A methodology for constructing of philosophy ontology based on philosophical texts

Jung-Min Kim *, Byoung-Il Choi, Hyo-Phil Shin, Hyoung-Joo Kim

OOPSLA Lab. School of Computer Engineering, Seoul National University, San 56-1, Shillim-dong, Kwanak-gu, Seoul, 151-742, South Korea

Abstract

An ontology is a crucial factor for the success of the Semantic Web and other knowledge-based systems in terms of share and reuse of domain knowledge. However, there are a few concrete ontologies within actual knowledge domains including learning domains. In this paper, we develop an ontology which is an explicit formal specification of concepts and semantic relations among them in philosophy. We call it a philosophy ontology. Our philosophy is a formal specification of philosophical knowledge including knowledge of contents of classical texts of philosophy. We propose a methodology, which consists of detailed guidelines and templates, for constructing text-based ontology. Our methodology consists of 3 major steps and 14 minor steps. To implement the philosophy ontology, we develop an ontology management system based on Topic Maps. Our system includes a semi-automatic translator for creating Topic Map documents from the output of conceptualization steps and other tools to construct, store, retrieve ontologies based on Topic Maps. Our methodology and tools can be applied to other learning domain ontologies, such as history, literature, arts, and music.
© 2006 Elsevier B.V. All rights reserved.

Keywords: Ontology building methodology; Philosophy ontology; Topic maps

1. Introduction

An ontology is an explicit formal specification of terms, which represent the intended meaning of concepts, in the domain and relations among them. The primary goal of an ontology is share and reuse of domain knowledge among people or machines, i.e. software agents. It is a crucial factor for the success of many knowledge-based applications like the Semantic Web, data warehouse, Knowledge Management System, E-Commerce, and E-learning [14].

However, it is difficult to find concrete ontologies within real-world knowledge domains because developing the actual ontologies requires huge efforts, hours, and cost. In this paper, we present a large ontology which conceptualizes knowledge in the philosophy domain. We call it a philosophy ontology. Our philosophy ontology is composed of many small ontologies and each of them represents pieces of philosophical knowledge, such as philosophers, texts of philosophy, terms of philosophy,

doctrines of philosophy, schools of philosophy, and so on. The outstanding characteristics of the philosophy ontology are externalization, formalization, and specification knowledge existing within the contents of texts. The philosophy ontology enables people to understand the meaning of main concepts within the contents of texts and semantic relations between them without reading the whole texts.

Ontology building is a labor-intensive work, thus, it requires an engineering method to formalize the building process and activities in each of the steps. In this paper, we present a methodology for building the philosophy ontology, which provides a standard building process, detailed activities, and the products to be output at each step for maintaining consistency and validation of ontologies within and between development teams. The building process of the philosophy ontology has 3 major steps like planning, conceptualization, and implementation and 14 minor steps within them.

We design a three-layered architecture for implementing the philosophy ontology, which is composed of Philosophy Reference Ontology (PRO), Philosophy Domain Ontology (PDO), and Philosophy Text Ontology (PTO). Because the

---

* Corresponding author.
  E-mail address: jmkim@oopsla.snu.ac.kr (J.-M. Kim).

philosophy ontology conceptualizes not only philosophical knowledge, but also textual information, PDO presents philosophical domain knowledge and PTO presents textual philosophical knowledge. Philosophical domain knowledge is general knowledge such as who Immanuel Kant is, who the author of "*German Ideology*" is, and what the relationships between Kant and Hegel are. Textual philosophical knowledge is implicit knowledge that can be learned through reading the whole texts including what practical reason means, and the relationships between practical reason and free will. PRO is the upper level ontology, which provides a schema and several templates for maintaining consistency of PDO and PTO.

For convenience in creating ontologies with a formal language, we develop a semi-automatic translator for creating XTM (XML Topic Maps) [13] documents from the ontology specification, which is created during conceptualization of the philosophy ontology. XTM is the standard formal language that defines XML syntax for Topic Maps [2,6] and gives a specific Topic Map data model. A Topic Map is defined as a collection of Topic Map documents, thus, we should write XTM documents which correspond with philosophy ontology specifications. Topic Maps are an ISO standard (ISO 13250) and used as powerful access structures for dynamic collections of information resources. Currently Topic Maps are used to represent and build an ontology on the computer as well as RDF [10] and OWL [1].

In addition, we implement a topic map management system, in order to build philosophy ontology on the computer. Our system has several components to create, manage, store, retrieve, and export ontologies such as Topic Map Object Wrapper (TMOW), Storage Wrapper (SW), Topic Map Provider (TMP), Topic Map Cache Manager (TMCM), Topic Map Factory (TMF), and Topic Map Utilities (TMU) [4].

The contributions of the philosophy ontology built in this research are as follows. Firstly, it conceptualizes and externalizes knowledge of text contents to catch main concepts without reading the texts. Secondly, it provides guidelines for developing text-based ontologies in other learning domains such as, history, language, art, and so on. Thirdly, it can be used as a knowledge map of a digital library, knowledge portal, or document management system used to retrieve and explore semantic information.

In this paper, I will briefly review earlier methodologies for building ontologies and philosophy resources on the Internet in Section 2. In Section 3, the entire philosophy ontology development process and detail work in each step will be described. I will introduce each conceptualization of philosophy domain knowledge and philosophy text ontology in Sections 4 and 5. In Section 6, conceptualized ontology is translated into topic maps and implemented on the computer using a custom topic maps management system. Section 7 concludes this paper with discussion, summary, and future work.

2. Related work

Ontology engineering methodology has been extensively studied, in order to construct more efficient ontologies. This methodology can be divided into two distinct sections, forward engineering and backward engineering.

Forward engineering is a process that is composed of analyzing knowledge resources, extracting concepts from them, and establishing relations among concepts by domain experts. Enterprise Methodology [18], TOronto Virtual Enterprise (TOVE) [3], METHONTOLOGY [5], On-To-Knowledge [14], and Development 101 [8] are such examples of forward engineering. These different methods define ontology development phases from a higher level viewpoint [7].

Enterprise Methodology is composed of 4 phases including purpose identification, ontology building, evaluation, and documentation. TOVE methodology has a unique feature in that it provides the formulation of competency questions answered by a model built based on the specific ontology. The METHONTOLOGY methodology is based on IEEE standards for Developing Software Life Cycle Processes, in order to build their ontologies. On-To-Knowledge methodology is based on two-loop architecture, knowledge process and knowledge meta-process, for introducing and maintaining ontology-based knowledge management. The knowledge meta-process is a methodology of ontology development composed of five major steps with 13 sub-steps. The Development 101 methodology is a more detailed guideline for building ontology. It introduces how to design ontology, with a few fine-grained guidelines such as those required for class identification processing, and so on. This guideline originated from literature on object-oriented design.

