

Research Article

Canonical Sensor Ontology Builder Based on ISO/IEC 11179 for Sensor Network Environments: A Standardized Approach

Sukhoon Lee,¹ Dongwon Jeong,² Jangwon Gim,³ and Doo-Kwon Baik⁴

¹ Department of Computer and Radio Communications Engineering, Korea University, Anam-dong 1, 5-ga, Seongbuk-gu, Seoul 136-701, Republic of Korea

² Department of Statistics and Computer Science, Kunsan National University, 558 Daehangno, Gunsan, Jeollabuk-do 573-701, Republic of Korea

³ Department of Computer Intelligence Research, Korea Institute of Science and Technology Information (KISTI), 245 Daehak-ro, Yuseong-gu, Daejeon 305-806, Republic of Korea

⁴ Graduate School of Convergence IT, Korea University, Anam-dong 1, 5-ga, Seongbuk-gu, Seoul 136-701, Republic of Korea

Correspondence should be addressed to Doo-Kwon Baik; baikdk@korea.ac.kr

Received 19 November 2013; Accepted 11 February 2014; Published 30 March 2014

Academic Editor: Hwa-Young Jeong

Copyright © 2014 Sukhoon Lee et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

The advancement of sensor technology has led to an explosive increase in sensors. It causes semantic heterogeneity problems, and much research has focused on sensor ontology building to solve the problems. However, there are still remaining several issues, and one of the most critical issues is about a method for progressive and dynamic concepts management and reuse of sensor ontology. This paper proposes an ontology generation system based on ISO/IEC 11179-MDR (metadata registry). The proposed system is referred to as the Canonical Sensor Ontology Builder (CaSOB) and can create ontologies by reusing the common concepts registered in a canonical sensor ontology concept registry, an MDR. This paper defines a mapping model and processes to create ontology with the concepts registered in an MDR. Our proposal provides many advantages such as high standardization, consistent concept usage, and easy semantic exchange. Therefore, CaSOB facilitates the high quality sensor ontology creation and reduces the costs of sensor ontology integration and system development.

1. Introduction

With the advancement of sensor technology, the number of sensors and application domains have been tremendously increased. It leads to a huge amount of sensor data as well as many kinds of sensor types in sensor network worlds. However, those increments cause various kinds of heterogeneity problems such as heterogeneity between data types, formats, and units of measure. Although we should be able to interpret and use all sensor information from any sensors in any sensor networks, those heterogeneity issues make it difficult to use seamlessly and transparently sensor data from all sensor networks.

Many studies have been done to resolve those issues and this section describes only the OGC (Open Geospatial Consortium) effort, one of the representative approaches. OGC established SWE (Sensor Web Enablement) to store,

discover, access, and use all types of sensor information on the web [1]. Thus, the various sensor information defined according to SWE enables representing syntactic model consistently. Also, for representing semantics in the syntactic model, SSW (Semantic Sensor Web) appeared with extending the Semantic Web concept [2]. SSW represents the semantic sensor information such as space, time, and theme as a web ontology.

Many sensor ontologies have been developed to represent the sensor information in various domains [3–6]. The sensor ontologies are used as schemas of sensor data in a specific domain. Therefore, some sensor network systems which use the same sensor ontology in a specific domain guarantee syntactic model, semantic data interchange, and interoperability.

However, if we need to establish a new sensor network for a new domain or having different purposes, it is impossible to directly adopt the existing sensor ontology and we need

to modify the existing ontology or add a new ontology. The modification and addition of ontologies cause another heterogeneity problem at conceptual (ontology schema) level [7, 8]. In other words, when we build a new ontology or reuse existing ontologies for adoption in other domains or for other purposes, we would customize the existing sensor ontologies [9]. It causes another semantic inconsistency between sensor ontologies.

This paper presents a method for sensor ontology management and reuse based ISO/IEC 11179, Metadata Registry (MDR) 3rd Edition, for resolving the semantic heterogeneity issue aforementioned. MDR is an international standard which has been revised for ontology management and published in 2013. A metadata registry based on MDR can manage metadata and the relations among them [10–15]. Ontologies also are registered and managed in the metadata registry. An MDR instance manages common concepts from the registered ontologies, and thus it ensures that ontologies are defined well and reusable by diverse systems. Reusing common concepts for building ontology enhances the ontology quality and semantic interoperability between sensor networks. MDR provides only the structure of registration and the administration of an ontology, but it does not support the ontology creation facility for reusing common concepts. Therefore, this paper proposes a new framework for ontology management and building based on MDR, and the proposed framework is named CaSOB (Canonical Sensor Ontology Builder). In addition, this paper considers a triple statement as an ontology. The reason is that various ontology languages such as OWL and RDF are formed by the triple and most of the ontology stores and reasoning engines are based on the triple.

This paper is organized as follows. In Section 2, we introduce previous researches into solutions to the semantic heterogeneity problem of sensor ontologies. In Section 3, we explain the CaSOB framework, a mapping model, and ontology building processes. Our implementation is described in Section 4, while Section 5 presents the evaluation. Finally, we conclude this paper and discuss future work in Section 6.

2. Semantic Heterogeneity Problem of Sensor Ontologies

Much research on ontologies has been directed at increasing interoperability and solving the semantic heterogeneity problem among various systems. An ontology matching technique was proposed for semantic interchange among peers in Open Network Systems [16]. Castano et al. [17] describe an algorithm that matched ontologies using a linguistic and contextual matching model for Open Network Systems. Pirro et al. [18] propose a similarity checking method based on WordNet for building semantic links in a peer-to-peer network. Bakillah and Mostafavi [19] also propose an ontology matching method based on lexical relations and mapping rules for improving the interoperability of distributed geospatial web services.

Ontology matching methods ensure that ontology concepts connect together and correspond semantically after

ontologies are generated. These methods can match numerous ontologies automatically and semiautomatically, but they can experience problems in terms of high cost and inaccuracy. Therefore, research on ontology sharing and reuse is required, because these techniques can solve the semantic heterogeneity problem by creating ontologies based on common concepts. Matuszek et al. [20] propose a system for defining and reusing the common concepts, while Biebow and Szulman [21] proposed ontology building methods based on terminology from a linguistic perspective.

In the Semantic Sensor Web, various sensor ontologies have been developed such as CSIRO [22], MMI [23], and ISTAR [24]. Each of the ontologies is suitable for representing information of sensors and devices and exchanging the information between relevant services only in the same sensor network environment. However, the ontology cannot be available for the environment where we would develop a new sensor network system for other application domains or with other purposes.

Compton et al. [25] introduces various sensor ontologies from researches and projects in various domains and compares the corresponding domain, purposes, and features of each of the ontologies. Especially, Compton et al. evaluates the expressive power of the sensor ontologies from the perspectives of sensor device information, physical properties, observation quality, and domain concepts.

The CSIRO sensor ontology [26] is designed for a generic domain and is intended to be used in data integration, search, classification, and workflows. The CSIRO sensor ontology provides the high expressive power in sensor, physicalness, and observation perspectives. However, the CSIRO ontology does not provide concepts like sampled medium and time.

MMI (Marine Metadata Interoperability) [27] aims to increase interoperability for oceanographic devices, sensor, and sampler. The MMI device ontology has been developed for devices and sensors in ocean environment, but irrespective concepts such as sensor hierarchy, history, location, power supply, a field of view, unit of measure, and time have not been defined for oceanographic devices, sensors, and samplers.

