Enterprise Knowledge Integration by Semantic Web

Wendong Gu, Guoping Xia*, and Weijia You
1 School of Economics & Management, BeiHang University, Beijing
100083, China
gxia@buaa.edu.cn
* correspondening author

Abstract. In today's enterprise, separate knowledge in different operation systems without integration makes decision-making difficult. To solve this problem, an application framework, which supports enterprise knowledge integration (EKI), is put forward. The framework has a multi-layer architecture and is implemented by semantic web technologies based on enterprise's legacy systems. Four key technologies are studied: data semantic transformation, unified knowledge representation by RDF/RDFS, constructing knowledge chain and knowledge query. A referenced approach to develop the framework is also proposed. Finally, a practical application in a large-scale corporation of Hydropower Engineering is further provided to illustrate the framework.

1 Introduction

Nowadays, knowledge has emerged as the most important determinant of accelerating innovation, core competition and competitive advantage. Generally, there are some separate application systems in a company to manage the data and data flow, which are only capable of coping with operation-layered problems. Whereas, it is more valuable to transform data into knowledge, integrate dispersed knowledge and build up knowledge repository via knowledge chain.

Many scholars have made a lot of achievements in the theory and technology on Knowledge Management (KM). Therein, [1] defines some concepts about KM, analyzes connotations of Knowledge, KM and KM systems, and puts emphasis on arguing about the use of Information Technologies in KM activities; In order to achieve the goal of KM, i.e. Knowledge Integration, and to solve problems by the multiple vendor application systems running in one company, a solution is proposed in [2], which is based on the distributed relation databases, XML and metadata; Ontology is becoming a major issue recently, and [3] describes a system architecture and its implementation, which makes use of ontologies to organize, share and query

Please use the following format when citing this chapter:

Gu, W., Xia, G., You, W., 2006, in International Federation for Information Processing, Volume 205, Research and Practical Issues of Enterprise Information Systems, eds. Tjoa, A.M., Xu, L., Chaudhry, S., (Boston:Springer), pp.203-212.

knowledge objects; Production design is a knowledge-intensive work, and an Internet-based distributive knowledge integration system to meet the demand of product designers on various knowledge resources is proposed in [4], which is based on STEP standard and combines the artificial neural networks and active server pages techniques. The relative studies have accomplished many valuable results. However, most of them only focus on one special part of knowledge integration. The further work needs to focus on how to develop highly efficient knowledge integration applications in the companies from systematical point of view.

In this paper, by applying Semantic Web technology, an application framework of enterprise knowledge integration systems is put forward. The framework is multi-layered and based on legacy enterprise application systems. The rest of the paper is organized as follows: Section 2 gives an overview of the framework's main architecture and functionalities; Section 3 describes four key technologies; Section 4 presents a referenced approach to build a knowledge integration system; Section 5 shows a practical example to illustrate the framework; And finally, section 6 concludes the paper with additional comments and future work.

2 The Architecture of the Proposed Framework

The formal definition for Knowledge Integration was raised by Grant [5], who indicated that "Knowledge Integration (KI) is the first role of enterprise and the nature of enterprise capability". Enterprise Knowledge Integration (EKI) researches the methods of knowledge acquirement, storage, sharing and reuse, and aims to integrate various knowledge resources and to make company as an organic whole.

This paper proposes a framework of EKI by applying Semantic Web technology. The architecture of the framework is divided into 4 layers: data layer, application layer, knowledge integration layer and knowledge service layer.

(1) Data Layer. It is the basis of the architecture, storing all the data coming form enterprise systems. Under the framework, those data, which can be semantically transformed to be knowledge, are the focus of the data layer.

(2) Application layer. It is the collection of all the application systems. Most enterprises have some application systems already, such as manufacture management system (MMS), equipment and material management system (EMS), customer relationship management system (CRMS) and human resource management system (HRMS), etc. In the architecture showed in Fig. 1, knowledge is acquired from all these systems and then transformed to proper format.

(3) Knowledge integration layer. It is the kernel of the whole architecture, realized by semantic web technology. The final target of EKI is to share, reuse the knowledge and improve the knowledge innovation. One of the most important problems is how to analyze, understand and deal with the knowledge automatically by computer, which is the advantage of semantic web.

