On Transforming the ER Model to Ontology Using Protégé OWL Tool

Pasapitch Chujai, Nittaya Kerdprasop, and Kittisak Kerdprasop

***Abstract*—Ontology is a concept to organize domains that can be widely used in many fields. For building the OWL ontology, several existing data sources such as XML, relational databases were used. Most researchers try to map data in a format of relational database into the OWL ontology using OWL syntax, which sometimes is difficult, especially for a person who does not know this syntax, or uses mistaken work to create the OWL ontology. So, in this research, we propose a mechanism to construct OWL ontology in order to reduce the problem of lack of understanding about the OWL ontology syntax based on Entity Relationship Model (ER), which is a model for describing data in a conceptual level of database design. We demonstrate a step-by-step transformation of ER model into OWL ontology using a tool editor called Protégé. The evaluations of the building ontology will use FaCT++ and HermiT 1.3.8. The results have shown that the ability to convert each part of ER model is very accurate, fast and easy to use.**

***Index Terms*—ER model, OWL ontology, Protégé editor.**

I. Introduction

Ontologies [1] play an important role in semantic description for common understanding and classification of the documents in the knowledge domain [2]. They use a single concept for reducing ambiguous concepts or terminology and support the exchange of information retrieval; they are also critical to the development of the knowledge based systems. Therefore, the knowledge base of ontologies can be exchanged or reused and published for others to be used widely in various fields. These Ontologies are being applied in many applications such as Artificial Intelligence, E-Commerce, Knowledge Management, Information Retrieval, Semantic Web and Recommendation System.

The language used to describe ontologies will be depending on the format of this language. The popular pattern is a form that is used for describing resources on the web and in the human form easily understood such as RDFS and OWL. At present, most of the ontologies are constructed manually and use several existing data sources [3] such as XML, relational databases and web pages etc.

A majority of this research has been done on building ontologies using relational database. The uses of this OWL syntax are commonly found in many researches. They have tried to construct OWL ontologies by mapping several data source into OWL syntax. The source of these data most often comes from relational database, which can be found in the following research:

Upadhyaya and Kumar [4] proposed an approach of automatic mapping Extended E/R diagram into OWL by developing the prototype of ERONTO using Java and Jena 2.1. Igor and Marguerite [5] proposed an approach using automatic mapping X-Ray diffraction ER schemas into OWL Lite ontology. Ashok and colleagues [6] proposed an approach to use automatic mapping ER Model into OWL-S. In [7] Saeed and Reza proposed an approach to use automatic mapping relational database into OWL, considering all types of relationships between tables using Jena and MySQL and made evaluation of the ontologies by using FaCT++ and Pellet. Jiuyun and Weichong [8] proposed an approach constructing the OWL ontologies from XML document, in two steps: the first step is transforming an XML document into entity-relation model (XTR) then transforming entity-relation model into OWL ontologies (RTO) as second step. In [9] Nora and colleagues proposed an approach to generate automatically OWL ontologies from multiple XML data sources based on different XML schema design styles. This approach can automatically generate ontologies although the XML schema does not exist, using the source XML document. Irina and Ahto [10] proposed an approach to map the SQL to OWL ontologies. This approach is based on an analysis of tables, columns, relationships, and constraints.

Most of these researches try to map data in a format of relational database into the OWL ontologies using OWL syntax, which sometimes is difficult to achieve, especially for a person who does not know this syntax, or uses mistaken work to create the OWL ontologies. So in this research, we propose to construct ontologies by reducing the problem of lack of understanding related to the OWL ontologies syntax with a tool editor called Protégé based on ER model. This tool is popular for constructing ontologies in OWL as can be seen from some research such as in [11] used for developing University Ontology. They use utility plug-in for Protégé and Pellet reasoned for checking consistency.

The rest of this research is organized as follows: Section II gives details of the techniques that used. In Section III gives details of our proposed method. The experimental analysis will be presented in Section IV. Finally, the research is concluded and provides a future work in Section V.