The ISTAR ontology [28] has the purpose for task assignment and has been developed as part of a system to automatically select sensors for tasks based on their fitness for the task description. However, ISTAR has only concepts about sensors and physical features for the task assignment and does not consider concepts about observation and domain.

As a result, the previous approaches related to the sensor ontology do not provide a canonical sensor ontology concept management and building facility, and thus an additional heterogeneity issue continuously arises. It causes many other problems and is a barrier for seamless sensor semantics interpretation, progressive and systematic semantics management, common concept reusability, and so on. For addressing those problems, this paper proposes a new sensor ontology management and building framework based on MDR. The goal of the framework proposed in this paper is to enable the canonical management and reuse of the existing well-defined ontology.

3. The CaSOB Framework

Figure 1 shows the CaSOB framework. CaSOB contains four processes (Section 3.2) and it creates an ontology using a mapping model from MDR (Section 3.1). Various concepts are registered in a metadata registry for sharing and reusing as common concepts. Therefore the mapping model is defined using components in regions related by common concepts and data representations, including concepts region and data description region, from several areas of MDR. As a result, ontology is built as an ontology according to ontology schema model, so this paper considers only the schema level of ontology and it excludes the instance level.

3.1. Mapping Model

3.1.1. Preliminary Definition. MDR supports the registration and administration of common concepts. And it specifies a metamodel for the registration and administration of common concepts as an MDR specification. There are seven main regions. These regions are classified as regions that register and administer concepts, regions that name and define concepts, regions that represent the relations among concepts and components, and regions that specifically describe data. In this paper, we formally define the MDR model for two regions, that is, the concepts region and data description region.

Definition 1 (MDR concepts model). The MDR concepts model is a six-tuple

$$\text{MDR}_C = (CS, C, R, RR, LE, link), \quad (1)$$

where one has the following.

- (i) CS is a set of concept systems. A concept system represents a domain of concepts such as an application domain or knowledge domain.
- (ii) C is a set of concepts. A concept is a unit of knowledge created by a unique combination of characteristics. Some tuples of the MDR concepts model and the MDR data description model inherit C as a basic unit.
- (iii) R is a set of relations. A relation is a concept that represents the association among concepts.
- (iv) RR is a set of relation roles. A relation role is a role of a concept associated with a specific relation.
- (v) LE is a set of link ends. A link end is a pair of a concept and a relation role as one side of $link$, and LE is expressed by $\langle c \text{ and } rr \rangle$ where $c \in C$ and $rr \in RR$.
- (vi) $link: R \rightarrow LE \times LE$ represents the linkage of two link ends, which are associated with a relation. In this paper, one focuses generation of a triple statement as ontology, so a binary relation is only considered.

The MDR concepts model defines all the concepts and relations administered in MDR. As an example, Figure 2 shows that the MDR concepts model contains $link_{\text{“class-subsumption”}}$ ($\langle \text{“Sensor”, “superclassOf”} \rangle$, $\langle \text{“System”, “subclassOf”} \rangle$), where $\{\text{“Sensor”, “System”}\} \in C$, $\text{“class-subsumption”} \in R$, and $\{\text{“superclassOf”, “subclassOf”}\} \in RR$.

Definition 2 (MDR data description model). The MDR data description model is an eight-tuple

$$\text{MDR}_D = (CD, VM, VD, OC, Pr, DEC, DE, Dim), \quad (2)$$

where one has the following.

- (i) CD is a set of conceptual domains. A conceptual domain is a domain wherein data element concepts could have a shared conceptual level. A conceptual domain is also used to represent a set of value meanings (VM).
- (ii) VM is a set of value meanings. A value meaning is a general meaning of actual values.
- (iii) VD is a set of value domains. A value domain is a domain in the actual data level, which represents the actual values such as a data type.
- (iv) OC is a set of object classes. An object class is a general concept that can capture many instances.
- (v) Pr is a set of properties. A property is an attribute of objects or concepts.
- (vi) DEC is a set of data element concepts. A data element concept is a pair of a concept and a data element. And DEC is expressed by $\langle oc, pr \rangle$ where $oc \in OC$ and $pr \in Pr$.
- (vii) $DE \subseteq DEC \times VD$ is a set of data elements. The data element is the basic unit of data for MDR.
- (viii) Dim is a set of dimensionalities. A dimensionality is an aggregation of the units of measures, which is related to a conceptual domain, for example, length, height, or weight.

An ontology contains components that depend on the language and the description method. In this paper, we focused on building ontologies to improve interoperability among various applications or systems, so we concentrated on building ontology in schema level, rather than instance or individual level.

Definition 3 (ontology schema model). The ontology schema model is a six-tuple

$$\text{Onto} = (Cl, P, Rel, DT, relation, prop), \quad (3)$$

where one has the following.

- (i) Cl is a set of classes. A class is a concept that groups items sharing the same characteristics. Common characteristics are described logically.
- (ii) P is a set of properties. A property indicates the characteristics of a class. In the ontology schema model, a property is described by connecting a class to a data type.
- (iii) Rel is a set of relations in an ontology domain. A relation in an ontology domain describes the characteristics existing among classes. A relation is described by connecting one class to another class.

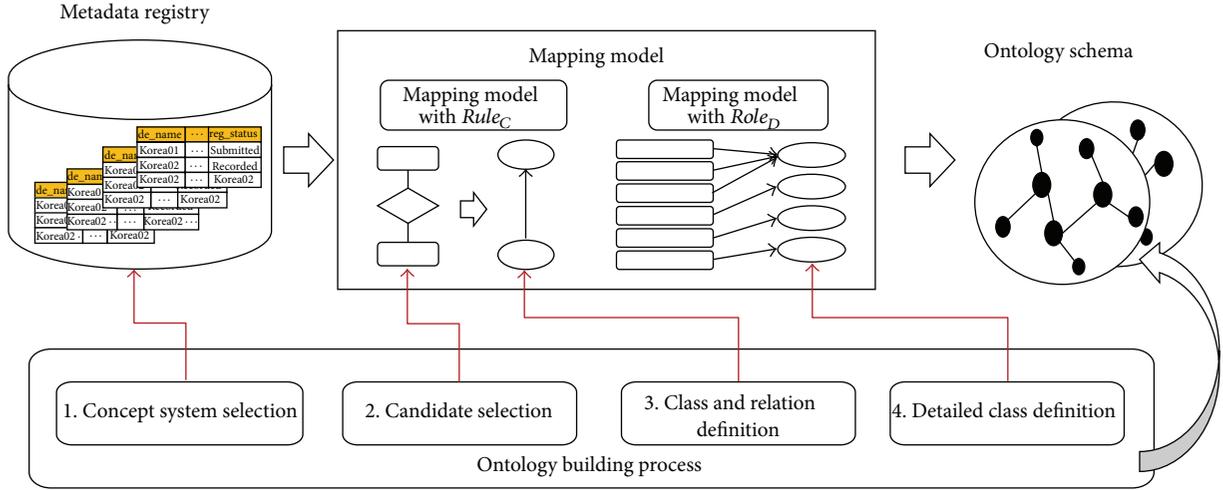


FIGURE 1: The CaSOB framework.