The ontology development techniques based on backward engineering are semi-automatic processes that extract noun or verb terms from semi-structured or free style text resources using natural language processing techniques and construct ontology using machine-learning algorithms [15,16]. Backward engineering applicable to knowledge domains, however, are not required that much because reference ontologies relevant to the domain's resources are rare and extraction of noun terms from resources is extremely difficult.

Philosophy web pages, which can be accessed through the Internet, can be classified into the following four types; (1) the philosophy text sites [11], which provide contents of texts written in the original language or translated into English. These websites are useful to people who read computer texts and extract some paragraphs from them. (2) Philosophy journal sites [9], which provide papers and philosophy articles directly or as hyperlinks. These websites support publication and reference of philosophy papers on the web for easy access. (3) Philosophy reference sites [17], which provide philosophy dictionaries or encyclopedias. These websites are useful in obtaining descriptions and explanations of philosophical terms, philosophers, texts of philosophy, philosophical subjects, and so on. (4) Philosophy meta-sites [12], which provide directories for accessing philosophy-related websites. These websites introduce information and meta-data of texts regarding philosophy, journals of philosophy, organizations of philosophy, and topics of philosophy.

The philosophy ontology here provides not only full contents of texts but also semantic knowledge such as concept thesaurus and semantic-based associations among philosophy concepts.

This method also constructs a semantic network through conceptualizing and interconnecting concepts.

3. Philosophy ontology development process

The process for building ontology is similar to the computer software development process like planning, modeling, implementing, testing, and maintaining. This process is defined as a top-layer process, which specifies the entire building process. However, more detailed steps and guidelines are required, in creating effective philosophy ontology.

Fig. 1 shows the detailed steps for building the proposed ontology. The ontology development process is composed of the following three major steps: planning, conceptualization, and implementation. Each of the major steps is composed of several minor steps.

3.1. Purpose and scope identification

The first step of building ontology is to identify the ontology's primary objectives, scope, granularity, and characteristics. The primary goal of the philosophy ontology is to provide semantic information for philosophy knowledge management and philosophy e-learning. The characteristics of the philosophy ontology are externalization, formalization, and specification knowledge existing within the contents of texts.

We defined two levels of philosophy ontology granularity. Philosophy ontology enables general users to retrieve and explore common knowledge of philosophy such as who *Immanuel Kant* is, who wrote *Critique of Practical reason*, and who the pupil of *Descartes* is. In addition, the philosophy ontology provides a finer granularity of semantic information which is knowledge of the contents of texts, such as the relationship between practical reason and pure reason, and the definition of practical reason.

3.2. Knowledge acquisition

Knowledge acquisition is defined as the step toward collecting information resources being analyzed by domain experts with a view to extract knowledge or concepts. Fundamental information resources for building the philosophy ontology cover philosophy-related web pages, philosophy dictionaries, encyclopedias, and philosophy texts. Under knowledge acquisition, above all, information resources are classified into seven categories according to period and geographical criteria. These seven categories are Korean philosophy, Chinese philosophy, Indian philosophy, Ancient western philosophy, Medieval western philosophy, Modern western philosophy and Contemporary western philosophy. In this classification the first three are categorized as oriental philosophy and the rest as western philosophy.

For achieving two levels of granularity of the philosophy ontology knowledge acquisition, two techniques, domain knowledge analysis and formal text analysis, are used. Domain knowledge analysis is used to extract general knowledge from
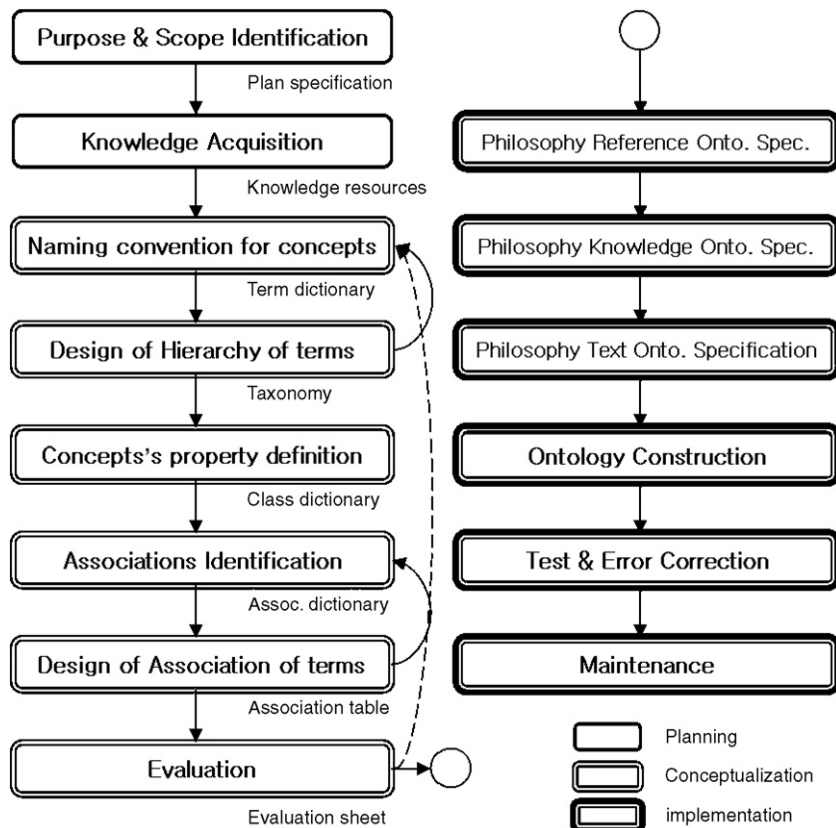

Fig. 1. Philosophy ontology development process.

philosophy resources on the Internet, philosophy dictionaries, and encyclopedias. Formal text analysis is used to find and externalize semantic knowledge from the contents of texts.

3.2.1. Domain knowledge analysis

General knowledge in the philosophy ontology can be classified into the following six categories; philosophers, texts of philosophy, terms of philosophy, branches of philosophy, schools of philosophy and doctrines of philosophy. For each category we selected the essential instances from collection of philosophical resources because philosophy consists of a vast knowledge resource from which we cannot extract all knowledge of philosophy. For example, we selected only well-known and influential philosophers according to the period and the geographical criteria, such as Yulkok, Wonhyo, and Yi Hwang as Korean philosophers, Confucius, Laozi, and Mencius as Chinese philosophers, Plato, Socrates, and Aristotle as Ancient western philosophers, Immanuel Kant and Hegel as Modern western philosophers, and so on. But we will expand the philosophy ontology to include more instances continuously because building ontology is a long-term project.

