

An Integrated Information Platform for Intelligent Transportation Systems Based on Ontology

Jun Zhai, Zhou Zhou, Zhiman Shi and Lixin Shen

School of Economics and Management, Dalian Maritime University, Dalian 116026,
P.R. China zhaijun_dlm@yaho.com.cn

Abstract. In semantic web environment, the integration of transportation information systems develops toward semantic integration. Ontology facilitates the integration of heterogeneous data sources by resolving semantic heterogeneity among them. In this paper, we propose the system architecture of integrated information platform for intelligent transportation systems (ITS) based on ontology, including three main layers: distributed heterogeneous data source layer, information integration layer, and application system layer. Its core is the information integration layer including the ontology server and the integrated database. Key technique applied in the ontology server is studies. Firstly, the domain ontology model for transport system is presented. Then we discuss the approach to semantic integration of XML documents based on ontology. At the end we conclude that ontology is a good tool to integrate traffic information among various heterogeneous traffic management systems on semantic level.

Keywords: *Application integration, Data integration, Ontology, Enterprise information architecture, Enterprise information integration (EII), Enterprise information systems (EIS), Semantic integration, Intelligent transportation systems (ITS)*

1. INTRODUCTION

Intelligent Transportation System (ITS), a developing conformation of transportation system in the information times, has been integrating a variety of advanced technologies, especially information technology [1].

Now many kinds of urban traffic management systems are established on their own data storage style (such as database or data files), which operate the data source directly, lacking data share and exchange. It is vital to solve that how to share and exchange the data among various transportation management systems, and how to utilize the data sufficiently and improve the level of urban traffic management. It is vital to solve the problems that how to share and exchange the data among various transportation management systems and how to utilize the data sufficiently to improve the level of urban traffic management. It is necessary to establish an Integrated

Information Platform (IIP) for ITS, which processes various data from different traffic data sources located in various traffic management systems. It can serve as a bridge of data share and exchange among different application systems and data sources, which provides traffic management decision-making support, information service function etc [2-4].

We found some projects similar to the integrated information platform in some countries. In 2001, the MOST (Ministry of Science and technology) of China appointed ten cities as ITS demonstration cities, including Beijing, Shanghai, Guangzhou, Shenzhen etc. The integrated information platform is an important part in the ITS planning of each city. Meanwhile many scholars and some companies have begun the research and development of the integrated information platform and achieved some valuable results. In other countries, such as England, Singapore, America, etc, there are also some similar research and implement project as IIP for ITS [5].

IIP for ITS mainly solves the heterogeneity among diverse information sources. Amit Sheth [6] has classified heterogeneity into four categories: system, syntax, structure and semantic. The system heterogeneity includes hardware and operating systems; the syntax heterogeneity includes different languages and data representations; the structure heterogeneity includes different data models; and semantic heterogeneity includes the semantics of user's information request and those of information sources. Many of the heterogeneity will overcome if XML/DTD is going to be adopted for data publishing, which is used in present research works. However the semantic heterogeneity remains [7].

In the WWW environment, the integration of current information systems turns to semantic level due to the more diversity and heterogeneity of data.

The semantics of diverse information sources are captured by their ontologies, i.e., the terms and relationships among them. Ontologies are emerging as an important tool for constructing sharable and reusable knowledge repositories and supporting their interaction [8-10].

In this paper, we present an IIP system architecture based on ontology to achieve the semantic integration for ITS. The remainder of the paper is structured as follows. In section 2 the IIP system architecture is described. In section 3 the domain ontology model for transportation system is introduced. In sections 4 we discuss the approach to semantic integration of XML documents based on ontology. Finally, in section 5 we present some conclusions and indicate directions for future work.

2. IIP SYSTEM ARCHITECTURE BASED ON ONTOLOGY

IIP forms an abstract homogeneous environment for the application system through integrating the distributed heterogeneous subsystems and providing transparent service based on semantic for users. The system architecture of IIP is shown as Figure 1, including three layers: distributed heterogeneous data source layer, information integration layer, and application system layer.