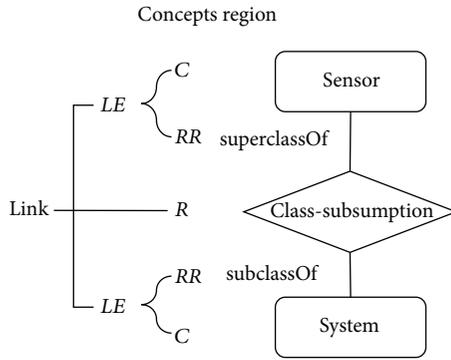


FIGURE 2: An example of the MDR concepts model.

- (iv) DT is a set of data types. A data type is a type of value given to a class property.
- (v) $relation: Rel \rightarrow Cl \times Cl$ is a triple statement expressed by (cl_1, rel, cl_2) , where $\{cl_1, cl_2\} \in Cl$ and $rel \in Rel$, such as an object property of OWL.
- (vi) $prop: P \rightarrow Cl \times DT$ is a triple statement expressed by (cl, p, dt) , where $cl \in Cl$, $p \in P$, and $dt \in DT$, such as a data type property of OWL.

3.1.2. CaSOB Mapping Model. The CaSOB mapping model defines the mapping rules based on the MDR models for the ontology schema model, as shown in the preliminary definition provided in Section 3.1.1. Data mapped by the CaSOB mapping model is used to build an ontology.

Definition 4 (mapping model). The mapping model is a two-tuple

$$MM = (Rule_C, Rule_D), \quad (4)$$

where one has the following.

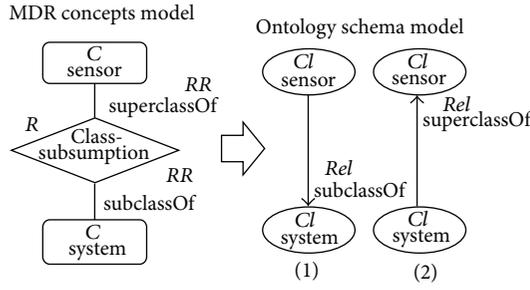
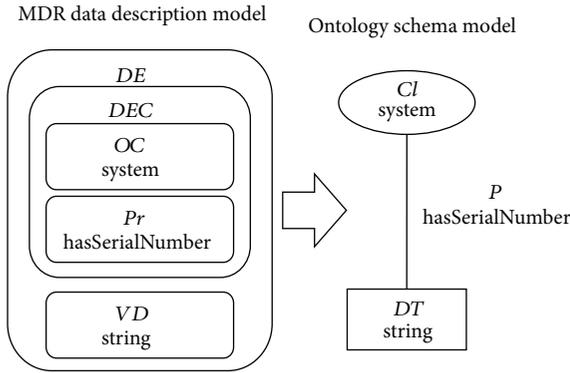
- (i) $Rule_C$ is a set of rules connecting the MDR concepts model with the ontology schema model. $Rule_C$ has three rules as follows:

- (a) $C \rightarrow Cl$,
- (b) $RR \rightarrow Rel$,
- (c) $link \rightarrow relation$; that is, $link_r(\langle c_1, rr_1 \rangle, \langle c_2, rr_2 \rangle) \rightarrow relation_{rr_2}(cl_1, cl_2)$ or $relation_{rr_1}(cl_2, cl_1)$, where $r \rightarrow rel$, $c_1 \rightarrow cl_1$, and $c_2 \rightarrow cl_2$ with $r \in R$, $\{c_1, c_2\} \in C$, $\{rr_1, rr_2\} \in RR$, $rel \in Rel$, and $\{cl_1, cl_2\} \in Cl$.

- (ii) $Rule_D$ is a set of rules connecting the MDR data description model and the ontology schema model. $Rule_D$ has eight rules as follows:

- (a) $CD \rightarrow Cl$,
- (b) $VM \rightarrow Cl$,
- (c) $Dim \rightarrow Cl$,
- (d) $OC \rightarrow Cl$,
- (e) $Pr \rightarrow P$,
- (f) $VD \rightarrow DT$,
- (g) $DEC \rightarrow Cl \times P$,
- (h) $DE \rightarrow prop$; that is, $(\langle oc, ch \rangle, vd) \rightarrow prop_p(cl, dt)$, where $oc \rightarrow cl$, $pr \rightarrow p$, and $vd \rightarrow dt$ with $de \in DE$, $oc \in OC$, $pr \in Pr$, $vd \in VD$, $p \in P$, $cl \in Cl$, and $dt \in DT$.

For $Rule_C$, C is specialized in CD , DE , OC , and so forth, in an MDR data description region, so C is mapped to Cl . Due to the fact that R is an undirected relation and RR has direction, RR is mapped to Rel . And $link$ has the same tuple LE and $relation$ also has the same tuple Cl ; hence, there are two types of results when $link$ is mapped into $relation$, which depends on RR . This is because the ontology schema model expresses the ontology as a triple statement with a direction such as (subject, predicate, object). However, the $link$ is able

FIGURE 3: The expected mapping results with $Rule_C$.FIGURE 4: A mapping example using $Rule_D$.

to map two *relations* by RR . Thus, a user needs to select a subject and an object when CaSOB uses rule (c) in $Rule_C$.

For example, Figure 3 shows two expected mapping results, (1) and (2), where *link* has two *LEs*, that is, $\{< \text{“Sensor”}, \text{“superclassOf”} >, < \text{“System”}, \text{“subclassOf”} >\} \in LE$, whereas the mapping model cannot automatically determine which *LE* becomes a subject or an object in *relation* of the ontology schema model. Thus, a user needs to select the subject and object of *relation*. Because the subclass is typically more used than superclass in ontology language, the user could select (1) as the mapping result, expressed by $relation_{subclassOf}(\text{“Sensor”}, \text{“System”})$.

The MDR data description model specifically describes the meaning of metadata and provides an explanation of the concepts defined in the MDR concepts model. The six tuples of the MDR data description model, that is, CD , DEC , OC , Pr , VM , and Dim , are inherited by C in the MDR concepts model. C is expressed differently depending on the property of the inherited tuples. Thus, each tuple is mapped to Cl , P , or DT , while most of the ontologies mapped by $Rule_D$ are triple statements such as $(Cl, P, \text{and } DT)$. CD , OC , VM , and Dim are only mapped to Cl in the six tuples, because they are entities which are connected by other classes. VD is mapped to DT although VD does not mean data type. However, VD contains a data type inside and DE contains VD , not a data type directly. Also if VD is mapped to DT , we can define some special data types like 2-alpha country code. Pr is mapped to P , because Pr indicates an attribute of classes. DEC contains OC and Pr , so DEC is

mapped to Cl and P both. Similarly DE contains DEC and VD , so DE could be mapped to (Cl, P, DT) statement.

Figure 4 shows a mapping example for $Rule_D$. The MDR data description model expresses DE as “System hasSerialNumber” with a “String” data type, where “System” $\in OC$, “hasSerialNumber” $\in Pr$, $\langle \text{“System”}, \text{“hasSerialNumber”} \rangle \in DEC$, and “String” $\in VD$. As a result, the ontology schema model constructs a $(\text{“System”}, \text{“hasSerialNumber”}, \text{“String”})$ statement, where “System” $\in Cl$, “hasSerialNumber” $\in P$, and “String” $\in DT$.