For each instance we acquire objective facts rather than subjective arguments from the collection of philosophical resources because knowledge in the philosophy ontology should be acceptable to most of domain experts. For example, we examine philosophical resources to acquire knowledge of philosophers with the following questions.

(1) What are his original name, English name, and Korean name?
(2) What is his biography?
(3) What are his main ideas?
(4) What are his active fields, schools, branches?
(5) What are his writings?
(6) Who are the philosophers related with him?

3.2.2. Formal text analysis

Domain experts analyze texts related to their major research fields. First of all, they look for basic information of the allocated texts, such as original title, Korean title, English title, author(s), published date, and so on. Afterward, they analyze their texts to answer the following questions.

(1) What are the main philosophical subjects (or issues)?
(2) What are the arguments of author(s)?
(3) What are the important philosophical terms included in the texts?
(4) What are the philosophers related with the contents of the texts?
(5) What are the texts of philosophy related with the contents of the texts?
(6) What are the branches of philosophy related with the contents of the texts?
(7) What are the schools of philosophy related with the contents of the texts?
(8) What are the doctrines of philosophy related with the contents of the texts?

The main philosophical subjects are used for identifying the central concepts of the text, which are importantly described through the whole text by author. These concepts are specialized with more specific concepts during conceptualization. The arguments of author(s) are also used to identify the central concepts of the text. Domain experts should identify not only philosophers, texts, terms, branches, and schools related with the contents of the texts, but also what types of relations are described in the contents of them.

3.3. Conceptualization

Once the previous knowledge acquisition step has been completed, the ontology designer, who may be the domain expert having knowledge of ontological engineering, has considerable unstructured and informal knowledge that must be formalized and organized. In the conceptualization steps, the ontology designers formalize, organize and structure acquired knowledge through the term definition of concepts. They organize hierarchy relation between broader concepts and narrow concepts and associate concepts with other semantically related concepts. Specifically, this phase builds the philosophy ontology using a set of intermediate representations that the domain expert, ontology designer and ontology implementer can understand. We explain a more detailed conceptualization process and a set of intermediate representations in Sections 4 and 5.

3.4. Implementation

For knowledge service based on semantic information, the conceptualization designed ontology must be transformed into a machine-interpretable data model using a formal representation language such as RDF/S, OWL and Topic Maps. A software system is required to interpret the transformed ontology's syntax, construct the ontology in the memory of a computer, and store it in the repository.

In Section 6, we describe the implementation process of the philosophy ontology using topic maps and the topic maps management system, enabling us to create, maintain, store, and retrieve the ontology.

4. Philosophy domain ontology conceptualization

The philosophy ontology is divided into the philosophy domain ontology and philosophy text ontology because it represents two kinds of knowledge, general knowledge of philosophy domain and content knowledge of philosophy texts. Firstly, for conceptualizing philosophy domain ontology, we determine the name of concepts, and design the hierarchical structure and associations for them.

4.1. Concept naming

It is important to get a comprehensive list of terms that correctly represent the intended meaning of concepts without any ambiguities. Domain experts are responsible for selecting or inventing adequate terms for concepts. They reference

Table 1
Naming convention for philosophy domain concepts

Rule	Rule description	Example
Naming form	Noun, proper noun or complex noun is used.	Philosopher, practical reason, Immanuel Kant
Singular or plural	Both can be used. But it should be consistent.	Doctrines of philosophy, schools of philosophy
Capitalization	First character should be capitalized.	Korean philosopher
Delimiter	Space is used rather than or dash.	Pure reason
Uniqueness	All terms should be unique.	Reason in the oriental, reason in the western
Length	Basically length of a term is not limited but it needs to decide max length of a term.	Ancient western philosopher
Abbreviation	Using abbreviated term is not recommended. Use full name as possible.	

Fig. 3. Hierarchical structure of the philosophy domain ontology.

philosophy dictionaries and encyclopedias to decide standard terms for representing the meaning of concepts. In addition, they need a naming convention to make consistent naming among them. Table 1 shows our naming convention for concepts in the philosophy domain ontology. During concept naming we build a glossary of terms including all terms of the philosophy ontology and their descriptions. Fig. 2 shows an example of the glossary of terms.

4.2. Concept hierarchy definition

There are three methods in developing concept hierarchy. These are defined as the top-down approach, bottom-up approach, and middle-out approach [7]. The top-down approach starts with one or more root concepts, represented by the most general concepts in the domain. Subsequently, it specializes the concepts until there are no more specialized concepts. We use this approach to develop the concept hierarchy in the philosophy domain ontology.

The bottom-up approach starts with the most specific concepts defined as the leaves of the hierarchy. Subsequently, related concepts are grouped i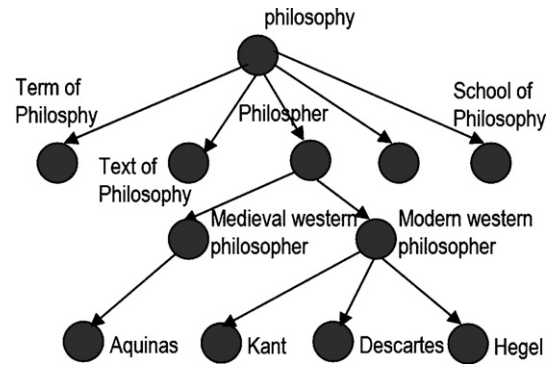nto more general concepts. A middle-out approach starts from outstanding concepts, considered as more important concepts by domain experts. Subsequently, these are generalized or specialized as appropriate. This approach is a combination of the top-down and bottom-up approaches.

According to the top-down approach the top-most concept, philosophy, in the proposed ontology is determined. Then, the philosophy concept is specialized by creating six sub-concepts: philosopher, text of philosophy, term of philosophy, branch of philosophy, doctrine of philosophy, and school of philosophy. We categorize philosophy domain concepts by these, because these sub-concepts are parts of the philosophy concept.

Subsequently, the philosopher concept has seven sub-concepts, identified in terms of period and geographic viewpoint: Korean philosopher, Chinese philosopher, Indian philosopher, Ancient western philosopher, Medieval western philosopher, Modern western philosopher, and Contemporary western philosopher. There is an "is–a" relationship, between the philosopher concept and its seven sub-concepts.

