AMCIS 2003 Proceedings

Americas Conference on Information Systems (AMCIS)

December 2003

# A Study on a Formal Ontology Model: Constructing a Customer Ontology in a CRM Context

Wang Hongwei
*Shanghai Jiao Tong University*

Jiang Fu
*Shanghai Jiao Tong University*

Wu Jiachun
*Shanghai Jiao Tong University*

# A STUDY ON A FORMAL ONTOLOGY MODEL: CONSTRUCTING A CONSUMER ONTOLOGY IN A CRM CONTEXT

**Wang Hongwei**
Research Center for Intelligent Management and Computer Simulation Applications
Shanghai Jiao Tong University
Kingdavid@citiz.net

**Jiang Fu**
Research Center for Intelligent Management and Computer Simulation Applications
Shanghai Jiao Tong University
Fjiang@mail.sjtu.edu.cn

**Wu Jiachun**
Research Center for Intelligent Management and Computer Simulation Applications
Shanghai Jiao Tong University
wjiachun@guomai.sh.cn

## Abstract

*Ontology is defined as an explicit specification of a conceptualization. The customer ontology is a specific application of ontology to customer domain, which can provide a unified view to customers for better sharing customer knowledge. In this paper, an formal ontology model is constructed using Description Logic, which is a 6-tuples including Term Set, Individual Set, Term Definition Set, Instantiation Assertion Set, Term Restriction Set, Term Comment Set. Based on the model, the issue on ontology validation is studied with the conclusion that the four kinds of term validation, including term satisfiability validation, term subsumption validation, term equivalence validation and term disjointness validation, can be reduced to the satisfiability validation, and satisfiability validation can be transformed into instantiation consistence validation, which can be decided by Tableau Algorithm. At last, the issue on the construction of customer ontology in the CRM context is discussed.*

## Introduction

Good customer relationships are at the heart of business success in today's customer-centered market. Customer relationship management (CRM) is a new business strategy used to analyze the patterns of users in order to develop stronger relationships with them. With the fierce market competition and the rapid IT development, more and more companies have been concerned about CRM to capture and preserve customers for maximizing their profits.

However, CRM systems are facing with several problems now. One of them is a lack of unified views to customers within the whole company. To improve the competitiveness, a company needs to recognize customers in multiple perspectives. Therefore, customer knowledge stored in various departments is required to be shared consistently. For customers, they certainly expect the best product or service, so they hope every department can fully know all the interactions they ever did with the company before. The other problem is the uncertainty of information while interacting with customers. Customers are not seen directly in the e-commerce environment. The information about customers is coming from their own utterances, their histories, and general market analysis, most of which is implicit and need to be made explicit for use. Moreover, some interactions are a series of questionnaires asking information about customers such as personal backgrounds, preferences and so on. Not only questionnaires itself but also users' replies are represented as natural language, which makes customer knowledge even more ambiguous and difficult to be recognized.

The term *ontology* has been originally used in philosophy sphere, where it indicates the systematic explanation of *Existence*. Now it is gaining a specific role in Artificial Intelligence as an explicit specification of a conceptualization (Guarino 1998). Nowadays, the importance of ontology is being recognized in research fields as diverse as knowledge engineering (Gaines 1998), knowledge representation (Guarino 1995), qualitative modeling (Borgo et al. 1997), database design (Van de Riet 1998), information modeling (Ashenhurst 1996), information integration (Mena 1998), semantic-based information retrieval (Wang 2003), semantic web, and knowledge management.

Construction of terms about customer (including concepts and relationships among them) is a promising way to a successful CRM in customer knowledge acquisition, retrieval and integration. Customer ontology is a specific application of ontology to the customer management domain. With customer ontology, all the departments of a company can make an agreement with customer knowledge consistently. Unfortunately, up to now, there is still not such a report about deep studies on customer ontology, especially in the Chinese CRM context. Some projects, such as Edinburgh Enterprise Ontology (EO) (Uschold et al. 1998) and Toronto Virtual Enterprise (TOVE) (Fox and Gruninger 1998), explored enterprise concept modeling. In these projects, customer concepts have been presented, but they are too general to satisfy the need of customer management in CRM. So an elaborate ontological model of customer with proper granularities is required.

The research on customer ontology depends on two key issues: the formal model of ontology and the domain knowledge of CRM. In section 2, a formal ontology model is proposed. In section 3, the ontology validation is studied. In section 4, the construction of customer ontology is discussed.

## Formal Ontology Model

### *Ontology and Ontology Interpretation*

Most of ontology models are based on first order logic (FOL), e.g. Ontolingua (Gruber 1993), CycL (Gruber 1990), and LOOM (MacGregor 1991). Although FOL has an expressive power, its reasoning processes are more complex and most of them are even undecidable, which is hard to validate ontology model. In this paper, we build an ontology model using Description Logics (DL) (Baader et al. 2002). DL is the name for a family of knowledge representation formalisms that represent the knowledge of an application domain by first defining the relevant concepts of the domain, and then using these concepts to specify properties of objects and individuals occurring in the domain. DL is equipped with a formal and logic-based semantics allowing inferring implicitly represented knowledge from the knowledge that is explicitly contained in the knowledge base. Although DL has a less expressive power than FOL, its inference procedures are more efficient, which is suitable for ontology validation (Wang 2003).

**Def. 1:** Given a term description language *L*, an *ontology model* (or *ontology*, for short) is a 6-tuple, written as

$$O=<T, X, TD, XD, TR, TC>$$

where *T* is a *Term Set*, *X* is an *Individual Set*, *TD* is a *Term Definition Set*, *XD* is an *Instantiation Assertion Set*, *TR* is a *Term Restriction Set* and *TC* is a *Term Comment Set*. Elements in *T* are also called *atomic term*, which falls into two categories: *atomic class term* (or *atomic class*, for short) and *atomic property term* (or *atomic property*, for short).

**Def. 2:** Given an ontology $O=<T, X, TD, XD, TR, TC>$, an *ontology interpretation* is a 2-tuple, written as

$$I=<\Delta^I, -^I>$$

where $\Delta^I \neq \emptyset$ is the domain of the interpretation, and $-^I$ is an interpretation function, which assigns to every atomic class *C* in *T* a set $C^I \subset \Delta^I$, and to every atomic property *P* in *T* a binary relation $P^I \subset \Delta^I \times \Delta^I$, and to every individual *a* in *X* an element $a^I \in \Delta^I$.

Figure 1 gives an example of a family ontology using the model defined in Def.1.