II. Background

A. Ontology Knowledge Based

Ontologies [12] are a concept that organizes the knowledge of a domain in a general way and provides a

Manuscript received December 13, 2014; revised March 20, 2014. This work was supported in part by grant from Suranaree University of Technology through the funding of Data Engineering Research Unit.

The authors are with the School of Computer Engineering, Suranaree University of Technology, Nakhon Ratchasima 30000, Thailand (e-mail: pasapitchchujai@gmail.com).

DOI: 10.7763/IJCTE.2014.V6.914

shared understanding of this domain, sharing and representing its conceptualization [13]. The characteristics of ontologies are a hierarchical data structure such as tree, graph etc., their concepts are grouped with Taxonomy in the manner of Parent-Child.

The components of ontology knowledge base consists of five basic elements, which are described in detail below:

1) The concepts are a domain of knowledge that can be clearly described.
2) Properties are the properties that apply to explain in details the concepts.
3) The relationship is a pattern of expressing relationships between concepts.
4) The axiom is a condition or a logic that is used to transform the relation between concepts and concepts or concepts and properties for conversion into the correct meaning.
5) The instance is a vocabulary with definitions in ontologies.

Ontologies can be applied in several fields such as semantic web, information retrieval, knowledge management and e-commerce. These ontologies provide many benefits in particular for the solution of a relational database that cannot find the information that have been related.

B. *Web Ontology Language (OWL)*

OWL [14] is the language used to describe ontologies that determines the relation between information of an interesting domain development into standard ontology language by the World Wide Web Consortium (W3C) [15]. To describe ontologies, OWL can describe them in form of classes, properties of classes and relation between classes and their instance. OWL syntax consists of different elements such as

owl: class for representing a group of individuals or a concept.

owl: Object Property for representing the link from individual or class to individual or class.

owl: Datatype Propertype represents the links from an individual or class to an XML Schema datatype value.

rdf: property is used for describing the relationships between instances or classes which are linked to a domain by *rdfs: domain* and a range by *rdfs: range*.

owl: individual for representing domain of objects and so on.

C. *Entity Relationship Model (ER Model)*

ER model is a model used for describing data in the conceptual level of database design. The well-formed ER model is based on Chen's original [16] and consists of three basic elements, which are described in detail below:

1) An entity is a thing that can be clearly identified.
2) A relationship is a relation between two or more entities.
3) An attribute is a characteristic of each entity or relationship.

D. *Protégé OWL Editor*

To transform ER model into ontology, this research used Protégé OWL editor tools (version 4.3) [17]. Protégé is a freeware developed by Stanford Center for Biomedical Informatics research at the Stanford University School of Medicine. This tool is used for developing ontologies and knowledge base, easy to use and support for building ontologies by RDF and OWL.

III. Methodology

In this paper we construct the ontology from ER model using by Protégé Editor- A framework is showed in Fig. 1.

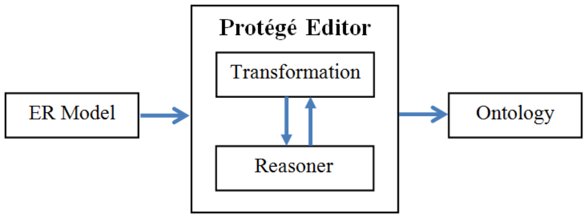


Fig. 1. The concept used for transforming ER model into Ontology using by Protégé Editor.

From Fig. 1, we used each component of the ER model transformed into each part of OWL ontologies by mapping rules and test consistency by reasoner which is a plug-in for Protégé OWL editor.

In addition to the research mentioned above, we used the work of [18], [19] aggregation to the manual of the OWL Protégé editor [20] to be applied for the transformation rules of our work. Thus, three types of transforming are distinguished:

A. *Transforming the Entities*

Each entity can be transformed into a class in the OWL ontology using the Classes Tab and detail shown in Table I.

