# Construct Ontology-based Enterprise Information Metadata Framework

Guoqiang Zhang and Suling Jia and Qiang Wang
School of Economics and Management, Beihang University, Beijing 100191, China
Email: zhanggq07@163.com, jiasuling@126.com, wang6965@sina.com

***Abstract*—The standardization and unification of metadata has a significant impact on enterprise's information integration and sharing. In this paper, information engineering and cluster analysis method is used to define the concepts and relationships of enterprises business meta-data, after that an algorithm is designed to map meta-data to OWL DL ontology and construct ontology-based enterprise information meta-data framework. The proposed framework can unify the definition of enterprise business information and provide a basis for information exchange and system interoperation, as well as provide a premise for knowledge management.**

***Index Terms*—Ontology, MetaData modeling, information framework**

## I. INTRODUCTION

With the improving of information systems and internet, massive amount of data has been collected in enterprises. However, because of generated for different reasons, objected to different business and layers, and followed to different planning and standards, the information has some serious problems which hinder the information sharing and interoperation, such as inconsistent data definition, segmented information system by sectors, data redundant and so on. To solve those problems, the enterprise should establish an enterprise information meta-data framework to unify and standardize the metadata of enterprise business information, which will convenient the enterprise information integration and systems interoperation, and ultimately realize the knowledge sharing.

Recently, ontology has been studied widely. The most famous definition of ontology is by Studer et al: "ontology is an explicit specification of conceptualization" [1]. Generally speaking, ontology is used to describe concepts and concepts' relationships in a domain or an even wider range. Ontology makes these concepts and relationships have unified and clear definition in the domain. Constructing ontology can improve the sharing of knowledge and interoperability of heterogeneous systems and participants. Being powerful to describe the real world, ontology can be used to describe the simple fact as well as beliefs, assumptions, projections and other abstract concepts; it also can be used to describe both static entities and temporal concept, such as events, activities, processes and so on. There are many kinds of ontology description: just a simple concept description, semantic network framework description of concepts and attributes, as well as rich semantics logic description. Today ontology has been widely used in Semantic Web, intelligent information retrieval, information integration, enterprise modeling, conceptual modeling and other fields. Therefore using ontology to unify the definition and description of the enterprise information metadata takes great significance for achieving enterprise knowledge management and heterogeneous systems communication.

In this paper, we give a brief literature review on enterprise metadata and ontology in part two; the construction method of metadata framework is propose in the third section; and then a brief ontology-based business metadata framework about transportation is constructed to demonstrate our method; the conclusion is given in section 5.

## II. ENTERPRISE METADATA AND ONTOLOGY CONSTRUCTION

In past decades, there have been some studies on enterprise metadata. Information engineering and subject database method have been used to establish enterprise data sources indicator system and define the information classification [2]. In another aspect, Common Warehouse Meta Model (CWM), proposed by OMG to support data warehouse application, is studied to establish a multi-level metadata model for supporting information sharing and data retrieval. These methods take an important part for information sharing and data warehouse constructing, but their main focuses are on the description and classification of metadata and take little attention to the semantic and logical relationship between metadata [3].

The foundation theory of ontology is Description logic (DL), which is an object-based formalization for knowledge representation. DL is a first-order logic decidable subset with specific semantics definition, and has a strong expressivity. A DL system contains four basic components: a structure set of concept and relationship; inferences regarding the relationships between classes (Tbox); whereas those reasoning concerned descriptions of individuals (Abox); reasoning mechanism based on TBox and ABox. A description logic system's capacity and reasoning ability depend on the choice of the above elements as well as different assumptions [4]. Description Logic has two basic elements, namely concepts and relations (Role). The former is explained as a subset of the field and the latter is explained as the relationship between individuals of the ontology, which is a kind of binary relation on the field collection. The DL can be used to describe the semantic and logic relationship of concepts more explicitly, which can be a good supplement for

The research is supported by National Science And Technology Project (2006BAG01A05), Humanities and Social Sciences Fund of Education Ministry (06JD6300001), and National Aviation Fund (2007ZG51078).


doi:10.4304/jsw.5.3.312-319

establishing and managing the business information metadata.

There are no unified standards for developing ontology; developers always use their familiar methods and experience to study and experiment their work. Now there are some valid method for the developing, such as Gruninger & Fox's "evaluation", Methontology, Mike Uschold & King's "skeleton", KACTUS, IDEF5, "Seven-Step" and so on [5]. Here we just introduce the "skeleton" and IDEF5 in detail.

