

An Investigation on Semantic Web–based Business Support: Ontology development and inference framework for the course recommendation

Jin S. Kim

School of Business Administration
Jeonju University, Jeonju 560–759, South Korea
E–mail: kimjs@jj.ac.kr

Abstract

As a fundamental knowledge source in a global learning environment, it is feasible to apply the relational database management systems (RDBMS), object–oriented database management systems (OODBMS), and other traditional DBMS. However, the traditional DBMSs are not feasible in semantic knowledge/ontology representation and inference. One of the reasonable ways to overcome the limitations is the semantic web–based business support framework. Especially, in this study, we focused on the development of semantic web ontology and natural language (NL)–based inference framework. To validate the efficiency of the proposed framework, we considered a reasonable scenario for course recommendation in a university.

Key Words : Inference, Knowledge management, Natural language, Ontology, OODB, RDBMS, Semantic web

1. Introduction

According to the improvement of the technology in knowledge description on the semantic web, a great number of web information resources are also available. In the global learning area, as an aftermath of the development, quite a number of global education courses are available on the Internet [1]. For the development of the courses, the most important work is a description of the information resources [2]. In addition, designing of the knowledge inference processes are required for the systems.

First of all, in this application, the knowledge managers should find a general consensus from all the members before the knowledge description and inference. A structured knowledge representation is then required. After the process, it will be available to design and construct the common knowledge management and inference framework acceptable to the customers.

Recently, semantic–web ontology is considered as one of acceptable foundations to develop the framework. Many researchers developed efficient mechanisms to develop the ontology domain users can share and annotate. There are some of the reasons that we should have focus on the development of the ontology [3].

- It can assist to share common understanding of the structure of information among people or software agents.
- It enables reuse of domain knowledge
- It makes domain assumptions explicit.
- It can separate domain knowledge from the operational knowledge.
- It provides an efficient way to analyze domain knowledge.

With the semantic web, to propose the ontology development and inference framework, we considered four technologies such as Normal Form of RDBMS, OWL, conceptual graph (CG), and NL.

The Normal Form is useful to extract the main ideas/knowledge from the domain users and the OWL is well known as a reasonable way to represent the semantic knowledge. The CG is a knowledge representation method developed by Sowa [4], and it has a direct mapping to and from NL and a graphic notation. Therefore, CG will be use to transform the web ontology into the CG Interchange Format (CGIF). Then the natural inference language PROLOG will be used to inference the knowledge. The combination of CG and PROLOG will lead us into the NL–based intelligent knowledge processing and theorem proving [12].

The remains of the study are as follows. First, the background technologies are summarized in section 2. A relatively complex example and the processes of ontology development are described in section 3.

접수일자 : 2010년 4월 20일

완료일자 : 2010년 7월 30일

The section 4 shows the knowledge extraction from the OWL documents and the NL-based knowledge inferences. Finally, some conclusions and further research topics are put forward in section 5.

2. Background Technologies

2.1. A genetic model for ontology organization in a university

As one of foundational researches on semantic web-based course management, Li and Ling [5] proposed a genetic model for ontology organization. In the research, they translated XML documents into semantic web files. The model has four special operators such as *inheritance*, *block*, *atavism* and *mutation*. With the model, they developed ontologies for course management in a university. Fig. 1 shows the ontology and operators used in the study.

Where, the ontologies reuse the primitives of ontology languages RDF, RDFS, and OWL based on the *inheritance* operator. For that reason, in 'Person' ontology, the names of root nodes ('RDF', 'RDFS', and 'OWL') were came from the names of primitive ontology languages. In addition, the lower level ontologies reuse the concepts of higher level ontologies. For example, the 'per' , and 'emp' are namespaces referring to Person and Employee ontologies.