3.2. Ontology Building Process

3.2.1. Concept System Selection. The first process for building ontology is to decide a domain of an ontology. If domains are different from each other, ontology meanings and relations are different, even if they have the same terminology. A concept system includes the domain and scope of the data. Thus, CaSOB determines the domain of an ontology by selecting a concept system from CS.

In the first process shown in Figure 5, the concept system “CSIRO Sensor Ontology” in CS is selected if a user wants to build an ontology in a sensor domain defined by CSIRO.

3.2.2. Candidate Selection. In the candidate selection process, candidates are selected to build an ontology. Tuples of the MDR concepts model, such as C , R , and *link*, may be selected as candidates. The candidates are the same as the set of terms used when ontologies are created.

C and *link* are selected from CS as candidate C and candidate *link*, respectively. *link* contains C , R , RR , and LE so both candidate C and candidate *link* can be selected from one *link*. The selected *link* becomes candidate *link*. C included in *link* is automatically selected as candidate C .

The concept system “CSIRO Sensor Ontology” is selected during the first step of Figure 5. The next step is to select the terms contained at the domain. Candidates are selected such as “Sensor,” “SensorGrounding,” and “supports.”

3.2.3. Class and Relation Definition. In the third process, Cl and Rel of the ontology schema model are defined by the candidates. A set of candidate concepts is defined for Cl , while a set of candidate links is defined for Rel .

However, CaSOB does not know the difference of RR for C , such as the “Subject” and “Object” in an actual system. It is difficult to automatically determine roles simply by using candidate *link*. Thus, a user has to determine the subject and object of *relation* in the ontology schema model.

In the third step shown in Figure 5, Cl and Rel are defined using the selected terms. $\{\text{“Sensor”}, \text{“SensorGrounding”}\} \subset Cl$ are defined, while “supports” $\in Rel$ is defined between “Sensor” as a subject and “SensorGrounding” as an object. In this process, it is possible to define concepts and relation over the domains. To define Cl and Rel in other domains, we need to come back to the first process; choose “MMI Ontology,” and select candidates for selecting terms defined in “MMI Ontology” domain. Then, Figure 5 shows $\{\text{“System”}\} \subset Cl$

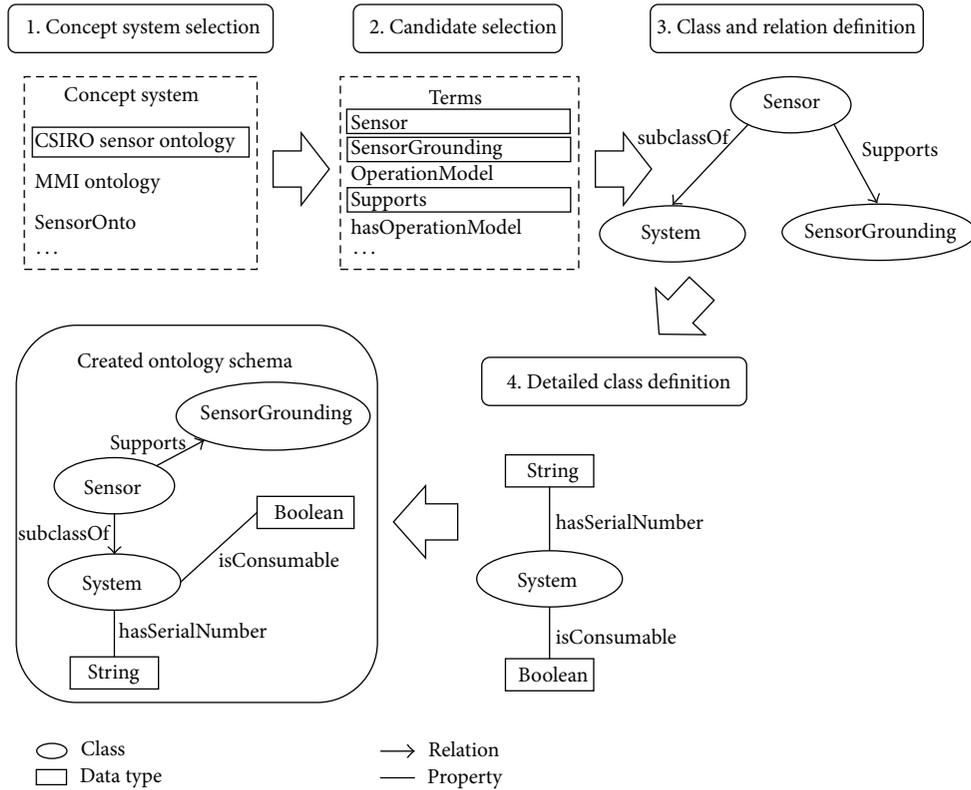


FIGURE 5: An example of the ontology building process.

is also defined, and “subclassOf” $\in Rel$ is defined between “Sensor” as a subject and “System” as an object.

3.2.4. Detailed Class Definition. In the final process, P and DT are defined to decide properties of classes, and additionally CI and Rel are defined by the mapping model.

Because “System” $\in CI$ was defined in the previous step, it is given that “hasSerialNumber” $\in P$ with “String” $\in DT$ and “isConsumable” $\in P$ using “Boolean” $\in DT$ in the final step of Figure 5.

4. Implementation

This section describes a metadata registry and CaSOB implementation. To make easy connection and mapping model, we need to implement a metadata registry based on MDR specification. The metadata registry is implemented by MySQL database, which is a RDBMS (relational database management system). For testing, we need to store datasets in the metadata registry, so we select several sensor ontologies as the dataset. Then we assume that the selected ontologies contained common concepts, so the datasets stored in the metadata registry also contain common concepts. CaSOB is developed in C# for a user interface, database connection, and the ontology building processes.

4.1. A Metadata Registry Implementation. The metadata registry is implemented so the MDR metamodel could collect

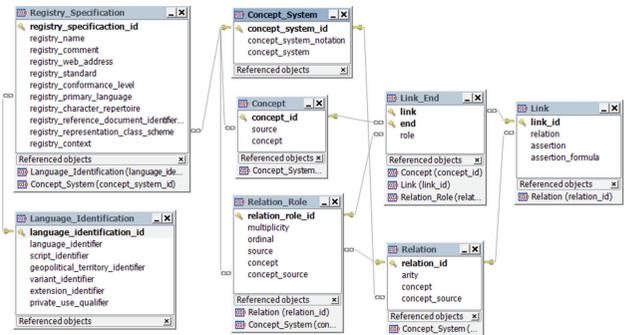


FIGURE 6: Implementation of the concepts region.

data directly as required, to import the administered metadata and create an ontology. Thus, we initially develop the metadata registry by implementing the metamodel of MDR. Section 4.1 presents the MDR-based system structured by RDBMS.

Each component in the metamodel is defined as RDBMS table, while properties of each component are defined as table attributes. Moreover, associations between components are defined using foreign key referential integrity.