Mike Uschold & King's "skeleton", established on the enterprise ontology, is a set of terms and definitions between commercial enterprises. Currently this enterprise ontology is built by the Artificial Intelligence Research Institute of Edinburgh University and its partners IBM, Lloyd's Register, Logica UK Limited and Unilever. The main steps are described as following:

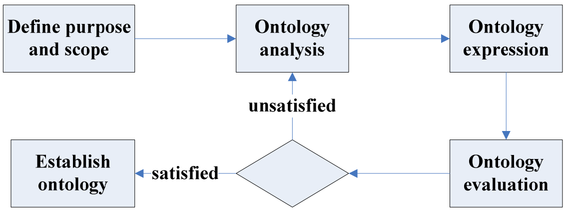


Figure 1. the main setps of "skeleton" apporach

(1)Define the purpose and scope of the ontology: the established ontology must correspond with the research domain or task. The boarder the domain the larger the ontology, so the research scope must be limited.

(2)Ontology analysis: analysis and define the meaning of all terms and relationships. Participation of experts is required in this step. The more understanding in this area, the more perfect ontology built.

(3)Express the ontology: a semantic model is always used.

(4)Evaluate the ontology: ontology is needed to evaluate for its clarity, consistency, integrity, and scalability. Clarity means that all of the terminology should be defined without ambiguity; consistency refers to the logical relationship between terms should be consistent; the integrity of the ontology says that the concept and relationship should be complete, which is hard to achieve and need to be constantly improved; scalability means the ontology applications could be expanded and new concept can be added.

(5)Build ontology: ontology should be examined following all the above criteria, the satisfied one should be stored as documents and unsatisfied should be switched to the second step.

"IDEF5 Ontology Description Capture Method"(IDEF5), proposed by the United States company Knowledge Based Systems Inc. (KBSI), is used to describe and acquire enterprise ontology [6]. Charts language (IDEF5 Schematic Language) and detail description language (IDEF5 Elaboration Language) are used to collect and formalize the concept, attributes and relationships, which will be the main framework of ontology. Constructing ontology through IDEF5 needs five major steps: (1) Define subject and organize teams. (2) Collect data. (3) Analysis data. (4) Develop the ontology preliminarily. (5) Optimize and verify.

These methods are good attempts for knowledge discovery and ontology construction, and play an important role in specific projects. But all of them are exploratory studies and have not been improved as an industry standard. Some of them based on the scientific domains and issues appropriate for the enterprise ontology construction. Even as the "skeleton" and IDEF5, which are appropriate for enterprise ontology, are just a guidance. There are many work needed to do before constructing a specific ontology. So how to combine the academic ontology construction with enterprise’s actual business data is a hot topic of current research.

The expressivity of ontology depends on the chosen of description languages, one of which is OWL. OWL is proposed by the World Wide Web Consortium (W3C), and it has become an international standard semantic Web language [6]. The OWL Web Ontology Language is designed for use by applications that need to process the content of information instead of just presenting information to humans. OWL facilitates greater machine interpretability of Web content than that supported by XML, RDF, and RDF Schema (RDF-S) by providing additional vocabulary along with a formal semantics.

OWL provides three increasingly expressive sublanguages designed for use by specific communities of implementers and users: OWL Lite, OWL DL and OWL FULL. OWL Lite supports those users primarily needing a classification hierarchy and simple constraints. OWL DL supports those users who want the maximum expressiveness while retaining computational completeness (all conclusions are guaranteed to be computable) and decidability (all computations will finish in finite time). OWL Full is meant for users who want maximum expressiveness and the syntactic freedom of RDF with no computational guarantees [7, 8]. W3C has proposed two types of grammar to ensure the completeness and give the powerful expression of OWL: ① exchange syntax, namely RDF/XML Graphs using a set of RDF triples in XML serialization format to express an ontology in order to publish and share Web Ontology; ②abstract syntax, facilitates access to and evaluation of the language. This particular syntax is a high-level abstract syntax for both OWL DL and OWL Lite and has a frame-like style, where a collection of information about a class or property is given in one large syntactic construct, instead of being divided into a number of atomic chunks (as in most Description Logics) or even being divided into even more triples. The same ontology when using different grammars, have the same formal semantics [9].