***O*-Family**=<{Person, Female, Male, Woman, Man, Mother, Father, Parent, hasChild, hasHusband, Wife, Grandmother, MotherWithoutSon, MotherWithManyChildren}, {Alice, Tom, Mary}, {Woman≡Person ⊓ Female, Man≡Person ⊓ Male, Mother≡Woman ⊓ ∃hasChild.Person, Father≡Man ⊓ ∃hasChild.Person, Parent≡ Father ⊔ Mother, Wife≡Woman ⊓ ∃hasHusband.Man, Grandmother≡ Mother ⊓ ∃hasChild.Parent, MotherWithoutSon≡ Mother ⊓ ∀hasChild.ØMan, MotherWithManyChildren≡Mother ⊓≥3,hasChild}, {Man(Tom), Woman(Alice), hasHusband(Alice, Tom), hasChild(Alice, Mary)}, {Female ⋒Male}, Ø>

**Figure 1. A Family Ontology Model**

## *Term Set, Term Definition Set and Term Comment Set*

*Term Set* comprises a group of atomic terms, denoted as $T=\{t_1, t_2,\ldots, t_n\}$, where $t_i \in T$ ($i$=1, 2,…, $n$) is an atomic term.

*Term Comment Set* is to describe the meaning of atomic terms in the natural language, denoted as $TC=\{tc(t_1), tc(t_2),\ldots, tc(t_n)\}$, where $t_i \in T$ is an atomic term in $T$, and $tc(t_i)$ is a description of $t_i$ in the natural language.

However, atomic terms can only express very limited and simple contents, because of their less expressive powers. So we adopt the *term constructors* here from DL to build term formulae for representing more complex contents.

Given a term description language $L$, the expression satisfying the following syntax is called an $L$-based term formula.

$$D, E \rightarrow C \mid \top\text{½^½Ø}C \mid D \sqcap E \mid \forall P.D \mid \exists P.\top \qquad (1)$$

where $C$ is an atomic class, $D$ and $E$ are two $L$-based formula, and $P$ is an atomic property. Tab.1 gives the ontology interpretations to term constructors.

**Table 1. Basic Term Constructors and Their Ontology Interpretations**

| Constructors Name | Term Constructors Syntax | Ontology Interpretation |
|---|---|---|
| Universal Class | $\top$ | $\top^I=\Delta^I$ |
| Empty Class | $\bot$ | $\bot^I=\emptyset$ |
| Atomic Class Negation | $\neg C$ | $(\neg C)^I=\Delta^I\backslash C^I$ |
| Intersection | $D \sqcap E$ | $(D \sqcap E)^I=D^I \cap E^I$ |
| Property Value Restriction | $\forall P.D$ | $(\forall P.D)^I=\{a\in\Delta^I \mid \forall b.\ (a, b)\in P^I \rightarrow b\in D^I\}$ |
| Limited Existential Quantification | $\exists P.\top$ | $(\exists P.\top)^I=\{a\in\Delta^I \mid \exists b.\ (a, b)\in P^I\}$ |

Table 1 shows that the expressive power of term formulas depends on the types of term constructors the language $L$ supports. In Formula (1), $L$ only supports six basic constructors listed in Tab.1, so $L$ is also called *basic term description language*, written as $L_B$, and the corresponding term formulas built by $L_B$ is called $L_B$-based term formulas.

Obviously, to obtain more expressive languages, we should add term constructors to $L_B$. Tab.2 gives another set of term constructors, called *extension term constructors*, where $P$ is an atomic property, and $D$ and $E$ are two term formulas.

**Table 2. Extension Term Constructors and Their Ontology Interpretations**

| Constructors Name | Term Constructors Syntax | Ontology Interpretation |
|---|---|---|
| Union | $D \sqcup E$ | $(D \sqcup E)^I = D^I \cup E^I$ |
| Non-atomic negation | $\neg D$ | $(\neg D)^I = \Delta^I \backslash D^I$ |
| Full Existential Quantification | $\exists P.D$ | $(\exists P.C)^I = \{a \in \Delta^I \mid \exists b.\ (a, b) \in P^I \wedge b \in C^I\}$ |
| At-least number restriction | $\geq n,P$ | $(\geq n,P)^I = \{a \in \Delta^I \mid \lvert\{b \mid (a, b) \in P^I\}\rvert \geq n\}$ |
| At-most number restriction | $\leq n,P$ | $(\leq n,P)^I = \{a \in \Delta^I \mid \lvert\{b \mid (a, b) \in P^I\}\rvert \leq n\}$ |

There are three kinds of relationships between terms, i.e. subsumption, equivalence, and disjointness.

**Def. 3 (Subsumption):** Given term description language $L$ and an ontology $O$=<$T$, $X$, $TD$, $XD$, $TR$, $TC$>, $D$ and $E$ are two $L$-based term formulas. We say $D$ is *subsumed* by $E$, if $D^I \subseteq E^I$ holds for any ontology interpretation $I$, denoted as $DME$.

**Def. 4 (Equivalence):** Given term description language $L$ and an ontology $O$=<$T$, $X$, $TD$, $XD$, $TR$, $TC$>, $D$ and $E$ are two $L$-based term formulas. We say $D$ and $E$ are *equivalent*, if $D^I = E^I$ holds for any ontology interpretation $I$, denoted as $D \equiv E$.

**Def. 5 (Disjointness):** Given term description language $L$ and an ontology $O$=<$T$, $X$, $TD$, $XD$, $TR$, $TC$>, $D$ and $E$ are two $L$-based term formulas. We say $D$ and $E$ are *disjoint*, if $D^I \cap E^I = \emptyset$ holds for any ontology interpretation $I$, denoted as $D \not\cap E$.

**Property 1:** Let $D$ and $E$ be two term formulas, $P$ be a property. Then we have the following equivalence relationships:
(i) $D \sqcap \neg D \equiv \bot$; (ii) $D \sqcup \neg D \equiv T$; (iii) $D \sqcup E \equiv \neg(\neg D \sqcap \neg E)$;
(iv) $\neg(\exists P.D) \equiv \forall P.\neg D$; (v) $\neg(\forall P.D) \equiv \exists P.\neg D$; (vi) $\neg(\geq(n+1),P) \equiv \leq n,P$
(vii) $\neg(\leq n,P) \equiv \geq(n+1),P$

**Property 2:** Let $D$ and $E$ be two term formulas, $P$ be a property. Then the following subsumption relationships hold:
(i) $D \sqcap EMD$; $D \sqcap EME$; (ii) $DMD \sqcup E$; $EMD \sqcup E$; (iii) $\forall P.DM$"$P.E$, iff $DME$;
(iv) $\exists P.DM\exists P.E$, iff $DME$; (v) $\geq n,PM\geq m,P$, if $n \geq m$; (vi) $\leq n,PM\leq m,P$, if $n \leq m$

**Def. 6 (Term Definition Item):** A term definition item actually is an equivalence relationship between two terms, written as $C \equiv D$, where $C$ is an atomic class term called *definiendum*, and $D$ is a term formula called *definiens*.