Figure 6 shows the table and foreign key references of the concepts region in MDR. The “Concept” table includes the attributes “concept” and “source.” “concept” is an attribute where the name held by the concept of MDR is expressed using strings. “source” is an attribute that expresses a concept

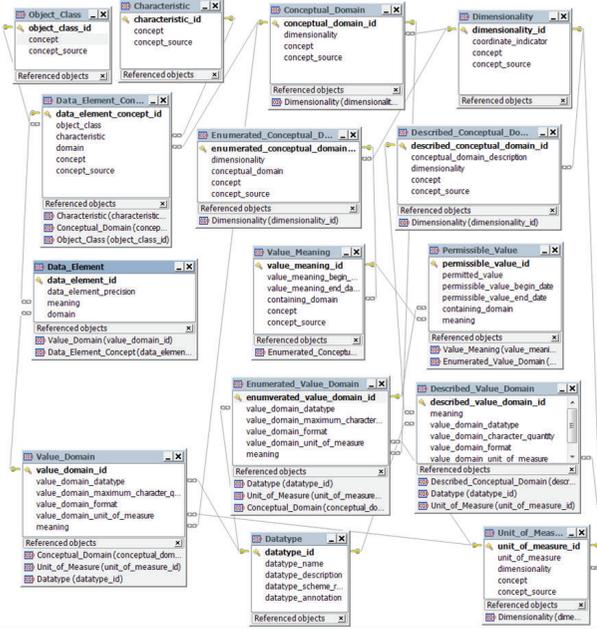


FIGURE 7: Implementation of the data description region.

system including concepts using foreign key references. Moreover, the “Link” table has a “relation” attribute for the “Relation” table. The “Link_End” table has the attributes “end” and “role” for the “link” attribute, where “end” means the concept, “role” means the relation role, and “link” means the link. Thus, one link has at least two link ends.

In Figure 7, the data description region is implemented as tables and foreign key references. Seven components in the data description region inherit the concept in the concepts region, so the components express the inheritance using each of the attributes of “concept” and “concept source” held in the “Concept” table.

4.2. CaSOB Implementation. In Section 4.2, the ontology building process of CaSOB is presented with the implementation screen, which is based on the four processes of ontology creation.

4.2.1. Concept System Selection (Process 1). If the domain of the ontology to be created has been determined, the system searches and selects the concept system (CS) of the corresponding domain. In Figure 8(a), CS of Process 1 is selected in the upper right corner of the screen as “CSIRO Sensor Ontology” to create the sensor ontology.

4.2.2. Candidate Selection (Process 2). The process selects each candidate concept and each candidate link from the concepts (C) and links (*link*), including the concept system selected in Process 1. Type currently indicates the seven components inheriting the concept.

In Figure 8(a), “support {Sensor (supportedBy), SensorGrounding (supports)}” is selected from links. The selected *link* is added to candidate link and contained concepts

“Sensor” and “SensorGrounding” are automatically added to candidate concepts.

4.2.3. Class and Relation Definition (Process 3). The process defines the classes (Cl) and relations (Rel) of the ontology schema model from the selected candidates. A user selects the desired candidate concept to define Cl and selects candidate links to set the domain and range of Rel. An R including the candidate link is mapped to a Rel of the ontology. The domain and range are set during the selection of a candidate link with the roles of a subject and an object, respectively. A user selects each candidate to define the Cl and Rel of the ontology schema model.

In Figure 8(b), to define “System” $\in C$ in other domains, “MMI Ontology” $\in CS$ is selected. And “System” is selected as candidate concepts and “class-subsumption {Sensor (superclassOf), System (subclassOf)}” is selected as candidate links. The marked candidate link expresses “class-subsumption” which has (“Sensor”, “subclassOf”, “System”)

4.2.4. Detailed Definition of Classes (Process 4). This process defines classes (Cl) and properties (P) of the classes using the candidate concept and the type of concepts. Additional relations (Rel) are also created.

In this process, a user selects the candidate concepts. Each concept has a type and has different mechanism to build ontologies according to the mapping model (MM). A user defines Cl, P, DT, and additional Rel of the ontology schema model. When a candidate concept is selected, the data that can be selected changes depending on its type so an ontology is created containing the relations among components that were defined in the previous section.

For example, Figure 8(c) shows a screenshot of the detailed definition of Cl in the ontology schema model. The selected candidate concept “System” has data element concept (DEC) type. “hasSerialNumber” $\in Pr$ becomes P with “String” $\in DT$ for “System” $\in Cl$ from “System, hasSerialNumber, String” $\in DE$ associated with “System, hasSerialNumber” $\in DEC$.

4.2.5. The Created Ontology. Figure 8(d) shows the complete ontology for all processes. The complete ontology is mainly comprised of classes (Cl), properties (P), and relations (Rel). P of each specific Cl has a data type (DT). Rel has two Cls with the roles of domain and range.

CaSOB provides a general ontology for a target ontology. It means that CaSOB does not provide specific ontology languages such as RDF, OWL, SKOS, and Topic Map. Thus, a user needs to describe the complete ontology using a specific ontology language. Algorithm 1 shows the ontology created in Figure 8(d) as described using OWL.

5. Evaluation

5.1. Qualitative Evaluation. We conducted a comparative evaluation study for ontology creation. This section first

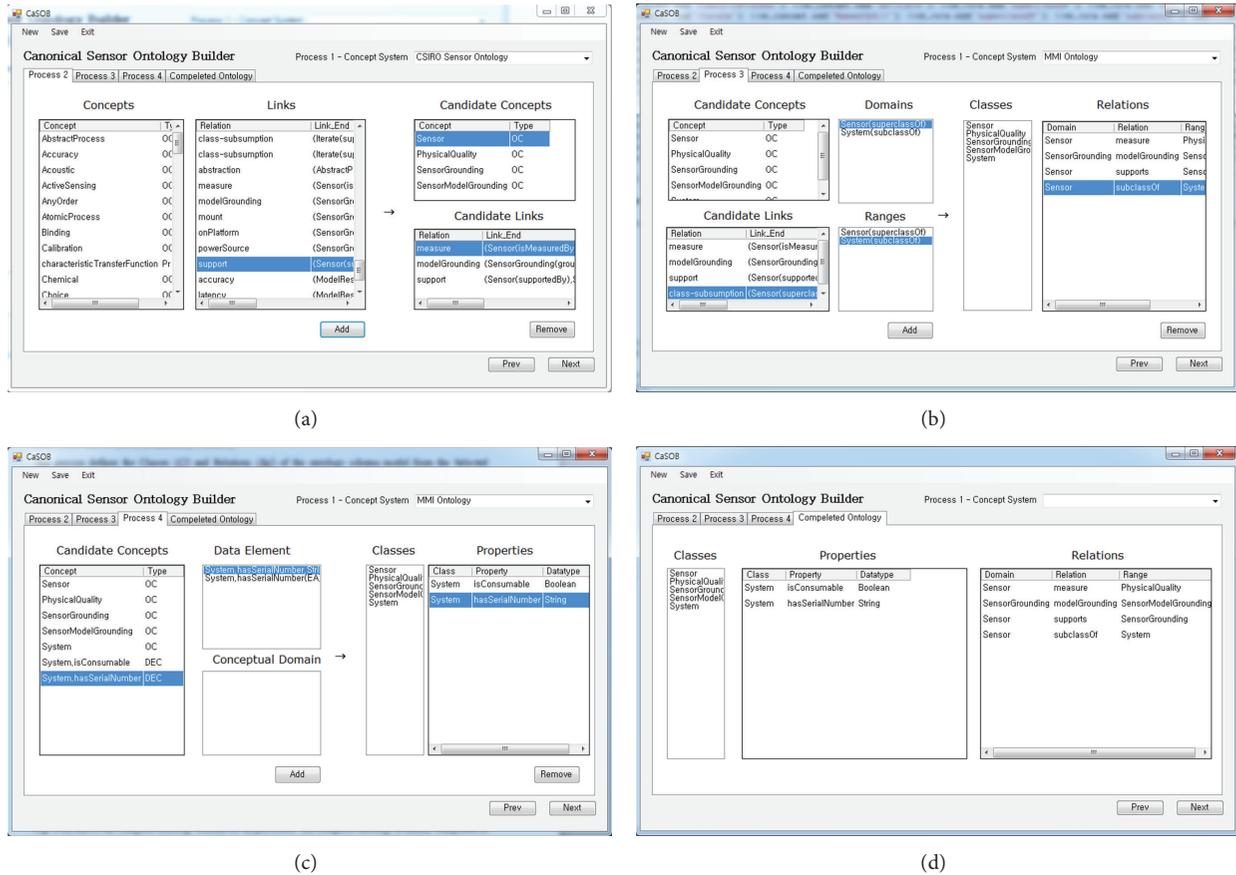


FIGURE 8: Screenshots of CaSOB.