For expressing the enterprise information metadata ontology, the formal definition of ontology must be given first. This paper studies the translation on the schemas level and not related to instances, so we introduce the definition by author as follows [10]:

**Definition 1**: One ontology is defined as $O=<C,A>$, in which C and A are two disjoint collections. C symbols as the construct set of ontology concept, the elements of C called concept include class concept $C_c$ and attribute concept $C_p$, so C can be expressed as $C=<C_c,C_p>$. A is a finite set of assertions, also known as a collection of terminology axiom. It is an axiomatic set describing the structure of the field and corresponds to TBox in DL. The elements of A known as axiom include class axioms $A_c$ , class restrictions $A_r$, property axioms $A_p$, so the axioms A can be expressed as $A = < A_c, A_r, A_p >$.

## III. CONSTRUCT ENTERPRISE INFORMATION METADATA FRAMEWORK ONTOLOGY

Certain methodology guidance is needed for construct the enterprise information meta-data ontology, with analysis we draw the "skeleton" method to practice constructing ontology. For the specific issues in the constructing process we will design corresponding algorithms to resolve.

### *A. Define the purpose and scope*

This paper mainly discusses how to construct application about enterprise operation, so the scope of the ontology is the metadata of enterprise's business data. Through the research we want to introduce a reference for the enterprise information resource planning and information management, at the same time support interoperability for different information systems.

### *B. Analysis and define the metadata and relationship*

The most important step of constructing the framework is analysis and defining the meaning of the metadata and their relationships, information engineering is appropriate for this. In the paper, we introduce the cluster method to analysis the business activities and entities to establish the hierarchy relationship between the entities [11]. This framework is organized in top-down logical sequence according with the "information domain", "information collection", and "information item".

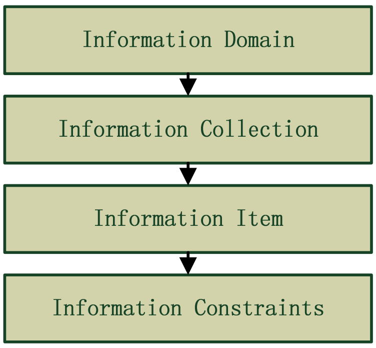


Figure 2. The metadata structure

Information domain corresponds to the main domain of business or organization activities in enterprise, and it is an abstraction of organization function rather than a replica of organization and should be kept stable. Information collection is a set of relevant information items or a subset of information sets, those information items or sets can be combined to describe a complete business processes. For example in the *human resources information* domain, *human basic information* is an information collection, describing the enterprise or organization staff's basic information. Information item, used to describe properties of the business process or object, is the atom data elements and corresponds to the fields or columns in relational database. Constraints are specific restriction on Information item, such as type, length, range, mandatory, and so on. The metadata framework of "Domain - Collection - Item - Constraints" is the foundation of our ontology-based framework to be constructed.

To get the relationship of defined concept, some definition and Algorithm are needed. Here we introduce the information engineering theory to analysis the activities and entities, at the same time, K-means cluster method is used to classify the defined entities.

**Definition 2**:

1) $A=\{a_1,a_2...a_n\}$ : the collection of all operational activities in system;
2) $E=\{e_1,e_2...e_m\}$ : the collection of all relevant entities in system;

Thus the corresponding mappings between operational activities and entities can be fund and analyzed and the E-A matrix $m_{ij}$ can be established:

$$\begin{pmatrix} m_{11} & \cdots & m_{1n} \\ \vdots & \ddots & \vdots \\ m_{m1} & \cdots & m_{mn} \end{pmatrix}$$

Where:

$$m_{ij} = \begin{cases} 0 & \text{entity } e_i \text{ does't participate in acitivity } a_j \\ 1 & \text{entity } e_i \text{ participates in acitivity } a_j \end{cases}$$

This is a 0-1 matrix, which can express the relationships between business operation activities and entities clearly.

**Definition 3**:

1) The amount of activities in which entity $e_i$ is participated symbols as A( $e_i$ );
2) The amount of activities in which entity $e_i$ and $e_j$ are both participated symbols as A( $e_i$ , $e_j$ );
3) $R(e_i,e_j) = A(e_i,e_j) / A(e_i)$ symbols the extent of affinity between entity $e_i$ and $e_j$ 。 The bigger the $R(e_i,e_j)$ , the more the compatibility is and the more the relevance between entity $e_i$ and $e_j$ is.

