语义<sup>[16,17]</sup>; OWL DL 底层的形式基础是描述逻辑 SHOIN(D)<sup>[18]</sup>. 模型论语义基于解释(interpretation), OWL DL 本体的解释定义如下.

**定义 6.** 一个 OWL DL 本体  $O=(ID_O, axiom_O)$  的一个解释  $I=(\Delta^I, \Delta_b^I, \cdot^I)$  由两个互不相交的有限解释域: 个体解释域  $\Delta^I$  和数据值解释域  $\Delta_b^I$  以及一个解释函数  $\cdot^I$  来指定. 其中, 函数  $\cdot^I$

- ① 将每个类  $C \in CID_O$  映射为集合  $C^I \subseteq \Delta^I$ ;
- ② 将每个数据值域(数据类型)  $D \in DRID_O$  映射为集合  $D^I \subseteq \Delta_b^I$ ;
- ③ 将每个对象属性  $R \in OPID_O$  映射为集合  $R^I \subseteq \Delta^I \times \Delta^I$ ;
- ④ 将每个数据类型属性  $U \in DPID_O$  映射为集合  $U^I \subseteq \Delta^I \times \Delta_b^I$ .

集合  $C^I$  的元素称为类  $C$  的实例, 即个体(individual); 集合  $D^I$  的元素称为数据值域(数据类型)  $D$  的实例, 即数据值(data value). 除了对以上原子概念的解释, 解释函数  $\cdot^I$  可进一步扩展到描述和公理(即概念表达式), 相应模型论语义由表 1 给出.

实际上, 表 1 的“模型论语义”列给出了 OWL DL(抽象语法)本体的解释针对本体中公理(及其构造子)应满足的语义条件. OWL 本体的可满足性由其模型(model)<sup>[18]</sup> 确保, 见以下定义 7.

**定义 7.** 设  $I$  是 OWL DL 本体  $O$  的一个解释, 当  $I$  满足  $O$  的所有公理的语义条件时, 称  $I$  是  $O$  的一个模型. 进一步地, 一个本体是可满足的, 当且仅当它存在一个模型.

## 3 从 ER 模式到 OWL DL 本体的翻译

### 3.1 翻译算法

一个 ER 模式与一个 OWL DL 本体间存在着优雅的对对应关系(见下文定理 1 及其证明), 基于以上形式化, 将一个 ER 模式翻译成 OWL DL(抽象语法)本体的形式方法由下列算法 1 给出.

**算法 1.** 从 ER 模式到 OWL DL 本体的翻译算法.

输入: ER 模式  $S=(L_S, isa_S, att_S, rel_S, card_S)$

输出: 由翻译函数  $\phi$  定义的 OWL DL(抽象语法)本体  $O=\phi(S)$ , 其中,  $ID_O=\phi(L_S)$ ,  $axiom_O=\phi(isa_S, att_S, rel_S, card_S)$

步骤:

1. 从字母表到标识符集的翻译  $ID_O=\phi(L_S)$ (可选的, 各标识符中的片断标识符可通过交互, 由领域专家重命名):

- 1.1. 对每个实体符号  $E \in E_S$ , 创建一个类标识符  $\phi(E) \in CID_O$ ,  $\phi(E)$  的片断标识符与  $E$  同名;
- 1.2. 对每个联系符号  $R \in R_S$ , 创建一个类标识符  $\phi(R) \in CID_O$ ,  $\phi(R)$  的片断标识符与  $R$  同名;
- 1.3. 对每个属性符号  $A \in A_S$ , 创建一个数据类型属性标识符  $\phi(A) \in DPID_O$ ,  $\phi(A)$  的片断标识符与  $A$  同名;
- 1.4. 对每个域符号  $D \in D_S$ , 映射为一个预定义的 XML Schema 数据类型标识符  $\phi(D) \in DRID_O$ ;
- 1.5. 对每个角色符号  $U \in U_S$ , 创建一对互逆对象属性的标识符  $\phi(U) \in OPID_O$  和  $V=inv_\phi(U) \in OPID_O$ ,  $\phi(U)$  的片断标识符与  $U$  同名;

2. 从 ER 模式构造子元素到 OWL DL 本体公理的翻译  $axiom_O=\phi(isa_S, att_S, rel_S, card_S)$ :

2.1. 对每一对实体  $E_1, E_2 \in E_S$ , 且  $E_1 isa_S E_2$ , 创建一个类公理:

$Class(\phi(E_1) \text{ partial } \phi(E_2))$  或

等价形式  $subClassOf(\phi(E_1) \phi(E_2))$  (i)

2.2. 对每个实体  $E \in E_S$ , 且  $att_S(E)=[A_1:D_1, \dots, A_h:D_h]$ ,

2.2.1. 创建一个类公理:

$Class(\phi(E) \text{ partial restriction}(\phi(A_1)$

$allValuesFrom(\phi(D_1)) \text{ cardinality}(1) \dots$

$restriction(\phi(A_h) \text{ allValuesFrom}(\phi(D_h))$

$cardinality(1))$  (ii)

2.2.2. FOR  $i=1, 2, \dots, h$  DO

创建一个属性公理(仅当  $A_i$  是唯一属性(单属性键)时, Functional 才出现):

$DatatypeProperty(\phi(A_i) \text{ domain}(\phi(E)) \text{ range}(\phi(D_i))$

$[Functional])$  (iii)

