PAPER *Special Section on Data Engineering*

XMDR+: An Extended XMDR Model for Supporting Diverse Ontological Relations

Jeong-Dong KIM[†a)], **Jiseong SON**[†], ***Student Members**, and* **Doo-Kwon BAIK**[†*], ***Nonmember***

SUMMARY Metadata registry (MDR) is based on the international standard ISO/IEC 11179. The committee of ISO/IEC JTC 1/SC 32, which had standardized the MDR, has started to improvise the MDR, and the improvised version is named extended MDR (XMDR). However, the XMDR does not fully support the ontology concept, and no method is available for mapping ontology registrations onto registries. To overcome the limitations of the outdated XMDR, this paper proposes an extended XMDR (XMDR+) framework. The XMDR+ framework provides a method for mapping of ontology registrations between the metadata registry and ontologies. To improve the functions of the XMDR, we have proposed herein a framework that is capable of defining a model that manages the relations not only among ontological concepts but also among instances, and guarantees the management and storage of their relationships for supporting valid relations of the ontologies.

key words: *metadata, semantic Web, ontology, MDR, OWL*

1. Introduction

The recent instances of progress in the field of information technology have led researchers to focus on service-oriented computing [1]. In addition, as mobile networks and computing technologies continue to advance, the demand for gaining accessibility to diverse services on a single platform unbounded by hardware and software types has led to the development of ubiquitous computing. Within the framework of ubiquitous computing, Semantic Web technology [2], [3] serves as the key to interoperability [4], which enables operators to provide various services irrespective of the time, location, or device used. Thus, studies are being conducted on metadata in different fields to selectively collect useful contents from the large amount of data existing in the Semantic Web environment [5]–[9] and to thereby provide and manage useful services [10]–[12].

The importance of using metadata functions as a guideline for defining data [13] has resulted in considerable attention being drawn to this topic lately. Accordingly, several fields that define standards for the management of data have come up [14]. Its typical examples used in various areas are the Dublin Core Metadata Initiative (DCMI) [15], which is used for bibliographical search purposes; J. metadata [16], which is used in the network area; MPEG-7 [17], which is closely related to multimedia; ebXML Registry Information Model (ebXML RIM) [18], which is useful in e-business; Universal Description, Discovery, and Integration (UDDI) [19], which is used for web-based services; and Metadata Registry (MDR: ISO/IEC 11179) [13], which serves as a vehicle for independent management of metadata in each domain.

Among the registries and saving devices, the MDR is a global standard system developed by ISO/IEC JTC 1/SC 32 to manage metadata in an organized manner and to define mechanisms, processes, and operations designed to generate and manage metadata. It also enables cross-reference and reuse of data used in various registries among different applications and organizations by predefining and later managing them [13], [20], [21]. As a result, interoperability is guaranteed. The MDR thus facilitates system integration and information sharing through its featured interoperability in fields such as e-business [18], transportation systems [22], medical systems [23]–[28], and environmental systems [29]. To facilitate better system integration and information sharing, the metadata must be integrated so as to manage heterogeneous aspects involving the semantics, syntax, and representation of data. In this context, in order to ensure and manage interoperability, the ultimate aim of the MDR lies in standardizing these three categories and sharing the details and meanings of metadata. In other words, to realize information integration and sharing and to provide a variety of useful services, the MDR allows metadata used in registries to be defined beforehand during the design of a metadata schemas and also allows the reuse of the existing data elements in other systems/registries, thereby guaranteeing accuracy and reuse of data meanings. Previous studies, however, have revealed the following shortcomings of metadata registries:

- **Shortcoming Concerning MDR**: The MDR-based system or registry is useful only for those who are able to conduct direct references to data sources. The system or the registry, however, is unable to extract the meanings of the data elements saved in the MDR or to automatically interpret them.
- **Shortcoming Concerning XMDR**: The XMDR (eXtended Metadata Registry: ISO/IEC 11179-Part 3, Edition 3) project defines a model for managing concept level relations of ontology technologies. It fails, however, to incorporate stances of the instance level. This failure arises from the inability to accurately manage the metadata meanings, resulting in the failure to guarantee accurate data on semantic interoperability in

Manuscript received June 6, 2010.
Manuscript revised September 25, 2010.
[†]The authors are with the Department of Computer Science and Engineering, Korea University, Seoul, Korea.
[*]Corresponding author.
a) E-mail: kjd4u@korea.ac.kr
DOI: 10.1587/transinf.E94.D.515

terms of storing, managing, and operation of the data to which ontology technologies are applied.

- **Inability to Conduct Mapping for Ontology Registration**: The metadata registry should be able to accommodate a previously defined ontological schema in order to use information in a more accurate and sufficient manner and to develop ontological resource-based services for a registry. In turn, to register such ontological schema, it is essential to save and manage ontological resources in a consistent manner by defining the mapping relations of registries and ontological resources.

The first shortcoming concerning the MDR arises from the inability to define relations between metadata and the data of different registries. If designers or developers adopt different definitions for the schema of metadata, a registry is unable to extract meanings or automatically interpret the relations of data elements that refer to the same concept. Consequently, interoperability is not ensured. Being aware of this problem, the XMDR project is being developed at the Lawrence Berkeley National Laboratory to address the failure to define relations between metadata and data of different registries [30]. Second, the XMDR project was originally aimed at resolving the first problem. To achieve this aim, it defines the elements saved in the MDR meta-model, such as classes, relations among them, data types, and a characteristic specified within classes by means of ontological techniques; hence, it enhances the machine readability of already defined metadata elements. However, it fails to support semantic interoperability because the semantic accuracy of data is not guaranteed while storing and managing the data related to ontological instances. This failure can be attributed to the technological scope being limited to the concept level. The third problem, which is related to ontology registration, arises because of the absence of a consistent mapping method for connecting ontological concepts to metadata registries. By using the notion of a concept system to define registry characteristics and ontological concepts, separate models are designed for various domains. This concept system-based approach, however, is unable to address the mapping issue related to the characteristics of metadata registries and ontological concepts. Therefore, it is necessary to ensure that the mapping is carried out such that the registration and management of ontological schemas take into account the attributes of the concept system.