TABLE I: The Relation between Entity and Classes

ER Component	**Protégé Component**
Strong Entity	Classes Tab
Weak Entity	Classes Tab Subclass of strong entity
Specifies each entity is not overlapping	Disjoint

B. *Transforming the Relationship*

Each relationship can be transformed into an object property in the OWL ontology using the Object Properties Tab and detail shown in Table II.

TABLE II: The Relation between Relationships and Object Properties

ER Component	**Protégé Component**
Relationships	Object Properties Tab
Binary Relationship without Attributes	The way of transformation should be dividing into two object properties, • the first should be the same of the relationships which show in ER model, and • the second should be an inverse property of the former
Binary Relationship with Attributes	The steps are as below: 1. Take the relationship's name in ER model to create the third class. 2. Take attributes of relationship to create datatype which set domains to the third class. 3. Breaks relationship into

ER Component	Prot ég é Component
	• 1:1 and 1:1 (From 1:1) • 1:N and M:N (From 1:N) • N:1 and 1:N (From N:1) • (N:1 and 1:N) or (1:N and N:1) or (M:N and M:N) (From M:N) 4. Do everything using the same Binary Relationship without Attributes.
Relationship from Entity A to Entity B	Domains : set to entity A Ranges : set to entity B
1:1 relationship	Set Function property and Set max cardinality to one
1:N relationship	Set Function property and Set max cardinality to one of inverse property
N:1 relationship	Set Function property and Set max cardinality to one with the same name or relationship in ER model
Specifies each relationship is not overlapping	Disjoint

C. Transforming the Attributes

Each attribute can be transformed into a datatype property in the OWL ontology using the Data Properties Tab. Datatype properties describe relationships between individual (or class) and type of data, details shown in Table III.

TABLE III: THE RELATION BETWEEN ATTRIBUTES AND DATA PROPERTIES

ER Component	Prot ég é Component
Attributes	Data Properties Tab
Domain of Attributes	Domains
Type of Data (real, integer etc.)	Ranges
Single valued Attribute (Null)	Functional datatype property
Single valued Attribute (Not Null)	Functional datatype property and Set min cardinality to one.
Multi valued Attribute (Null)	Data Properties Tab
Multi valued Attribute (Not Null)	Set min cardinality to one.
Key Attribute	Functional datatype property, Set max cardinality to one and Set the uniqueness by inverse functional property
Composite Attribute	Sub property of data property
Specifies that each attribute is not overlapping	Disjoint

IV. EXPERIMENTAL ANALYST

For describing the method that is mentioned above, we have demonstrated by a case study that is a simple ER model, described as follows - A relationship between student and advisor is showed in Fig. 2 and each step can be addressed in details as the following:

A. Transforming Entities

This step is using Classes Tab to map each entity in ER model into a class in OWL ontology, creating the name of class using the name of an entity directly.

Example 1 Strong and weak entities are explained as details below:

1) The strong entity '*Student*' is mapped into a '*Student*' class.
2) The weak entity *'GradStudent'* is dependent on student entity, then uses subclass for creating '*GradStudent*' subclass.
3) Specifies that each entity is not overlapping by making them disjoint from one another.

The results are demonstrated in Fig. 3, respectively.