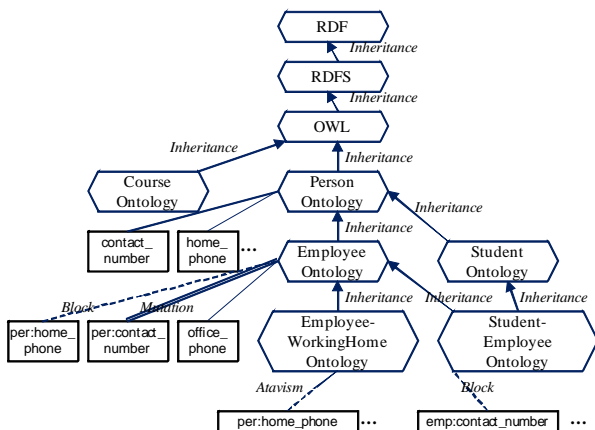


Fig. 1. The ontology hierarchy [5]

After the construction of the hierarchy of ontologies, they proposed the stepwise translations. The translations were composed of three sequential steps such as *Semantic Translation*, *Structural Translation*, and *Schematic Translation*. Using the

translations they translated XML documents/files into the semantic web files.

First, the *Semantic Translation* from an XML file to a semantic web file means that the XML elements, attributes and values were replaced with concepts from ontologies.

Second, the *Structural Translation* refers to the translation of an XML file to a file complying with the Subject-Verb-Object (SVO) format.

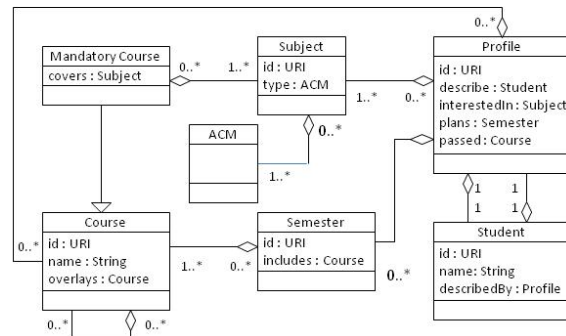
Third, the *Schematic Translation* means that some features of the XML schema are translated to the RDF, RDFS and OWL languages.

However, there are some limitations. Firstly, in the study, they just proposed a logical procedures and rules for the translations. Secondly, they didn't prove the proposed mechanism with an acceptable real-world example. Thirdly, the proposed ontology hierarchy is just represented by using the generic terms not the ontology-specified languages and terms.

2.2. Domain knowledge and metadata development for course recommendation

Hsu [6] proposed a mechanism which can support a novel recommendation for students in a university. The basic concepts were based on dynamic link generation and XML link language (XLink). Using the mechanism, he proposed Semantic XLink Recommendation System (SXRS).

Unfortunately, the pure XLink lacks knowledge and semantic representation to cope with the computer-interpretable operations. To overcome the limitations, he combined semantic web technologies and knowledge manipulation technologies (i.e. XLink base, knowledge base, search engine, and inference engine). Therefore, he implemented three different approaches: *XLink-based metadata*, *ontology-based reasoning*, and



rule-based inference.

Fig. 2. The UML diagram for SelCourse [6]

Using those approaches, he represented how multiple-layered Semantic Web stack, including XLink, OWL, and SWRL, can be mapped into XLink Base, Ontology Base, and Rule Base respectively. Then he applied the Web-based SXRS in assisting students to plan semester courses. In the application, SelCourse ontology was developed to assist the students and designed by using the Unified Modeling Language (UML) diagram. Fig. 2 shows the ontology.

Nevertheless, the UML diagram and XLink couldn't support an intelligent inference at a semantic level. To overcome the limitation, he just proposed pre-defined inference rules arbitrarily in a fixed form. It's just like the traditional *If-Then rule*. Table 1 shows the example of the rules.

Table 1. The example rules for managing ontology

OWL expression	<pre><owl:Class rdf:ID=" MandatoryCourse" > <rdfs:subClassOf rdf:resource=" Course" /> <owl:Class></pre>
Semantic meaning	The MandatoryCourse is a subclass of the Course.
Rule expression	If MandatoryCourse(x) then Course(x)

The rules are not expandable according to the expansion of the domain ontology. In addition, it is strongly depended on the pre-defined inference process. All the pre-defined rules should be changed simultaneously if the process or target domain is revised.

2.3. Conceptual Graph (CG) and ontology

CG, developed by Sowa [4], is one of knowledge representation methods. Its basic concepts were derived from the Charles Peirce's Existential Graphs and Semantic Networks of AI [7]. The essential functions are direct mapping to and from NL and a graphic notation designed for human readability. It could express the meanings in a form that is logically precise, humanly readable, and computationally tractable. Therefore, most of popular graphic notations and structures ranging from type hierarchies to entity-relationship or state transition diagrams can be viewed as special cases of CGs [7].