To improve the functions of the metadata registry, this paper proposes a framework that is capable of defining a model that manages the relations not only among ontological concepts but also within the instance level and is also capable of consistently keeping and managing data through the mapping management function for ontology registration. The framework proposed applies ontology technologies for semantic extraction of data elements and interpretation of their relations. Furthermore, it is capable of storing information on ontological support and manages ontological concepts, instances, and their diverse relations. To support these functions, which would ultimately provide accurate information, the framework executes a variety of functions such as the utilization of the concept level defined in the XMDR, delineation of the instance-managing model for the instance level, specification of the management model that incorporates the concept and instance levels, and, thereby, provision of efficiency of semantic relations between the concept and instance levels. In the end, the semantic differences between registries are resolved, which eventually leads to improved interoperability of data.

The rest of this study is organized as follows: Sect. 2 describes the MDR and XMDR and reviews relevant issues, Sect. 3 defines the proposed XMDR+ (Extended eXtended Metadata Registry) framework, Sect. 4 describes its experiment and evaluation, and Sect. 5 contains the conclusions and recommendations for future work.

2. Related Work

2.1 Metadata Registry (ISO/IEC 11179; MDR)

The MDR is a framework used for achieving interoperability of databases and is an international standard developed by ISO/IEC JTC 1/SC 32 [13]. This international standard is divided into 6 parts that specify the basic attributes and administration of metadata. Items registered include individual data elements and related metadata items such as data element concepts, value domains, taxonomies, and identification of responsible parties. The framework lays emphasis on semantic information such as definitions of data elements, definition of each value meaning, and stewardship responsibilities. The focus of these metadata registries is data administration, semantics management, and data standardization. Traditional MDR supports data sharing, data reporting, system development, and dissemination of information that describes data products.

The MDR consists of two main parts: the *Conceptual Level* and the *Representational Level*. The conceptual level contains two components: *Data_Element_Concept* (*DEC*; *ex*:*County Name*) and *Conceptual_Domain* (*CD*; *ex*:*Countries*). The *CD* plays the role of a container, which defines the applicable domain of the *DEC*, both the *CD* and the *DEC* consist of a combination of an object concept and its corresponding properties. The *DEC* is described independently of any particular representation. The representation level, and the expression of the concepts in the conceptual level, contains the following components: *Value_Domain* (*VD*; *ex*:*ISO 3166 2*, *3-Alpha Code*, ...) and *Data_Element* (*DE*; *ex*:*County Name 3-Alpha Code='KOR'*). The *VD* is a set of permissible values for the *DEC*. The *DE* is the product obtained when a permissible value is applied to the *DEC*, and the *DE* includes descriptions such as definition, type, and length. Figure 1 shows the conceptual model of the main components for MDR.

The existing MDR was originally designed to manage metadata between table fields of a relational database. In other words, the MDR adopts a structure that enables the management of relations between tables and metadata of the data saved in each table in the relational database. As a result, it facilitates sharing and exchange of data, and generates and manages metadata. In this context, the features of the Semantic Web technology are (i) a storing structure suitable for the management of graph data, (ii) definition of various properties of the concepts, (iii) support for semantic inference, and (iv) URI management techniques that are vital to web ontology.

However, the existing MDR does not support any of these functions. Therefore, to resolve the semantic differences of the schemas uniformly defined among registries at ISO/IEC JTC 1/SC 32, a meta-model has been defined by including various ontological attributes and considering storing and managing ontological resources; the Semantic Web technology has been applied to this meta-model to enable a registry to automatically understand the attributes and to enhance the semantic interoperability of data. Figure 2 illustrates why alterative characteristics are required in the MDR for supporting Semantic Web.

2.2 eXtended Metadata Registry (ISO/IEC 11179-Part 3, Edition 3; XMDR)

The XMDR defines various classes and concepts to incorporate the notion of ontology technologies into it. On the basis of these definitions, efforts are being made to design a storing structure and a protocol that would facilitate the application of the ontology technologies. Currently, the XMDR being developed at ISO/IEC JTC 1/SC 32 is not limited to any particular application area; in fact, it is applicable to various applications and domains for independent storing and managing. The XMDR supports data description using a triple (subject, predicate, and object) set for ontological resources [30]. Under the XMDR scheme, the relations between concepts are defined in the ontology language (i.e., RDF, RDF-S, and OWL) to help a system/registry to automatically understand the meanings of the data elements specified in the registries. Thus, the XMDR is intended to understand data, infer their meanings, and more clearly define the meanings of the data elements defined in the MDR-based registries.

Figure 3 illustrates the conceptual model of the key relationships applied for managing the triple set in XMDR. It also shows an appropriate example for the management of ontological resources. The XMDR defines that *Registered_Item* consists of various classes such as *Concept* (maps to *univ:Professor*, *univ:Student*) and *Relation* (maps to *advise*). Among them, the *Relation* class forms an inheritance relationship with the *Concept* class and serves as a property reference for the *Binary_Relation* class. In addition, to define the relations among concepts and their semantic specifications, the relations between *Concept* classes are defined by means of classes such as *Relation*, *Relation_Role*, *Link*, and *Link_End*. In other words, the *Relation* class expresses the relationships among *Concept* classes, where *Link* class is an association of the *Relation* (i.e., an individual n-tuple, which comprises the corresponding relation) and the *Link_End* class is an association that identifies a particular role that a *Concept* plays in a *Link* and *Relation_Role* (maps to *Subject*, *Predicate*, and *Object*). Thus, *Link* and *Link_End* classes distinguish between names and describe one or more elements of a *Relation*. Furthermore, to define *Classification_Scheme* between concepts taxonomy and partonomy were used, and the ideas of predicate and axiom were used to specify the relations. Therefore, the structure was designed so as to accommodate the ontology technologies. Ontology that constitutes the Semantic Web mainly consists of concept and instance levels. The XMDR was designed considering the concept level alone, ignoring the

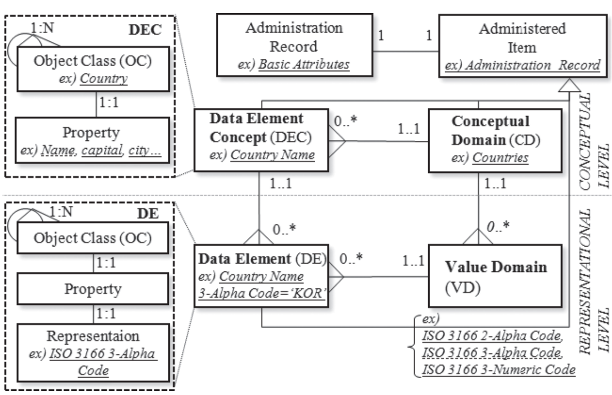

Fig. 1 MDR conceptual model and example.