2.3. 对每个联系  $R \in R_S$ , 且  $rel_S(R)=[U_1:E_1, \dots, U_k:E_k]$ ,

2.3.1. 创建一个类公理:

$Class(\phi(R) \text{ partial restriction}(\phi(U_1)$

$allValuesFrom(\phi(E_1)) \text{ cardinality}(1) \dots$

$restriction(\phi(U_k) \text{ allValuesFrom}(\phi(E_k))$

$cardinality(1))$  (iv)

2.3.2. FOR  $i=1, 2, \dots, k$  DO

创建一个属性公理:

$ObjectProperty(V_i \text{ domain}(\phi(E_i)) \text{ range}(\phi(R))$

$inverseOf \phi(U_i))$  (v)

创建一个类公理:

$Class(\phi(E_i) \text{ partial restriction}(V_i \text{ allValuesFrom}(\phi(R)))$

(vi)

2.4. 对每个关联实体和联系的角色  $U \in U_S$ , 且  $rel_S(R)=[\dots, U:E, \dots]$ ,

2.4.1. 创建一个属性公理:

ObjectProperty( $\phi(U)$  domain( $\phi(R)$ ) range( $\phi(E)$ ))  
(vii)

2.4.2. 若  $m = \text{mincard}_s(E, R, U) \neq 0$ , 则创建一个类公理:

Class( $\phi(E)$  partial restriction( $V$  minCardinality( $m$ )))  
(viii)

2.4.3. 若  $n = \text{maxcard}_s(E, R, U) \neq \infty$ , 则创建一个类公理:

Class( $\phi(E)$  partial restriction( $V$  maxCardinality( $n$ )))  
(ix)

2.5. 对每一对符号  $X, Y \in E_s \cup R_s$ , 同时  $X \neq Y$  且  $X \in R_s$ , 创建一个类公理:

DisjointClasses( $\phi(X)$   $\phi(Y)$ ) (x)

值得指出的是, 为了使本体简洁, OWL DL 抽象语法鼓励合并关于相同属性的限制描述(由 restriction 构造子定义)、关于相同类的类公理(由 Class 构造子定义)以及关于相同属性的属性公理(由 ObjectProperty/DatatypeProperty 构造子定义), 合并后语义保持不变<sup>[6]</sup>. 据此, 以上算法 1 中, 公理(viii)和(ix)中的属性限制描述可合并到公理(vi)中关于相同属性的限制描述中, 从而导致公理(vi), (viii)和(ix)的合并; 进一步地, 公理(vi)又可合并到关于相同类的公理(ii)中; 如果这个类还有公理(i), 那么又可进一步合并到公理(i)中. 因此, 算法 1 实现时, 公理的创建可采用“添加”式, 即如果关于一个类的公理(包括相关属性限制描述)尚不存在, 则创建之; 如果已经存在, 则新的构造子添加到类公理中. 第 4 节的案例中说明了这种做法.

算法 1 的时间性能一定程度上取决于表示 ER 模式的数据结构. 现有主要的 ER 建模工业工具通常将一个 ER 图(即 ER 模式)表示成一个 XML 文件, 并对 ER 构造子元素进行了合理的组织, 因而可认为用 XML 解析器一次扫描、解析 ER 模式文件可获得算法 1 中各操作步处理所需的全部输入数据. 因此, 对算法 1 的时间复杂度可做以下一般分析. 由于算法中第 1 步的全部标识符创建可以在第 2 步的公理创建中直接进行, 因此可认为算法的基本操作为公理的创建. 设一个 ER 模式  $S$  的规模  $N = N_E + N_A + N_R + N_U$ , 其中,  $N_E$  是实体符号集  $E_s$  的基数,  $N_A$  是属性符号集  $A_s$  的基数,  $N_R$  是联系符号集  $R_s$  的基数,  $N_U$  是角色符号集  $U_s$  的基数. 实体间的 ISA 联系是单射和非循环的, 极端情况下全部实体构成一个层次结构, 因此第 2.1 步中公理(i)的创建次数至多是  $N_E - 1$  次. 显然, 第 2.2.1 步中公理(ii)的创建次数是  $N_E$  次; 第 2.2.2 步中公理(iii)

的创建次数是  $N_A$  次; 第 2.3.1 步中公理(iv)的创建次数是  $N_R$  次; 第 2.3.2 步中公理(v)和(vi)的创建次数均是  $N_U$  次; 第 2.4.1 步中公理(vii)的创建次数是  $N_U$  次; 第 2.4.2 步中公理(viii)和第 2.4.3 步中公理(ix)的创建次数均至多是  $N_U$  次. 以上全部公理的创建次数总和至多为  $2N_E + N_A + N_R + 5N_U - 1$ . 第 2.5 步中公理(x)的创建次数为  $N_R \times (N_R - 1) / 2 + N_R \times N_E = N_R \times (N_E + N_R / 2 - 1 / 2)$  次. 因此, 最坏情况下, 算法的基本操作总执行次数  $T = N_R \times (N_E + N_R / 2 - 1 / 2) + 2N_E + N_A + N_R + 5N_U - 1 < N^2 + 5N$ . 故算法 1 的时间复杂度低于  $O(N^2)$ .