Fig. 3 shows the hierarchy structure of the philosophy domain ontology, organized into four levels. The root concept and intermediate concepts are considered as a class and leaves are an instance. For example, the modern western philosopher concept has few philosophers who had lived in the age of the modern philosophy such as *Immanuel Kant*, *Descartes*, *Hegel*, and so on.

Abbreviation	Using abbreviated term is not recommended. Use full name as possible	

Korean name	English name	Description
철학자	Philosopher	A person who is involved in the creation or study of philosophica knowledge.
철학분야	Branch of philosophy	A subfield of philosophy concerned with a specific philosophy problem.
철학이론	Doctrine of philosophy	A doctrine of philosophy proposed by a philosopher or a philosophical school.
철학용어	Term of philosophy	A person who is involved in the creation or study of philosophica knowledge.
철학학파	School of philosophy	A group of philosophers who share common doctrines of philosophy.
철학문헌	Text of philosophy	A text written by a philosopher, or a text which contains some philosophical knowledge.

Fig. 2. An example of the glossary of terms in the philosophy ontology.

CID	philosopher			LEVEL		1.1
Concept Type	Class					
Concept Name	Korean name: 철학자		English name: Philospher			
Super Class	philosphy					
Sub Class / Instance	kor_philosphy anc_philosphy					
Properties	Name	Data / Reference	Cardinality	Domain	Attribute Type	
	Korean name	Data	1	String	Person name	
	English name	Data	1	String	Person name	
	Web page	Reference	0..n	String	Web resource	
Relationships	Name	Origin	Ori_Role	Destination		Dest_Role
	author of	philsopher	author	phil_text		book
	teacher of	philsopher	teacher	philsopher		student
	has main idea	philsopher	creator	phil_doctrines		doctrine

Fig. 4. A concept's definition in the concept dictionary.

4.3. Concept's property definition

A concept of an ontology looks like a class in object-oriented design. A concept has a name, representing the intended meaning of itself. However, a concept's name alone will not provide enough information that is useful knowledge to learn from ontology. Thus, a concept must consist of one or more properties describing the meaningful attributes which belong to it. The properties of domain concepts are defined and documented in the concept-property table, depicted in Fig. 4.

In Fig. 4, the philosopher concept has the following properties: English name, original name, biography, biographical sketch, figure, and so on. Similar to a class, all sub-concepts of a concept inherit the properties of that concept. For example, all the properties of the concept philosopher will be inherited to all sub-concepts of philosopher, including Korean philosopher, Chinese philosopher, and Indian philosopher. We will add additional properties, hanja (Chinese character) name, to the Korean philosopher and Chinese philosopher.

4.4. Concept association definition

An association is a binary relation between two concepts, such as "*synonym*", "*disjoint of*", "*author of*", "*contribute to*", "*pupil of*", and so on. An association is a kind of semantic information-based relation that explains how a concept would

Association	Concept1	Concept2
author of	Immanuel Kant	Critiques of Practical Reason
author of	Immanuel Kant	Critiques of Pure Reason
related philosopher of	David Hume	Immanuel Kant
related philosopher of	John Locke	Immanuel Kant
contributed doctrine of	Idealism	Immanuel Kant
contributed branch of	Epistemology	Immanuel Kant
contributed branch of	Metaphysics	Immanuel Kant

Association of Philosophy Domain Ontology

Fig. 5. Association table of philosophy domain ontology.

be related to another concept. Ontology has superiority over a taxonomy and thesaurus because of the associative relationships established between concepts.

An association can be classified into two types, explicit and implicit. An explicit association is an obvious relationship that domain experts can identify and specify with ease. For example, the association "*author of*" is established between a philosopher and a text and the association "*pupil of*" is established between one philosopher and another philosopher. An implicit association is a hidden relationship that can be identified from analyzing revealed explicit associations. For example, the association "*author of*" has an inverted relationship, "*written by*" and if the association "*opponent of*" exists between text A and text B, the same relationship is established between philosopher A and philosopher B, who are the authors of both texts A and B.

Fig. 5 shows the association table of the philosophy domain ontology. We defined six associations between a philosopher and other concepts which belong to the philosophy concept. These associations are inherited to all philosopher instances, such as Immanuel Kant, Plato, Descartes, Karl Marx, and so on.

5. Philosophy text ontology conceptualization

The philosophy text ontology organizes and specifies knowledge contained in the contents of texts. Thus the ontology conceptualization process starts by extracting and naming of concepts from texts. After naming of concepts, definition of concept hierarchy, property, and associations follow.

5.1. Concept naming

Naming rules for concepts in the text ontology follows the naming convention of the domain ontology described in the previous chapter. However, most of all the text ontology's concept names take the form of noun phrases, such as "*features of practical reason*", "*major difference between practical reason and pure reason*", "*combine of virtue and happiness*", and so on.

One of the naming rules is uniqueness which means that any term in the philosophy ontology must have a unique name. According to the rule, we should rename the duplicated names of concepts which have different meaning. But in the philosophy ontology, the rule is not always applied because meaning, definition, or usage of a term can be differently described in several texts of philosophy. In this case, it is not appropriate that we rename the duplicated name with a unique new name because it may violate the intended meaning of a term which the author wants to explain within his texts.

For example, Immanuel Kant explains the philosophical meaning of reason within his texts, "*Critique of Practical Reason*" and "*Critique of Pure Reason*", whereas Hegel explains different meaning of reason within his texts. In this case, for preserving the uniqueness rule, it is not appropriate to rename reason with "*reason of Kant*" or "*reason of Hegel*".

We use scoping rather than renaming to preserve duplicated names and their different meaning of a philosophical term. Scoping restricts the semantic range of duplicated names into texts which describe definition, usage, and philosophical meaning of them. We aggregate and relate these duplicated names with general definition of the term to enable users to easily understand general and specialized definition, meaning, and usage of it. Fig. 6 shows an example of the glossary of scoped terms. From the example, we know that Descartes defines the meaning of liberty in his book "*Meditation*" and Kant also defines differently the meaning of liberty in his book "*Critique of Practical Reason*". When users wish to understand various concepts of liberty, they can retrieve all liberty words scoped with names of texts and the general concept as well as their specialized concepts of liberty.

5.2. Concept hierarchy definition

A middle-out approach should be used to develop a concept hierarchy of the text ontology because a text describes particular subjects and includes specific concepts rather than general concepts. For example, "*Critique of Practical Reason*" written by Kant describes several philosophical issues, practical reason, autonomy, moral law, free will, and so on. These issues are the main concepts that are emphasized in the text.

According to the guideline of a middle-out approach, we should first consider finding the central concepts in the texts. To obtain these concepts, domain experts analyze the table of contents, term glossary or index, and preface of the texts. They finally determine the main concepts from the obtained candidates through analyzing the contents of the texts.