Normally, the knowledge represented by the CG is divided into two parts such as *Terminological Knowledge* and *Assertional Knowledge*. Where, the *Terminological Knowledge (Support)* contains the 'ground' vocabulary. Secondly, the *Assertional Knowledge* consists of a set of CGs built by means of the *Terminological Knowledge* [8].

The ground vocabularies in *Terminological knowledge* were used to build the domain knowledge base: the type of *Concepts*, the *Instances* of these types, and the types of *Relations (Conceptual Relationships)*. The concepts could be linked by means of relations and the support contains the set of *Conceptual Relation Types*.

Kayed and Colomb [7] suggested that the using of CGs facilitates in two ways:

- By embedding CGs in Web-XML documents to represent knowledge.
- By building a browser-based XML like language that can communicate with a collection of ontological components that use CG Interchange Format (CGIF) as their native language.

In their researches, however, they just focused on the knowledge representations by using the CG and CGIF, not the combination with intelligent inference.

To overcome the limitations founded in previous researches [5, 6], we propose a new framework for the development of semantic web ontology and intelligent inference. The key technologies used in our proposal are Normalization of data, OWL and CG-based ontology constructions, and NL-based knowledge inference.

3. Construction of the ontology

3.1. A relatively complex example

In this section, we have used the Protog 3.1.1 to develop the ontology and it was stored in OWL format. The faculties, staffs, students, other members, and objects were engaged to construct the ontology. They suggested the general view of ontology in educations. Fig. 3 shows some members/objects engaged in the development of the ontology and the ontology hierarchy.

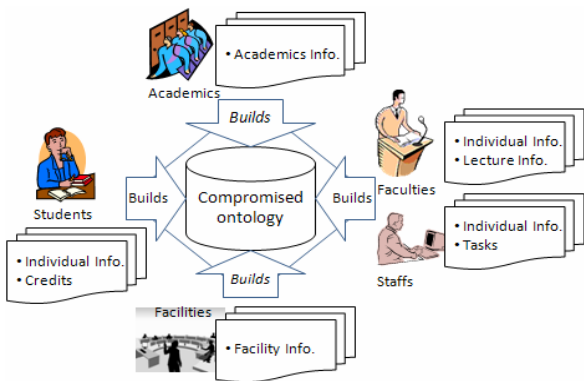


Fig. 3. The members/objects of the ontology

We expect that the ontology could support effective interoperation between heterogeneous knowledge sources and users in a university. For that, in this study, we proposed the hybrid process of normalization and hierarchical division of the ontology. Fundamentally, it is based on the normalization of the traditional RDBMS [9, 10].

3.2. Normalization of the ontology

3.2.1. Conversion to 1st normal form

Step 1: Eliminate the repeating groups

We can start the process by removing the repeated groups and finding the key attributes. In the first step, we presented the data gathered from heterogeneous knowledge resources in a tabular format (Table 2). Each cell in the Table 2 has a single value and there are no repeating groups (1st Normal Form).

Table 2. The 1st Normal Form

Dept	Major	Std_ID	Sex	Blood Type	Std_Name
School of Business	Business	200131367	M	A	YongLee
School of Business	Business	200831008	F	B	JeePark
School of Business	Business	200731068	F	AB	HwanKim
School of Business	Business	200231241	M	A(-)	JunKim
School of Business	Business	200831044	F	O	MiSeo
SSN	Fac_ID	Fac_Name	Major2	Class	
821102-*****	10035	Kim, Sang	Accounting and Tax	MIS	
880719-*****	11319	Jin, Hye	Business Management	Prin. of Accounting	
881217-	10065	Choi,	Business	Acc.	

*****		Dong	Management	Theory
830706-*****	10560	Park, Jong	Logistics and Trade	Marketing Mgmt
890516-*****	10458	Oh, Sang	Chinese Lang. and Culture	OB

Step 2: Identify the primary key (PK)

However, 'Dept (Department)' or 'Major' could not be a PK because it did not uniquely identify all the remaining entity attributes. For example, 'School of Business' identify all of the students. As a proper PK, it has to identify other attributes uniquely. 'Std_ID' or 'SSN' could be the PK for the student. Following dependency shows the example of PK.