(4) Knowledge service layer. It means all kinds of specific services supplied for knowledge workers such as knowledge search engine, knowledge map, etc. The intelligent services can help knowledge workers fulfill their work effectively.

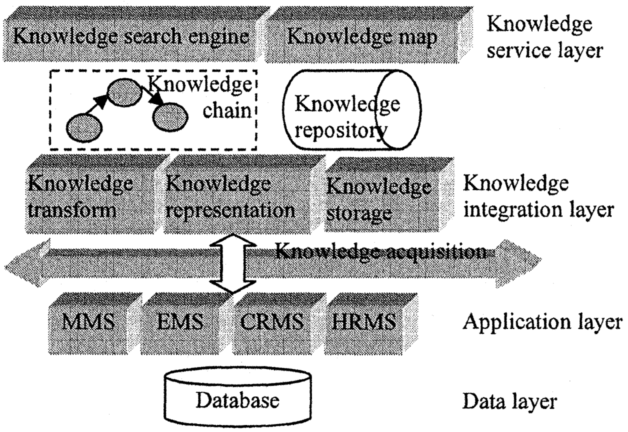

Fig. 1. The Architecture of EKI Application Framework.

3 Key Technologies of the Proposed Framework

Semantic Web [6] was put forward by Tim Berners-Lee. And the researches on semantic web have reaped some achievements already, such as RDF. Researches of next phase should be on the themes of how to put them into application. To realize the framework proposed in this paper, four key technologies are involved.

3.1 Semantically Transforming Data to Knowledge

XML has been the de facto standard for data exchange and transformation. Nearly all the relational databases and information systems can provide business-describing documents easily in the format of XML. However, XML is not strictly a semantic language, lacking strict model, theory and logic. The understanding to a XML document relies on the programs parsing XML. While using semantic web to do system integration, we can attach semantic information to the data. In the semantic web architecture, the language for knowledge representation, such as RDF, are all based on XML, so it is feasible to semantically transform the XML data to ontology [7]. This paper proposes the process of semantically transformation of data, the crucial part of EKI, which covers three layers, data layer, business system layer and knowledge integration layer, see Fig. 2.

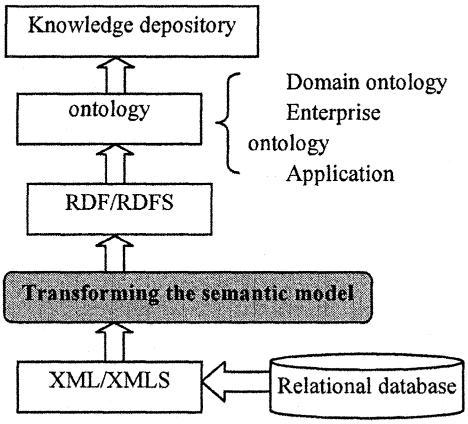

Fig. 2. semantically transforming data to knowledge

3.2 Unified Representation of Knowledge by RDF/RDFS

One of the most important problems, which occur when integrating the enterprise data from various kinds of separate application systems in the knowledge layer, is semantic non-interoperability. Ontology, which is defined by Gruber [8] as an explicit specification of a conceptualization, is considered to be a useful approach to achieve semantic interoperability. For knowledge integration based on ontology, the interoperability must be in the same domain context or the knowledge share must be from semantic transforming agent. Therefore, the representation of domain knowledge must be from domain ontology, which is constructed in each domain. Based on Perez's research [9], domain ontology can be represented as follows.

$DO = \{V, I, R, F^D, A^D\}$, where

DO, the collection of domain ontology;

V, the concept collection of the domain, is usually organized by sorts;

I, the cases collection of the domain while case means the realization of some concepts;

R, the collection of the relationships between concepts, which should subject to $VnR = ¢$ and $R \to V \times V$;

FD, the collection of the functions available in the domain;

AD, the collection of the axioms of the domain.

RDF (Resource Description Framework) is recommended by W3C to describe resource and their semantic criteria. RDF, taking XML as its realizing language, provides a simple and powerful data model. Three types of objects are included: resource, which could be any information except basic data types; property, which describes specific aspect, character and relationship of resource; statement, which is composed of resource, its property and its value. RDF data model could be the basic model for any complicated one. In EKI, domain knowledge, not so complicated and so many as the information on the internet, is more concentrated and structured and therefore is very fit for being represented by RDF. The ontologies representation in the proposed framework is shown in Fig. 3.