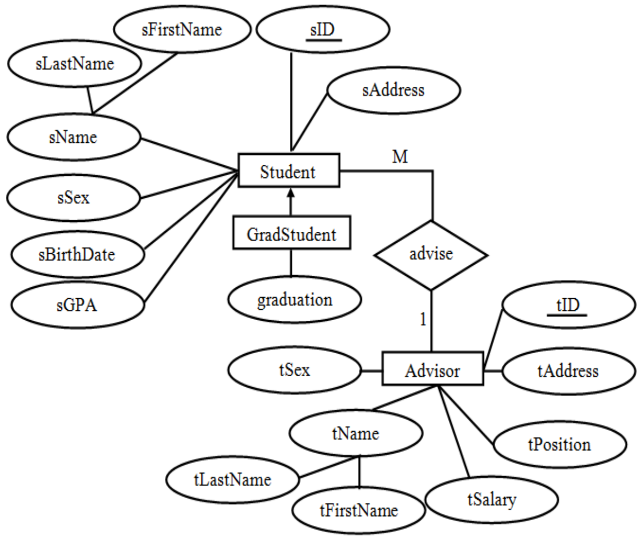
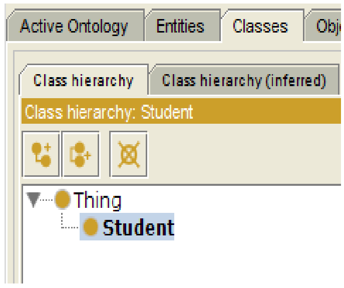
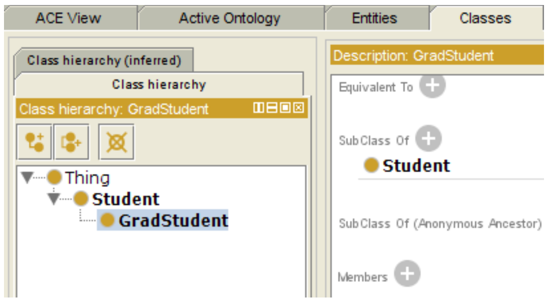


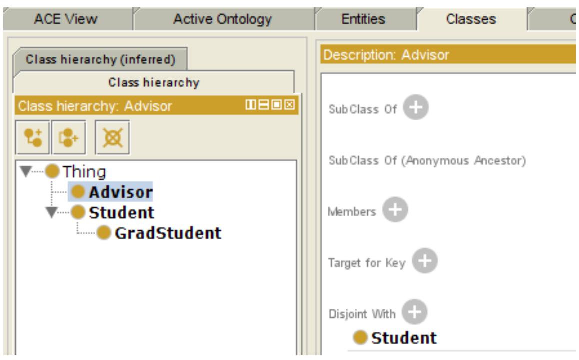
Fig. 2. ER Model of relationship between student and advisor.



(a) Stong entity Student.



(b) Weak entity GradStudent that is a subclass of Entity Student.



(c) Disjoint between classes advisor and student.

Fig. 3. Transforming entities into class using class tab.

B. Transforming Relationship

This step uses Object Properties Tab to map relationship in ER model into two Object properties in OWL ontology.

The way of transformation should be dividing into two object properties, the first should be same of the relationships which show in ER model, and the second should be an inverse property of the former, then specified domains and ranges.

Object properties will link between individual (or class) and individual (or class), from the domain (entity) to the range (entity).

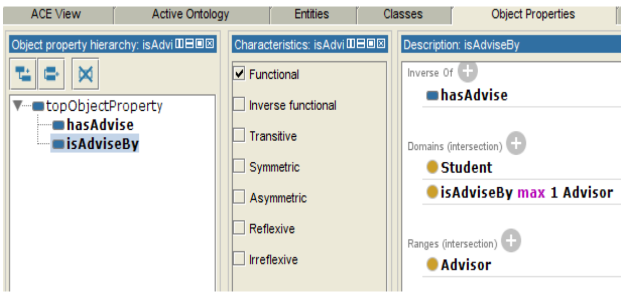
Relationship in OWL will be described in binary form dividing into the following groups: One to One Relationship (1:1), One to Many or Many to One relationship (1:N or N:1), and Many to Many relationship (M:N); this can be specified by using cardinality restrictions.

Since n-ary relationship cannot be translated in OWL ontology, so we have to create a third entity, then break relationship and set restrictions.