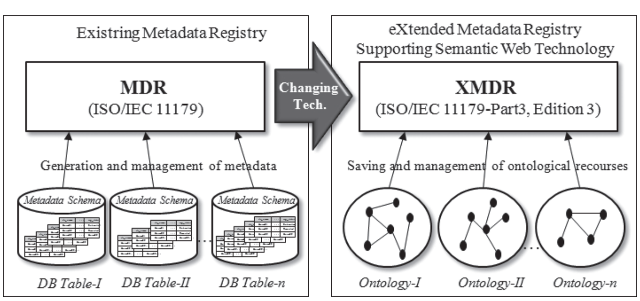

Fig. 2 Storing and management of ontological resources.

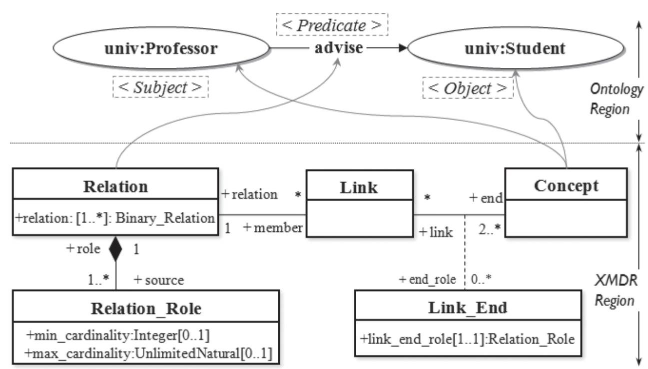

Fig. 3 XMDR conceptual model and example.

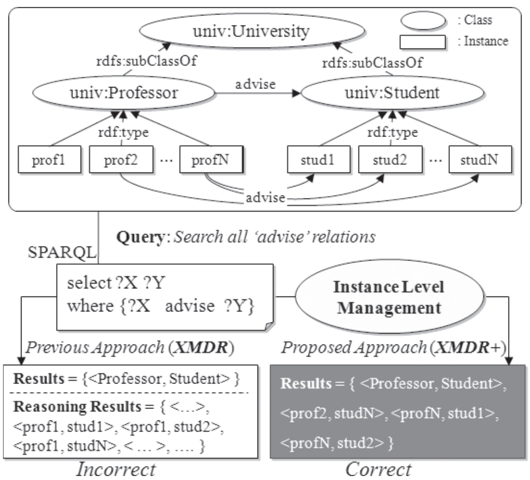

Fig. 4 Example of the XMDR problem.

domain-range relations at the instance level and the relations between properties of instances.

Figure 4 clearly depicts the example of the existing XMDR problem as well as the motivation and aim of our work. In Fig. 4, if the SPARQL query is "*Search all 'advise'*" such as **select ?X ?Y where {?X advise ?Y}**, the existing XMDR leads to a lack result such as {*<Professor, Student>*}. However, inference based on this is incorrect, resulting in the set {*<Professor, Student>*, *<prof1*, *stud1>*, *<prof1*, *stud2>*, *<prof1*, *studN>*, *<prof2*, *stud1>*, *<prof2*, *stud2>*, *<prof2*, *studN>*, *<profN*, *stud1>*, *<profN*, *stud2>*, *<profN*, *studN>*}. An incorrect result is obtained because of the fact that the query does not consider the basic characteristics of ontologies, it fails to guarantee correct answers to queries. The correct results are {*<Professor, Student>*, *<prof2*, *studN>*, *<profN*, *stud1>*, *<profN*, *stud2>*}. In order to address this problem, a new framework has been proposed to manage both the concept level of ontology and the instance level.

3. Proposed Framework

In the previous section, we discussed the inherent problems of the structure of the XMDR and the XMDR itself, which is the extended form of the MDR, during the application of ontology technologies in the Semantic Web environment. To resolve these issues, in this study, the concept of the XMDR has been extended to define a new framework that can manage even the instance level of ontology technologies. Therefore, it is ultimately aimed at guaranteeing interoperability of data among registries.

3.1 XMDR+ Framework

The XMDR+ framework proposed herein sets forth a meta-model that can manage the instance level as well as the concept level and further presents a mapping model for ontology registration. To manage the concept level, the concept model defined in the existing XMDR is first used, and then,

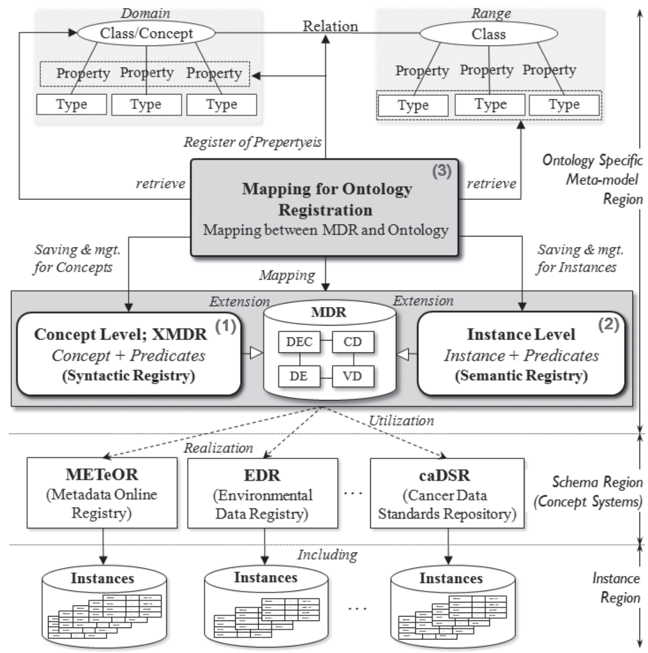

Fig. 5 Conceptual framework of XMDR+.

a new model that can manage the instance level is added to it. To define a meta-model incorporating both the concept and instance levels, definitions of the data relations are supported with respect to the classes defined in the existing MDR, relations among the classes, instances of classes, and relations among the instances. Consequently, the data meanings become machine-readable.