### 3.2 翻译的语义保持性

由于本文翻译方法基本上是通过结合

① 从 ER 模式到 ALUNI DL 知识库的语义保持的映射规则<sup>[13]</sup>;

② 比 ALUNI 更具表达性的 SHOIN( $D$ ) DL 知识库与 OWL DL 本体之间的语义对应<sup>[18]</sup>而导出的, 因此, 直觉上它应是语义保持的翻译. 在形式证明它的语义保持性之前, 我们先定义一个新概念. 由定义 4 中 ER 模式  $S$  的合法数据库状态条件(3)可知:  $R^B$  中的任一实例  $r$  是一个定义在  $[U_1, U_2, \dots, U_k]$  上的函数. 根据函数的性质,  $S$  的每个合法数据库状态都隐含了如下限制<sup>[13]</sup>: 对于任意的  $r_1, r_2 \in R^B$ , 其中  $R \in R_s$  且  $\text{rel}_s(R) = [U_1 : E_1, \dots, U_k : E_k]$ , 必有  $(\bigwedge_{1 \leq i \leq k} \forall e \in E^B. (r_1[U_i] = e \leftrightarrow r_2[U_i] = e)) \rightarrow (r_1 = r_2)$ . 因此, 文献[13]引入了 ALUNI 描述逻辑知识库的关系描述模型(relation-descriptive model)的概念. 同样, 我们这里也仅需考察 OWL DL 本体  $\phi(S)$  的具有相应限制的模型——关系描述模型, 见定义 8.

**定义 8.** 设  $\phi(S)$  是从 ER 模式  $S = (L_s, \text{isa}_s, \text{att}_s, \text{rel}_s, \text{card}_s)$  翻译而得的一个 OWL DL(抽象语法)本体,  $I$  是本体  $\phi(S)$  的一个模型. 对于任意的  $s_1, s_2 \in (\phi(R))^I$ , 其中  $R \in R_s$  且  $\text{rel}_s(R) = [U_1 : E_1, \dots, U_k : E_k]$ , 如果  $(\bigwedge_{1 \leq i \leq k} \forall f \in \Delta^I. (\langle s_1, f \rangle \in (\phi(U_i))^I \leftrightarrow \langle s_2, f \rangle \in (\phi(U_i))^I)) \rightarrow (s_1 = s_2)$ , 则称  $I$  是  $\phi(S)$  的一个关系描述模型.

**定理 1.** 算法 1 从 ER 模式  $S$  到 OWL DL(抽象语法)本体  $\phi(S)$  的翻译方法是语义保持的, 即

(1) 对  $S$  的任一合法数据库状态  $B$ , 必存在映射  $\alpha: B \rightarrow I$ , 使得  $I = \alpha(B)$  是  $\phi(S)$  的一个模型.

(2) 对  $\phi(S)$  的任一关系描述模型  $I$ , 必存在映射  $\beta: I \rightarrow B$ , 使得  $B = \beta(I)$  是  $S$  的一个合法数据库

状态.

证明. 首先证明(1). 给定  $S$  的一个数据库状态  $B = (\Lambda^B, \Theta^B)$ , 我们定义  $\phi(S)$  的一个解释  $I = \alpha(B)$  如下:

$$\textcircled{1} I \text{ 的个体域 } \Delta^I = \Lambda^B \cup \bigcup_{R \in R_S} R^B.$$

$$\textcircled{2} I \text{ 的数据值域 } \Delta_b^I = \bigcup_{D \in D_S} \Delta_{\text{act}}^B = \bigcup_{D \in D_S} \{d \in B(D) \mid D \in D_S \wedge \exists A \in A_S, e \in \Lambda^B. \langle e, d \rangle \in A^B\}.$$

$$\textcircled{3} \text{ 对每个符号 } X \in E_S \cup A_S \cup R_S, \text{ 令} \\ (\phi(X))^I = X^B \quad (1)$$

$$\textcircled{4} \text{ 对每个域符号 } D \in D_S, \text{ 令} \\ (\phi(D))^I = \Delta_{\text{act}}^B = \{d \in B(D) \mid D \in D_S \wedge \\ \exists A \in A_S, e \in \Lambda^B. \langle e, d \rangle \in A^B\} \quad (2)$$

$\textcircled{5}$  对每个联系  $R \in R_S$  且  $rel_S(R) = [U_1 : E_1, \dots, U_k : E_k]$ , 令

$$(\phi(U_i))^I = \{\langle r, e_i \rangle \in \Delta^I \times \Delta^I \mid r \in R^B \wedge \\ e_i \in E_i^B \wedge r[U_i] = e_i\}, \quad i = 1, 2, \dots, k \quad (3)$$

假定  $B$  是  $S$  的一个合法数据库状态, 下面证明  $I$  必是  $\phi(S)$  的一个模型, 即  $I$  满足  $\phi(S)$  中所有的公理(即算法 1 中的公理(i)~(x))(注意, 由于  $B$  是有限结构, 因此  $I$  也是有限的).

(a) 设实体对  $E_1, E_2 \in E_S$ , 且  $E_1 \text{ isa}_S E_2$ . 根据合法数据库状态(定义 4)条件(1)有  $E_1^B \subseteq E_2^B$ , 又由式(1)有  $(\phi(E_1))^I \subseteq (\phi(E_2))^I$  (即解释  $I$  满足公理(i)的语义条件——下同, 均略). 故  $I$  满足  $\phi(S)$  的公理(i).