A term definition item $C \equiv D$ means $C$ is defined in term of $D$. Given an ontology $O$=<$T$, $X$, $TD$, $XD$, $TR$, $TC$>, *Term Definition Set TD* is such a set that consists of term definition items subject to the following restrictions, written as $TD = \{C_1 \equiv D_1, C_2 \equiv D_2, \ldots, C_n \equiv D_n\}$. Where $C_i \in T$, $D_i$ is a term formula, and every term in $D_i$ is from $T$.

(i) for any $i$, $j$ ($i \neq j, 1 \leq i \leq n, 1 \leq j \leq n$), $C_i \neq C_j$ holds.
(ii) if there exist $C_1' \equiv D_1'$, $C_2' \equiv D_2'$, …, $C_m' \equiv D_m'$ in $TD$, and $C_i'$ occurs in $D_{i-1}'$ ($1 < i \leq m$, $m \leq n$), then $C_1'$ must not occur in $D_m'$.

Restriction (i) guarantees that each term in $T$ can be defined at most once to avoid the logical conflict caused by defining a term in multiple times; Restriction (ii) is intended to avoid cyclic definitions.

**Def. 7 (Model of Term Definition Item):** Given an ontology $O$=<$T$, $X$, $TD$, $XD$, $TR$, $TC$>, if there exists an ontology interpretation $I$ satisfying a term definition item $A$ of $TD$, then $I$ is called a *model* of $A$. If $I$ is a model of all term definition item of $TD$, then we say $I$ is a *model* of $TD$.

**Def. 8 (Defined Term & Primitive Term):** Given an ontology $O$=<$T$, $X$, $TD$, $XD$, $TR$, $TC$>, atomic terms of $T$ can be divided into two sets: the *defined terms* occurring in the *definiendum* of term definition item of $TD$, written as $T_d$, and the *primitive terms* occurring only in the *definiens*, written as $T_p$.

In Figure 1, $T_d$={Woman, Man, Male, Mother, Father, Parent, Wife, MotherWithManyChildren, MotherWithoutSon, Grandmother}, $T_p$={Person, Female, hasChild, hasHusband}.

**Def. 9 (Expansion of term definition item):** Let $TD=\{C_1 \equiv D_1, C_2 \equiv D_2, \ldots, C_n \equiv D_n\}$ be a Term Definition Set, we expand each term definition item in $TD$ through an iterative process by replacing each occurrence of a defined term in the *definiense* with the primitive terms it stands for. Since no cycle term definition is allowed in $TD$ (done by Restriction ( ) above), the process eventually stops and we end up with a Term Set $T'=\{C_1 \equiv D_1', C_2 \equiv D_2', \ldots, C_n \equiv D_n'\}$, where $D_i'$ contains only primitive terms and no defined terms. We say that $D_i'$ is the *expansion* of $D_i$ with respect to $TD$, written as $Exp(D_i)$, and $C_i \equiv D_i'$ is the *expansion* of $C_i \equiv D_i$ with respect to $TD$, and $T'$ is the *expansion* of $T$ with respect to $TD$, written as $Exp(T)$.

**Proposition 1:** Suppose that $O=<T, X, TD, XD, TR, TC>$ is an ontology, where $TD=\{C_1 \equiv D_1, C_2 \equiv D_2, \ldots, C_n \equiv D_n\}$, and $E$ is a term formula. If $I$ is a model of $TD$, then $E^I=Exp(E)^I$ holds.

**Proof:** Since $I$ is a model of $TD$, we can conclude $C_i^I=D_i^I$, for any term definition item $C_i \equiv D_i$ in $TD$. Then replace one of defined term $C_j$ occurring in $E$ with $D_j$ and obtain a new term formula $E'$. We have $E^I=E'^I$ since $C_i^I=D_i^I$. Moreover, $Exp(E)$ can be obtained through the above replacing process in finite times until all defined terms are replaced with primitive terms, so $E^I=Exp(E)^I$ holds.

**Proposition 2:** Given an ontology $O=<T, X, TD, XD, TR, TC>$, where $TD=\{C_1 \equiv D_1, C_2 \equiv D_2, \ldots, C_n \equiv D_n\}$, and $S$ is a term formula. If $I$ is a model of $Exp(TD)$, then there must exists a model of $TD$ $I'$, such that $S^{I'}=Exp(S)^I$.

**Proof:** Let $T_p=\{B_1, B_2, \ldots, B_m\}$ be the Primitive Term Set. Since $TD=\{C_1 \equiv D_1, C_2 \equiv D_2, \ldots, C_n \equiv D_n\}$, we have the Defined Term Set $T_d=\{C_1, C_2, \ldots, C_n\}$. Suppose $Exp(TD)=\{C_1 \equiv D_1', C_2 \equiv D2', \ldots, C_n \equiv D_n'\}$, i.e. $D_i'=Exp(D_i)$. If $I$ is a model of $Exp(TD)$, then $C_i^I=D_i'^I$ holds for any term definition item in $Exp(TD)$. Then we use $I$ to build a new ontology interpretation $I'$, such that $B_i^{I'}=B_i^I$ for any primitive term $B_i$; $C_i^{I'}=D_i'^I$, for any defined term $C_i$. With the new interpretation $I'$, we have $S^{I'}=Exp(S)^I$ for any term formula $S$, which result in $D_i^{I'}=Exp(D_i)^I$. Moreover, since $C_i^{I'}=D_i'^I=Exp(D_i)^I$, we can conclude $C_i^{I'}=D_i^{I'}$, i.e. $I'$ is a model of $TD$.

## *Individual Set and Instantiation Assertion Set*

*Individual Set* is a set of individuals. *Instantiation Assertion Set* consists of class instantiation assertions, property instantiation assertions and individual inequality assertions.

A class instantiation assertion, written as $C(a)$, states that individual $a$ belongs to class $C$. a property instantiation assertion, written as $P(a, b)$, states that there exists a relation $P$ between $a$ and $b$, and $b$ is called the value of $a$ about property $P$. Individual inequality assertion, written as $a \not\doteq b$, means that the two objects denoted by $a$ and $b$ are distinct.

Given an ontology interpretation $I$, if a class instantiation assertion $C(a)$ holds, then $a^I \in C^I$. If a property instantiation assertion $(a, b)$ holds, then $(a^I, b^I) \in P^I$. If an individual inequality assertion $a \not\doteq b$ holds, then $a^{I_1} \not\doteq b^I$.

**Def. 10 (Model of Instantiation Assertion):** Given an ontology $O=<T, X, TD, XD, TR, TC>$, if there exists an ontology interpretation $I$ making an instantiation assertion $\alpha$ holds, then $I$ is said to be a model of $\alpha$. If $I$ is a model of all the instantiation assertions in $XD$, then $I$ is called a model of $XD$.