describes our qualitative evaluation using various comparative items. This evaluation involved the comparative evaluation of three ontology building approaches which are user-defined approach, ontology reuse approach, and our proposed MDR based approach. Table 1 shows the qualitative evaluation results.

The first factor for evaluation is standardization level. It means whether the created ontology could be used as a common or general ontology or not. User-defined approach is hard to become a common standard approach because of experts who can create ontologies based on their own knowledge. Hence created ontologies by user-defined approach cannot share their concept. On the other hand, since ontology reuse approach and MDR based approach use ontologies already created, which can contain common concepts, created ontology using these ontology reuse and MDR based approaches can become standardized ontologies. That is why each standardization level of these two approaches becomes high.

The second evaluation factor is ontology quality. Ontology quality means whether created ontologies are well defined and whether the quality of created ontologies is guaranteed. In case of user-defined approach, a created ontology contains personal view point without general or common perspective. Therefore, the created ontology cannot guarantee the quality of ontologies. CSIRO, MMI, and ISTAR ontologies are popular ontologies. These ontologies are one of

the results from each big project. These ontologies are verified by a lot of experts whether concepts and relationships of these ontologies are well defined or not. If we create a new ontology reusing these well-defined ontologies, we can guarantee the ontology quality. In MDR based approach, ontologies and common concepts are controlled by MDR organizations that follow a standardized procedure for managing common concepts and reflect a standardized lifecycle for creating ontologies. It means that the MDR-based ontology reuse approach enables high-quality ontology creation.

Limitation of expression means how the expressive power is restricted when ontologies are created. User-defined approach has no limitation to express knowledge based on ontologies because a user selects their own terms and the user can make relationships following user's opinion. Ontology reuse approach just uses existing ontologies. It is not allowed to change existing concepts and terms when concepts or terms are incorrect from target domain. Thus, the expression of ontology is very strict. MDR based approach reuses registered ontologies in the metadata registry. The MDR based approach also permits changing other ontologies when there are no concepts or terms that are required in existing ontologies. Nevertheless, the MDR approach is strict because it is not allowed to define unregistered ontologies.

The final factor of qualitative evaluation is creation cost. Creation cost means convenience of creating ontologies and time for building ontologies. In user-defined approach,

```

<?xml version="1.0"?>
<Ontology xmlns="http://www.w3.org/2002/07/owl#"
  xml:base="http://software.korea.ac.kr/SwsysSensorOntology.owl"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xml="http://www.w3.org/XML/1998/namespace"
  >
  <Prefix name="xsd"
    IRI="http://www.w3.org/2001/XMLSchema#"/>
  <Prefix name="owl"
    IRI="http://www.w3.org/2002/07/owl#"/>
  <Prefix name="rdf" IRI="http://www.w3.org/1999/02/22-rdf-syntax-ns#"/>
  <Prefix name="rdfs" IRI="http://www.w3.org/2000/01/rdf-schema#"/>

  <Declaration>
    <Class IRI="#PhysicalQuality"/>
    <Class IRI="#Sensor"/>
    <Class IRI="#SensorGrounding"/>
    <Class IRI="#SensorModelGrounding"/>
    <Class IRI="#System"/>
    <ObjectProperty IRI="#subclassOf"/>
    <ObjectProperty IRI="#measure"/>
    <ObjectProperty IRI="#modelGrounding"/>
    <ObjectProperty IRI="#supports"/>
    <DataProperty IRI="#isConsumable"/>
    <DataProperty IRI="#hasSerialNumber"/>
  </Declaration>
  <ObjectPropertyDomain>
    <ObjectProperty IRI="#subclassOf"/>
    <Class IRI="#Sensor"/>
  </ObjectPropertyDomain>
  <ObjectPropertyRange>
    <ObjectProperty IRI="#subclassOf"/>
    <Class IRI="#System"/>
  </ObjectPropertyRange>
  <ObjectPropertyDomain>
    <ObjectProperty IRI="#measure"/>
    <Class IRI="#Sensor"/>
  </ObjectPropertyDomain>
  <ObjectPropertyRange>
    <ObjectProperty IRI="#measure"/>
    <Class IRI="#PhysicalQuality"/>
  </ObjectPropertyRange>
  <ObjectPropertyDomain>
    <ObjectProperty IRI="#modelGrounding"/>
    <Class IRI="#SensorGrounding"/>
  </ObjectPropertyDomain>
  <ObjectPropertyRange>
    <ObjectProperty IRI="#modelGrounding"/>
    <Class IRI="#SensorModelGrounding"/>
  </ObjectPropertyRange>
  <ObjectPropertyDomain>
    <ObjectProperty IRI="#supports"/>
    <Class IRI="#Sensor"/>
  </ObjectPropertyDomain>
  <ObjectPropertyRange>
    <ObjectProperty IRI="#supports"/>
    <Class IRI="#SensorGrounding"/>
  </ObjectPropertyRange>
  <DataPropertyDomain>
    <DataProperty IRI="#isConsumable"/>
    <Class IRI="#Person"/>
  </DataPropertyDomain>

```

```

<DataPropertyRange>
  <DataProperty IRI="#isConsumable"/>
  <Datatype abbreviatedIRI="xsd:boolean"/>
</DataPropertyRange>
<DataPropertyDomain>
  <DataProperty IRI="#hasSerialNumber"/>
  <Class IRI="#Person"/>
</DataPropertyDomain>
<DataPropertyRange>
  <DataProperty IRI="#hasSerialNumber"/>
  <Datatype abbreviatedIRI="xsd:string"/>
</DataPropertyRange>
</Ontology>

```

ALGORITHM 1: An ontology described using OWL.

TABLE 1: Qualitative evaluation of ontology building approaches.

Approach	Standardization level	Ontology quality	Limitation of expression	Creation cost
User-defined	Low	Low	No limitation	High
Ontology reuse	High	Medium	Very strict	Low
MDR based (our proposal)	High	High	Strict	Medium

experts directly define concepts and terms, and it requires high cost to survey domain knowledge and collect reference documents for verification. Ontology reuse approach uses existing ontologies without change if the existing ontologies coincide with concepts and terms in target domain. Thus, the cost for ontology creation is low. MDR based approach uses ontologies registered in a metadata registry. When ontologies are created, users need to search terms and their relations to apply the search result to new ontologies. Users also need a customization task to create link between domains and ranges. Although this approach requires medium cost which is higher cost than the cost of ontology reuse approach, the cost of MDR approach is lower than user-defined approach.