At the same time we can get the following conclusion:

4) A( $e_i$ , $e_j$ )＝A( $e_j$ , $e_i$ );
5) $0 \le \mathrm{R}(e_i,e_j) \le 1$.

At the view point of relevance between entities, the ultimate goal of relevance analysis is to establish relationships between entities, and those relationships could be able to reflect the extent of relevance between entities. Using relevance analysis the distance between entities must defined firstly, however we can obviously see that $R(e_i, e_j)$ and $R(e_j, e_i)$ are not equal, $\mathrm{R}(e_i, e_j)$ can not be directly used as an index to measure the relevance between any two entities.

**Definition 4**: for any two entities $e_i$ and $e_j$, the extent of their relevance could be defined as $\mathrm{I(e_i,e_j)=A(e_i,e_j)(1/A(e_i)+1/A(e_j))}$, same as $\mathrm{I(e_i,e_j)=R(e_i,e_j)+R(e_j,e_i)}$。

According to this definition we can get that $\mathrm{I(e_i,e_j)=I(e_j,e_i)}$, and $0 \le \mathrm{I(e_i,e_j)} \le 2$, the bigger the $\mathrm{I(e_i,e_j)}$, the more the relevance extent is. Similarly for the extent of relevance between three and more than three entities we can give the following definition: $\mathrm{I(e_1,\cdots,e_i)=A(e_1,\cdots,e_i)[\sum 1/A(e_i)]}$.

**Definition 5**: for any two entities $e_i$ and $e_j$, the distance between them can be defined as $\mathrm{D(e_i,e_j)=2\text{-}I(e_i,e_j)}$. According to this distance definition, we can realize the entities classification using the k-means cluster method.

Based on the above definitions, we introduce k-means algorithm [12] to analysis and classify the entities.

**Algorithm 1**: K-means cluster Algorithm for entities classification

**Input**: information entities set $e_i$; the distance matrix between entities $\mathrm{D(e_i,e_j)}$, the expected classification numeric parameter k

**Output**: { $C_i$ }: the K clusters after clustering

**Steps**:

FOR *i=1* to *k* DO

Select $\mathrm{r_i}$ as the primitive cluster $\mathrm{C_i}$ 's center randomly, $\mathrm{r_i} \in D$

End FOR

While *no change occur in cluster* $\mathrm{C_i}$ DO

FOR *i=1* TO *k* DO //forming cluster

$$\mathrm{C_i} = \{x \in D \mid d(r_i, x) \le d(r_j, x)\}, \forall j = 1, \cdots, k,\ j \ne i$$

End FOR

FOR *i=1* TO *k* DO // Recalculating the cluster center

$r_i = \frac{1}{n_i}\sum C_i$ *// the mean value in cluster* $\mathrm{C_i}$, *every parameter in* $r_i$ *equals to the mean value of corresponding* $\mathrm{C_i}$ *parameter.*

End FOR

END While

Based on the above definitions and Algorithm, we can classify the metadata concepts and provide a basis for constructing our ontology-based enterprise information metadata framework.

*C. Express the framework with OWL DL*

Before constructing, referring to the ER model definition by Calvanese et al. [13], we give the following formal definition of our metadata framework:

**Definition 6**: An information metadata framework model is a ternary equation $\mathrm{F} = (l_f, isa_f, att_f)$, where:

a) $l_f = E_f \cup A_f \cup D_f$ is a finite alphabet partitioned into a set $E_f$ of entity concept symbols contain the defined information domain and information collection, a set $A_f$ of attribute symbols denoted the information item, a set $D_f$ of domain symbols; each domain symbol $D \in D_f$ has an associated predefined basic domain $D^{B_D}$, and we assume the various basic domains to be pairwise disjoint.

b) $isa_f \subseteq E_f \times E_f$, used to express the hierarchy or inheritance relationship between entities, named IS-A relationship, which is injective and acyclic.

c) $att_f$ is a function that maps each entity symbol in $E_f$ to an $A_f$ -labeled tuple over $D_s$. For each entity $E \in E_f$ has $att_f(E) = [A_1 : D_1, \ldots, A_n : D_n]$ (where $[A_i : D_i]$ means the domain of attribute $A_i$ is $D_i$). We assume attributes to be single-valued and mandatory, but we could also easily handle multi-value attributes with associated cardinalities. If an attribute $A \in A_f$ satisfies $A \xrightarrow{F} E$, A is called a key of E, and we assume that all keys of E are single attribute for simplicity.