- $Std_ID, Fac_ID \rightarrow Std_Name, Dept, Major, Fac_Name, Fac_Dept, ATitle, Class \dots$

Step 3: Identify all the dependencies

Here, all the identified dependencies are as follows.

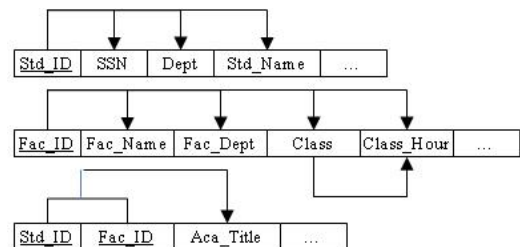
- $Std_ID, SSN \rightarrow Dept, Std_Name, Major, Sex, \dots$
- $Fac_ID \rightarrow Fac_Name, Fac_Dept, ATitle, Class, \dots$

3.2.2. Conversion to 2nd Normal Form

Step 1: Write each key component on a separate line

After the conversion in the 1st normal form, we can write the key components on a separate line and the composite key on the last line just as follows.

- Line 1: Std_ID
- Line 2: Fac_ID
- Line 3: Std_ID, Fac_ID



- Student ($Std_D, SSN, Dept, Std_Name, \dots$)
- Faculty ($Fac_D, Fac_Name, Dept, \dots$)
- Academics ($Std_D, Fac_ID, ATitle, \dots$)

Fig. 4. The 2nd Normal Form

Step 2: Assign corresponding dependent attributes

The PKs are dependent on other attributes. The dependencies for the original key components are found by examining the arrows in the dependency diagram shown in Fig. 4.

3.2.3. Conversion to 3rd Normal Form

Step 1: Identify each new determinant

For every transitive dependencies founded in the 2nd normal form, we could write its determinant as a PK and made a new table. Where, we had three different transitive dependencies/determinants.

Step 2: Identify the dependent attributes

To clarify the dependencies among attributes, we identified all the attributes that are dependent on each determinant in Fig. 4.

Step 3: Remove the dependent attributes from transitive dependencies

Finally, we eliminated all dependent attributes shown in Fig. 4. Therefore, we could get the four different tables and attributes as follows.

- Student (*Std_ID, SSN, Dept, Std_Name, ...*)
- Faculty (*Fac_ID, Fac_Name, Fac_Dept, ...*)
- Class (*Class_ID, Class_Hour, ...*)
- Academics (*Std_ID, Fac_ID, ATitle, Class, ...*)

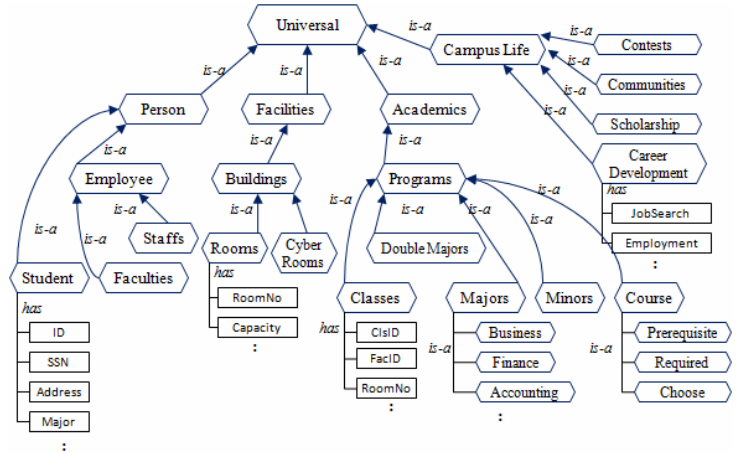
3.3. Hierarchical division of the ontology

After the *Normalization* of the knowledge, we could get some tables (the initial form of ontology). Where, each 'table' matched with the 'class' in ontology. Then the 'column (attribute)' could be transformed into 'property' in the ontology. Finally, the 'tuple (or record)' matched with the 'instance' of the class [9, 10]. In addition, the PK will be used as a 'trigger variable (or key question variable in a query)' introduced in the next section. For example, the Std_ID will be used to inference the knowledge from the student class and other related/referred classes.