2.1 Distributed Heterogeneous Data Source Layer

The distributed heterogeneous data source layer is mainly consisted of various traffic management systems, such as traffic signal control system, emergency management system, public transport management system, video monitoring system, and variable message sign system etc, which are the main data sources of IIP. The compatible data sources of IIP include relational database, real-time database, web data, XML documents etc. As the data sources of IIP, various traffic management systems can share and exchange data through integrated transportation information platform on semantic level. The main types of traffic data are shown in Table 1.

Table 1. The Main Kinds of Traffic Data

Type	Content
Space geographical data	The city transportation concerned spatial data and spatial attribute data, which include the elements of city-road-net such as road, crossing and park.
Traffic infrastructure data	Hardware infrastructure data exclude space geographical data, which include communicate, examination and control infrastructure in transportation.
Traffic management operation data	The real-time data of traffic flow, traffic signal control, electric polite, equipment fault etc.
Historical data	The historical data of traffic flow, accident and peccancy.
Other concerned data	The transportation concerned data (such as transfer info, weather report and schedules) and other traffic service concerned yellow page information (such as hotel, train, service station etc).

2.2 Information Integration Layer

The information integration layer is the key of integrated information platform, which mainly includes two parts: ontology server and integrated database. The main functions of ontology server are as follows:

1. To achieve the semantic upgrade of data sources. The server uses the software reverse engineering approach to extract an initial ontology from given data sources and their application programs. Ontology designers will refine this initial ontology. Thus every data source generates own local ontology. Ontology reasoning then is used to check ontology consistency and to merge ontologies to obtain the global ontology, which provides the user access to the data with a uniform query interface to facilitate the formulation of a query on all the data sources. The server includes the mapping database, which records the mapping information between ontologies, and ontology and data sources. These mappings are primitive concept mappings.

2. To achieve the semantic annotation to the integrated database. A set of XML schema or DTDs derived from the global ontology. The global ontology defines the semantics of the terms used to model the data. The set of XML schema or DTDs is used to mark up the data in the integrated database. When an application or agent requires data from the information platform, it will construct a set of queries over the encoded data source. If the application or agent does not understand the tags of the XML schema or DTDs associated with the data, it will query the ontology server. The ontology server is capable of mapping terms, returning relationships between terms and attributes associated with the terms.
3. To manage the ontology. Ontology editor and browser are tools that allow users to visually manipulate ontologies. The editor and browser display ontology in a tree-like structure with nodes denoting concepts. The details of the concepts can be displayed by highlighting nodes. In the editor mode, definitions can be modified, nodes can be added or removed. Ontology relationships can also be displayed.

In section 3 and section 4 we discuss the ontology model and integration approach, which are the key technique applied in the ontology server.

2.3 Application System Layer

Achieving the semantic integration, the IIP can deeply, synthetically utilize various traffic data and information. Based on a large amount of traffic data stored in IIP integrated database, with the Data Fusion, Data Mining technology, some application systems may be developed, such as information service system, decision-making support system etc, to provide information service for traffic managers, travelers and other people. Through these ways, traffic data and information can be used more efficiently, and the intelligent level of urban traffic management can be improved as well.

3. THE DOMAIN ONTOLOGY MODEL FOR ITS

Gruber [11] defines ontology as an explicit specification of a conceptualization, i.e. an abstract and simplified representation of real-world entities. An ontology can be viewed as a model of a domain that defines the concepts existing in that domain, their properties and the relationships between them and is typically represented as a knowledge base.

An ontology (O) organizes domain knowledge in terms of concepts (C), properties (P) and relations (R) and can be formally defined as follows.