(b) 设实体  $E \in E_S$ , 且  $att_S(E) = [A_1 : D_1, \dots, A_h : D_h]$ . 根据定义 4 条件(2)及式(1)和(2)有  $\forall e \in E^B = (\phi(E))^I$ , 存在唯一的  $\langle e, d_i \rangle \in A_i^B = (\phi(A_i))^I$ ,  $d_i \in B(D_i) = (\phi(D_i))^I$ ,  $i = 1, 2, \dots, h$ , 即有

$$(\phi(E))^I \subseteq \bigcap_{i=1}^h \{e \mid \forall d_i. \langle e, d_i \rangle \in (\phi(A_i))^I \rightarrow$$

$$d_i \in (\phi(D_i))^I \wedge \#\{d_i \mid \langle e, d_i \rangle \in (\phi(A_i))^I\} = 1\}.$$

故  $I$  满足  $\phi(S)$  的公理(ii). 又根据数据库状态定义(定义 3)有  $A_i^B \subseteq \Lambda^B \times \bigcup_{D_i \in D_S} B(D_i)$ ,  $i = 1, 2, \dots, h$ ; 进一步地, 根据定义 4 条件(2)有  $\forall e \in E^B$ , 存在唯一的  $\langle e, d_i \rangle \in A_i^B$ ,  $d_i \in B(D_i)$ ,  $i = 1, 2, \dots, h$ . 因此有  $A_i^B \subseteq E^B \times B(D_i)$ ,  $i = 1, 2, \dots, h$ . 再由式(1)和(2), 有  $(\phi(A_i))^I \subseteq (\phi(E))^I \times (\phi(D_i))^I$ ,  $i = 1, 2, \dots, h$ ; 同时, 当  $A_i^B$  具有函数性时,  $(\phi(A_i))^I$  显然也具有函数性,  $i = 1, 2, \dots, h$ . 故  $I$  满足  $\phi(S)$  的公理(iii).

(c) 设角色  $U \in U_S$ , 且  $rel_S(R) = [\dots, U :$

$E, \dots]$ . 根据定义 4 条件(3)有: 对  $\forall r \in R^B$ ,  $r = [\dots, U : e, \dots]$ , 其中  $e \in E^B$ ; 由式(3)有  $(\phi(U))^I = \{\langle r, e \rangle \in \Delta^I \times \Delta^I \mid r \in R^B \wedge e \in E^B \wedge r[U] = e\}$ , 即  $(\phi(U))^I \subseteq R^B \times E^B$ . 又由式(1)有  $(\phi(U))^I \subseteq (\phi(R))^I \times (\phi(E))^I$ . 故  $I$  满足  $\phi(S)$  的公理(vii).

(d) 设联系  $R \in R_S$ , 且  $rel_S(R) = [U_1 : E_1, \dots, U_k : E_k]$ . 根据定义 4 条件(3)及式(1)和(2),  $\forall r \in R^B = (\phi(R))^I$ , 有  $r = [U_1 : e_1, \dots, U_k : e_k]$ ,  $e_i \in E_i^B = (\phi(E_i))^I$ ,  $i = 1, 2, \dots, k$ . 故  $r$  是一个定义在  $[U_1, U_2, \dots, U_k]$  上的函数. 由式(3), 存在唯一的  $e_i = r[U_i] \in E_i^B = (\phi(E_i))^I$  使得  $\langle r, e_i \rangle \in (\phi(U_i))^I$ . 故

$$(\phi(R))^I \subseteq \bigcap_{i=1}^k \{r \mid \forall e_i. \langle r, e_i \rangle \in (\phi(U_i))^I \rightarrow \\ e_i \in (\phi(E_i))^I \wedge \#\{e_i \mid \langle r, e_i \rangle \in (\phi(U_i))^I\} = 1\} \quad (4)$$

故  $I$  满足  $\phi(S)$  的公理(iv). 对  $i = 1, 2, \dots, k$ , 因为属性  $V_i$  是属性  $\phi(U_i)$  的逆, 故有  $V_i^I = ((\phi(U_i))^I)^-$ ; 根据(c)中的证明有  $(\phi(U_i))^I \subseteq (\phi(R))^I \times (\phi(E_i))^I$ , 故有  $V_i^I = ((\phi(U_i))^I)^- \subseteq (\phi(E_i))^I \times (\phi(R))^I$ . 故  $I$  满足  $\phi(S)$  的公理(v). 考察  $\forall r \in (\phi(R))^I$ , 由式(4)知:  $\exists e_i \in (\phi(E_i))^I$ , 使得  $\langle r, e_i \rangle \in (\phi(U_i))^I$  即  $\langle e_i, r \rangle \in V_i^I$ ,  $i = 1, 2, \dots, k$ . 故  $(\phi(E_i))^I \subseteq \{e_i \mid \forall r. \langle e_i, r \rangle \in V_i^I \rightarrow r \in (\phi(R))^I\}$ . 故  $I$  满足  $\phi(S)$  的公理(vi).

(e) 设角色  $U \in U_S$  且  $rel_S(R) = [\dots, U : E, \dots]$ . 若  $m = \text{mincard}_S(E, R, U) \neq 0$ , 考察  $\forall e \in E^B = (\phi(E))^I$ , 根据定义 4 条件(4)有  $\#\{r \in R^B \mid r[U] = e\} \geq \text{mincard}_S(E, R, U) = m$ , 也即  $(\phi(E))^I \subseteq \{e \mid \#\{r \in R^B \mid r[U] = e\} \geq m\}$ . 由式(3)可知  $(\phi(E))^I \subseteq \{e \mid \#\{r \in R^B \mid \langle r, e \rangle \in (\phi(U))^I\} \geq m\}$ . 因  $I$  满足  $\phi(S)$  的公理(vi), 故  $(\phi(E))^I \subseteq \{e \mid \#\{r \in R^B \mid \langle e, r \rangle \in V^I\} \geq m\}$ . 故  $I$  满足公理(viii). 同理可证  $I$  满足  $\phi(S)$  的公理(ix).