Then we found the *Hierarchy* to represent relationships among the *classes*. We call the process as *Hierarchical Division*. It could help us to represent the *Inheritances* among classes. During the division, common attributes could be distributed into the *Parent/Child classes*. Finally, after the

Normalization and *Hierarchical Division*, we could get the hierarchy of the ontology shown in the Fig. 5.

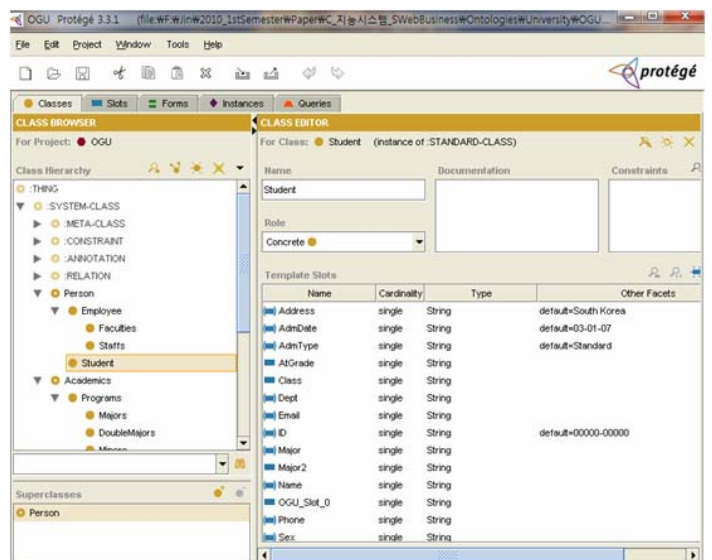
Fig. 5. The hierarchy of the ontology



3.4. Ontology construction by using Protégé

The ontology was represented by using the OWL recommended by the W3C and the development processes were supported by the editing tool Protégé (<http://protege.stanford.edu/>). The editor is an integrated supporting tool widely used in developing ontology or knowledge-based systems [1]. Fig. 6 shows the editing windows of the system.

Fig. 6. The ontology developed by Protégé 3.3.1



4. Knowledge Inference

In this section, we applied the Natural Language-based Query Framework (NLQF) proposed by Kim [11].

4.1. Knowledge Extraction from OWL documents

For inference, according to the framework proposed by Kim [11], the knowledge stored in the OWL web document was translated into the NL and CG form.

In this study, we used the NL-based knowledge based design and inference tool PROLOG+CG [12] to manage the CG and NLQF (Natural Language Query Framework)-based inference. Table 3 shows the simple example of the transformed NLQF knowledge.