After the formalize definition, we can see the main relationship between entities is hierarchy or subsumption, and the main relationship between the entities and attributes is composing or one-to-many.

There are semantics-preserving correspondences between metadata framework and OWL DL ontology. Based on the above formalization and referring to the ER model translation [10], we propose a formally translation algorithm to express the framework.

**Algorithm 2**: Translation algorithm for the metadata framework to OWL DL ontology.

**Input**: metadata framework model $F = (l_f, isa_f, att_f)$

**Output**: OWL DL-based Ontology $O = <C, A>$

**Steps**:

1. Translation from the alphabet $l_f$ to the ontology concept identifier set $\phi(l_f)$ , (the name of identifier can be renamed by domain experts interactively):
   a) For every entity $E \in E_f$ create a class concept identifier $\phi(E)$ having the same name as E;
   b) For every attribute $A \in A_f$ create a data type property identifier $\phi(A)$ having the same name as A;
   c) For every domain $D \in D_s$ create an object property identifier $\phi(D)$ having the same name as D;
   d) For every basic domain $D^{B_D} \in D^{B_D}{}_f$ map to a predefined RDF Schema data type $\phi(D^{B_D})$ .
2. Translation from framework to OWL DL ontology:
   a) For every entity $E_i \in E_f$ create class axiom:

      Class( $\phi(E)$ )

   b) For a pairs of entities $E_1, E_2 \in E_f$ and $E_1 isa_s E_2$ create IS_A relationship class axiom :

      SubClassOf( $\phi(E_1)$ $\phi(E_2)$ )

      This expression denotes the hierarchy relationship between entities;

   c) For every domain $D \in D_f$ if D is just defined as the data type then create a map to the corresponding predefined RDF Schema data type; if D is defined having a specific range (such as gender value only in male or female) then create class axiom:

      EnumeratedClass( $\phi(D)$ )

   d) For every entity $E \in E_f$ and $att_f(E) = [A_1 : D_1, \ldots, A_k : D_k]$ create class axiom:

      Class ( $\phi(E)$ partial restriction ( $\phi(A_i)$ allValuesFrom ( $\phi(D_i)$ ) cardinality (1))…);

      For every attribute $A_i \in A_f$ create property axiom (only if $A_i$ is a key of E, the tag "Functional" could be used):

      DatatypePropery( $\phi(A_i)$ domain ( $\phi(E)$ ) range ( $\phi(D_i)$ ) [Functional]

   e) For concept in same classification, create disjoint axiom:

      DisjointClasses($X_1$ $X_2$)

Obviously, all the elements of an enterprise business metadata framework can be translated to OWL DL ontology via the algorithm, which make it possible for knowledge management on information metadata.

## IV. AN TRANSPORTATION CASE OF THE FRAMEWORK

In order to demonstrate and verify the method, based on a project of one Chinese city's Transportation Committee's basic information planning we test it in practical applications. Due to space and presentation convenience, we simplify the original content, choose typical data for introduction.

### A. Analysis and define metadata and relationships

Firstly we use the cluster method to analysis the entities in scope of the Transportation Committee management. The $e_i (i = 1, \cdots 9)$ symbols *Urban Road*, *Highway*, *Track*, *Practitioners*, *Vehicle*, *Transport Company*, *Transport Junction*, *Bus Line* and *Traffic Enforcement*. $a_i (i = 1, \cdots, 6)$ symbols main operational activities of the Committee such as *approval*, *credit*, *traffic enforcing* and *road working* and so on [11].

The primitive E-A matrix is depicted in Fig.3:

| | $a_1$ | $a_2$ | $a_3$ | $a_4$ | $a_5$ | $a_6$ |
|---|---|---|---|---|---|---|
| $e_1$ | 1 | 1 | 0 | 0 | 0 | 1 |
| $e_2$ | 1 | 1 | 0 | 0 | 0 | 0 |
| $e_3$ | 1 | 1 | 0 | 0 | 0 | 0 |
| $e_4$ | 0 | 0 | 1 | 1 | 1 | 0 |
| $e_5$ | 0 | 0 | 1 | 1 | 1 | 0 |
| $e_6$ | 1 | 1 | 1 | 1 | 1 | 1 |
| $e_7$ | 0 | 0 | 0 | 0 | 0 | 1 |
| $e_8$ | 0 | 0 | 0 | 0 | 0 | 1 |
| $e_9$ | 0 | 0 | 0 | 0 | 1 | 0 |