Definition (Ontology) – An Ontology O is a triplet of the form $O = (C, P, R)$, where:

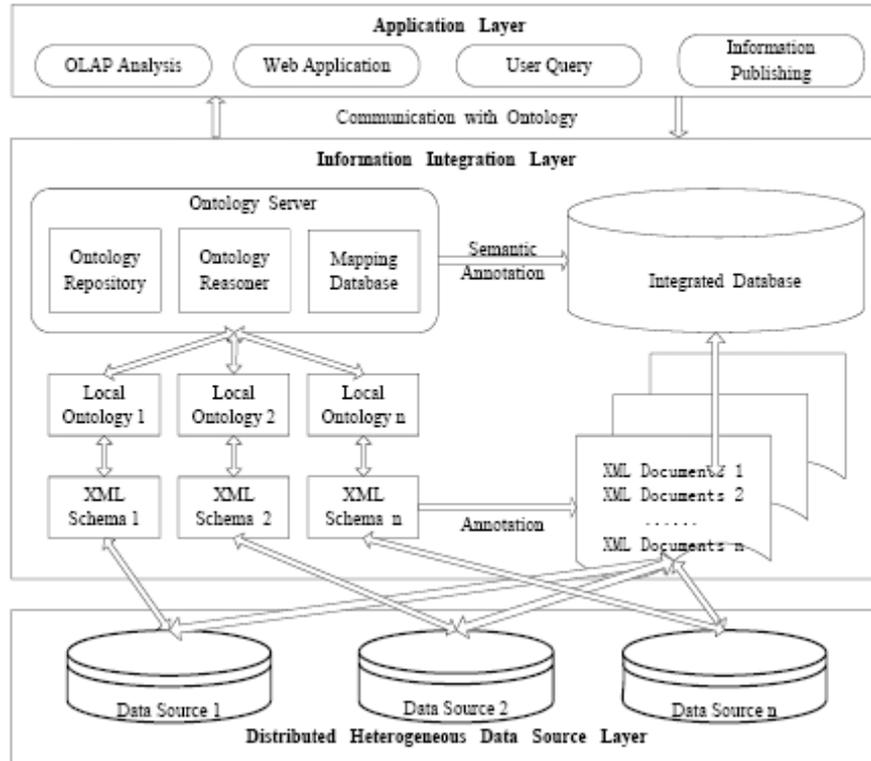


Figure 1. IIP System Architecture Based on Ontology for ITS

1. C is a set of concepts defined for the domain. A concept is often defined as a class in an ontology.
2. P is a set of concept properties. A property $p \in P$ is defined as an instance of a ternary relation of the form $p(c, v, f)$, where $c \in C$ is an ontology concept, v is a property value associated with c and f defines restriction facets on v . Some of the restriction facets are – type (f_t), cardinality (f_c), and range (f_r). The type facet f_t may be any one from the standard data types supported by ontology editors i.e., $f_t \in \{\text{Boolean, integer, float, string, symbol, instance, class ...}\}$. The cardinality facet f_c defines the upper and lower limits on the number of values for the property. The range facet f_r specifies a range of values that can be assigned to the property.
3. $R = \{r \mid r \subseteq C \times C \times R_t\}$ is a set of binary semantic relations defined between concepts in O . $R_t = \{\text{one-to-one, one-to-many, many-to-many}\}$ is the set of relation type.

A set of basic relations is defined as $R_b = \{\approx, \uparrow, \nabla\}$ which have the following interpretations:

1. For any two ontological concepts $c_i, c_j \in C$, \approx denotes the equivalence relation. $c_i \approx c_j \Rightarrow c_i$ is equivalent to c_j . The synonym relation of natural language is

modeled in an ontology using the equivalence relation. If two concepts c_i and c_j are declared equivalent in an ontology then instances of concept c_i can also be inferred as instances of c_j and vice-versa.

2. \uparrow denotes the generalization relation. $c_i \uparrow c_j \Rightarrow c_i$ is a generalization of c_j . When an ontology specifies that c_i is a generalization of c_j , then c_j inherits all property descriptors associated with c_i , and these need not be repeated for c_j while specifying the ontology.
3. $c_i \nabla c_j \Rightarrow c_i$ has part c_j . In an ontology, a concept which is defined as aggregation of other concepts is expressed using the relation ∇ .