Table 3. The NLQF knowledge

<p>Universal > Person, Facilities, Academics, CampusLife, Action. Person > Employee, Student. Employee > Faculties, Staffs. Facilities > Buildings. Buildings > Rooms, CyberRooms. Rooms > Computers. Academics > Programs, Classes. Programs > Majors, DupleMajors, Minors, Course. Majors > Business, Finance, Accounting, Production, HRM, MIS. CampusLife > Contests, Communities, Scholarship, CareerDevelopment. Course > Prerequisite, Required, Choose. Attribute > Phone, Address, Grade, ID, Major. hasPhone > Phone. hasAddress > Address. hasGrade > Grade. hasID > ID. hasMajor > Major. Action > Lecture, Credit, Graduate, Manage. . . .</p> <p>StdAddress = Jeonju, Kunsan, Iksan. Student = MiSeo, JunKim, HwanKim, JeePark, YongLee. StdGrade = Freshman, Sophomere, Junior, Senior.</p> <p>Prerequisite(Freshman, Business, (MA134, ST214, BA100)). Prerequisite(Sophomere, Business, (MA135, EC215)). Prerequisite(Junior, Business, (AC210, FI310)).</p>	<p>Prerequisite(Senior, Business, (MS110, IB120)). . . .</p> <p>Choose(Freshman, Business, (AD101, BA100)). Choose(Sophomere, Business, (AD102, BA101)). Choose(Junior, Business, (AD103, BA102)). Choose(Senior, Business, (AD104, BA103)). . . .</p> <p>Required(Freshman, Business, (AC221, AC222, FI361)). Required(Sophomere, Business, (MG257, MG301, MK301)). Required(Junior, Business, (MG375, EC225, MA139)). Required(Senior, Business, (MG376, BI230, MA140)). . . .</p> <p>Credit(S, M, G, P, R, C) :- // Graduation(ID, Major, Grade, Prerequisite, Required, Choose). [Student: x]<-MEMB-[hasID: S], [Student: x]<-MEMB-[hasGrade: G], Prerequisite(G, M, P), Required(G, M, R), Choose(G, M, C). . . .</p> <p>[Student: x]<-MEMB-[hasAddress: y]:- [Person:x]<-AGNT-[Property]-hasAddress->[Address: y].</p> <p>[Student: x]<-MEMB-[hasGrade: y]:- [Person:x]<-AGNT-[Property]-hasGrade->[Grade: y]. . . .</p> <p>[Person:MiSeo]<-AGNT-[Property] hasAddress->[Address: Jeonju]. [Person:JunKim]<-AGNT-[Property]-asAddress->[Address: Kunsan]. [Person:HwanKim]<-AGNT-[Property] hasAddress->[Address: Iksan]. [Person:JeePark]<-AGNT-[Property]-hasAddress->[Address: Jeonju]. [Person:YongLee]<-AGNT-[Property]-hasAddress->[Address: Kunsan]. . . .</p> <p>[Person:MiSeo]<-AGNT-[Property]-hasGrade->[Grade: Freshman].</p>
-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

```
[Person:JunKim]<-AGNT-[Property]-hasGrade-
>[Grade: Sophomere].
[Person:HwanKim]<-AGNT-[Property]-
hasGrade->[Grade: Junior].
[Person:JeePark]<-AGNT-[Property]-hasGrade-
>[Grade: Senior].
[Person:YongLee]<-AGNT-[Property]-hasGrade-
>[Grade: Senior].
...

[Student : x] -
  -hasAddress->[Address],
  -hasPhone->[Phone],
  -hasPicture->[Picture],
  -hasID->[ID],
  -hasMajor->[Major].
-----
```

4.2. Knowledge Inference

In this study, the Normal Form, ontology hierarchy, OWL, and CG were used to represent the semantic knowledge and their relationships logically. Then the PROLOG+CG was used as an efficient tool to manage the semantic knowledge and inference. Table 4 shows the NLQF queries and the result of inference executed by the system.

Table 4. The NL queries and their inference result

```
-----
Query-1:
?- Credit(BA201031127, x, M, G, P, R, C).

Descriptions of Query-1:
"Show me the course recommendation
(P: Prerequisite), (R: Required), (C: Choose)
for the man his/her student ID is 'BA201031127'
(Student: x)
in Grade (G), and Major in (M)"

Answer-1 for Query-1:
{x=MiSeo, M=Business, G=Freshman,
P=(MA134, ST214, BA100), R=(AC221, AC222,
FI361), C=(AD101, BA100)}

Descriptions of Answer-1:
His/Her name is 'MiSeo'
He/She is a 'Frsehman' and Major in 'Business.'
The course recommendation for 'MiSeo' is
Prerequisite: MA134, ST214, BA100,
Required: AC221, AC222, FI361,
Choose: AD101, BA100
```

```
Query-2:
?- Classes(MA134, C, F, R).
```

```
Descriptions of Query-2:
What is the name (C) of the class 'MA134' ,
Who is the faculty (F) for the class, and
Where is the classroom (R).
```

```
Answer-2 for Query-2:
{C=PrinciplesOfBusiness,F=Kim_Sang, R=JHall101}
```

```
Descriptions of Answer-2:
The name of the class is 'Principles Of
Business' ,
The name of the faculty is 'Kim_Sang' , and
The classroom is 'JHall 101.'
```

In Table 4, through the description, human users can understand the results of inferences.

Where, we could find that the trigger variables and their values Std_ID = BA20103127 and ClassID = MA134 were used in each query. As result of the inference with the values, we could get the list of recommended classes and the detailed information for the classes.