Figure 3. The primitive E-A matrix

After the distance calculation the distance matrix of entities is shown in Fig.4 following:

After clustering with different k values, we found that if k=4 the result is the most reasonable. With analysis and expert's modification; we got the following conclusion:

1) Entity *Urban Road*, *Highway*, *Track* share the same classification called *Traffic Infrastructure Information*;
2) Entity *Practitioners*, *Vehicle*, *Transport Company* share the same classification called *Traffic Management Information*;

| | $e_1$ | $e_2$ | $e_3$ | $e_4$ | $e_5$ | $e_6$ | $e_7$ | $e_8$ | $e_9$ |
|---|---|---|---|---|---|---|---|---|---|
| $e_1$ | 0 | 0.33 | 0.33 | 2 | 2 | 0.5 | 0.67 | 0.67 | 2 |
| $e_2$ | | 0 | 0 | 2 | 2 | 0.67 | 2 | 2 | 2 |
| $e_3$ | | | 0 | 2 | 2 | 0.67 | 2 | 2 | 2 |
| $e_4$ | | | | 0 | 0 | 0.5 | 2 | 2 | 0.67 |
| $e_5$ | | | | | 0 | 0.5 | 2 | 2 | 0.67 |
| $e_6$ | | | | | | 0 | 0.63 | 0.63 | 0.63 |
| $e_7$ | | | | | | | 0 | 0 | 2 |
| $e_8$ | | | | | | | | 0 | 2 |
| $e_9$ | | | | | | | | | 0 |

Figure 4. the distance matrix

3) *Transport Junction* and *Bus Line* are both in the *Transport Services* class;
4) Entity *Traffic Enforcement* as an independent class.

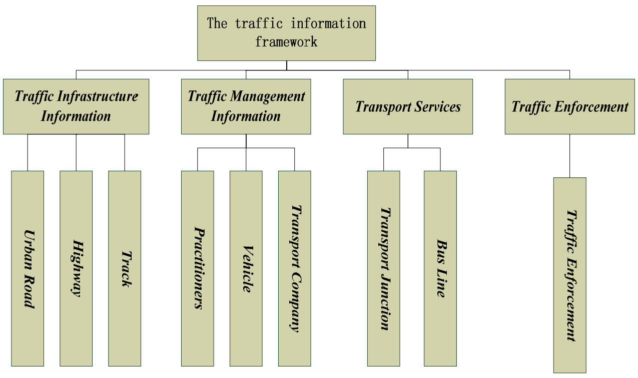


Figure 5. The metadata framework of our case

From the results we can see the classification is in line with human's understanding and can provide a better basis for the enterprises metadata classification. This cluster analysis reveals the relationship between entities and supports us do the following work.

### *B. Expression in OWL DL*

Obtained the concepts and relationships, the ontology description of our application can be specified. In order to guarantee the readability and intelligibility, we use both schemas to illustrate the validity, feasibility and machine readability of the results.

In last section the analyzed entities are as follows: {"*Urban Road*", "*Highway*", "*Track*", "*Traffic Infrastructure Information*", "*Staff*", "*Vehicle*", "*Traffic Enforcement*", "*Traffic Management Information*", "*Transport Junction*", "*Bus Line*", "*Transport Services*", "*Transport Company*"}. For refining the entities and their attributes, {"*Urban Road*"} is divided as {"*Road Profile*", "*Road Surface*" ...}, in which the {"*Road Profile*"} includes {"*Road Basic Information*", "*Road Segment Information*"}. Every concept corresponds to a class, the leaf class has attributes. The top class can be described as:

Classs("*Traffic_Infrastructure_Information*")

The expression in exchange syntax is:

```
<owl:Class rdf:about="#Traffic_Infrastructure_Information">
   <owl:disjointWith>
     <owl:Class rdf:about="#Traffic_Management_Information"/>
   </owl:disjointWith>
   <owl:disjointWith>
     <owl:Class rdf:ID="Human_Rescource"/>
   </owl:disjointWith>
   <owl:disjointWith>
     <owl:Class rdf:ID="Document_Management"/>
   </owl:disjointWith>
   <owl:disjointWith>
     <owl:Class rdf:ID="Organization_Information"/>
   </owl:disjointWith>
   <owl:disjointWith>
     <owl:Class rdf:ID="Fixed_Assets"/>
   </owl:disjointWith>
   <owl:disjointWith>
     <owl:Class rdf:ID="Transport_Services"/>
   </owl:disjointWith>
   <owl:disjointWith>
     <owl:Class rdf:ID="Business_Management_Information"/>
   </owl:disjointWith>
</owl:Class>
```