(f) 设一对符号  $X, Y \in E_S \cup R_S$ , 同时  $X \neq Y$  且  $X \in R_S$ . 若  $Y \in E_S$ , 则显然有  $X^B \cap Y^B = \emptyset$ , 由式(1)有  $(\phi(X))^I \cap (\phi(Y))^I = \emptyset$ . 若  $Y \in R_S$ , 根据定义 4 条件(3)有  $\forall x \in X^B$ , 有  $x = [U_1 : e_1, \dots, U_k : e_k]$ ,  $U_1, U_2, \dots, U_k \in U_S$ ;  $\forall y \in Y^B$ , 有  $y = [W_1 : d_1, \dots, W_l : d_l]$ ,  $W_1, W_2, \dots, W_l \in U_S$ . 由定义 2 知, 每个角色是针对特定的联系定义的, 故  $[U_1, U_2, \dots, U_k] \neq [W_1, W_2, \dots, W_l]$  且  $x \neq y$ . 因此  $X^B \cap Y^B = \emptyset$ , 即  $(\phi(X))^I \cap (\phi(Y))^I = \emptyset$ . 故  $I$  满足  $\phi(S)$  的公理(x).

最后证明(2). 给定  $\phi(S)$  的一个解释  $I = (\Delta^I, \Delta_b^I, \cdot^I)$ , 我们定义  $S$  的一个数据库状态  $B = \beta(I)$  如下:

$$\textcircled{1} \Delta^B = \Delta^I - \bigcup_{R \in R_S} (\phi(R))^I$$

② 对每个符号  $X \in E_S \cup A_S$ , 令

$$X^B = (\phi(X))^I \quad (5)$$

③ 对每个域符号  $D \in D_S$ , 令其基本域

$$B(D) = (\phi(D))^I \quad (6)$$

④ 对每个联系  $R \in R_S$ , 且  $rel_S(R) = [U_1: E_1, \dots, U_k: E_k]$ , 令

$$R^B = \{ [U_1: e_1, \dots, U_k: e_k] \mid e_1, e_2, \dots, e_k \in \Delta^I \wedge (\bigwedge_{i=1}^k (\exists r \in (\phi(R))^I. \langle r, e_i \rangle \in (\phi(U_i))^I)) \} \quad (7)$$

假定  $I$  是  $\phi(S)$  的一个关系描述模型, 下面证明  $B$  必是  $S$  的一个合法数据库状态, 即  $B$  满足  $S$  中的所有完整性约束条件(即定义 4 中的条件(1)~(4)).

(a) 因  $I$  满足  $\phi(S)$  的公理(i), 故对每一对实体  $E_1, E_2 \in E_S$  且  $E_1 isa_S E_2$ , 有  $(\phi(E_1))^I \subseteq (\phi(E_2))^I$ . 又由式(5)知  $E_1^B \subseteq E_2^B$ . 故  $B$  满足  $S$  的完整性约束条件(1).

(b) 因  $I$  满足  $\phi(S)$  的公理(ii), 故对每个实体  $E \in E_S$  且  $att_S(E) = [A_1: D_1, \dots, A_h: D_h]$ , 有  $(\phi(E))^I \subseteq \bigcap_{i=1}^h \{ e \mid \forall d_i. \langle e, d_i \rangle \in (\phi(A_i))^I \rightarrow d_i \in (\phi(D_i))^I \wedge \# \{ d_i \mid \langle e, d_i \rangle \in (\phi(A_i))^I \} = 1 \}$ ,  $i=1, 2, \dots, h$ . 因此, 由式(5)和(6)知, 对  $\forall e \in (\phi(E))^I = E^B$ , 必存在唯一的  $\langle e, d_i \rangle \in (\phi(A_i))^I = A_i^B$ ,  $d_i \in (\phi(D_i))^I = B(D_i)$ ,  $i=1, 2, \dots, h$ . 故  $B$  满足  $S$  的完整性约束条件(2).

(c) 因  $I$  满足  $\phi(S)$  的公理(iv), 故对每个联系  $R \in R_S$  且  $rel_S(R) = [U_1: E_1, \dots, U_k: E_k]$ , 有  $(\phi(R))^I \subseteq \bigcap_{i=1}^k \{ r \mid \forall e_i. \langle r, e_i \rangle \in (\phi(U_i))^I \rightarrow e_i \in (\phi(E_i))^I \wedge \# \{ e_i \mid \langle r, e_i \rangle \in (\phi(U_i))^I \} = 1 \}$ ,  $i=1, 2, \dots, k$ . 因此, 由式(7)知, 对每个联系  $R$  必满足  $\forall r \in R^B$ , 有  $r = [U_1: e_1, \dots, U_k: e_k]$ ,  $e_i \in E_i^B$ ,  $i=1, 2, \dots, k$ . 故  $B$  满足  $S$  的完整性约束条件(3).