Figure 5 shows the conceptual model of the newly proposed XMDR+ framework. It incorporates the basic concepts required for the management of the metadata defined by the existing MDR. It further uses the concept system defined in the XMDR for the registration and management of the ontological schema used in various domains. The *Concept_System* retains the meta-model defined in the XMDR framework. Thus, it should be able to register and manage schemas of the models in various domains. Therefore, *Concept_System* is used. For the abovementioned reasons, the XMDR has been extended to enable it to save and manage the concept level of *Concept_System*-based ontology to produce the XMDR+ framework that can also manage the ontological instance level. The newly proposed XMDR+ framework consists of the following three regions:

- **Instance Region**: In this region, the actual data values are saved. It shows the physical data saved in the database tables. It represents the data corresponding to the instance level of ontology, which becomes the body of the actual values constituting ontology. Therefore, the instance region of the newly proposed framework represents the area where the ontological instances are saved and managed, which have been generated in accordance with the schema design defined in the schema/registry region.
- **Schema Region**: The schema/registry region defines

the meta-model for the design of the registry schema of metadata. The meta-model defined serves as the basis for a model constructed for sharing and exchange of data/metadata of systems/applications of various regions, which in turn are defined based on metadata registries. Notions shown in Fig. 5, such as METeOR, EDR, and caDSR indicate the registries that contain the schemas defined based on metadata registries. The model used for the design of each schema in this study is termed *Concept_System*. Therefore, the registry put to use in this region is based on the existing basic registry model used for the metadata. Consequently, it retains the identical schemas pertaining to semantics, syntax, and representation. Furthermore, it becomes possible, upon application of ontology technologies, to delineate various properties of the schemas that adopt different naming methods. Therefore, the interpretation of semantic relations of the metadata leads to the maintenance of the interoperability of data whenever metadata are generated or added to the registry. In other words, systems/applications are defined on the basis of the meta-model of schemas/registries. Thus, the following components enable machine reading of previously defined data elements: concepts/classes, relations among concepts, and the data types and properties defined within the concepts.

- **Ontology Specific Meta-model Region**: This region refers to the area where the existing metadata registry is extended to save and manage ontology technologies. This region classifies the concept and instance levels of ontology, manages their relations with ontological data, and defines the mapping relations between the registries and ontology. Figure 5-(2) indicates the level at which the management of added instances and their properties is undertaken. It is called the instance level, and in this study, it is termed as semantic registry. Figure 5-(3) illustrates the registration model that defines the procedure for carrying out the mapping of concepts of the metadata registry and ontology. To register the ontological schema used in various fields, it is necessary to map the relations of the metadata registry and the ontological concept. By defining them, it becomes possible to consistently manage mapping relations between registries and ontological concepts and to ensure efficiency of the semantic relations between the concept and the instance level. As a result, more accurate information can be provided. Figure 5-(1) indicates the level where the meta-models are defined in the existing XMDR. This level constitutes the concept level. In short, it is the region for storing and managing ontological concepts, and therefore, it is termed as syntactic registry.

The XMDR+ framework proposed enables, through the aforementioned three regions, semantic consistency of data in each registry, guaranteed interoperability, integration and sharing of data, and provision of knowledge-based services. Therefore, the newly proposed framework guarantees completeness of data by expressing and managing the diverse properties of ontology.

3.2 Design of Meta-Model for Ontology Management

It is necessary to save and manage not only the ontological concept level but also the instance level in order to resolve the semantic differences in the metadata schemas arising between registries and to ensure interoperability of data. The XMDR incorporates the "relations meta-models" for defining the ontological concept level relations [30]. In this subsection, the relations meta-model has been modified and supplemented. Then, the framework has been proposed, which supports the relations between the concept and instance levels and defines the relations among instances retained at the instance level. The higher concept level in Fig. 5-(1) represents the model constructed for defining the relations among the concepts specified in the XMDR. As discussed above, however, the accuracy and reliability of the answers to the user inquiries is not guaranteed unless the instance level is considered. Being aware of this issue, we have added a relation-defining model, as shown in Fig. 5-(2). The new approach helps create a model that is capable of not only defining the inter-concept level relationships but also managing the instance level. Then, as proposed herein, a new framework has been created, which improves the efficiency and accuracy of ontological concepts.

To define the relations among concepts at the concept level, three classes are used: *Link*, *Link_End*, and *Relation_Role*. Moreover, three additional classes are used to specify the characteristics of ontology: *Concept_System*, *Ontology*, and *Relation*. Further, seven classes are defined for the instance level shown in Fig. 5-(2) to support the delineation of relations among instances and concept-instance relations at the instance level. As illustrated in Fig. 6, instance relations are defined by means of *Instance_Property*, *Instance_Link_End*, *Instance*, *Instance_Relation*, and *Instance_Relation_Role* to further define inter-instance relations. In this manner, it becomes possible to save and manage the relations among instances. Then, the ID values of concepts and instances are set forth through the former, and those of *Concept_Instance_Link* and *ConceptProperty_InstanceProperty_Link* are set forth through the latter. Finally, it becomes possible to define the concept-instance relations. As discussed above and shown in Fig. 6, by additionally defining the concept-instance level relations and inter-instance relations and by using the notions that are different from the concept and instance levels, it becomes possible to save and manage all of the ontological characteristics, generate accurate responses to user inquiries, and eventually define the framework that is capable of storing and managing the resources of the ontological concepts, which guarantees reliability.