**Def. 11 (Expansion of Instantiation Assertion):** Given an ontology $O=<T, X, TD, XD, TR, TC>$. $C(a)$ is a class instantiation assertion in $XD$, and $Exp(C)$ is an expansion of $C$ with respect to $TD$. $Exp(C)(a)$ is said to be the expansion of $C(a)$ with respect to $TD$. Through transforming each class instantiation assertion into the form of expansion, we can get a new Instantiation Assertion Set $XD'$. The new set $XD'$ is called the expansion of $XD$ with respect to $TD$, denoted as $Exp(XD)$.

## *Term Restriction Set*

Given an ontology $O=<T, X, TD, XD, TR, TC>$, the Term Restriction Set $TR$ is a set of term relationships in the form of subsumption, equivalence or disjointness, which is intended to restrict the logical relationship between terms in $T$. Let $D$ and $E$ be two term formulas, then the meaning of the three kinds of relationship are as follows:

(i) $D$M$E$ states that every instance of the class denoted by $D$ is also an instance of the class denoted by $E$.
(ii) $D \equiv E$ states that every instance of the class denoted by $D$ is also the instance of the class denoted by $E$, and vice versa.

(iii) $D \Cap E$ states that the two classes denoted by $D$ and $E$ respectively do not have any instance in common.

$Exp(D)\text{M}Exp(E)$, $Exp(D)\equiv Exp(E)$ and $Exp(D) \Cap Exp(E)$ are called the expansion of $D\text{M}E$, $D\equiv E$ and $D \Cap E$ respectively.

**Def. 12 (Model of Term Restriction):** Given an ontology $O=<T, X, TD, XD, TR, TC>$, if there exists an ontology interpretation $I$ satisfying a term relation $R$ in $TR$, then we say $I$ is a model of $R$. If $I$ is a model of all the term relation in $TR$, then $I$ is called a model of $TR$.

**Def. 13 (Expansion of Term Restriction Set):** Given an ontology $O=<T, X, TD, XD, TR, TC>$. For convenience, we here assume $TR=\{D_1\text{M}E_1, D_2\equiv E_2, D_3 \Cap E_3\}$. If each term relation in $TR$ has been transformed into the expansion form, a new Term Restriction Set $TR'=\{Exp(D_1)\text{M}Exp(E_1), Exp(D_2)\equiv Exp(E_2), Exp(D_3) \Cap Exp(E_3)\}$ is obtained. $TR'$ is said to be the expansion of $TR$, written as $Exp(TR)$.

**Def. 14 (Model of Expansion of Term Restriction Set):** Given an ontology $O=<T, X, TD, XD, TR, TC>$, if there exists an ontology interpretation $I$ satisfying all the term interpretations in $Exp(TR)$, then we call $I$ a model of $Exp(TR)$.

**Proposition 3:** Let $O=<T, X, TD, XD, TR, TC>$ be an ontology. If $TD$ and $TR$ have a model $I$ in common, then $I$ is also a model of $Exp(TR)$.

**Proof:** Let $TR$ be $\{D_1\text{M}E_1, D_2\equiv E_2, D_3 \Cap E_3\}$ for the sake of simplicity, then we have $Exp(TR)=\{Exp(D_1)\text{M}Exp(E_1), Exp(D_2)\equiv Exp(E_2), Exp(D_3) \Cap Exp(E_3)\}$. If $TD$ and $TR$ have a common model $I$, then $D_1^I\subseteq E_1^I$, $D_2^I=E_2^I$ and $D_3^I\cap E_3^I=\varnothing$ hold since $I$ is a model of $TR$. Moreover, $I$ is also a model of $TD$, so we can conclude that $C^I=Exp(C)^I$ holds for any term formula $C$ according to Proposition 1. With the above conclusion, we can further obtain that $Exp(D_1)^I\subseteq Exp(E_1)^I$, $Exp(D_2)^I=Exp(E_2)^I$, $Exp(D_3)^I\cap Exp(E_3)^I=\varnothing$. So $I$ is a model of $Exp(TR)$.

**Proposition 4:** Let $O=<T, X, TD, XD, TR, TC>$ be an ontology, where $TD=\{C_1\equiv D_1, C_2\equiv D_2,\ldots, C_n\equiv D_n\}$. $T_p=\{B_1, B_2,\ldots, B_m\}$ is the set of primitive terms in $TD$, and $S$ is a term formula. If $I$ is a model of $Exp(TR)$, then there must exist a common model $I'$ of both $TD$ and $TR$, such that $S^{I'}=Exp(S)^I$.

**Proof:** Since $TD=\{C_1\equiv D_1, C_2\equiv D_2,\ldots, C_n\equiv \text{D}_n\}$, the set of defined terms in $TD$ is $T_d=\{C_1, C_2,\ldots, C_n\}$. Suppose that $Exp(TD)=\{C_1\equiv D_1', C_2\equiv D_2',\ldots, C_n\equiv D_n'\}=\{C_1\equiv Exp(D_1), C_2\equiv Exp(D_2),\ldots, C_n\equiv Exp(D_n)\}$ and $TR=\{D_1\text{M}E_1, D_2\equiv E_2, D_3 \Cap E_3\}$, we can get $Exp(TR)=\{Exp(D_1)\text{M}Exp(E_1), Exp(D_2)\equiv Exp(E_2), Exp(D_3) \Cap Exp(E_3)\}$. If $I$ is a model of $Exp(TR)$, then $Exp(D_1)^I\subseteq Exp(E_1)^I$, $Exp(D_2)^I=Exp(E_2)^I$, $Exp(D_3)^I\cap Exp(E_3)^I=\varnothing$. Then we use $I$ to construct a new ontology interpretation $I'$, such that $B_i^{I'}=B_i^I$, for any primitive term $B_i$; $C_i^{I'}=D_i'^I$, for any defined term $C_i$; $a^{I'}=a^I$, for any individual $a$. With the new constructed interpretation $I'$, we can conclude $S^{I'}=Exp(S)^I$ for any term formula $S$, which result in $D_i^{I'}=Exp(D_i)^I$. Moreover, since $C_i^{I'}=D_i'^I=Exp(D_i)^I$, we can conclude $C_i^{I'}=D_i^{I'}$, i.e. $I'$ is a model of $TD$. Furthermore, according to the above conclusion: $Exp(D_1)^I\subseteq Exp(E_1)^I$, $Exp(D_2)^I=Exp(E_2)^I$, and $Exp(D_3)^I\cap Exp(E_3)^I=\varnothing$, we can obtain that $D_1^{I'}\subseteq E_1^{I'}$, $D_2^{I'}=E_2^{I'}$ and $D_3^{I'}\cap E_3^{I'}=\varnothing$, i.e. $I'$ is a model of $TR$.

## Ontology Validation

### *Term Validation*

Ontology Validation includes term validation and instantiation validation. Term validating includes: term satisfiability validation, term subsumption validation term equivalence validation and term disjointness validation.