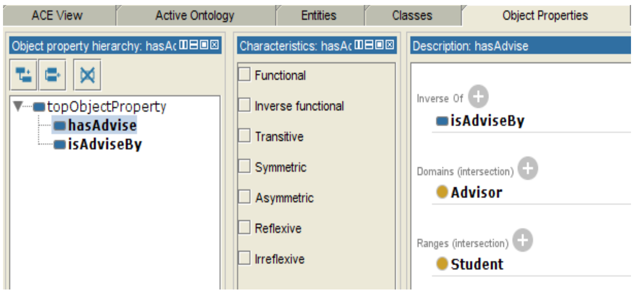
Example 2 Relationship *'advise'* is explained as details below:

1) The relationship *advise'* is mapped to an object property with prefix 'has' to *'hasAdvise'*, and inverse object property is *'isAdvideBy'*,
2) Specifies a domain and a range such as relationship from student to advisor, mapping into *'isAdviseBy'* object properties with *'student'* as domains and *'advisor'* is ranges.
3) Set cardinality restrictions.
4) Set invert property.

The results are demonstrated in Fig. 4.



(a) Relationship from student to advisor as 1:1.



(b) Relationship from advisor to student as 1:N.

Fig. 4. Transforming relationship between entity student and advisor into object property using object properties tab.

C. Transforming Attributes

This step uses Data Properties Tab to map attributes in ER model into Data properties in OWL ontology.

Example 3 for primary key is explained as details below:

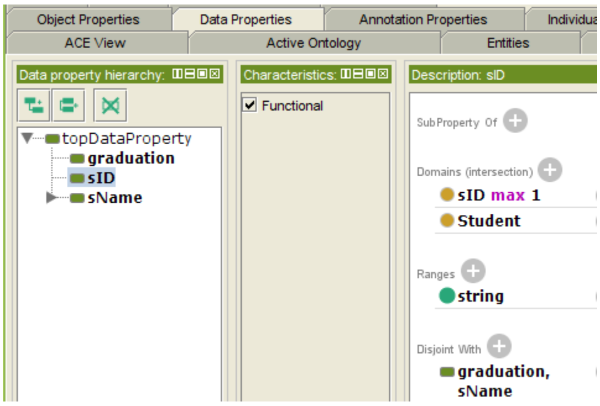
1) The primary key of student entity, '*sID*', is mapped to a datatype property '*sID*' with *'Student'* as domains and *'string'* as ranges.
2) Specified functional, cardinality to one.
3) Set disjoint.

The results are demonstrated in Fig. 5(a).

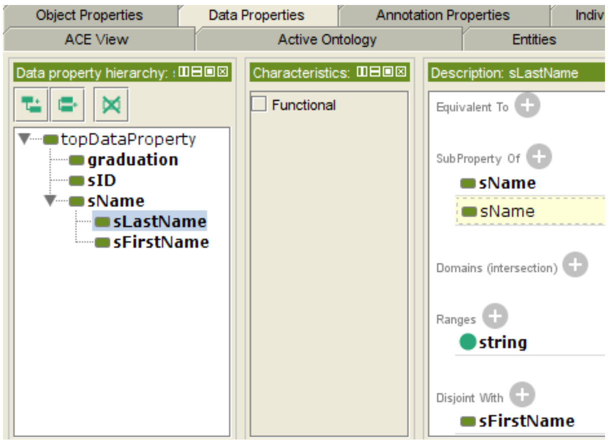
Example 4 for composite attribute as explained in details below:

1) The composite attribute key of student entity, '*sName*', is mapped to a datatype property '*sName*' with *'Student'* as domains and *'string'* as ranges.
2) The attribute *'sFirstName'* and *'sLastName'* are mapped to a sub property of *'sName'* property, set *'string'* as range.
3) Set disjoint.

The results are demonstrated in Fig. 5(b).



(a) The primary key of student entity.



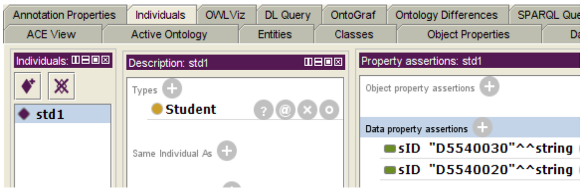
(b) The composite attribute of student entity

Fig. 5. Transforming attribute into data property using data properties tab.