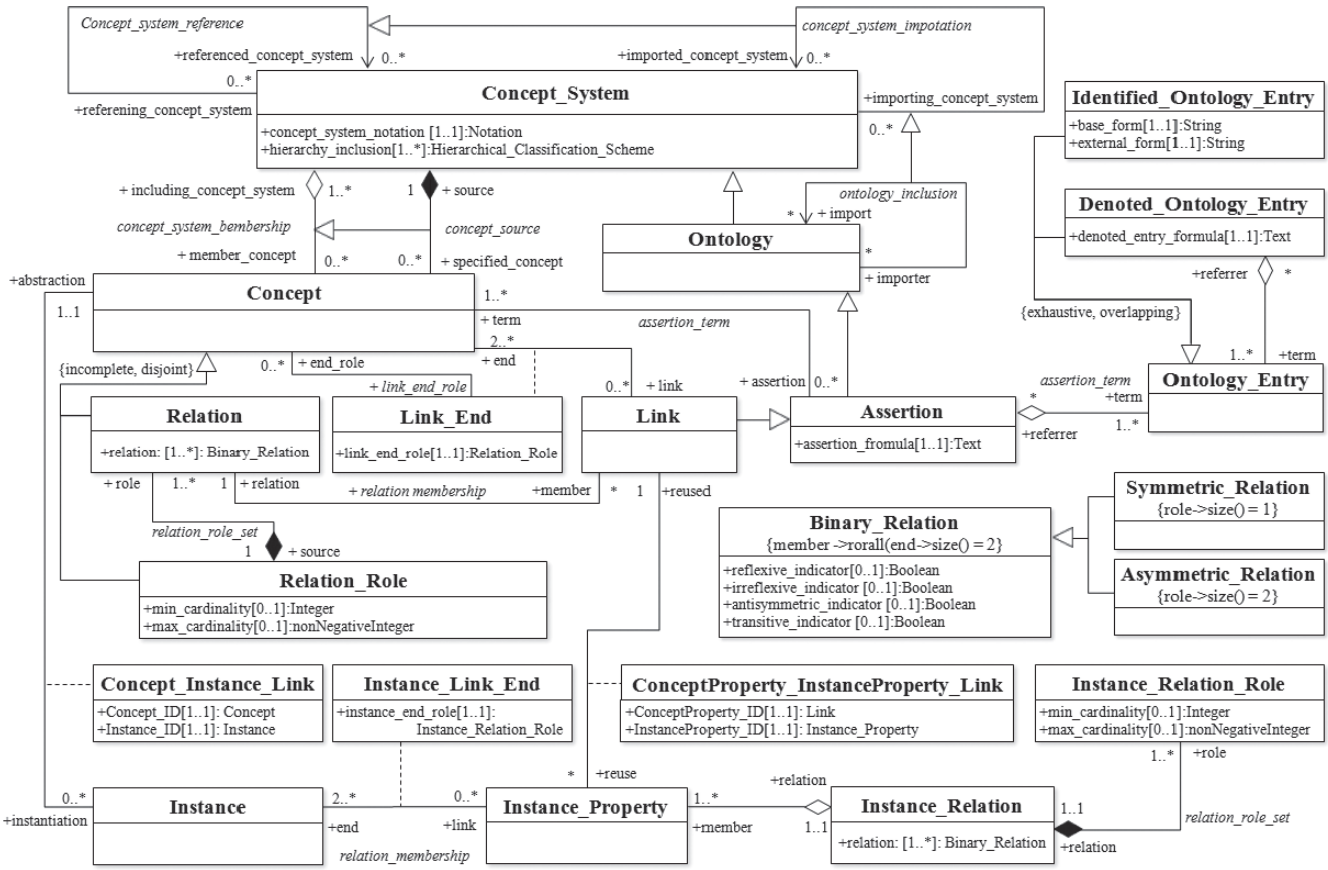

Fig. 6 Ontology-based meta-model for management of metadata registry.

3.3 Mapping between Metadata Registry and Ontology

The *Concept_System* meta-model is generic enough to also support registration of such built-in constructs in an OWL concept system (e.g., an OWL concept system can be used to define OWL ontological relations such as *rdf:type*, *rdfs:range*, and *owl:disjointWith*). Thus, the OWL built-in constructs are most naturally described as *Ternary* relations and others as variable *Arity* relations with two roles. It is also reasonable to describe *ObjectProperties* as relations rather than relation roles. These constructs are reused from RDF-S, which in turn is defined on top of RDF, and most of the OWL datatypes are taken from XML schema. Thus, the OWL built-in constructs are naturally described as *Ternary* relations and the others as variable *Arity* relations having two roles. It is also reasonable to describe *ObjectProperties* as relations rather than relation roles. These constructs are reused from RDF-S, which in turn is defined on RDF, and most of the OWL data types are taken from the XML schema.

Many other OWL built-in constructs do not have corresponding elements built into the *Concept_System* meta-model, as summarized in Table 1. Table 1 illustrates the description type of the metadata registry, which is semantically identical to *Conceptual_Domain* (*CD*), *Reference_Concept* (*RC*), *Value_Meaning* (*VM*), *Data_Element_Concept* (*DEC*), *Object_Class* (*OC*), *Unit_of_Measure* (*UM*), *Relation* (*R*), and *Characteristic* (*C*) of ontological concepts (i.e., classes). *DEC* and *Property* (*P*) are identical to the fields

Table 1 Mapping for OWL built-in constructs.

OWL (Ontology) constructs type	Metadata Registry description type
Ontology	*Concept_System* (*C*, *RC*, *VM*, *CD*, *OC*, *P*, *R*, *UM*, *VD*, *DE*, *DEC*)
imports	importation
minCardinality, max Cardinality, cardinality, FunctionalProperty, InverseFunctionalProperty	multiplicity
TransitiveProperty	transitivity
metaclasse Class, ObjectProperty, DatatypeProperty, Property	*Concept* (*P*, *C*, *RC*, *VM*, *CD*, *OC*, *UM*, *CEC*)
classes Thing and Noting	*Concept* (*C*, *RC*, *VM*, *CD*, *OC*, *UM*, *CEC*)
datatypes	*Concept* (*VM*, *DE*)
equivalentClass, equivalentProperty, sameAs	*Binary_Relation* (symmetric, transitive)
differentFrom, complementOf, disjointWith	*Binary_Relation* (symmetric, intransitive)
subClassOf, subPropertyOf	*Binary_Relation* (asymmetric, transitive)
type, range	*Binary_Relation* (asymmetric, intransitive)
AllDifferent, distinctMembers	*Relation* (variable Arity, 1 role)
intersectionOf, oneOf, unionOf	*Relation* (variable Arity, 2 role)
allValuesFrom, sameValuesFrom, hasValue	*Relation* (Arity=3, 3 role)

where the data types of a database are not defined, namely, the generalized and abstract properties. Therefore, they are mapped onto the ontological properties.

4. Experiment and Evaluation

4.1 Experiment

The experiment performed in this study uses a data set created by the University-Bench Artificial Data Generator (UBA), an ontology creation tool provided by Lehigh University [31]. The UBA is an ontology generation tool, which has been developed to evaluate the results of the SWAT project [32]. UBA creates data in LUBM(n, s), where n and s are the number of universities and the seed value, respectively (such as LUBM(1, 0), LUBM(5, 0), LUBM(10, 0), LUBM(20, 0), and LUBM(50, 0)). LUBM(1, 0) represents the ontology of one university, containing information such as departments, professors, colleges, and graduate schools. In this experiment, we have used the LUBM(1, 0) and LUBM(5, 0) sets; because the other three LUBM sets were too large, they were excluded from the experiments (LUBM(10, 0), LUBM(20, 0), and LUBM(50, 0). Nevertheless, two ontology data sets (LUBM(1, 0) and LUBM(5, 0)) are adequate in size (number of statements are 103, 074, and 645,649) to be used in our experiment because the objective of this research is simply to examine whether our proposed framework can manage ontology at the instance level. In other words, the proposed framework exhibits a performance that would ultimately provide accurate information such as the delineation of the instance-managing model for the instance level and specification of the management model that incorporates concept and instance levels; a good performance of the framework would ensure efficient semantic relations.