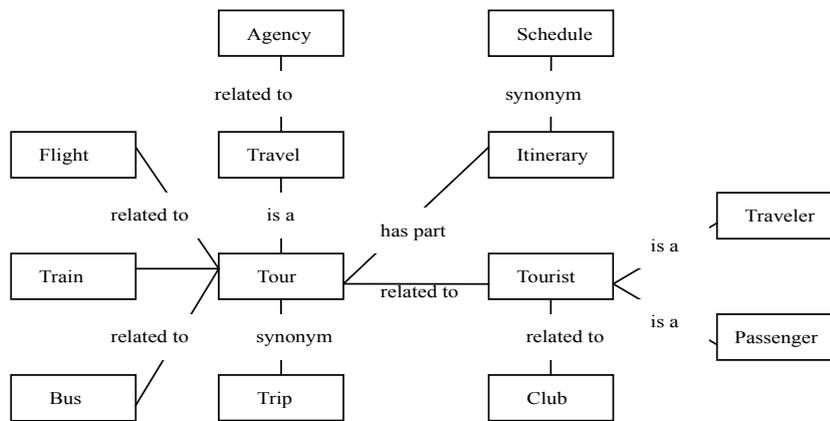


Figure 2. Partial Tourist Information Services Domain Ontology

The individual information services are the vital part of the traveler information services, which assemble correlative information from multi-data-sources (weather bureau, news, agency etc.) for satisfying the personal needs of travelers. The typical application is the travel information afforded for the tourist, in which the partial travel domain ontology is shown in Figure 2. The main concepts include tourist, agency, trip, itinerary etc. The concepts can be related in various ways. The four most commonly used are synonym (equivalent), is-a (generalization), whole-part (has part) and related-to where the semantics of the first three are described above. “Related-to” is used for generic associations between concepts.

In RDFS the concepts of domain ontology are represented as follows:

```
<rdf:Class rdf:ID="Agency"></rdf:Class>
<rdf:Class rdf:ID="Travel product"></rdf:Class>
<rdf:Class rdf:ID="Tour">
  <rdf:SubClassOf rdf:resource="#Travel product"/>
</rdf:Class>
```

The relations are represented as follows:

```
<rdf:Property rdf:ID="has part">
  <rdf:domain rdf:resource="#Tour">
  <rdf:range rdf:resource="# Itinerary">
```

```
</rdf:Property>  
<rdf:Property rdf:ID="synonym">  
  <rdfs:domain rdf:resource="#Tour">  
  <rdfs:range rdf:resource="#Trip">  
</rdf:Property>
```

The property is represented as follows:

```
<rdf:Property rdf:ID="Agency Name">  
  <rdfs:domain rdf:resource="#Agency">  
  <rdfs:range rdf:resource="#String">  
</rdf:Property>
```

The instance of a concept is represented in RDF as follows:

```
<rdf:Description rdf:ID="Dalian Agency">  
  <rdf:type rdf:resource="#Agency"/>  
  <Agency Name>Dalian Agency</Agency Name>  
</rdf:Description>
```

4. THE APPROACH TO SEMANTIC INTEGRATION OF XML DOCUMENTS BASED ON ONTOLOGY

XML is becoming the standard for data interchange on the web and a web-friendly technology for IIP information exchange [4]. However, XML and its schema languages do not express semantics but rather structure, such as nesting information [7]. RDF (Resource Description Framework) and RDFS (RDF Schema) is a data model and support mechanism for representing meta-data of schemas. XML and RDF are the current standards for establishing semantic interoperability on the Web. RDF better facilitates interoperation because it provides a data model that can be extended to address ontology representation techniques [12].