The middle or non-leaf classes which have IS-A relationships can be described as:

SubClassof("*Urban_Road*" "*Traffic_Infrastructure_Information*")

In exchange syntax:

```
<owl:Class rdf:about="#Urban_Road">
   <owl:disjointWith rdf:resource="#Highway"/>
   <owl:disjointWith rdf:resource="#Track"/>
   <rdfs:subClassOf rdf:resource="#Traffic_Infrastructure_Information"/>
</owl:Class>
```

The leaf classes contain attributes; we take "*Road Basic Information*" for example:

Classs("*Road_Basic_Information*" partial restriction("*Road_Code*" allValuesFrom("&xsd;nonNegativeInteger") cardinality(1))…)

In exchange syntax:

```
<owl:Class rdf:ID="Road_Basic_Information">
   <rdfs:subClassOf>
     <owl:Restriction>
```

```
            <owl:cardinality
        rdf:datatype="http://www.w3.org/2001/XMLSch
        ema#int" >1
            </owl:cardinality>
            <owl:onProperty>
              <owl:FunctionalProperty
rdf:ID="Road_Code"/>
            </owl:onProperty>
          </owl:Restriction>
        </rdfs:subClassOf>
        <rdfs:subClassOf>
          <owl:Restriction>
            <owl:onProperty>
              <owl:DatatypeProperty
 rdf:ID="Startpoint_Name"/>
            </owl:onProperty>
            <owl:cardinality
 rdf:datatype="http://www.w3.org/2001/XMLSchema#int
 "
            >1</owl:cardinality>
          </owl:Restriction>
        </rdfs:subClassOf>
        <rdfs:subClassOf>
          <owl:Restriction>
            <owl:onProperty>
              <owl:DatatypeProperty
 rdf:ID="Roade_Name"/>
            </owl:onProperty>
            <owl:cardinality
 rdf:datatype="http://www.w3.org/2001/XMLSchema#int
 "
            >1</owl:cardinality>
          </owl:Restriction>
        </rdfs:subClassOf>
      <rdfs:subClassOf
rdf:resource="#Road_Profile"/>
    </owl:Class>
```

Create data type property axiom for "*Road Code*", expressed as:

DatatypeProperty("*Road_Code*""
domain("*Road_Basic_Information*"))
range("&xsd;nonNegativeInteger")) Functional

In exchange syntax:

```
    <owl:FunctionalProperty rdf:about="#Road_Code
">
     <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#Datatype
Property"/>
    <rdfs:range
rdf:resource="http://www.w3.org/2001/XMLSchema#stri
ng"/>
    <rdfs:domain
rdf:resource="#Road_Basic_Information"/>
     </owl:FunctionalProperty>
```

After step-by-step analysis and constructing, all elements in metadata framework can be translated to ontology, the ultimate ontology-based enterprise information meta-data framework as shown in Fig.6:

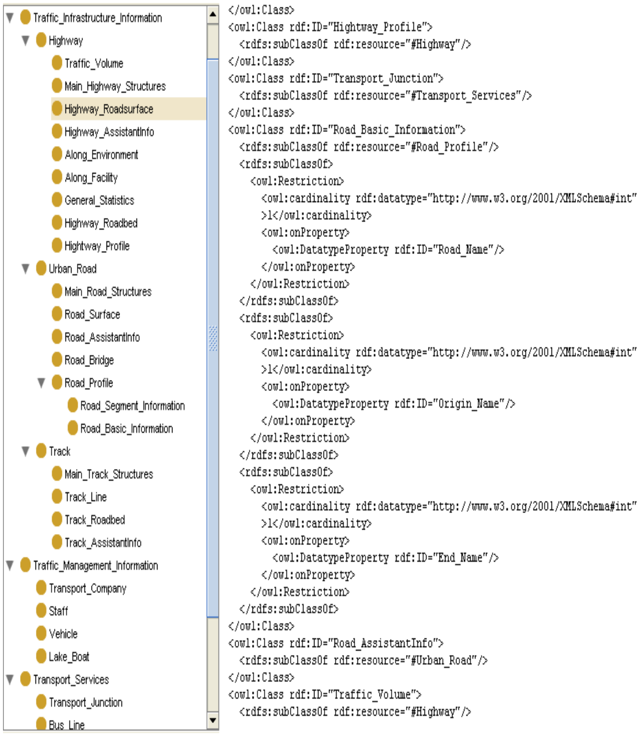


Figure 6. The metadata framework of our case

## V. CONCLUSIONS

The standardization and unification of enterprise metadata has a crucial impact on business-to-business information integration and sharing. In this paper information engineering and cluster analysis is used to define and classify the metadata concepts and their relationships, based on which a metadata framework is established. Then an algorithm is designed to map the components in the metadata framework to OWL DL-based ontology.