We performed an experiment under the following system environment: CPU: Intel Core Duo (2.53 GHz), Memory Size: 3 GB, Heap Memory Size: 2 GB, Hard Disk Size: 300 GB, Platform: Windows Server OS, Language: Java (JDK 1.6.0), DBMS: Oracle 9i Enterprise Edition Release 9.2.0.1.0.

In Fig. 7, the XMDR+ framework designed by using Protégé is shown [33], [34]. Table 2 lists the number of ontological meta-schema metrics such as classes, properties,

Table 2 Number of ontological meta-schema metrics.

Ontology Metrics	Numbers	Ontology Sub-metrics	Numbers
Class	83	Subclass	213
Object property	141	Inverse object properties	64
		Functional object properties	65
		Inverse functional object property	55
		Object property domain	141
		Object property range	141
Data property	85	Functional data property	79
		Data property domain	85
		Data property range	85
		Entity annotation	155

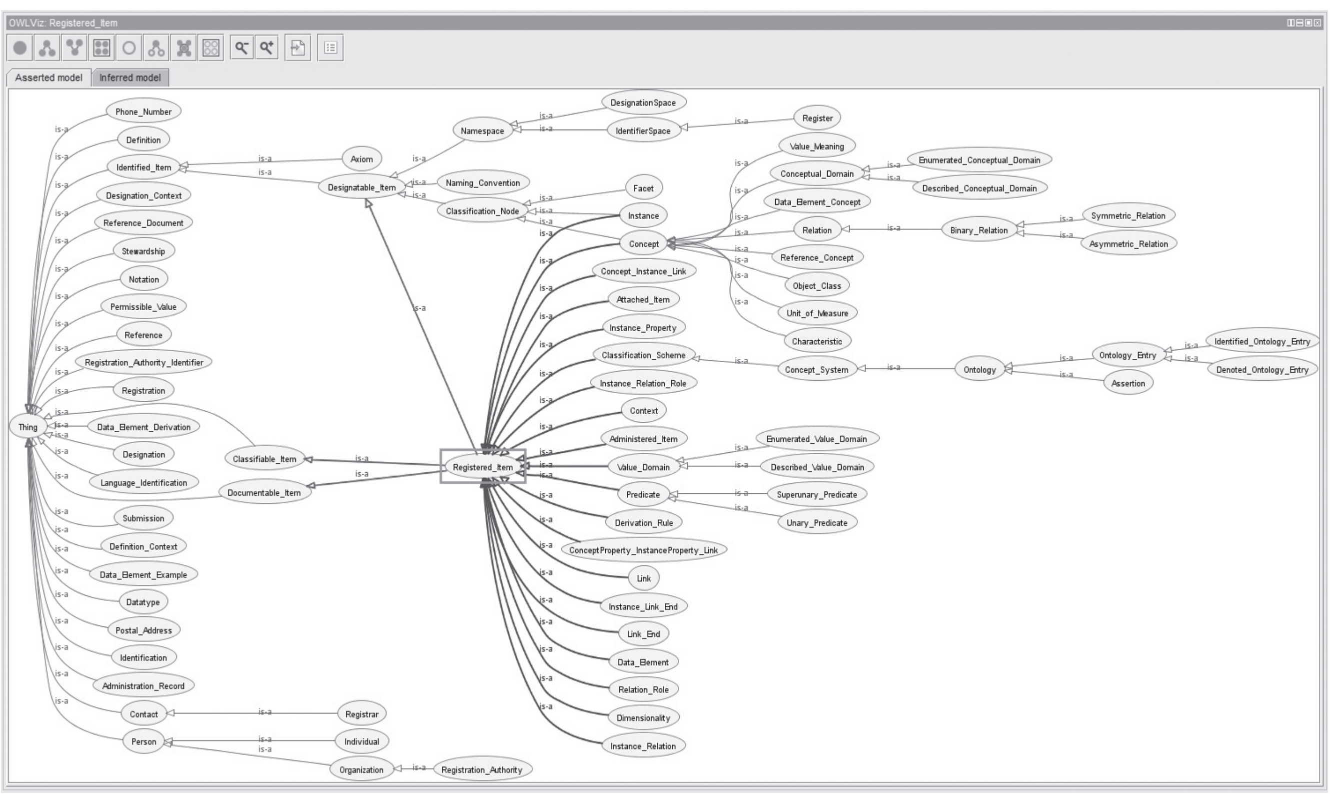

Fig. 7 Design of XMDR+framework by protégé.

and their instances.

The query patterns are defined considering the following factors: simple target search (classes, instances), hierarchy (class and property hierarchy), and instance search for one or more classes. According to these considerations, the defined query patterns are as listed in Table 3. Table 4 lists seven results obtained for the query patterns. Q-P 1, Q-P 4, and Q-P 5 are the queries concerning definitions and relations of the ontological concept level alone. They generate the same answers in both the XMDR+ and the XMDR. The queries concerning Q-P 2, Q-P 3, Q-P 6, and Q-P 7 are the search results on the instances and their hierarchies at the instance level. Therefore, the XMDR+ shows the same results as those produced manually, whereas the XMDR fails to produce the correct answers to queries. As per the previous discussion, it is not possible to ensure completeness and accuracy of the answers to the queries when the relations among instances and among properties are ignored not only at the ontological concept level but also at the instance level. In the absence of results upon questioning or in the event of data loss, there is no reliability, however excellent the registry may be. Hence, to produce reliable answers to the queries, it is essential to provide definitions of the relations of instances and those of their properties at the instance level. As a result, the problems of the XMDR are resolved through comparison of the accuracies of the queries of the XMDR and the XMDR+. On comparison, it is revealed that the XMDR+ produces better answers.

Table 3 Query patterns.