(d) 因  $I$  满足  $\phi(S)$  的公理(viii), 故对角色  $U \in U_S$  且  $rel_S(R) = [\dots, U: E, \dots]$ , 有  $\forall e \in (\phi(E))^I$ ,  $(\phi(E))^I \subseteq \{ e \mid \# \{ r \in R^B \mid \langle e, r \rangle \in V^I \} \geq m \}$ ; 又由  $I$  满足  $\phi(S)$  的公理(v), 有  $V^I = ((\phi(U))^I)^-$ . 故  $(\phi(E))^I \subseteq \{ e \mid \# \{ r \in R^B \mid \langle r, e \rangle \in (\phi(U))^I \} \geq m \}$ , 因此由式(5)知,  $\forall e \in E^B$ ,  $\# \{ r \in R^B \mid r[U] = e \} \geq mincard_S(E, R, U) = m$ . 同理可证明:  $\forall e \in E^B$ ,  $\# \{ r \in R^B \mid r[U] = e \} \leq maxcard_S(E, R, U)$ . 故  $B$  满足  $S$  的完整性约束条件(4). 定理得证. 证毕.

## 4 算法实现与案例

算法 1 已在基于 Java 2 Platform Standard Edition (J2SE) v1. 4. 2 平台自行开发的 ER2WO 工具中实现. ER2WO 是在 Windows 2000/XP 平台独立运行的程序, 它读入由 PowerDesigner 9. 5 产生并输出的 XML 格式 ER 模式(ER 图)文件后, 经自动解析、翻译, 产生并输出相应的 OWL DL 本体(抽象语法和交换语法). 逻辑上, ER2WO 由 4 个功能模块所组成: ① ER 模式解析与可视化: 打开并解析 XML 格式的 ER 模式文件, 在内存形成相应数据结构, 并在屏幕上输出 ER 元素(实体、属性、联系、角色)后准备响应用户的鼠标点击事件(如可选的元素改名); ② 模式词汇改名: 接受 ER 元素的新名输入, 实现改名, 并在屏幕回显新名; ③ ER 向 OWL 翻译: 自动地将 ER 模式翻译成 OWL DL 抽象语法本体, 并保存到 .txt 文件; ④ OWL 语法转换: 自动地将 OWL DL 抽象语法本体转换为等价的交换语法本体, 并保存到 .owl 文件.

我们先后用 ER2WO 进行了 15 个 ER 模式到 OWL DL 本体的自动翻译实验(部分早期的案例已发布于 <http://cse.seu.edu.cn/people/ysdong/graduates/~zmxu/ER2WO/>). 实验结果显示: 算法 1 的运行速度很快, 即使是对大规模 ER 模式也是如此(我们将 ER 模式的实体、属性、联系、角色的数量总和定义为问题的规模  $N$ ); 同时, 算法 1 实际运行时间与问题规模  $N$  的关系与以上理论分析吻合(稍优于  $N^2$  曲线).

为了验证算法 1 的有效性以及完全自动的机器翻译的可实现性, 这里给出一个小规模的例子. 图 1 是用 PowerDesigner 9. 5 设计的 ER 图(ER 模式) univ 的截屏. 需指出的是, PowerDesigner 在 ER 建模方面有以下特点: ① 为了与其支持的 UML 类图风格保持一致, 在 ER 图中, 关联实体和联系的角色基约束是放置在被关联实体端显示的(这与教科书中的通常做法正好相反), 例如图 1 中基约束  $card_{univ}(Faculty, Teaching, teacher) = (1, 5)$  (而不是  $(1, 1)$ ), 这表示任何教员作为教师课时最少教 1 门、最多教 5 门课程; ② 不支持多元联系; ③ 不支持联系的属性. 另外, 图 1 中实体间的“ISA”表示 ISA 联系, 属性后的“pi”表示主键属性, 最大基数“n”表示  $\infty$  (即无约束), 域符号 A、VA20、TXT 和 I 依次表示预定义基本域 Characters、Variable char-



acters(20)、Text 和 Integer;其它构造子及语义是显然的。

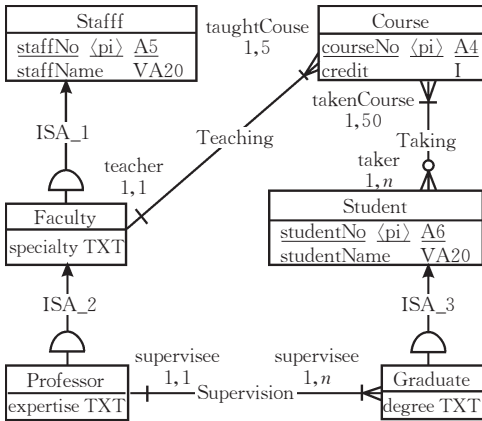


图 1 PowerDesigner 工具产生的 ER 模式 univ

ER 模式 univ 显然符合本文的 ER 模式形式定义,其字母表  $L_{univ}$  组成如下:

实体符号集  $E_{univ} = \{Staff, Faculty, Professor, Student, Graduate, Course\}$ ;

属性符号集合  $A_{univ} = \{staffNo, staffName, specialty, expertise, studentNo, studentName, degree, courseNo, credit\}$ ;

角色符号集合  $U_{univ} = \{teacher, taughtCourse, taker, takenCourse, supervisor, supervisee\}$ ;

联系符号集合  $R_{univ} = \{Teaching, Taking, Supervision\}$ ;

域符号集表示成  $D_{univ} = \{Characters, Variable characters(20), Text, Integer\}$ .