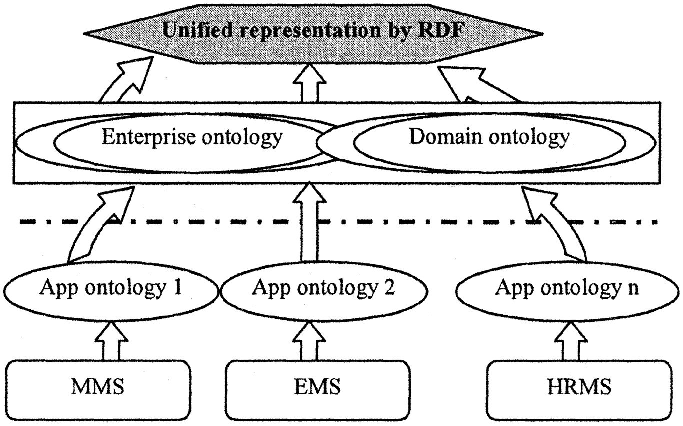

Fig. 3. unified representation by RDF

3.3 Construct Knowledge Chain

Knowledge chain is the crucial task of EKI, which includes two important elements -- knowledge nodes and relationships. Knowledge nodes are combined by various kinds of relationships. Knowledge chain is generally based on work flow of application systems.

In this architecture, knowledge can be communicated on the level of semantics by using RDF in describing unified enterprise knowledge. RDF statement can be represented by a triple <subject, predicate, object>, which can be described easily by directed graph. Therefore, RDF is suitable for the language of knowledge chain, by which all the related knowledge of the domain could be connected on the level of semantics, so that knowledge share and reuse can be greatly improved.

3.4 Knowledge Query

According to the proposed architecture, knowledge query is actually the semantic query to the ontologies represented by RDF. Ontology depository is usually composed by enterprise ontologies and application ontologies. Enterprise ontologies, which provide common concepts collection, are the share vocabularies for different application systems. Application ontologies provide ontologies used by each application.

Different concepts can be used in different systems, which can be interoperated by ontologies mapping and agents. Three steps are followed when doing a knowledge query. First is enterprise ontologies query. Second is transformation from enterprise ontologies to application ontologies. At present, query language based on RDF and XML have been well developed.

RDQL is the ontology query language recommended by W3C. Clause with SELECT is used to recognize the variable returned from the query. Clause with FROM is used to specify the URI used by the RDF model. Clause with WHERE is used to specify the triple in the RDF. Clause with AND is used to specify the Boolean expression. Clause with USING is used to provide abbreviation of URI.

XQeuery, which provides a unified method to search, retrieve XML based on XPath, is the query language for XML recommended by W3C. Both RDQL and XML are similar to SQL in syntax. Therefore, it is easy to transform between them. The procedure of knowledge query is showed in Fig.4.

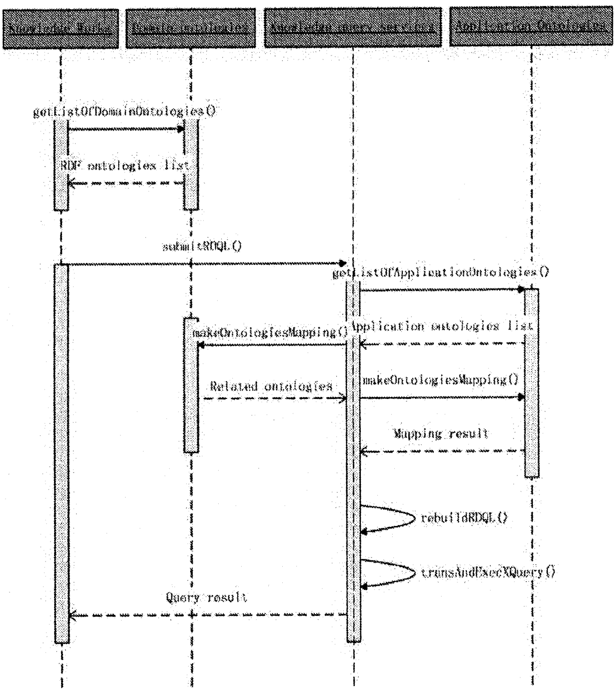

Fig. 4. the process of knowledge query

4 The Developing Approach of the Proposed Framework

The framework proposed above can be developed by eight universal steps which can also be expanded according to your own requirements. For example, Case Based Reasoning (CBR) application is an important content when building an EKI system for supporting those enterprises which put emphasis on products design.