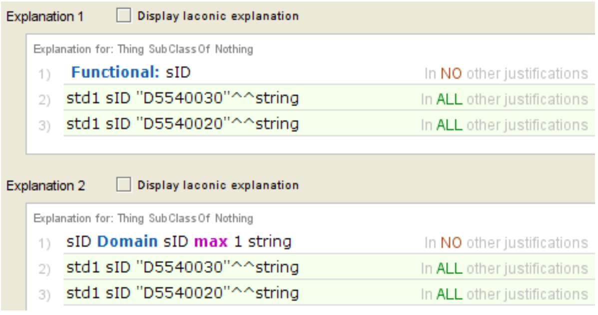
D. Test the Consistency by FaCT++ and HermiT 1.3.8 Reasoner

After constructing the ontology by mapping each element of ER model into each element of OWL ontology using by Protégé Editor, we test the consistency of data by inserting some instance at Individuals Tab. The evaluations of the building OWL ontology will use FaCT++ and HermiT 1.3.8 reasoner.

Example 5 for consistency of primary key, we test by inserting data at *'sID'* more than one row. The results show that OWL ontology is inconsistent in Fig. 6.



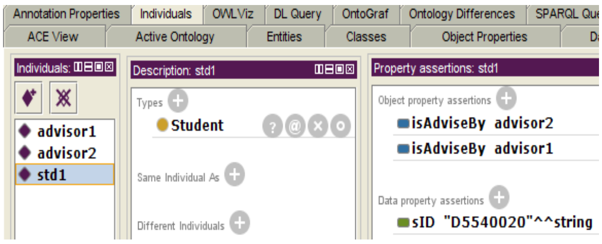
(a) Insert data in more than one row in PK.



(b) The results from reasoner.

Fig. 6. The evaluations of PK.

Example 6 for consistency of 1 to 1 relationship – one student has only one person as advisor. We test by inserting the list of advisors more than one row for student *'std1'*. The results show that OWL ontology is inconsistent in Fig. 7.



(a) Insert data in more than one row in relationship.

(b) The results from reasoner.

Fig. 7. The evaluations of relationship.

V. Conclusion and Future Work

In this research, we proposed an approach to building OWL ontology from relational database based on ER model using an ontology editor called Protégé. We can transform each part of an ER model: entities, relationship and attributes into each part of OWL ontology. We demonstrated a step-by-step transformation of ER model into OWL ontology. The evaluations for building ontology will use FaCT++ and HermiT 1.3.8 reasoner. The results have shown that the ability to convert each part of ER model is very accurate, fast and easy to use without understanding OWL ontology syntax. For future work, we are interested by the approach of constructing OWL ontology from requirements specification which are written in Natural Language into OWL ontology using by Controlled Natural Language.

References

[1] G. K. Saha, "Web Ontology Language (OWL) and semantic web," *ACM Ubiquity*, vol. 8, pp. 1-24, September 2007.

[2] V. C. Storey, D. Dey, H. Ullrich, and S. Sundaresan, "An ontology-based expert system for database design," *Data and Knowledge Engineering*, vol. 28, no. 1, pp. 31-46, 1998.

[3] N. Cullot, R. Ghawi, and K. Yétongnon, "DB2OWL: A tool for automatic database-to-ontology mapping," in *Proc. 15th Italian Symp. Adv .Database*, 2007, pp. 491-494.

[4] S. R. Upadhyaya and P. S. Kumar, "ERONTO: a tool for extracting ontologies from extended E/R diagrams," in *Proc. the 2005 ACM Symposium on Applied Computing*, ACM, 2005.

[5] I. Myroshnichenko and M. C. Murphy, "Mapping ER schemas to OWL ontologies," in *Proc. IEEE International Conference on Semantic Computing, ICSC'09*, 2009, pp. 324-329.

[6] A. Kumar, G. Mahadevan, S. K. Guptha, and G. Krishna, "Composition of semantic web services based on ER-model," in *Proc. International Conference on Computational Techniques and Artificial Intelligence (ICCTAI'2011)*, 2011.