#### Term Satisfiability Validation

Given an ontology $O=<T, X, TD, XD, TR, TC>$ and a term formula $D$, if there exists an ontology interpretation $I$, such that $D^I\neq\varnothing$, then $D$ is said to be *satisfiable* and *unsatisfiable* otherwise. If there exists a model of $TR$, such that $D^I\neq\varnothing$, then $D$ is *satisfiable* with respect to $TR$ and *unsatisfiable* otherwise. If there exists a common model of both $TR$ and $TD$, such that $D^I\neq\varnothing$, then $D$ is said to be *satisfiable* with respect to $TR$ and $TD$, or else *unsatisfiable* with respect to $TR$ and $TD$.

In the development of domain ontology, a new term is constructed, possibly in term of others that have been defined before. During this process, it is necessary to find out whether the newly defined term makes sense or whether it is contradictory. From a logical point of view, a term makes sense for us if there is some interpretation that satisfies both *TD* and *TR* (that is, a model of *TD* and *TR* in common) such that the term denotes a nonempty set in that interpretation.

### Term Subsumption Validation

Given an ontology *O*=<*T*, *X*, *TD*, *XD*, *TR*, *TC*> and two term formulas *D* and *E*, if $D^I \subseteq E^I$ holds for all the ontology interpretation *I*, then *D* is said to be *subsumed* by *E*, or *E* subsumes *D*, denoted as ╞*D*M*E*. If in all the models *I* of *TR*, $D^I \subseteq E^I$ holds, then we say *TR* entails that *D* is *subsumed* by *E*, denoted as *TR*╞*D*M*E*. If $D^I \subseteq E^I$ holds in all the common models *I* of both *TR* and *TD*, then we say *TR* and *TD* jointly entails that *D* is *subsumed* by *E*, denoted as (*TR*+*TD*)╞*D*M*E*.

### Term Equivalence Validation

Given an ontology *O*=<*T*, *X*, *TD*, *XD*, *TR*, *TC*> and two term formulas *D* and *E*, if $D^I = E^I$ holds for all the ontology interpretation *I*, then we say *D* and *E* are *equivalent*, written as ╞*D*≡*E*. If $D^I = E^I$ holds in all the models *I* of *TR*, then we say *TR* entails that *D* and *E* are *equivalent*, written as *TR*╞*D*≡*E*. If in all the common models *I* for both *TR* and *TD*, $D^I = E^I$ holds, then we say *TR* and *TD* jointly entails that *D* and *E* are *equivalent*, written as (*TR*+*TD*)╞*D*≡*E*.

### Term Disjointness Validation

Given an ontology *O*=<*T*, *X*, *TD*, *XD*, *TR*, *TC*> and two term formulas *D* and *E*, if $D^I \cap E^I = \varnothing$ holds for all the ontology interpretation *I*, then we call *D* and *E* are *disjoint*, written as ╞*D* ⋒ *E*. If in all the models *I* of *TR*, $D^I \cap E^I = \varnothing$ holds, then we call *TR* entails that *D* and *E* are *disjoint*, written as *TR*╞*D* ⋒ *E*. If $D^I \cap E^I = \varnothing$ holds in all the common models *I* for both *TR* and *TD*, then we call *TR* and *TD* jointly entails that *D* and *E* are *disjoint*, written as (*TR*+*TD*)╞*D* ⋒ *E*.

**Proposition 5 (Reduction to Subsumption):** Given an ontology *O*=<*T*, *X*, *TD*, *XD*, *TR*, *TC*>, for any two term formulas *D*, *E*, we have

(i) *D* is unsatisfiable with respect to *TR* and *TD*, iff (*TR*+*TD*)╞*D*M⊥;
(ii) *TR*+*TD*)╞*D*≡*E*, iff (*TR*+*TD*)╞*D*M*E* and (*TR*+*TD*)╞*E*M*D*;
(iii) *R*+*TD*)╞*D* ⋒ *E*, iff (*TR*+*TD*)╞(*D* ⊓ *E*)M^.

**Proposition 6 (Reduction to Unsatisfiability):** Given an ontology *O*=<*T*, *X*, *TD*, *XD*, *TR*, *TC*>, for term formulas *D*, *E*, we have:

(i) (*TR*+*TD*)╞*D*M*E*, iff *D* ⊓ ¬*E* is unsatisfiable with respect to *TR* and *TD*;
(ii) (*TR*+*TD*)╞*D*≡*E*, iff both *D* ⊓ ¬*E* and ¬*D* ⊓ *E* are unsatisfiable with respect to *TR* and *TD*;
(iii) (*TR*+*TD*)╞*D* ⋒ *E*, iff *D* ⊓ *E* is unsatisfiable with respect to *TR* and *TD*.

Proposition 5 and 6 can be proved easily, so proofs are omitted here. These two propositions imply that all the four kinds of term validation can be reduced to the (un)satisfiability or subsumption.

**Proposition 7:** Given an ontology *O*=<*T*, *X*, *TD*, *XD*, *TR*, *TC*>. *D* and *E* are two term formulas. *Exp*(*D*) is the expansion of *D* with respect to *TD*, and *Exp*(*E*) is the expansion of *E* with respect to *TD*. We have:

(i) *D* is satisfiable with respect to *TR* and *TD*, iff *Exp*(*D*) is satisfiable with respect to *Exp*(*TR*);
(ii) (*TR*+*TD*)╞*D*M*E*, iff *Exp*(*TR*)╞*Exp*(*D*)M*Exp*(*E*);
(iii) (*TR*+*TD*)╞*D*≡*E*, iff *Exp*(*TR*)╞*Exp*(*D*)≡*Exp*(*E*);
(iv) (*TR*+*TD*)╞*D* ⋒ *E*, iff *Exp*(*TR*)╞*Exp*(*D*) ⋒ *Exp*(*E*).

**Proof:**

(i) **Sufficient Condition:** If *Exp*(*D*) is satisfiable with respect to *Exp*(*TR*), then there must exist such a model *I* of *Exp*(*TR*) that satisfies $Exp(D)^I \neq \varnothing$. According to Proposition 4, if *I* is a model of *Exp*(*TR*), then there must exist a common model *I′* of both

*TD* and *TR*, such that $D^{I'}=Exp(D)^{I}$. So we can get $D^{I'}\neq\varnothing$ from $Exp(D)^{I}\neq\varnothing$ and $D^{I'}=Exp(D)^{I}$. That is *D* is satisfiable with respect to *TR* and *TD*.

(i) **Necessary Condition:** If *D* is satisfiable with respect to *TR* and *TD*, then there must exist a common model *I* of both *TR* and *TD*, such that $D^{I}\neq\varnothing$. Since *I* is a model of *TD*, $D^{I}=Exp(D)^{I}$ holds based on Proposition 1. Therefore we have $Exp(D)^{I}\neq\varnothing$. Moreover, according to Proposition 4, we can conclude that *I* is also a model of *Exp*(*TR*). So *Exp*(*D*) is satisfiable with respect to *Exp*(*TR*).