After acquiring the main concepts, more general concepts are derived from the main concepts using a bottom-up approach. General concepts must be used to collect semantically related

Korean name	English name	Description	Scope	General Term
자유	Liberty	Descartes describes two concepts of liberty in this text. One is liberty from forced passions and another is liberty from inevitability	Meditation of Descartes	Liberty
자유	Liberty	The true liberty of a man, as it speaks pure perfection, is when by the right use of the faculty of free will, ...	Critique of Practical Reason of Kant	Liberty

Fig. 6. An example of the glossary of scoped terms in the philosophy ontology.

main concepts. For example, "*reason*" concept aggregates specific concepts, which describe the issues related with reason. These specific concepts are described in the different texts of philosophy, such as "*pure reason*" and "*practical reason*" of Kant, "*social reason*" of Hegel, and so on.

We identify general concepts through analyzing the structure of a main concept name. If a main concept's name was a complex noun or noun phrase, we try to find a noun where its meaning was restricted by other adjectives or nouns. For example, we know that "*practical*" restricts the meaning of "*reason*" from "*practical reason*", and "*practical reason*" is a specific meaning of the "*reason*" concept.

In addition, more specific concepts are derived from the main concepts using a top-down approach. These specific concepts organize sub-trees rooted by each of the main concepts. Specialized concepts are extracted from analyzing the contents of those pages, because a main concept is explained over many pages. The leaves of text ontology are specific concepts, representing the meaning included in one or more paragraphs because a paragraph is a meaningful unit that can be summarized and conceptualized by domain experts.

Fig. 7 shows an example of a hierarchical relation of a main concept and its specialized concepts. In this figure, CID represents the concept identifier, which may be manually specified by an ontology implementer or automatically generated by an ontology management system. Resource ID indicates one or more paragraphs in a text resource, referenced by a concept.

5.3. Concept's property definition

All concepts of the philosophy text ontology have two different kinds of properties, explanation and quotation. A concept's explanation property is a description of the intended meaning of the concept and the quotation property is a reference to particular paragraphs within philosophy texts. An explanation property is divided into an internal explanation and external explanation. An internal explanation is a short description, which exists inside the text ontology, and the external explanation is a reference to certain paragraphs of explanation articles, which are written by domain experts through analyzing philosophy texts. Explanation articles are used to provide more understandable information to novices in the philosophy domain.

We digitalized all explanation articles and some texts, of which copyrights are owned, with HTML and PDF formats. An external explanation property and quotation property has a hyperlink to these digitalized files. These files and other information resources such as images, web pages, and movie clips construct a knowledge base of philosophy ontology.

5.4. Concept association definition

In the philosophy text ontology, an association is defined to represent the semantic relationship between concepts in a text or different texts. In the index of a printed book, two terms are related with only 'see also' relationship. This indicates that some semantic relationships exist between two terms but these relationships do not represent obvious relationships.

We define semantic relations which exist between concepts in the text ontology, such as "*be identical to*", "*be opposed to*", "*complementary to*", "*sequence to*", "*cause and result*", and so on. Fig. 8 shows the concept association table of the text ontology.

In contrast to associations of domain ontology, text ontology's associations need to explain the meaning of

Concept Hierarchy of Philosophy Text Ontology

Text : The Critique of Practical Reason(Immanuel Kant)

CID	Concepts Hierarchy	Resource ID
c1	Practical Reason	
c11	Speculative reason and practical reason	P1.1.2.1
c111	Equivalence of speculative reason and practical reason	P1.1.3.1
c112		
c113	Use of reason	P1.1.3.5
c1131	Speculative use of reason	P1.1.3.8
c1132	Practical use of reason	P1.1.4.1
c1133	Comparing speculative use to practical use	P1.1.4.3
c12	Superiotiry of practical reason	P1.1.5.1
c121	Semantic of superiority	P1.1.5.2
c122	When speculative reason has superiority	P1.1.5.6
c123	When practical reason has superiority	P1.1.6.5
c2	Moral Law	
c21	Origin of moral law	P1.2.1.3
c22	Moral law and reason	P1.2.2.1
c23	Phase of moral law	P1.2.3.3
c24	Moral law and happiness	P1.2.4.1
c241	Critique of moral sensibility	P1.2.4.4
c242	Principles of self-happiness	P1.2.4.9
c243	Difference between principles of self-happiness	P1.2.5.11
c244	and moral law	P1.2.4.15

Fig. 7. Hierarchy structure of philosophy text ontology.

Association of Philosophy Text Ontology

Association	Concept1	Concept2
be identical to	Socialism theory	Communism theory
be identical to	Historic materialism	Materialistic view of history
be opposed to	Historic materialism	Ideological view of history
assertion and foundation	Importance of materialistic production activity	Foundation of societary life
be identical to	Doctrine of science	Positivism
prescription of	Foundation	Social structure
mutual action	Productivity	Production

Fig. 8. Association table of the philosophy text ontology.

semantic relationships. For example, we can describe why historic materialism is opposed to the ideological view of history, from Fig. 8. These associations are called reified associations, which posses explanation and quotation properties to describe the meaning of semantic relations.

6. Philosophy ontology implementation

After conceptualization, ontology implementers should design ontology structure and specify the conceptualized ontology using a machine-understandable formal language