[7] S. M. Sedighi and R. Javidan, "A novel method for improving the efficiency of automatic construction of ontology from a relational database," *International Journal of Physical Sciences*, vol. 7, no. 13, pp. 2085-2092, 2012.

[8] J. Xu and W. Li, "Using relational database to build OWL ontology from XML data sources," in *Proc. CISW 2007, International Conference on Computational Intelligence and Security Workshops*, December 2007, pp. 124-127.

[9] N. Yahia, S. A. Mokhtar, and A. Ahmed, "Automatic generation of OWL ontology from XML data source," *arXiv preprint arXiv: 1206.0570*, 2012.

[10] I. Astrova and A. Kalja, "Mapping of SQL relational schemata to OWL ontologies," in *Proc. the 6th WSEAS International Conference on Applied Informatics and Communications,* Elounda, Greece, 2006.

[11] N. Malviya, N. Mishra, and S. Sahu, "Developing university ontology using protégé OWL Tool: Process and reasoning," *International Journal of Scientific & Engineering Research*, vol. 2, issue 9, September 2011.

[12] G. Yang, S. Mukherjee, and I. V. Ramakrishnan, "On precision and recall of multi-attribute data extraction from semistructured sources," in *Proc. Third IEEE International Conference on Data Mining*, ICDM, IEEE, 2003, pp. 395-402.

[13] M. Uschold and M. Gruninger, "Ontologies: Principles, methods and applications," *Knowledge Engineering Review 11.2*, 1996, pp. 93-136.

[14] G. Antoniou and F. V. Harmelen, "Web ontology language: Owl," *Handbook on Ontologies*, Springer Berlin Heidelberg, pp. 67-92, 2004.

[15] P. Chen, "The entity-relationship model — toward a unified view of data," *ACM Transactions on Database Systems (TODS)*, vol. 1, issue 1, March 1976.

[16] W3C. Web Ontology Language (OWL). [Online]. Available: http://www.w3.org/

[17] Protégé [Online]. Available: http://protege.stanford.edu/

[18] O. Vasilecas, D. Bugaite, and J. Trinkunas, "On approach for enterprise ontology transformation into conceptual model," in *Proc. International Conference on Computer Systems and Technologies, CompSysTech*, vol. 6, 2006.

[19] T. Halpin, "Ontological modeling: part 6," *Business Rules Journal*, vol. 12, no. 2, February 2011.

[20] M. Horridge. A practical guide to building OWL ontologies using Protégé 4 and CO-ODE tools edition 1.3. [Online]. Available: http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/

Pasapitch Chujai is currently a doctoral student at the School of Computer Engineering, Suranaree University of Technology, Thailand. She received her bachelor degree in computer science from Ramkhamhaeng University (RU), Thailand in 2000, and master degree in computer and information technology from King Mongkut's University of Technology Thonburi (KMUTT), Thailand in 2004. Her current research includes ontology,

recommendation system.

Nittaya Kerdprasop is an associate professor at the School of Computer Engineering, Suranaree University of Technology, Thailand. She received her bachelor degree in radiation techniques from Mahidol University, Thailand in 1985, master degree in computer science from the Prince of Songkla University, Thailand in 1991 and doctoral degree in computer science from Nova Southeastern University, U.S.A in 1999. She is a member of ACM and IEEE Computer Society. Her research of interest includes knowledge discovery in databases, artificial intelligence, logic programming, and intelligent databases.

Kittisak Kerdprasop is an associate professor and chair of the School of Computer Engineering, Suranaree University of Technology, Thailand. He received his bachelor degree in mathematics from Srinakarinwirot University, Thailand in 1986, master degree in computer science from the Prince of Songkla University, Thailand in 1991 and doctoral degree in computer science from Nova Southeastern University, U.S.A. in 1999. His current research includes data mining, artificial intelligence, functional and logic programming languages, computational statistics.