Step 1, to construct ontologies. First, domain ontologies construction, which is lead by domain experts, is to describe the concepts of the domain and their relationships. Second, enterprise ontologies construction, which is lead by enterprise's knowledge workers, is to describe concepts of the enterprise and their relationships. Third, application ontologies construction, which is lead by system developers, is to describe the concepts of specific business and their relationships.

Step 2, to define ontology mapping. In this process, mapping tables are usually used to record the relationships between ontologies.

Step 3, to store data in XML. Define the XML schemas for the data, and transform the data from the relational database to the format of XML by XML Schema.

Step 4, to transform data between XML/XML schema and RDF/RDF schema. This is the foundation of knowledge query.

Step 5, to represent knowledge in RDF.

Step 6, to construct knowledge chain and build knowledge depository.

Step 7, to develop knowledge services, including knowledge search engine, knowledge map, etc.

Step 8, to query knowledge. The query here includes initiative query, which returns the results according to the conditions submitted by the user, and automatic query, also called knowledge recommendation which recommends the useful knowledge to the user automatically by intelligent agent technology.

5 Application of the Proposed Framework

In the hydropower engineering, the project management is a complicated work, which involves plans, contracts, people, materials, financial affairs, and so on. Much correlative information is produced during the process. The following section describes the applying of the proposed framework in a famous large-scale hydropower Chinese corporation.

5.1 The Case Model

The systems include project construction management system, contract management system, human resource management system, and equipment and material management system. Thereinafter, a simplified data mode of the systems is described. It presents the relationship of Engineering, Contract, People, and Equipment, which are the core business entities. As fig. 5, Engineering, Contract, Project Manager, Equipment are application ontologies; Project Owner, Project Classification, Project Area, Project Manager, and Unit Work are enterprise ontologies; All 18 parts including Area code, Work code, etc, which compose the hydropower engineering patent code specification, are domain ontologies.

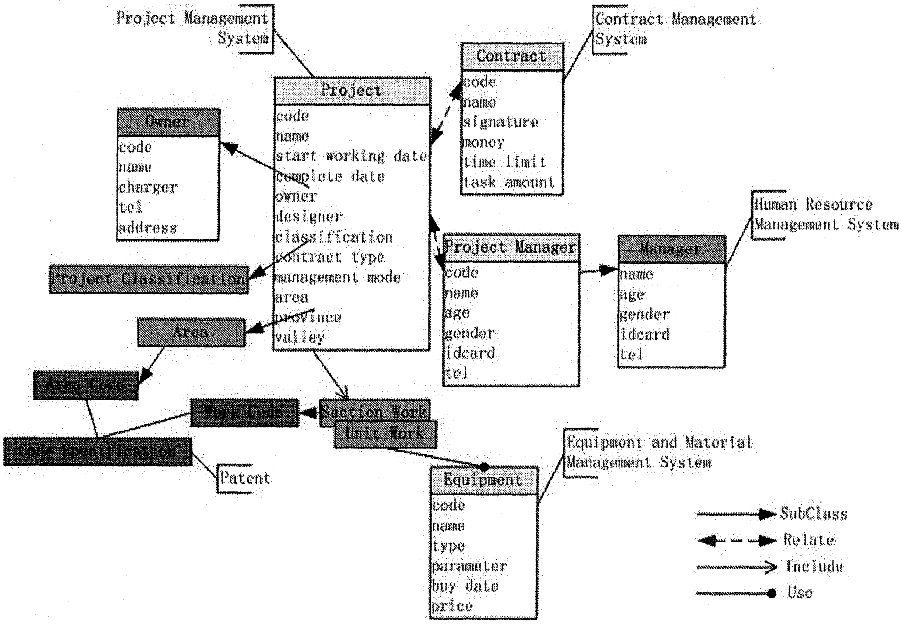

Fig. 5. a simplified case model

5.2 Ontologies Representation by RDF

The ontologies of the model can be represented by RDF. For example, the RDF representation of the Project ontology is shown in Table 1.