Query Patterns	Description of Queries
Q-P1	*Search for Class:* select ?X where { ?X ?Y Uv::http://www.w3.org/2002/07/owl#Class }
Q-P2	*Search for Class-Instance:* select ?X where { ?X Uv::http://www.w3.org/1999/02/22-rdf-syntax-ns#type Uv::http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#UndergraduateStudent }
Q-P3	*Search for Instance-Instances:* select ?X where { ?Y Uv::http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#emailAddress ?X. ?Y Uv::http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#advisor Uv::http://www.Department0.University0.edu/AssistantProfessor7 }
Q-P4	*Search for Class-Hierarchical:* select ?X where {?X Uv::http://www.w3.org/2000/01/rdf-schema#subClassOf Uv::http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#Professor}
Q-P5	*Search for Property- Hierarchical:* select ?X where {?X Uv::http://www.w3.org/2000/01/rdf-schema#subPropertyOf Uv::http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl#degreeFrom }
Q-P6	*Search for Property-Hierarchical-Instance:* select ?X ?Y where { ?Y Uv::http://www.w3.org/2000/01/rdf-schema#subPropertyOf Uv::http://www.lehigh.edu/zhp2/2004/0401/univ-bench.owl#degreeFrom. ?X ?Y Uv::http://www.University241.edu }
Q-P7	*Search for Class -Hierarchical-Instance:* select ?Y where {?X Uv::http://www.w3.org/2000/01/rdf-schema#subClassOf Uv::http://www.lehigh.edu/~zhp2/2004/0401/univ-bench.owl #Professor. ?Y Uv::http://www.w3.org/1999/02/22-rdf-syntax-ns#type ?X }

4.2 Performance Comparison among Registries

In this section, the metadata registries that serve as the base technology for various fields have been qualitatively compared without being bound by particular area types. As discussed above, DCMI [14], J. metadata [15], MPEG-7 [16], and ebXML RIM [17] are the metadata registries that are specialized for particular domains. Because of their nature, they are inappropriate for comparison with the registries developed in our study. Therefore, the existing MDR and the XMDR are compared with the newly proposed XMDR+ to assess registry performances.

Table 5 sums up the results of the qualitative performance assessment of the three registry types (i.e., MDR, XMDR, and XMDR+). Ten criteria were used for the evaluation.

As shown in Table 5, the MDR focuses on registration and management of metadata information without ontological relations. The MDR performs well in terms of *Accuracy of Queries* but fails to accommodate various characteristics of ontological concepts because the current version of the MDR was developed in a backdrop without Semantic Web technologies. Therefore, the MDR does not support *Instance Management*, *Relations among Instances*, *Class-Instance Association*, and *Mapping Method* between ontological concepts and metadata registry. Moreover, it is infe-

Table 4 Comparison of experiment query results.

Query patterns	Results of Queries					
	LUBM (1, 0)			LUBM (5, 0)		
	Manually	XMDR (Previous approach)	XMDR+ (Proposed approach)	Manually	XMDR (Previous approach)	XMDR+ (Proposed approach)
Q-P1	44 Classes	43 Classes	44 Classes	43 Classes	43 Classes	43 Classes
Q-P 2	5916 Instances	-	5916 Instances	36682 Instances	-	36682 Instances
Q-P 3	6 Instances	-	6 Instances	6 Instances	-	6 Instances
Q-P 4	6 Classes	6 Classes	6 Classes	6 Classes	6 Classes	6 Classes
Q-P 5	3 Properties	3 Properties	3 Properties	3 Properties	3 Properties	3 Properties
Q-P 6	14 Properties	-	14 Properties	46 Properties	-	46 Properties
Q-P 7	447 Instances	-	447 Instances	2811 Instances	-	2811 Instances

Table 5 Qualitative comparison.

Registries / Comparative Items	MDR (Previous approach)	XMDR (Previous approach)	XMDR+ (Proposed approach)
Class Management	Partially supports	Supports	Supports
Instance Management	Does not support	Does not support	Supports
Relations among Classes	Partially supports	Supports	Supports
Relations among Instances	Does not support	Does not support	Supports
Class-Instance Association	Does not support	Does not support	Supports
Mapping Method	Does not support	Does not support	Supports
Semantic Integration Capacity	Low	Medium	High
Accuracy of Queries	High	Low	High
Complexity	Low	Medium	High
Management Cost	Low	Medium	High

rior to the proposed model in terms of the functions of *Class Management* and *Relations among Classes*. However, it is better than in terms of the other registries with respect to *Complexity* and *Management Cost*.

The XMDR mainly supports the registration and management of syntactic (partially semantic) information. However, it fails to support *Instance Management*, *Relations among Instances*, and *Class-Instance Association* because it only deals with the concept level of ontologies. Furthermore, to register such ontological schema, it is required to consistently save and manage ontologies by defining the mapping method for registries and ontological concepts. However, the XMDR fails to address the mapping issue when the characteristics of metadata registries and ontological concepts are considered. Therefore, it is inferior to the proposed XMDR+ framework with respect to *Semantic Integration Capability*, *Accuracy of Queries*, and *Mapping Method* of answers.

The proposed XMDR+ framework, which is designed to manage the concept level as well as the instance level, supports *Class Management*, *Instance Management*, *Relations among Classes*, *Relations among Instances*, *Class-Instance Association*, and *Mapping Method*. It further guarantees totality of inquiries. In other words, it ensures that inquiries are searched in the target items. Thus, it is superior to the other registries in terms of *Semantic Integration Capacity* and *Accuracy of Queries* of inquiries. To define a meta-model incorporating both the concept and instance levels, the definitions of data relations are supported with respect to the classes defined, inter-class relations, instances of classes, and inter-instance relations. Consequently, data meanings become machine-readable. Despite the excellent functions of the proposed framework, it has a more complex structure than the MDR and XMDR, raising concerns of *Complexity* and *Management Cost*, i.e., it costs more because of its complex structure. In this study, however, the focus is on a framework that is capable of ensuring the validity of significant relations among data and providing accurate information.

5. Conclusion and Future Work

Current registries such as MDR and XMDR do not support registration of metadata of ontological resources. Therefore, the current metadata registries need a more practically extended model and a methodology that ensures ontology registration for management of relations among instances for ontology technologies. This paper presented the XMDR+ framework; we also attempted to design a meta-model that manages not only concepts but also instances; furthermore, we defined a method for mapping relations between the metadata registry and ontological techniques. Consequently, the XMDR+ framework provides support for the metadata registry not only at the concept level but also at the instance level, supports the encoding registry, ensures interoperability between different metadata sets using various ontology properties, and facilitates flexibility of the registry.

Next, we plan to enhance the registration procedures for the mapping between ontological concepts and metadata registry in detail. Furthermore, we plan to improvise the XMDR+ to realize semantic interoperation in concrete domains such as e-government and e-business.

Acknowledgments

This work was supported by the Second Brain Korea 21 Project and the basic Science Research Program through the National Research Foundation of Korea (NRF), funded by the Ministry of Education, Science and Technology (2010-0013781). The authors would like to thank the anonymous reviewers for their constructive comments and suggestions that helped us to improve our work.