In this section, we discuss the integration approach of XML sources into the global ontology. The ontology integration process contains two steps: schema transformation and ontology merging. In the first step, we use RDFS to model each XML source as a local RDF ontology to achieve a uniform representation basis for the ontology-merging step. The key operation is the preservation of the nesting structure of the XML documents. In the second step, we merge all the local RDF schemas to generate the global ontology. In this process, additional domain-related knowledge (e.g., inheritance) may be introduced. During the merging process, a mapping table is produced to contain the mapping information between the global RDF ontology and local RDF ontologies.

4.1 Schema Transformations

Taking into account XML elements, attributes and their relationships, the transformation from XML to RDF can further include element-level transformation and structure level transformation.

The element-level transformation defines the basic classes and properties of the local RDF ontology according to the transformation correspondences shown in Table 2. No new RDF metadata needs to be defined here because `rdfs: Class` and `rdfs: Property` is enough for the specifications of classes and properties.

The structure-level transformation encodes the hierarchical structures of the XML schema into the local RDF ontology. The encoding involves two relationships: element-attribute relationship and element-subelement relationship. Following the element-level transformation, it is natural to encode the element-attribute relationship as a class-to-literal relationship, and the element-subelement relationship as a class-to-class relationship in RDFS. We define a new RDFS predicate `rdfx: contain` to represent class-to-class relationships. Specifically, we add a new property with its domain being one class (converted from the parent element), its range being the other class (converted from the subelement), and its name being `rdfx: contain`.

Table 2. Element-level Transformation

XML Schema concepts	RDF Schema concepts
Attribute	Property
Simple-type element	Property
Complex-type element	Class

4.2 Ontologies Merging

The process of ontology merging takes multiple local ontologies (encoded in RDFS) as the input and returns a merged ontology as the output. Ontology merging and ontology alignment are widely pursued research topics. In this paper we do not intend to introduce a new technique for ontology merging. Instead, we utilize existing techniques to generate the integrated ontology from the local ontologies. In particular, we use the approach that provides the following functionalities [13]:

1. Merging of classes where multiple conceptually equivalent classes are combined into one class.
2. Merging of properties where multiple conceptually equivalent properties of a class are combined into one property.
3. Merging relationships between classes where conceptually equivalent relationships from one class c_1 to another class c_2 are combined into one relationship (i.e., an RDF property taking c_1 as its domain and c_2 as its range).
4. Copying a class and/or its properties if the same or equivalent class/property does not exist in the target ontology.
5. Generalizing related classes into a more general superclass. The superclass can be obtained by searching an existing knowledge domain or reasoning over a thesaurus.

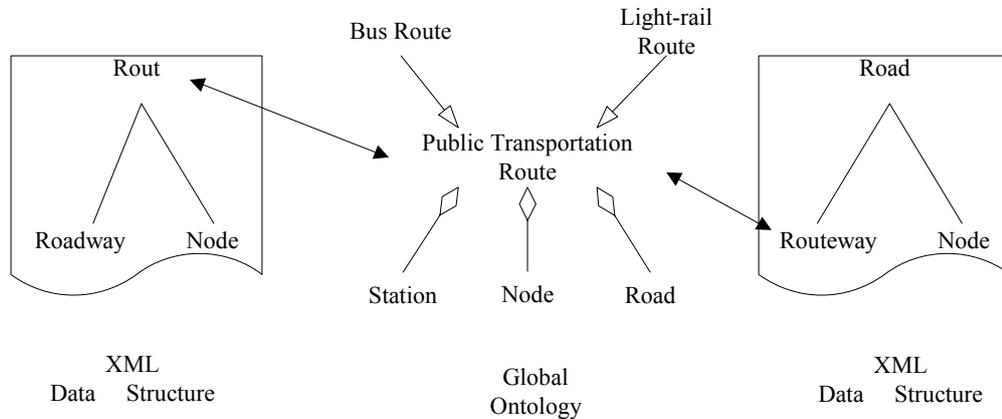


Figure 3. Global Ontology and XML Documents

Figure 3 gives an example, in which the elements “Rout” and “Routeway” of two XML data structures are semantically integrated to the “Public Transportation Route” concept of the global ontology.