Table 1. The Project ontology

Class	Property
<rdfs:Class rdf:about=”GZB:Project”> <rdfs:subClassOf rdf:resource=” rdfs#Resource”/> </rdfs:Class> <rdfs:Class rdf:about=”GZB:Owner”> <rdfs:subClassOf rdf:resource=” rdfs#Resource”/> </rdfs:Class> …	<rdf:Property rdf:about=”GZB:Project Name”> <rdfs:domain rdf:resource=”GZB:Project”/> <rdfs:range rdf:resource=”rdfs#Literral”/> </rdf:Property> <rdf:Property rdf:about=”GZB:Project Code”> <rdfs:domain rdf:resource=”GZB:Porject”/> <rdfs:range rdf:resource=”rdfs#Literral”/> </rdf:Property> <rdf:Property rdf:about=”GZB:Owner”>
Relation	<rdfs:domain

```
<rdf:Description
  rdf:about=”GZB:Project”>
 <GZB:Include
    resource=”GZB:Section
Work”/>
</rdf:Description>
<rdf:Description
  rdf:about=”GZB:Section Work”>
  <GZB:Include
    resource=”GZB:Unit Work”/>
</rdf:Description>
...
```

```
rdf:resource=”GZB:Project”/>
<rdfs:range rdf:resource=”GZB:Owner”/>
</rdf:Property>
<rdf:Property
  rdf:about=”GZB:Project Area”/>
<rdfs:domain
rdf:resource=”GZB:Project”/>
<rdfs:range rdf:resource=”GZB:Area”/>
</rdf:Property>
...
```

5.3 RDQL Query

The query can be made by RDQL. For example, based on the above case model, make the semantic query: “How about the price of the same Unit Work (concrete for foundation) in different projects”? The RDQL query of the question is:

```
SELECT    ?project, ?price, ?item
WHERE     (?project rdf:type GZB:Project)
   AND    (?price rdf:Property ?project)
  AND     (?project GZB:Include ?item)
  AND     (?item rdf:type GZB:Unit Work)
  AND     (?item eq “concrete for foundation”)
USING     GZB FOR http://localhost/gzb-schema#
          rdf FOR http://www.w3.org/1999/02/22-rdf-syntax-ns#
```

6 Conclusions

The knowledge in an enterprise is always discrete, out of order, and allocated in separate application systems. Knowledge integration is for the knowledge users to better the knowledge sharing and reuse. However, EKI is such a complicated systemic engineering that there are still many problems to be further researched, such as how to do knowledge representation and semantic query by a more powerful language of OWL. So are the semantic similarity calculation methods, and the knowledge inference technologies, etc.

References

1. A. Maryam and E.L. Dorothy, Review: Knowledge Management And Knowledge Management Systems: Conceptual Foundations And Research Issues, *MIS Quarterly* **25**(1), 107-136 (2001)

2. C. Wang, T. Chen, Y. Chen and H. Chu, Design Of A Meta Model For Integrating Enterprise Systems, *Computers in Industry* **56**(3), 305-322 (2005)

3. A. Fernandes and A. M. de C. Moura, An Ontology-Based Approach For Organizing, Sharing And Querying Knowledge Objects On The Web, Proceedings of the 14th International Workshop on Database and Expert Systems Applications (2003).
4. S. Zhou and K. Chin, Internet-Based Distributive Knowledge Integrated System for Product Design, *Computers in Industry* **50**, 195-205 (2003).
5. R Grant. Prospering in Dynamically-Competitive Environment: Organizational Capability as Knowledge Integration, *Organization Science* **7**(4), 375–87 (1996).
6. T. Berners-Lee, J. Hendler, and O. Lassila, The Semantic Web, *Scientific American* **284**(5), 34-43(2001).
7. Z. Wu and J. Yuanping, A Study of Approaches To XML Data Semantic Mapping, *Computer Engineering and Application.* **28**, 88-91 (2003).
8. T. Gruber, Towards Principles for the Design of Ontologies Used for Knowledge Sharing, *International Journal of Human-Computer Studies* **43** (5/6), 907-928 (1995).
9. A. Gomez-Perez and A. Benjamins, Overview of Knowledge Sharing and Reuse Components: Ontologies and Problem-Solving Methods. In the Proceedings of the IJCAI-99 Workshop on Ontologies and Problem-solving Methods (KRR5), Stockholm, Sweden (1999).