Fig. 9. Three-layered architecture of philosophy ontology.

```
        <baseName><base Name String>Korean name</baseNameString></baseName>
    </topic>
    <topic id="biography">
        <baseName><baseNameString>Biography</baseNameString></baseName>
    </topic>
    <topic id="figure">
        <baseName><baseNameString>Figure</baseNameString></baseName>
    </topic>
    <topic id="webpage">
        <baseName><baseNameString>Webpage</baseNameString></baseName>
    </topic>
    <topic id="philosopher">
        <instanceOf><topicRefhref="#philosophy"/></instanceOf>
        <baseName><baseNameString>Philosopher</baseNameString></baseName>
        <occurrence>
            <instanceOf><topicRefhref="#korean_name"/></instanceOf>
            <resourceDate>철학자</resourceData>
        </occurrence>
        <occurrence>
            <instanceOf>>topicRefhref="#biography"/></instanceOf>
            <resourceData></resourceData>
        </occurrence>
    </topic>
```
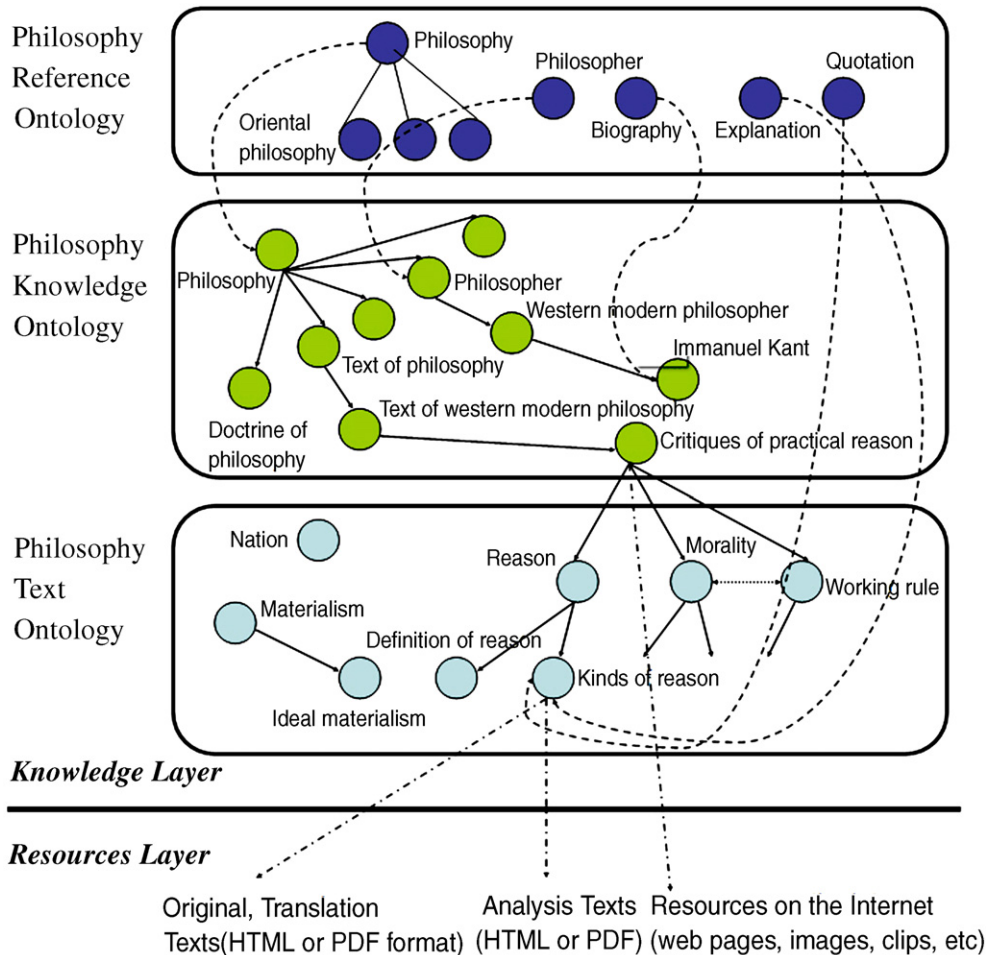
Fig. 10. An XTM representation of the philosopher concept and some of its properties.

such as RDF, OWL, or Topic Maps. We use Topic Maps to specify our philosophy ontology. Finally, our topic map management system interprets Topic Map documents and constructs our philosophy ontology.

6.1. Three-layered architecture of philosophy ontology

The implemented philosophy ontology is composed of three ontologies, Philosophy Reference Ontology (PRO), Philosophy

Domain Ontology (PDO), and Philosophy Text Ontology (PTO). Fig. 9 shows a three-layered architecture of the philosophy ontology. PRO provides a schema and several templates, which are referenced by PDO and PTO to build valid ontologies. We defined upper level concepts, which categorize PDO philosophy domain concepts and property concepts, referenced by PDO and PTO concept properties. For example, in Fig. 9, Philosophy, Philosopher, and Modern western philosopher of PRO are upper level concepts which categorize

```
Hegel.xtm
    <topic id="hegel">
        <instanceOf> <topicRef xlink:href="philosophy.xtmp#mod_philosophers"/> </instanceOf>
        <baseName> <baseNameString>Georg Wilhelm Friedrich Hegel</baseNameString> </baseName>
        <occurrence>
            <instanceOf> <topicRef xlink:href="philosophy.xtmp#bio"/> </instanceOf>
            <resourceData>Born in Stuttgart and educated in Tübingen, Georg … </resourceData>
        </occurrence>
        …
    </topic>
    <association>
        <instanceOf> <topicRef xlink:href="philosophy.xtmp#teacher_of"/> </instanceOf>
        <member>
            <roleSpec> < topicRef xlink:href="philosophy.xtmp#teacher"/> </roleSpec>
            <topicRef xlink:href="kant.xtm#kant"/>
        </member>
        <member>
            <roleSpec> < topicRef xlink:href="philosophy.xtmp#pupil"/> </roleSpec>
            <topicRef xlink:href="kant.xtm#hegel"/>
        </member>
    </association>
</association>

Kant.xtm
    <topic id="kant">
        <instanceOf> <topicRef xlink:href="philosophy.xtmp#mod_philosophers"/> </instanceOf>
        <baseName> <baseNameString>Immanuel Kant</baseNameString> </baseName>
        …
    </topic>
```
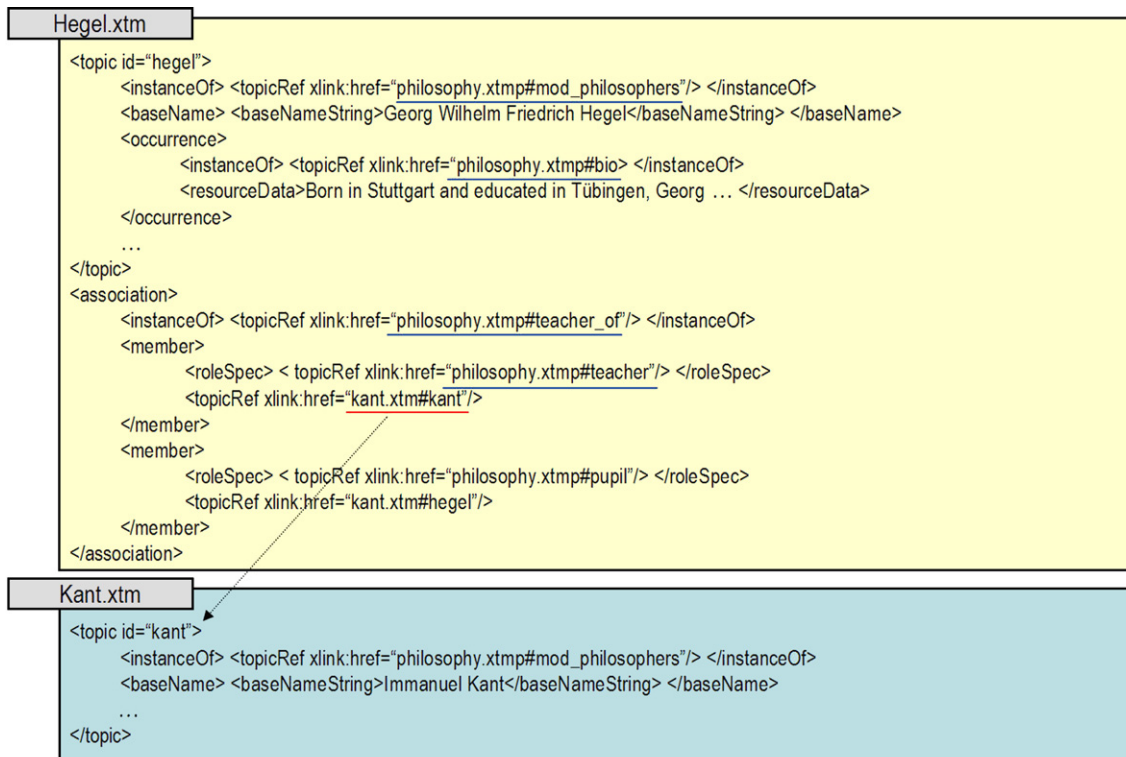
Fig. 11. An example of an association between Kant topic and Hegel topic for representing that Kant is teacher of Hegel.
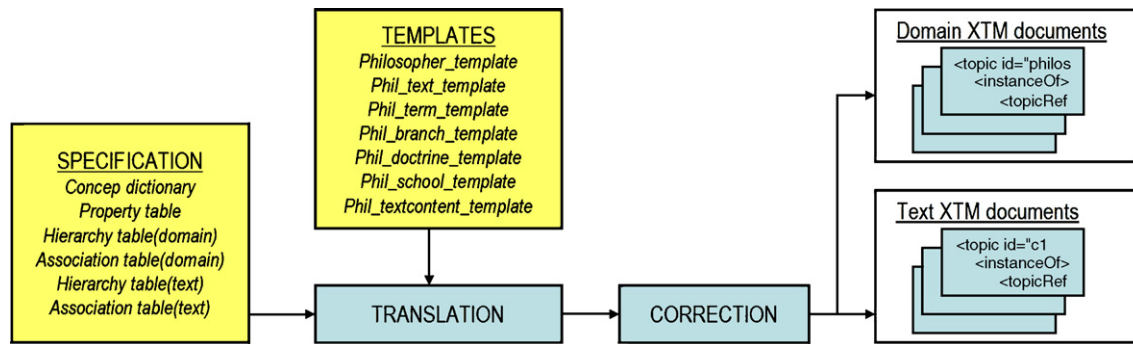
Fig. 12. Translation process for creating XTM documents from the conceptualization specification.