5.2. Quantitative Evaluation: Simulation Approach. This section describes our quantitative analysis, which clearly shows the advantages of the proposed system, CaSOB. Many systems and studies have addressed interoperability between ontologies. However, most focus on interoperability issue among ontologies, which have already been created with their own ontology schema definition rules. Our approach is implemented in the step before ontology creation, which aims to reuse common concepts and increase interoperability. This paper defines existing approaches as user-defined ontology schema creation methods and we conducted a quantitative evaluation to compare our method with the user-defined schema creation method. The metadata registry also provides no schema creation method because its purpose is to register and share ontologies. In other words, our proposed method,

CaSOB, is based on the MDR, but the metadata registry is excluded from the quantitative evaluation.

An appropriate comparative item was determined for the evaluation. The goal of this paper is to minimize the semantic heterogeneity of ontologies via the reuse of common concepts. Therefore, this paper conducted a quantitative evaluation of the semantic heterogeneity rate. We can estimate the approximate ontology integration cost using the semantic heterogeneity rate, so this paper also includes the ontology integration cost as a comparative item.

The evaluation methodology is summarized as follows:

- (i) comparative targets: our proposed method, CaSOB, and the user-defined ontology schema creation method,
- (ii) comparative items: the semantic heterogeneity rate, that is, the number of heterogeneous semantics (concepts) shared among the created ontologies; the ontology integration cost, that is, the cost required to integrate a set of created ontologies.

The key symbols and notations are defined as follows:

- (i) O : a set of ontologies created with a target method;
 $O = \{o_1, o_2, o_3, \dots, o_n\}$,
- (ii) o_i : a created ontology consisting of one or more semantics, that is, a set of concepts such as classes, relations, properties, and data types,
- (iii) C : a set of common concepts registered and used for creating an ontology,

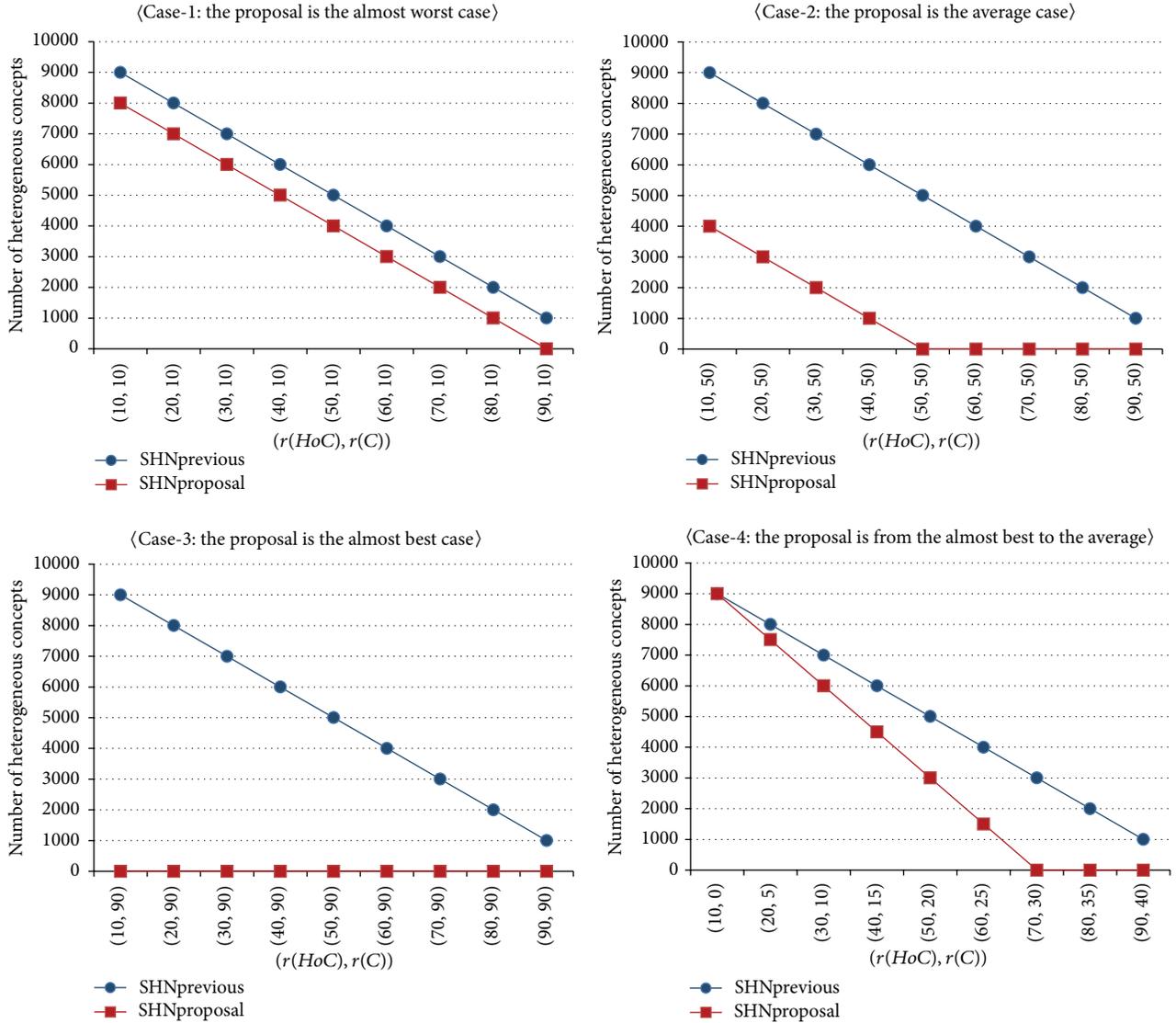


FIGURE 9: Comparative evaluation results for the number of semantic heterogeneities.

- (iv) HeC_i : a set of heterogeneous concepts in the i th ontology among the created ontologies,
- (v) HoC : a set of homogeneous concepts among the created ontologies,
- (vi) $n(O)$: the number of created ontologies,
- (vii) $n(o_i)$: the number of concepts in the i th ontology,
- (viii) $n(C)$: the number of common concepts used to create an ontology,
- (ix) $n(HeC_i)$: the number of heterogeneous concepts in the i th ontology,
- (x) $r(HoC)$: the average ratio of homogeneous concepts,
- (xi) $r(C)$: the average ratio of used common concepts,
- (xii) t : semantic interpretation time between two concepts; $0 < t < 1$,
- (xiii) SHN : the number of heterogeneous concepts,
- (xiv) OIC : the ontology integration cost.

For the evaluation, we assume that n -people generate an ontology in the same domain. In this case, the homogeneous concept means concepts which are commonly contained by most of the ontologies. For example, the concept such as sensor could be a homogeneous concept even though each of the sensor ontologies is developed by different people. On the other hand, the heterogeneous concept means discordant concepts with each of the ontologies. We assume that the number of heterogeneous concepts is the number of total concepts without an average ratio of homogeneous concepts.

Figure 9 shows the comparative evaluation result for the semantic heterogeneity as SHN . The evaluation determines the variation in the number of heterogeneous concepts depending on the change in the ratio of homogeneous concepts and the ratio of used common concepts.

