On Transforming the ER Model to Ontology Using Proté gé 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 Proté gé. 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, Proté gé 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:


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 Proté gé 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 Proté gé 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
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.

$owl: Datatype Property$ for representing a group of individuals or a concept.

$rdf: property$ 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.](image)

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>
<thead>
<tr>
<th>ER Component</th>
<th>Protégé Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>Strong Entity</td>
<td>Classes Tab</td>
</tr>
<tr>
<td>Weak Entity</td>
<td>Classes Tab</td>
</tr>
<tr>
<td>Specifies each entity is not overlapping</td>
<td>Disjoint</td>
</tr>
</tbody>
</table>

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>
<thead>
<tr>
<th>ER Component</th>
<th>Protégé Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>Relationships</td>
<td>Object Properties Tab</td>
</tr>
<tr>
<td>Binary Relationship without Attributes</td>
<td>The way of transformation should be dividing into two object properties, (\text{the first should be the same of the relationship which show in ER model, and}) the second should be an inverse property of the former</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Binary Relationship with Attributes</th>
<th>The steps are as below:</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Take the relationship’s name in ER model to create the third class.</td>
<td></td>
</tr>
<tr>
<td>2. Take attributes of relationship to create datatype which set domains to the third class.</td>
<td></td>
</tr>
<tr>
<td>3. Breaks relationship into</td>
<td></td>
</tr>
</tbody>
</table>
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>
<thead>
<tr>
<th>ER Component</th>
<th>Protégé Component</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attributes</td>
<td>Data Properties Tab</td>
</tr>
<tr>
<td>Domain of Attributes</td>
<td>Domains</td>
</tr>
<tr>
<td>Type of Data (real, integer etc.)</td>
<td>Ranges</td>
</tr>
<tr>
<td>Single valued Attribute (Null)</td>
<td>Functional datatype property</td>
</tr>
<tr>
<td>Single valued Attribute (Not Null)</td>
<td>Functional datatype property and Set min cardinality to one.</td>
</tr>
<tr>
<td>Multi valued Attribute (Null)</td>
<td>Data Properties Tab</td>
</tr>
<tr>
<td>Multi valued Attribute (Not Null)</td>
<td>Set min cardinality to one.</td>
</tr>
<tr>
<td>Key Attribute</td>
<td>Functional datatype property. Set max cardinality to one and Set the uniqueness by inverse functional property</td>
</tr>
<tr>
<td>Composite Attribute</td>
<td>Sub property of data property</td>
</tr>
<tr>
<td>Specifies that each attribute is not overlapping</td>
<td>Disjoint</td>
</tr>
</tbody>
</table>

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.

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 ‘isAdviseBy’;
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.

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).

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.
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


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 (KMU/T), Thailand in 2004. Her current research includes ontology.

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
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.