5. Conclusions

The main purpose of this study is the development of the semantic web ontology and NL-based inference framework for college student. Then the prototype ontology and inference system is focused on the course recommendation. To validate the efficiency of the proposed framework, we examined a reasonable example for course recommendation and showed the ontology and results of the NL-based inference.

In the background researches, we have inspected ontology development for course management/recommendation and CG-based knowledge representation/inference. Then the Faculties, Staffs, and Students in a university were engaged into the study to share their semantic knowledge for course management.

During the construction of the ontology, we have proposed the hybrid process of normalization and hierarchical division of the ontology. It is based on the Normal Form of the traditional RDBMS.

Through the study, therefore, we could propose the integrated view for the development of ontology used in course management. Then the effectiveness of the proposed framework was validated through the experiment. However, the study has some limitations and further research topics.

First, in the study, we could not suggest a flexible transformation mechanism for the OWL and CG, and vice versa. For the flexible knowledge management, we have to revise the transformation mechanism.

Second, in real-world example, intelligent agent executes the knowledge management and inferences. Therefore, it is required to support the agent.

Third, there are some OWL-based inference mechanisms. However, in this study, we did not consider that researches. In the further research, we should compare the effectiveness of the mechanisms and technologies.

Acknowledgements

This work was financially supported by LG Yonam Foundation, South Korea.

References

- [1] Gladun, J. Rogushina, F. Garca-Snchez, R. Martnez-Bjar, and J.T. Fernndez-Breis, "An application of intelligent techniques and semantic web technologies in e-learning environments," *Expert Systems with Applications*, vol.36, no.2, pp. 1922-1931, 2009.
- [2] H. Yao and L. Etzkorn, L., "Automated conversion between different knowledge representation formats," *Knowledge-Based Systems*, vol. 19, pp. 404-412, 2006.
- [3] N.F. Noy and D.L. McGuinness, *Ontology development 101: A guide to creating your first ontology*, Stanford University, <http://ksl.stanford.edu>, 2007.
- [4] J.F. Sowa, *Conceptual structures: Information processing in minds and machines*, Addison-Wesley, Reading, MA, 1984.
- [5] Li and T.W. Ling, "From XML to semantic web," *Proceedings of the Database Systems for Advanced Applications (DASFAA) 2005*, 10th International Conference, Beijing, China, pp. 582-587, April 17-20, 2005.
- [6] I-Ching Hsu, "SXRS: An XLink-based recommender system using semantic web technologies," *Expert Systems with Applications*, vol. 36, no. 2, pp. 3795-3804, 2009.
- [7] A. Kayed and R.M. Colomb, "Using BWW model to evaluate building ontologies in CGs formalism," *Information Systems*, vol. 30, pp. 379-398, 2005.
- [8] J. Dibia-Barthlemy, O. Haemmerl, and E. Salvat, "A semantic validation of conceptual graphs," *Knowledge-Based Systems*, vol. 19, pp. 489-510, 2006.
- [9] P. Rob and C. Coronel, *Database Systems: Design, Implementation, and Management* (7th Edition), Course Technology, Thomson, 2007.
- [10] K. Lee and T. Whang, "A novel method for matching between RDBMS and domain ontology," *Journal of Korea Multimedia Society*, vol. 9, no.12, pp. 1552-1559, Dec. 2006.
- [11] J.S. Kim, "A natural language query framework for the semantic web," *Journal of Korean Institute of Intelligent Systems*, vol. 18, no. 1, pp. 127-132, 2008.
- [12] A. Kabbaj and U. Peterson, PROLOG+CG version 2.0, <http://www.hum.aau.dk/~ulrikp> or <http://prologpluscg.sourceforge.net>, 2007.

저 자 소개



김진성 (Kim, Jin Sung)

김진성 (金珍成)은 현재, 전주대학교 경영학부 부교수로 재직 중이다. 주요 관심분야는 퍼지이론과 인공지능 기법을 이용한 지능형의사결정지원과 시맨틱 웹 서비스 등이다.

TEL : 063-220-2932
FAX : 063-220-2787
E-mail : kimjs@jj.ac.kr