real philosopher concepts of PDO, such as Kant, Hegel, Mill, and so on. In addition, Biography, Figure, Explanation, and Quotation of PRO are property concepts, which are used to scope properties of concepts of PDO and PTO.

6.2. Formalization of philosophy ontology

To implement the philosophy ontology on the computer, we should translate conceptualized informal specifications into formal representations of Topic Maps, XTM (XML Topic Maps). XTM is the standard formal language that defines XML syntax for Topic Maps and gives a specific Topic Map data model. A Topic Map is defined as a collection of Topic Map documents, thus, we should write XTM documents, which correspond with philosophy ontology specifications.

All philosophy ontology concepts are translated into topics of Topic Map documents. Properties are translated into topics and occurrences because a property itself is a concept and specifies a certain value to attribute of a concept. Fig. 10 shows an XTM representation of a philosophy concept, and its properties are depicted in Fig. 4.

A hierarchy relation is represented with an *instanceOf* XTM element. For example, in Fig. 10, a philosopher topic has an *instanceOf* element referenced by a philosophy topic. We know that the philosopher is a part of philosophy. An *instanceOf* element within an *occurrence* element references a property type of *resourceData* element's value. In addition, an association relation is represented with an *association* element of XTM. Fig. 11 shows an example of *association* element of XTM representing that Kant is teacher of Hegel.

We developed a semi-automatic translator for creating XTM documents from an ontology specification, which is created during conceptualization of the philosophy ontology using the above mapping rules. Input data for translation is ontology specification and templates. Templates are composed

of the following seven files: philosopher_template, phil_-text_template, phil_term_template, phil_branch_template, phil_doctrine_template, phil_school_template, and phil_text-content_template. First, our translator creates uncompleted XTM documents, which have empty properties from input data. Second, domain experts fill empty properties with identified values and manually correct the errors occurring in translated documents. For example, two XTM documents in Fig. 11 are similar in their element structure and attributes because both are created from the philosopher_template file (Fig. 12).

6.3. Building and exploring of philosophy ontology

Ontology can be constructed as a large ontology, including all concept or many separated small ontologies, which include concepts relevant to different themes and may be merged to build an integrated ontology. Our philosophy ontology is composed of many separated small Topic Maps including kant.xtm, kant_practical.xtm, kant_practical_km.xtm, hegel.xtm, hegel_right.xtm, hegel_right_km.xtm, and so on.

Our topic maps management system interprets formalized XTM documents and constructs Topic Maps in memory. Our system has several components used to create, manage, store, retrieve, and export Topic Maps such as Topic Map Object Wrapper (TMOW), Storage Wrapper (SW), Topic Map Provider (TMP), Topic Map Cache Manager (TMCM), Topic Map Factory (TMF), and Topic Map Utilities (TMU).

The philosophy ontology includes the results of concep-tualization of thirty-six famous oriental and western philosophy texts. These texts consist of three ancient western texts, two medieval western texts, twelve modern western texts, ten contemporary western texts, and nine oriental texts such as, *Nicomachean Ethics* by *Aristotle*, *Republic* by *Plato*, *Summa theologiae* by *Aquinas*, *Theory of Science* by

Table 2
The statistics of the philosophy ontology

	Philosopher	Text of philosophy	School of philosophy	Branch of philosophy	Doctrine of philosophy	Term of philosophy
Number of topic maps	208	89	102	58	78	587
Number of topics	221	15 367	110	65	80	599
Number of associations	1782	5322	621	398	430	320

Fig. 13. Philosophy knowledge portal based on the philosophy ontology.

Fichte, and so on. We will analyze and conceptualize more texts, which are essential in studying philosophy for experts and non-experts.