(ii) **Sufficient Condition:** Let *I* be a common model of both *TD* and *TR*, then according to Proposition 3, it is also a model of *Exp*(*TR*) such that $D^{I}=Exp(D)^{I}$ and $E^{I}=Exp(E)^{I}$. Since $Exp(TR)\models Exp(D) \mathrm{M} Exp(E)$ and *I* is a model of *Exp*(*TR*), we can conclude $D^{I'}\subseteq E^{I'}$. That is for any common model *I* of both *TD* and *TR*, we have $D^{I}\subseteq E^{I}$, therefore $(TR+TD)\models D \mathrm{M} E$.

(ii) **Necessary Condition:** First, let *I* be a model of *Exp*(*TR*), then according to Proposition 4, there must exist a common model *I′* of both *TD* and *TR* such that $D^{I'}=Exp(D)^{I}$ and $E^{I'}=Exp(E)^{I}$. Then, since $(TR+TD)\models D \mathrm{M} E$ and *I′* is a common model of both *TD* and *TR*, we can conclude $D^{I'}\subseteq E^{I'}$. That is for any model *I* of *Exp*(*TR*), we have $Exp(D)^{I}\subseteq Exp(E)^{I}$, therefore $Exp(TR)\models Exp(D) \mathrm{M} Exp(E)$.

(iii) According to Proposition 5(ii), $(TR+TD)\models D\equiv E$ is equivalent to both $(TR+TD)\models D \mathrm{M} E$ and $(TR+TD)\models E \mathrm{M} D$. According to Proposition 7(ii), $(TR+TD)\models D \mathrm{M} E$ is equivalent to $Exp(TR)\models Exp(D) \mathrm{M} Exp(E)$ and $(TR+TD)\models E \mathrm{M} D$ is equivalent to $Exp(TR)\models Exp(E) \mathrm{M} Exp(D)$. Therefore, $(TR+TD)\models D\equiv E$ is equivalent to both $Exp(TR)\models Exp(D) \mathrm{M} Exp(E)$ and $Exp(TR)\models Exp(E) \mathrm{M} Exp(D)$. That is $Exp(TR)\models Exp(D)\equiv Exp(E)$.

(iv) According to Proposition 5(iii ), $(TR+TD)\models D \pitchfork$ is equivalent to both $(TR+TD)\models D \mathrm{M} \neg E$ and $(TR+TD)\models E \mathrm{M} \neg D$. According to Proposition 7(ii), $(TR+TD)\models D \mathrm{M} \neg E$ is equivalent to $Exp(TR)\models Exp(D) \mathrm{M} \neg Exp(E)$, and $(TR+TD)\models E \mathrm{M} \neg D$ is equivalent to $Exp(TR)\models Exp(E) \mathrm{M} \neg Exp(D)$. Therefore, $(TR+TD)\models D\equiv E$ is equivalent to both $Exp(TR)\models Exp(D) \mathrm{M} \neg Exp(E)$ and $Exp(TR)\models Exp(E) \mathrm{M} \neg Exp(D)$. That is $Exp(TR)\models Exp(D) \Cap Exp(E)$.

Proposition 7 states that the term validation can be decided by validating the expansion of the term.

## *Instantiation Validation*

Def. 15 (Consistence of Instantiation Assertion): Given an ontology *O*=<*T*, *X*, *TD*, *XD*, *TR*, *TC*> and an instantiation assertion $\alpha$. If there exists such an ontology interpretation *I* that is a model of $\alpha$, then we say $\alpha$ is consistent, and inconsistent otherwise. If *I* is not only a model of $\alpha$, but also a common model of both *TD* and *TR*, then we say $\alpha$ is consistent with respect to *TD* and *TR*. If *I* is a model of *XD*, then we say *XD* is consistent. If *I* is not only a model of *XD*, but also a common model of both *TD* and *TR*, then we say *XD* is consistent with respect to *TD* and *TR*.

**Proposition 8:** Given an ontology *O*=<*T*, *X*, *TD*, *XD*, *TR*, *TC*> and a class instantiation assertion *C*(*a*), we have:

(i) *C*(*a*) is consistent with respect to *TD* and *TR*, iff *Exp*(*C*)(*a*) is consistent with respect to *Exp*(*TR*).
(ii) *XD* is consistent with respect to *TD* and *TR*, iff *Exp*(*XD*) is consistent with respect to *Exp*(*TR*).

**Proof:**
**Sufficient Condition:** If *Exp*(*C*)(*a*) is consistent with respect to *Exp*(*TR*), then there exists a model *I* of *Exp*(*TR*) satisfying $a^{I}\in Exp(C)^{I}$. Since *I* is a model of *Exp*(*TR*), there must exits a common model *I′* of both *TD* and *TR* such that $a^{I'}\in a^{I}$ and $C^{I'}=Exp(C)^{I}$ according to Proposition 4. So we can obtain $a^{I'}\in C^{I'}$, i.e. *C*(*a*) is consistent with respect to *TD* and *TR*.

**Necessary Condition:** If *C*(*a*) is consistent with respect to *TD* and *TR*, then there exists a common model *I* of both *TD* and *TR* such that $a^{I}\in C^{I}$. Since *I* is a model of *TD*, we have $C^{I}=Exp(C)^{I}$ according to Proposition2.2.1, which imply that $a^{I}\in Exp(C)^{I}$. Moreover Proposition 3 states that *I* is also a model of *Exp*(*TR*). So we can conclude that *Exp*(*C*)(*a*) is consistent with respect to *Exp*(*TR*).

Proposition 8 states that the consistence of an instantiation assertion is equivalent to that of the expansion of the instantiation assertion.

**Proposition 9:** Given an ontology *O*=<*T*, *X*, *TD*, *XD*, *TR*, *TC*> and a term formula *C*, we have: *C* is satisfiable with respect to *TD* and *TR*, iff *C*(*a*) is consistent with respect to *TD* and *TR*, where *a* is an arbitrarily chosen individual name.

**Proof:**
**Sufficient Condition:** If $C(a)$ is consistent with respect to *TD* and *TR*, then there exists a common model *I* of both *TD* and *TR*, such that $a^I \in C^I$, that is $C^I \neq \emptyset$, so *C* is satisfiable with respect to *TD* and *TR*.

**Necessary Condition:** If *C* is satisfiable with respect to *TD* and *TR*, then there exists a common model *I* of both *TD* and *TR*, such that $C^I \neq \emptyset$. It mean that there exists at least one individual in Individual Set, which belongs to the class denoted by *C*. Let *a* be the name of the individual, we have $a^I \in C^I$, so $C(a)$ is consistent with respect to *TD* and *TR*.