用 ER2WO 工具将 ER 模式 univ 自动翻译成 OWL DL 本体  $univ\_OWL = \phi(univ)$  后的输出结果见图 2 所示(图中未显示简单的 DisjointClasses 公理).下面对输出进行简要说明.

关于从 ER 模式字母表到本体标识符集的翻译  $ID_{univ\_OWL} = \phi(L_{univ})$ . 本体标识符集  $ID_{univ\_OWL}$  组成如下:

类标识符集  $CID_{univ\_OWL} = \{Staff, Faculty, Professor, Student, Graduate, Course, Teaching, Taking, Supervision\}$ ;

数据类型属性标识符集  $DPID_{univ\_OWL} = \{staffNo, staffName, specialty, expertise, studentNo, studentName, degree, courseNo, credit\}$ ;

数据值域标识符集(预定义的 XML Schema 数据类型标识符)  $DRID_{univ\_OWL} = \{xsd:string, xsd:integer\}$ ;

对象属性标识符集  $OPID_{univ\_OWL} = \{teacher, inv\_teacher, taughtCourse, inv\_taughtCourse, taker, inv\_taker, takenCourse, inv\_takenCourse, supervisor, inv\_supervisor, supervisee, inv\_supervisee\}$ .

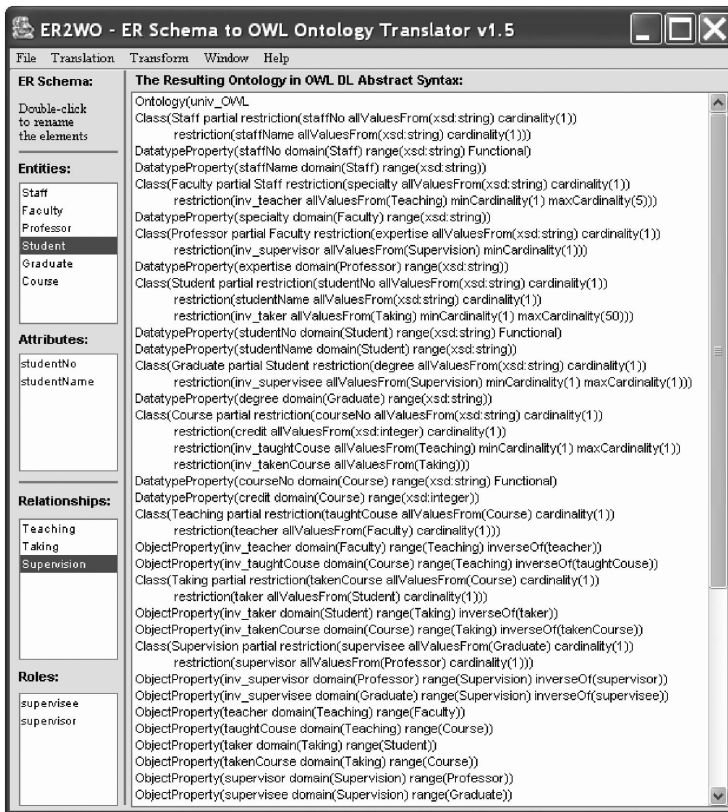


图 2 ER2WO 工具将 ER 模式翻译后输出的 OWL DL 本体

关于从 ER 模式构造子元素到本体公理的翻译

$axiom_{univ\_OWL} = \phi(isa_{univ}, att_{univ}, rel_{univ}, card_{univ})$ . 本体公理序列  $axiom_{univ\_OWL}$  (除了 DisjointClasses 公理) 组成如下:

9 条类公理 Class(...);

9 条数据类型属性公理 DatatypeProperty(...);

12 条对象属性公理 ObjectProperty(...).

对照算法 1 可看出, 以上输出就是算法理论上应有的结果. 这说明了算法的实现是正确的, 自动机器翻译是可实现的.

## 5 结 语

本文提出了一种从 ER 模式到 OWL DL 本体的语义保持的翻译方法. 依照该方法, 用户不仅可方便地将 ER 模式手工翻译成 OWL DL 本体, 而且还可开发相应软件工具, 实现机器自动翻译. 本文贡献在于不仅提出了从 ER 模式到 OWL DL 本体的自动翻译方法, 而且理论上证明了该翻译方法的语义保持性.

语义 Web 的成功与兴旺不仅取决于能否快速、低代价地开发出满足应用需求的 Web 本体, 而且有赖于实现不同知识表示系统间的语义互操作, 以便共享和复用现有知识<sup>[19]</sup>. 因此, 本文方法具有广泛应用前景. 从直接意义来讲, 它可以将数据库中隐含的领域知识迁移到本体中, 从而形成一种有效的 Web 本体开发方法; 从广泛意义来讲, 它在数据密集型 Web 应用和语义 Web 之间搭建了一座语义互操作的桥梁.