Table 2 shows the statistics of the ontology philosophy. These statistics show numbers of topic maps, topics, and associations in terms of six main categories including philosopher, text of philosophy, term of philosophy, school of philosophy, branch of philosophy, and doctrine of philosophy of the philosophy ontology. The philosophy column text includes topic maps, topics, and associations related to texts of domain ontology and text ontology.

Fig. 13 shows our philosophy knowledge portal based on the philosophy ontology. Users can explore philosophy knowledge through selecting one main category displayed in the main flash. For example, if users want to retrieve descriptions and associations of *free will*, they first select Texts of Philosophy and then they select one sub-category, Modern Western Texts of Philosophy. They get a list of titles of modern western texts of

Fig. 14. Search page to retrieve knowledge relevant to input keyword using full-text searching.

philosophy and will find *Critique of Practical Reason*. If they select the title, then they can acquire meta-knowledge of the text like title, author, outline, main terms, and so on. Consequently they can find *free will* from the list of main terms and its description and association by selecting it.

Fig. 14 shows a search page to retrieve knowledge relevant to an input keyword using a full-text search technique. User can find *free will* and its related concepts without exploring a portal.

7. Conclusion

In this paper, we developed the philosophy ontology, which conceptualizes knowledge of the philosophy domain and contents of philosophy texts. We provide detailed guidelines for constructing text-based ontology. Our guideline is composed of 3 major steps and 14 minor steps. This guideline can also be applied to other learning domain ontologies such as history, literature, arts, music.

The philosophy ontology includes contents knowledge of 37 texts of philosophy but will be expanded to almost all the texts of oriental and western philosophy which are considered as essential philosophy texts by experts. It organizes a text-based knowledge base, which includes electronic texts and philosophy articles.

We developed a semi-automatic translator for creating XTM documents from the ontology specification, which is created during conceptualization of the philosophy ontology. Our translator creates uncompleted XTM documents with empty properties from input data, then domain experts fill empty properties with identified values and manually correct the errors in translated documents.

We designed a three-layered architecture of philosophy ontology to implement on the computer. The implemented philosophy ontology is composed of three ontologies, Philosophy Reference Ontology (PRO), Philosophy Domain Ontology (PDO), and Philosophy Text Ontology (PTO). Due to this layered architecture, we take advantage of reuse and share of ontologies among similar philosophy-related domains.

The presented topic map management system builds philosophy ontology from translated XTM documents and a philosophy knowledge portal to service semantic information to users. This system is composed of several components, which build, store, and retrieve Topic Maps, a data model of ontology.

The philosophy ontology provides not only easy concepts to novices and students but also more specialized detailed concepts from the contents of texts to domain experts.

References

[1] D.L. McGuinness, F. Harmelen. OWL Web Ontology Language Overview, W3C Recommendation, 10 February 2003, http://www.w3.org/TR/owl-features/.

[2] M. Graham, Topic Map technology—the state of the art, XML 2000 Conference and Exposition, Washington, USA, December 2000.

[3] M. Gruninger, M.S. Fox, Methodology for the Design and Evaluation of Ontologies, IJCAI-95 Workshop on Basic Ontological Issues in Knowledge Sharing, Montreal, August, 1995.

[4] J.M. Kim, C.M. Park, J.W. Jung, H.J. Lee, K.S. Min, H.J. Kim, K-Box: ontology management system based on topic maps, Journal of the Korea Information Science Society 10 (1) (2004 (February)).

[5] M.F. Lopez, A. Gomez-Perez, J.P. Sierra, Building a chemical ontology using methontology and the ontology design environment, IEEE Intelligent Systems 14 (1) (1999 (January)).

[6] B. Michel, B. Martin, N. Steve, ISO/IEC 13250 TopicMaps.

[7] N.F. Noy, D.L. McGuinness, Ontology Development 101: A Guide to Creating Your First Ontology, SMI technical report SMI-2001-0880, 2001.

[8] R. Mizoguchi, Tutorial on ontological engineering—Part2: ontology development, tools and languages, New Generation Computing, vol. 22 (11), OhmSha & Springer, 2004.

[9] Online papers in philosophy, http://opp.weatherson.net/.

[10] Ora Lassila, Ralph R. Swick. Resource Description Framework (RDF) Model and Syntax Specification, W3C Recommendation, 22 February 1999, URL: http://www.w3.org/TR/REV-rdf-syntax.

[11] Perseus Classics Collection, http://www.perseus.tufts.edu/cache/perscoll_Greco-Roman.html.

[12] Philosophy in Cyberspace, http://www-personal.monash.edu.au/~dey/phil/.

[13] S. Pepper, G. Moore. XML Topic Maps (XTM) 1.0, TopicMaps.Org.

[14] S. Staab, H.-P. Schnurr, R. Studer, Y. Sure, Knowledge processes and ontologies, IEEE Intelligent Systems, Special Issue on Knowledge Management 16 (1) (2001 (January/February)).

[15] S. Staab, R. Studer, Handbook on Ontologies, Springer, 2004, pp. 133–150.

[16] S. Staab, R. Studer, Handbook on Ontologies, Springer, 2004, pp. 173–190.

[17] Stanford Encyclopedia of Philosophy, http://plato.stanford.edu/.

[18] M. Uschold, M. King, Towards A Methodology for Building Ontologies, IJCAI-95 Workshop on Basic Ontological Issues in Knowledge Sharing, Montreal, August, 1995.

Jung-Min Kim is a Ph.D candidate of the school of CS&E in Seoul National University. He received his B.S. and M.S. degrees in computer science from Hongik University in 1992 and 1994, respectively. He has been in the Philosophy Ontology Construction Project of the Institute of Philosophy in Seoul National University for 4 years. His research interests include ontology, text mining and information retrieval.

Byoung-Il Choi is a senior researcher of the Institute of Philosophy in Seoul National University. He received his B.A. and M.A. degrees in the Department of Philosophy from Yeosei University. He received his Ph.D. degree in the Department of Philosophy from the UC-Berkeley in 1997. His research interests include Semantic Web, ontologies and Topic Maps.

Hyo-Phil Shin is an associate professor of the Department of Linguistics in Seoul National University. He received his B.A., M.A. and Ph.D. degrees in linguistics from Seoul National University in 1988, 1990 and 1994, respectively. He received his M.S. degree in computer science from the University of Missouri-Kansas City in 1997. His research interests include NLP, ontologies and information retrieval.

Hyoung-Joo Kim is a professor of the School of CS&E in Seoul National University. He received his B.S. degree in computer engineering from Seoul National University in 1982, his M.S. and Ph.D. degrees in computer science from the University of Texas at Austin in 1985 and 1988, respectively. His research interests include object-oriented systems, databases, and electronic commerce.