Proposition 9 states that the consistence of an instantiation assertion can be determined through checking the satisfiability of the term.

## Primary Study on Customer Ontology in CRM

The customer ontology is a specific application of ontology to the customer management domain. In the e-commerce, customers are not seen directly. The information about customers is coming from customers' own utterances, customers' histories, and general market analysis, etc. much of that is implicit and has to be made explicit for use. With the customer ontology, all the departments can make an agreement with customer information consistently and unambiguously within a company. The customer ontology can be regarded as a part of the enterprise ontology (Uschold 1998).

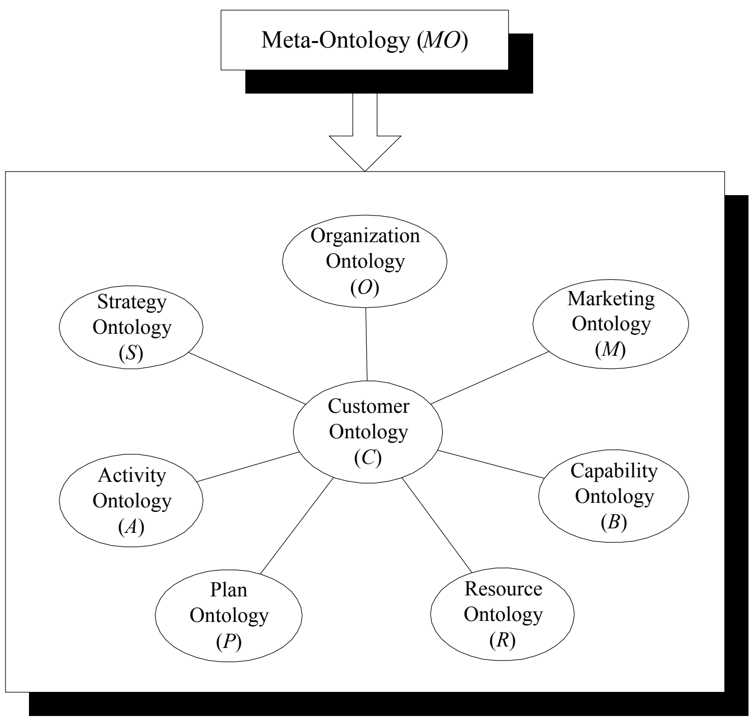


**Figure 2. The Hierarchy of the Enterprise Ontology**

Figure 2 shows the hierarchy of the enterprise ontology, where the customer ontology is on the center of the other six ontologies (the strategy ontology, the organization ontology, the marketing ontology, the activity ontology, the capability ontology, the resource ontology, and the plan ontology). All these ontologies are below the level of the meta-ontology, which provides basic terms / primitives (e.g. Entity, Relationship, Role) for defining the terms in other ontologies. Figure 3 gives enterprise meta-ontology based on the model presented above.

***O*-Meta**=<{Entity, Relationship, Role, Attribute, State Of Affairs, Achieve, Actor Role, Actor, Potential Actor, Time Point, Time Line, Time Interval}, ∅, ∅, ∅, {RelationshipMEntity, RoleMEntity, Actor RoleMRole, AttributeMRelationship, ActorMEntity, Potential ActorMEntity, TimeMEntity}, {*tc*(Entity)="a fundamental thing in the domain being modeled", *tc*(Relationship)="the way that two or more ENTITIES can be associated with each other", *tc*(Role)="the way in which an Entity participates in a Relationship", *tc*(Attribute)="a Relationship between two Entities with the property: within the scope of interest of the model, for any particular attributed Entity the Relationship may exist with only one value Entity", *tc*(State Of Affairs)="a situation; it consists of a set of Relationships between particular Entities; it can be said to hold, or be true",*tc*(Achieve)="the realization of a State Of Affairs", *tc*(Actor Role)="a kind of Role in a Relationship whereby the playing of the Role entails some notion of doing or cognition", *tc*(Actor)="an Entity that actually plays an Actor Role in a Relationship", *tc*(Potential Actor)="an Entity that can play an Actor Role in a Relationship", *tc*(Time Point)="a particular, instantaneous point in time", *tc*(Time Line)="an ordered, continuous, infinite sequence of Time Points", *tc*(Time Interval)="an interval of time specified as two Time Points and bounds on the distance between the two time points"}>

**Figure 3. Enterprise Meta-Ontology**

Recently, we are engaged in the work, *Construction of Customer Ontology in CRM*, which just begins with many ideas yet to be tested and many issues yet to be resolved. Up to now, a conceptual framework of customer domain has been primarily constructed. Figure 4 gives the model of customer ontology, in which the namespace mechanism is adopted for referring terms from other ontologies. For example, *MO* denotes the namespace for the meta-ontology. So, we can introduce terms defined in the meta-ontology, e.g. Entity, to the customer ontology in the form of *MO*: Entity. In the customer ontology, some terms are from the meta-ontology or the other six ontologies using namespaces. WordNet (Miller et al. 2003) is also a helpful tool in the process of term definitions. Now, we have designed a metadata set (called ONT) using RDF Schema, which provides necessary primitives for modeling domain-specific ontology (Wang 2003). With ONT, the customer ontology has been expressed in RDF syntax.