**致 谢** 感谢已毕业的硕士研究生曹潇、苏文萍在工具原型 Java 编程和大量案例研究中曾给予的协助!

## 参 考 文 献

- 1 Berners-Lee T., Hendler J. A., Lassila O.. The Semantic Web. *Scientific American*, 2001, 284(5): 34~43
- 2 Jacob E. K.. Ontologies and the Semantic Web. *Bulletin of the American Society for Information Science and Technology*, 2003, 29(4): 19~22
- 3 Li Shan-Ping, Yin Qi-Wei, Hu Yu-Jie, Guo Ming, Fu Xiang-Jun. Overview of researches on ontology. *Journal of Computer Research and Development*, 2004, 41(7): 1041~1052(in Chinese)

(李善平, 尹奇骅, 胡玉杰, 郭 鸣, 付相君. 本体论研究综述. 计

算机研究与发展, 2004, 41(7): 1041~1052)

- 4 Dean M., Schreiber G. eds. OWL Web ontology language reference. W3C Recommendation. 2004-02-10. <http://www.w3.org/TR/owl-ref/>
- 5 Maedche A., Staab S.. Ontology learning for the Semantic Web. *IEEE Intelligent Systems*, 2001, 16(2): 72~79
- 6 Benjamins V. R., Contreras J., Corcho O., Gómez-Pérez A.. Six challenges for the Semantic Web. *AIS SIGSEMIS Bulletin*, 2004, 1(1): 24~25
- 7 Gómez-Pérez A., Manzano-Macho D. eds. A survey of ontology learning methods and techniques. EU-IST Project OntoWeb (IST-2001-29243) Deliverable 1.5. <http://ontoweb.org/Members/ruben/Deliverable%201.5>, 2003-06-06
- 8 Reeve L., Han H.. Survey of semantic annotation platforms. In: *Proceedings of the 2005 ACM symposium on Applied Computing*, Santa Fe, New Mexico, 2005, 1634~1638
- 9 Bergman M. K.. The deep Web: Surfacing hidden value. *The Journal of Electronic Publishing*, 2001, 7(1), <http://www.press.umich.edu/jep/07-01/bergman.html>
- 10 Chang Kevin Chen-Chuan, He Bin, Li Chengkai *et al.* Structured databases on the Web: Observations and implications. *SIGMOD Record*, 2004, 33(3): 61~70
- 11 Teorey T. J., Yang D., Fry J. P.. A logical design methodology for relational databases using the extended Entity-Relationship model. *ACM Computing Surveys*, 1986, 18(2): 197~222
- 12 Hainaut J.-L.. Research in database engineering at the University of Namur. *SIGMOD Record*, 2003, 32(4): 124~128
- 13 Calvanese D., Lenzerini M., Nardi D.. Unifying class-based representation formalisms. *Journal of Artificial Intelligence Research (JAIR)*, 1999, 11: 199~240
- 14 Nardi D., Brachman R. J.. An introduction to description logics. In: Baader F., McGuinness D. L., Nardi D. *et al.*, eds. *The Description Logic Handbook: Theory, Implementation, and Applications*. Cambridge: Cambridge University Press, 2003, 5~44
- 15 Beckett D. eds. RDF/XML syntax specification (revised). W3C Recommendation. 2004-02-10. <http://www.w3.org/TR/rdf-syntax-grammar/>
- 16 Patel-Schneider P. F., Hayes P., Horrocks I. eds. OWL Web ontology language semantics and abstract syntax. W3C Recommendation. 2004-02-10. <http://www.w3.org/TR/owl-absyn/>
- 17 Horrocks I., Patel-Schneider P. F., Harmelen F. V.. From SHIQ and RDF to OWL: The making of a Web ontology language. *Journal of Web Semantics*, 2003, 1(1): 7~26
- 18 Horrocks I., Patel-Schneider P. F.. Reducing OWL entailment to description logic satisfiability. *Journal of Web Semantics*, 2004, 1(4): 345~357
- 19 Lakshmanan L. V. S., Sadri F.. Information integration and the Semantic Web. *IEEE Data Engineering Bulletin*, 2003, 26(4): 19~25



**XU Zhuo-Ming**, born in 1965, Ph.D., professor and Ph. D. supervisor. His research interests include databases, ontological engineering, Semantic Web and Web intelligence.

**DONG Yi-Sheng**, born in 1940, professor and Ph. D. supervisor. His research interests include databases, information systems and knowledge engineering.

**LU Yang**, born in 1977, Ph. D. candidate, lecturer. His research interests include operational research and cybernetics, network information security and Semantic Web.

## Background

The work for this paper is one of the central parts of the key project, Research on Languages and Supporting Software for the Semantic Web, supported by the Natural Science Foundation of Jiangsu Province of China under grant No. BK2003001, and was also funded by the Foreign Culture and Educational Experts Invitation Key-Projects Foundation of the State Administration of Foreign Experts Affairs of China under grant No. 20050360543. Its succedent related researches were further funded by the National Natural Science Foundation of China under grant No. 60573098.

Current information management infrastructures lack universal semantic interoperability, which is much worse on the Web. This enormously limits the abilities of computers that assist human users in computing tasks. As an extension of the current Web, the Semantic Web aims to create machine-processable Web content attached well-defined formal semantics. Semantic annotation of Web content with suitable OWL ontologies is a key enabling technology for achieving the goal. At present, however, how to semantically annotate

dynamic Web pages generated from databases, the greater majority of Web content, is still an open problem. The project team believes that it is of great importance to develop a formal method for acquiring domain knowledge from database schemata and then constructing OWL ontologies, no matter from the angle of ontology development or from the angle of semantic annotation of dynamic Web pages. The authors thus present a method for semantics-preserving translation from ER schema to OWL DL ontology. Theoretic analysis, algorithmic implementation and case study show that the method is feasible and effective and the fully automated machine translation is realizable. The proposed method bridge the gap between existing database applications or legacy systems and the Semantic Web. The research result has definite scientific significance and practical value to foster the application of the Semantic Web technology and to build a new generation of information management infrastructure with semantic interoperability.