5. CONCLUSIONS

Ontology-based integrated information platform is motivated by the following properties:

1. Precise description of data and information. IIP uses standardized terms of the ontology and its inherent semantics to provide a formal description of its data. Queries based on the uniform semantics are less prone to misinterpretation of local information semantics.
2. Dynamic support for multiple contexts or interpretation of data. Traditional schema-based database integration of systems requires costly updates to accommodate new semantics. Using the terms of ontology as meta-constructs or meta-attributes allows proper dynamic interpretation of the different contexts.

Our further researches lay on the automatic integration among different domain ontology, especially among the transportation ontology and other domain ontology such as the GIS ontology and urban ontology.

ACKNOWLEDGEMENTS

The authors would like to thank peer reviewers for commenting this article. This work is supported by the National Natural Science Foundation of China (Grant NO.70540005).

REFERENCES

1. L. Figueiredo, I. Jesus, J.A.T. Machado, J.R. Ferreira, and J.L.M.D. Carvalho, Towards the Development of Intelligent Transportation Systems, in *Proc. of 2001 IEEE Intelligent Transportation Systems Conference* (USA, 2001), pp.1206-1211.
2. W. Liu, D. Sun, W. Song, and L. Fu, A Virtual Common Information Platform For Intelligent Transportation Systems, in *Proc. of the 2004 IEEE Intelligent Transportation Systems Conference* (Washington, D.C., USA, October 36), pp.136-141, (2004).
3. Q. Shi and W. Zheng, Architecture Analysis of Common Information Platform for Intelligent Transportation Systems (ITS) and Its Construction Means, *Journal of Transportation Engineering and Information*. Volume 1, Number 1, pp.41-47, (2003).
4. R. Li, H. Lu, Z. Qian, and Q. Shi, Research of the Integrated Transportation Information Platform Based on XML, in *Proc. of the 8th International IEEE Conference on Intelligent Transportation Systems* (Vienna, Austria, 2005), pp.214-219.
5. R. Qiu, ITS development of Singapore, *ITS communication*. Volume 5, Number 1, pp.5-11, (2003).
6. A.P. Sheth, Changing Focus on Interoperability in Information Systems: from System, Syntax, Structure to Semantics, *Interoperation Geography Information Systems* (Academic Publishers, 1998), pp.5-30.
7. I.F. Cruz, H. Xiao, and F. Hsu, An Ontology-based Framework for XML Semantic Integration, in *Proc. of the International Database Engineering and Applications Symposium (IDEAS'04)* (2004), pp.50-58.
8. Z. Cui and P. O'Brien, Domain Ontology Management Environment, in *Proc. of the 33rd Hawaii International Conference on System Sciences* (2000), pp.1-9.
9. Y. Li, J. Zhai, and Y. Chen, Using Ontology to Achieve the Semantic Integration of the Intelligent Transport System, in *Proc. of 2005 International Conference on Management Science & Engineering (12th) (Volume III)* (2005), pp.2528-2532.
10. J.J. Samper, V.R. Tomás, J.J. Martínez, and L.V.D. Berg, An Ontological Infrastructure for Traveller Information Systems, in *Proc. of the 2006 IEEE Intelligent Transportation Systems Conference* (Toronto, Canada, 2006), pp.1197-1202.
11. T.R. Gruber, A Translation Approach to Portable Ontology Specification, *Knowledge acquisition*. Volume 5, Number 2, pp.199-220, (1993).
12. M. Klein, Interpreting XML Documents via an RDF Schema Ontology, in *Proc. of the 13th International Workshop on Database and Expert Systems Applications (DEXA '02)* (2002), pp.1-5.
13. F. Noy and M.A. Musen, PROMPT: Algorithm and Tool for Automated Ontology Merging and Alignment, in *Proc. of the Seventeenth National Conference on Artificial Intelligence and Twelfth Conference on Innovative Applications of Artificial Intelligence, AAAI/IAAI* (2000), pp.450-455.