***O*-Customer**=<{Customer, Relationship, Sale, Potential Sale, For Sale, Sale Offer, Vendor, Actual Customer, Potential Customer, High Profit Customer, Low Profit Customer, Reseller, Product, Asking Price, Sale Price, Market, Segmentation Variable, Market Segment, Market Research, Brand, Image, Feature, Need, Market Need, Promotion, Competitor, Field Sales, Mobile Sales, Inside Sales, Telesales, Loyalty, Satisfaction, Customer Description, Emotional Description, Physical Description, Corresponding Description, Purchase Behavior}, {Customer≡*M*:Customer, Vendor≡*M*:Vendor, Sale≡*M*:Sale, Product≡*M*:Product, Market≡*M*:Market, Segmentation Variable≡*M*: Segmentation Variable, Price≡*M*:Price, Mobile Sales≡Field Sales, Telesales≡Inside Sales}, ∅, ∅, {Customer≡Actual Customer ⊔ Potential Customer, RelationshipM*MO*:Relationship, SaleM*MO*:Relationship, VendorM*MO*:Role, Actual CustomerM*MO*:Role, Potential CustomerM*MO*:Role, ResellerMCustomer, High Profit CustomerMCustomer, Low Profit CustomerMCustomer, ProductM*MO*:Role, Asking PriceM*MO*:Role, Sale PriceM*MO*:Role, Segmentation VariableM*MO*:Attribute, NeedM*MO*:Entitiy, Market NeedMNeed, PromotionM*A*:Activity, CompetitorM*MO*:Role, Field SalesM*A*:Activity, Inside SalesM*A*:Activity, LoyaltyMRelationship, SatisfactionMRelationship, Customer Description≡Emotional Description ⊔ Physical Description ⊔ Corresponding Description}, {*tc*(Customer)="a *Person* who pays for products or services", *tc*(Relationship)="a state of connectedness between *Customers* and *Vendors* (especially in emotional connection)", *tc*(Sale)="an agreement between two *Legal Entities* to exchange one goods, services or quantity of money for another good, service or quantity of money", *tc*(Potential Sale)="a possible future *Sale*", *tc*(For Sale)="a situation whereby one *Legal Entity* offers to enter into a *Sale*", *tc*(Sale Offer)="a *For Sale* situation where a particular *Legal Entity* is being offered the *Product*", *tc*(Vendor)="the *Role* of the *Legal Entity* who offers a *Product*, *For Sale* for an *Asking Price* or agrees to exchange a *Product* for a *Sale Price* in a *Sale*", *tc*(Actual Customer)="the *Role* of the *Legal Entity* agreeing to exchange a *Sale Price* for a *Product* in a *Sale*", *tc*(Potential Customer)="any *Legal Entity* who may become an *Actual Customer*", *tc*(Product)="the *Role* of the good, service, or quantity of money, that is offered *For Sale* by a *Vendor* or agreed to be exchanged by the *Vendor* with the *Actual Customer* in a *Sale*", *tc*(Market)="all Sales and Potential Sales within a scope of interest", *tc*(Segmentation Variable)="any *Attribute* determinable from a *Sale* or *Potential Sale* in a *Market*", *tc*(Market Segment)="all *Sales* and *Potential Sales* in a *Market* having defined values of one or more *Segmentation Variables*", *tc*(Need)="a physical, psychological or sociological requirement of a *Customer*", *tc*(Promotion)="an *Activity* whose primary *Purpose* is to improve the Image of a *Product*, *Brand* and/or *Vendor*", *tc*(Competitor)=" a *Role* of a *Vendor* in a *Relationship* with another *Vendor* whereby one offers one or more *Products For Sale* that could limit the *Sales* of one or more *Products* of the other *Vendor*">

**Figure 4. Customer Ontology**

The relationship between a company and its customers involves continuous bi-directional communication and interaction. The relationship can be short-term or long-term, continuous or discrete, repeating or one-time, and attitudinal or behavioral. Even though customers have a positive attitude towards the company and its products, their buying behaviors are highly situational and vary with different background, e.g. national culture. So, in our study, such factor as customers' emotion is considered. We plan to conduct some investigations into China's companies in different business fields, such as insurance, telecom, retail, IT and so on. We will design a series of questionnaires for various customers to better learn their behaviors. We expect to find more key issues concerned by both companies and customers and get distinct features of China from other countries. Based on the empirical research finding, the customer ontology will be revised and refined into a deeper level.

## Conclusions and Future Work

In this paper, an ontology model is constructed using Description Logics. Based on this model, problems of ontology validation are studied. Finally, a model of customer ontology in CRM is discussed. There are still several issues required to be further done in the future, including refining the customer ontology into a deeper level, establishing the ontology-driven customer management system, and exploring how to improve efficiency of CRM using the customer ontology, in particular in Chinese contexts.

### *Acknowledgements*

This work was supported by the National Natural Science Foundation of China under Grant No.70271038. We would like to thank two anonymous reviewers and the track chairman for their valuable comments. We extend thanks also to Dr. Lan Jun and Dr. Hou Liwen, who provided useful insights on this paper.

### *References*

Ashenhurst, R. L. Ontological Aspects of Information Modeling. Minds and Machines, 1996, 6: 287-394.

Baader, F., D McGuinness, D Nardi and P P Schneider. The Description Logic Handbook: Theory, Implementation and Applications, Cambridge: Cambridge University Press, 2002.

Borgo, S., Guarino, N., and Masolo, C. An Ontological Theory of Physical Objects. In Proceedings of Qualitative Reasoning 11th International Workshop. 1997, Cortona, Italy, IAN-CNR, Pavia, pp. 223-231.

Gaines, B. Editorial: Using Explicit Ontologies in Knowledge-based System Development. International Journal of Human-Computer Systems, 1997, 46: 181.

Gruber, T. R. Ontolingua: A translation approach to portable ontology specifications. Knowledge Acquisition, 1993, 5(2): 199 20.

Guarino, N. Formal Ontology, Conceptual Analysis and Knowledge Representation. International Journal of Human and Computer Studies, 1995, 43(5/6): 625-640.

Guarino, N. Formal Ontology and Information Systems. In N. Guarino, editor, Proceedings of the 1st International Conference on Formal Ontologies in Information Systems, FOIS'98, Trento, Italy, pages 3--15. IOS Press, June 1998.

Fox, M.S., Gruninger, M., Enterprise Modeling, AI Magazine, AAAI Press, Fall 1998, pp. 109-121, 1998.

Lenat, Guha. Building Large Knowledge-based System: Representation and Inference in the CYC Project. London: Addison-Wesley Press, 1990

MacGregor. Inside the LOOM Classifier. SIGART Bulletin. 1991, 2(3): 70-76

Mena, E., Kashyap, V., Illarramendi, A., and Sheth, A. Domain Specific Ontologies for Semantic Information Brokering on the Global Information Infrastructure. In N. Guarino (ed.) Formal Ontology in Information Systems. IOS Press , 1998

Miller, George A., Richard Beckwith, Christiane Fellbaum,Derek Gross, and Katherine Miller. Introduction to WordNet: An On-line Lexical Database. Available at **http://www.cogsci.princeton.edu/~wn/5papers.pdf**

Uschold, Mike, Martin King, Stuart Moralee and Yannis Zorgios, The Enterprise Ontology, The Knowledge Engineering Review, Vol. 13, Special Issue on Putting Ontologies to Use (eds. Uschold. M. and Tate. A.), 1998.

Van de Riet, R., Burg, H., and Dehne, F. 1998. Linguistic Issues in Information Systems Design. In N. Guarino, editor, Proceedings of the 1st International Conference on Formal Ontologies in Information Systems, FOIS'98, Trento, Italy, IOS Press, June 1998.

Wang Hongwei, Wu Jiachun, Jiang Fu. Ontology based Metadata Model and Expression Using DAML. Journal of the China Society for Scientific and Technical Information, 2003, to appear (in Chinese).

Wang Hongwei, Wu Jiachun, Jiang Fu. Study on Formal Ontology Model and its Application to Semantic Retrieval. In Li Xiaoming (ed.) Advances of Search Engine and Web Mining in China, Beijing: Higher Education Press, 2003, 205-213. (in Chinese)

Wang Hongwei, Wu Jiachun, Jiang Fu. Study on Naive Ontology Model Based on Description Logics. System Engineering, 2003, to appear. (in Chinese)