The knowledge-based enterprise information metadata framework provides a more complete view of the enterprise information, from which the enterprise managers and decision makers can have a more complete view over information resources and get more support. At the same time, this framework standardizes and unifies the definition of concepts and relationships among different departments and businesses, and will improve the communication and interoperation of information and knowledge.

## ACKNOWLEDGMENT

The research was supported by National Science And Technology Project (2006BAG01A05), Humanities and Social Sciences Fund of Education Ministry (06JD6300001), and National Aviation Fund (2007ZG51078).

## REFERENCES

[1] Neches R, Fikes R E, Gruber T R, et al. Enabling Technology for Knowledge Sharing[J]. AI Magazing,1991,12(3):36-56

[2] Gao Fuxian. Information Resource Planning - information technology foundation engineering [M]. Tsinghua University Press, 2002.(In Chinese)
[3] OMG. Common Warehouse Metamodel Specification, v1.1. http://www.omg.org/technology/documents/formal/cwm.htm
[4] Borst W N. Construction of Engineering Ontologies for Knowledge Sharing and Reuse[D]. PhD thesis, University of Twente, Enschede, 1997
[5] Li Jing, Meng Liansheng. Ontology Construction of the comparative study. Technology of Library and Information [J],2004,(7):17-22(In Chinese)
[6] KBSI. IDEF5 Ontology Description Capture Overview. 2000-06-23. http://www.idef.com/idef5.html
[7] Deborah L. McGuinness, Frank van Harmelen. OWL Web Ontology Language Overview. W3C Recommendation 20040210. http://www.w3.org/TR/owl-features/
[8] Dean M., Schreiber G. eds. OWL Web ontology language reference. W3C Recommendation. 20040210. http://www.w3.org/TR /owl2ref/
[9] Patel-Schneider P. F., Hayes P., Horrocks I. eds. OWL Web ontology language semantics and abstract syntax. W3C Recommendation. 2004202210. http://www.w3.org/TR/2004/REC-owl-semantics-20040210/
[10] Guoqiang Zhang, Suling Jia. Ontology-based knowledge extraction for relational database schema[C]. 2009 Second International Symposium on Electronic Commerce and Security, 2009: 585~589
[11] Guoqiang Zhang, Suling Jia, et al. The Construction of Base Data Resource Index System on Transportation [A]. The symposium on CNAIS2007[C]. Kunming: 379-382(In Chinese)
[12] Liu Weiyi, Li Weihua, Yue kun. Intelligent Data Analysis[M]. Science Press, 2007. (In Chinese)
[13] Calvanese D., Lenzerini M., Nardi D., Unifying class-based representation formalisms. Journal of Artificial Intelligence Research (JAIR), 1999,11, pp:199-240
[14] Chang-Shing Lee, Y.-F.K., Yau-Hwang Kuo, Mei-Hui Wang, Automated ontology construction for unstructured text documents[J]. Data & Knowledge Engineering, 2007. 60(3). pp: 547-566.
[15] Sung-Shun Weng, H.-J.T., Shang-Chia Liu, Cheng-Hsin Hsu, Ontology construction for information classification[J]. Expert Systems with Applications, 2006. 31(1). pp: 1-12.

**Zhang Guoqiang**, born in 1982, is currently a Ph.D. candidate at School of Economics and Management, Beihang University, Beijing, China. His current research interests include information management and information system, knowledge management, and ontology. He has published over 5 papers refereed journals and conference proceedings.

**Jia Suling**, born in 1954, is professor of School of Economics and Management, Beihang University, Beijing, China. Her current research interests include information management and information system, system dynamics.

**Wang Qiang**, born in 1966, is associate professor of School of Economics and Management, Beihang University, Beijing, China. His current research interests include information management and information system, system dynamics.