References

[1] P.P. Micheal, T. Paolo, and L. Frank, "Service-oriented computing: State of the art and research challenges," Proc. IEEE Computer Society, pp.64–71, 2007.
[2] T. Berners-Lee, J. Hendler, and O. Lassila, "The semantic Web," Scientific American, vol.284, no.5, pp.34–43, May 2001.
[3] N. Shadbolt, W. Hall, and T. Berners-Lee, "The semantic Web revisited," IEEE Computer Society, IEEE Intelligent Systems, vol.21, no.3, pp.96–101, May/June 2006.
[4] H. Sato and A. Kanai, "A technical survey of data integration and information sharing using knowledge representation on the Web," IEICE Trans. Inf. & Syst., vol.E87-D, no.11, pp.2435–2445, Nov. 2004.
[5] H. Lan, F.P.S. Peter, B. Harold, T. Said, G. Benjamin, and D. Mike, "SWRL: A semantic Web rule language combining OWL and ruleML," World Wide Web Consortium (W3C), http://www.w3.org/Submission/SWRL/, 2004.
[6] HP Lab., "Jena semantic Web famework for Java," http://jena.sourceforge.net/, 2008.
[7] J. Broekstra, A. Kampman, and F.v. Harmelen, "Sesame: A generic architecture for storing and querying RDF and RDF schema," Springer-Verlag, Lecture Notes in Computer Science, vol.2342, pp.54–68, June 2002.
[8] OWL Web Ontology Language Overview, W3C, 2004.
[9] RDF Vocabulary Description Language 1.0: RDF Schema, 2004.

[10] M. Okabe, A. Yoshioka, K. Kobayashi, and T. Yamaguchi, "Organizational knowledge transfer using ontologies and a rule-based system," IEICE Trans. Inf. & Syst., vol.E93-D, no.4, pp.763–773, April 2010.
[11] A.Y. Cristiane, T.P.S. Marilde, and T.P.V. Marina, "Ontologuu-based meta-model for storage and retrieval of software components," Proc. VLDB Conference, 2005.
[12] J.D. Kim, D. Jeong, Y. Jing, and D. Baik, "UbiMDR: Supporting application-independent semantic interoperability for ubiquitous applications," International Conference on Computational Intelligence and Security, pp.1155–1158, Nov. 2006.
[13] ISO/IEC JTC 1/SC 32, ISO/IEC 11179: Information Technology-Metadata Registries (MDR)-Part 1–6, 2004.
[14] B. David, P.C.K. Hung, and M.B. Menachem, "Web metadata standards: Observations and prescriptions," Proc. IEEE Software, 2005.
[15] Dublin Core Metadata Initiative (DOMI). http://www.dublincore.org/, 2009.
[16] International Telecommunication Union (ITU), Metadata on cable networks, http://www.itu.int/, 2002.
[17] J.M. Martinez, MPEG-7 Overview, ISO/IEC/JTC1/SC29/WG11, http://www.hiariglione.org/MPEG/standards/mpeg-7/, July 2002.
[18] OASIS, ebXML Registry Information Model (RIM), http://www.oasisopen.org/specs/index.php/, 2008.
[19] Universal Description, Discovery, and Integration (UDDI), http://www.uddi.xml.org/, 2008.
[20] ISO/IEC JTC 1/SC 32, ISO/IEC 20944: Information Technology-Metadata Interoperability and Bindings (MDIB), http://metadata-stds.org/20944/index.html, 2008.
[21] ISO/IEC JTC 1/SC 32, ISO/IEC 20943, Information Technology-Procedures for Achieving Metadata Registry Content Consistency, http://metadata-stds.org/20943/index.html, 2003.
[22] U.S. Department of Transportation, U.S. Intelligent Transportation System (ITS), http://www.dot.gov/, 2008.
[23] Australian Institute of Health and Welfare, Metadata Online Registry (METeOR), http://meteor.aihw.gov.au/, 2009.
[24] US National Cancer Institute, Cancer Data Standards Repository (caDSR), http://ncicb.nci.nih.gov/NCICB/infrastructure/cacore_overview/cadsr, 2009.
[25] Australian National Health Data Committee, National Health Data Dictionary, http://www.aihw.gov.au/, 2007.
[26] Canadian Institute for Health Information, CIHI Data Dictionary, http://secure.cihi.ca/ddexternal/welcome.do, 2007.
[27] UK Cancer Grid, Cancer Grid Metadata Registry, https://129.67.148.193:8443/exist/rest/db/mdr/web/homepage.xquery, 2009.
[28] US Health organizations (multiple), US Health Information Knowledgebase (USHIK), http://ushik.ahrq.gov, 2009.
[29] Environmental Protection Agency (EPA), Environmental Data Registry (EDR), http://www.epa.gov/edr/, 2008.
[30] XMDR Project, Lawrence Berkeley National Laboratory, http://www.xmdr.org/, 2009.
[31] Y. Guo, Z. Pan, and J. Heflin, "LUBM: A benchmark for OWL knowledge base systems," J. Web Semantics, vol.3, no.2, pp.158–182, 2005.
[32] SWAT Projects- the Lehigh University Benchmark (LUBM), http://swat.cse.lehigh.edu /projects/lubm/, 2007.
[33] The Protégé-OWL 4.0, RDF/OWL editor, 2009. URL: http://protege.stanford.edu/overview/protege-owl.html, 2010.
[34] OWL Visualization (OWLViz), http://www.co-ode.org/downloads/owlviz/, 2009.

Jeong-Dong Kim received his M.S. degree from Korea University in 2008, focusing on computer science and in Korea. Since 2008, he has a Ph.D. student in the department of computer science and engineering, Korea University, Seoul, Korea. His research interests include metadata-based integration, Semantic Web, ontology engineering, social computing and access control.

Jiseong Son received her M.S. degree in Computer Engineering from Korea University in 2009. She is currently a Ph.D. student in Korea University in Seoul, Korea. Her research interests include Semantic Web data management, relational databases, query translation, and ontology engineering and access control.

Doo-Kwon Baik received his B.S. degree in Mathematics from Korea University, Seoul, Korea, in 1974. He received his M.S. and Ph.D. degrees in Computer Science from Wayne State University U.S.A., in 1983 and 1986, respectively. He is a full professor in the department of computer science and engineering, Korea University, Seoul, Korea. His research interests include modeling, simulation, data and software engineering.