$SHN_{proposal}$ considers the common concepts used to build an ontology. In other words, it is the result of an evaluation using our proposed method. $SHN_{previous}$ is an evaluation result using a user-defined schema creation method.

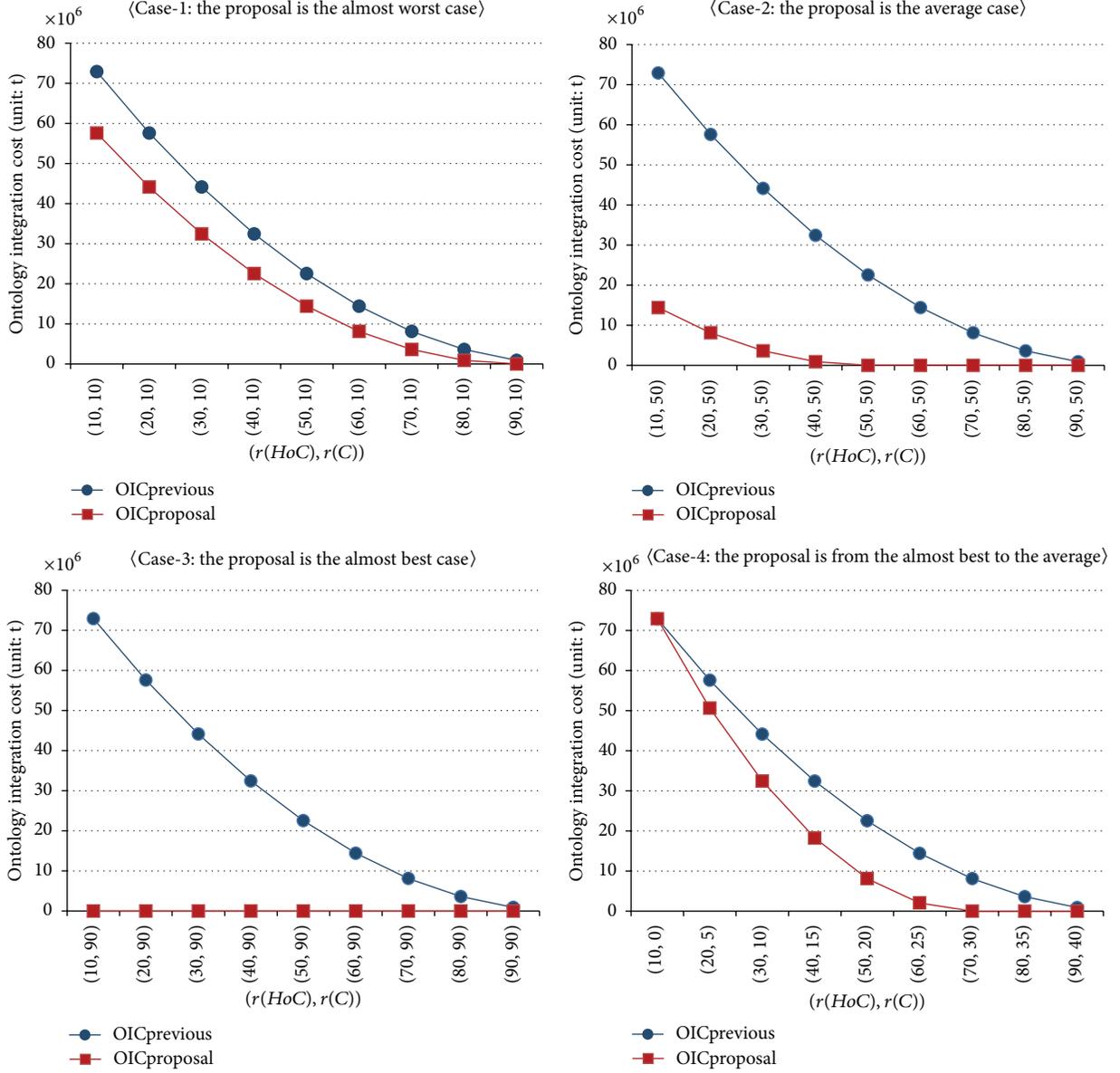


FIGURE 10: Comparative evaluation results for the ontology integration costs.

The number of heterogeneous concepts in $SHN_{previous}$ and $SHN_{proposal}$ is calculated as follows:

$$SHN = \sum_{i=1}^{n(O)} n(HeC_i). \quad (5)$$

Once $SHN_{previous}$ is determined, the number of heterogeneous concepts is

$$n(HeC_i) = \begin{cases} n(o_i) - (n(o_i) \times r(HoC)), & r(HoC) < 1, \\ 0, & \text{otherwise.} \end{cases} \quad (6)$$

And when $SHN_{proposal}$ is determined, the number of heterogeneous concepts is

$$n(HeC_i) = \begin{cases} n(o_i) \\ - ((n(o_i) \times r(HoC)) \\ + (n(o_i) \times r(C))), & r(HoC) + r(C) < 1, \\ 0, & \text{otherwise.} \end{cases} \quad (7)$$

Figure 9 shows that in all cases there is a decrease in the number of heterogeneous concepts, which is inversely proportional to the ratio of used common concepts. Thus, the results demonstrate the efficiency of the use of common concepts for heterogeneity problems.

Figure 10 shows the comparative evaluation results for the ontology integration costs. The ontology integration

cost is the time taken for the comparison of ontology integration. The magnitude of the comparison decreases with the increasing size of the homogeneous concepts and the common concepts. The ontology integration cost is calculated as follows:

$$OIC = \sum_{i=1}^{n(o)} \left(n(HeC_i) \times \left(\left(\sum_{j=1}^{n(o)} n(HeC_j) \right) - n(HeC_i) \right) \right) \times t. \quad (8)$$

When $OIC_{previous}$ is determined, the number of heterogeneous concepts uses the same equation as the $SHN_{previous}$. Similarly, when $OIC_{proposal}$ is determined, the number of heterogeneous concepts uses the same equation as the $SHN_{proposal}$.

Figure 10 shows that in all cases the decrease in the ontology integration cost is inversely proportional to the square of the ratio of the common concepts. Thus, the results show that a greater use of common concepts is efficient for ontology matching and integration.

6. Conclusion

In various domains of sensor network environment, sensor ontologies are developed to represent sensor information consistently and increase interoperability. However, these sensor ontologies are defined with specific purposes, and thus we cannot use the existing sensor ontologies as it is.

This paper proposed MDR based approach to solve the heterogeneity problems. MDR based approach develops ontologies using common concepts and metadata registered in a metadata registry. The approach guarantees a quality of ontologies and solves the heterogeneity problems. We proposed MDR based Ontology Builder (CaSOB) framework, described mapping model between MDR and ontology, and presented ontology building processes. For implementation, we developed the metadata registry with RDBMS and CaSOB system. In qualitative evaluation, we compared the proposed approach with user-defined approach and ontology reuse approach. As a result, the proposed approach shows advantages from ontology quality and creation cost. For quantitative evaluation we compared CaSOB system with user-defined ontology creation method, and CaSOB system efficiently solves heterogeneity problems rather than the user-defined ontology creation method in ontology matching and integration technique perspective.

The proposed approach still has several limitations which should be resolved. First, CaSOB does not handle constraints and objects (individuals) in instance level. Therefore, the proposed system should be extended to address this issue. CaSOB requires a manual processing by ontology designers for selecting proper concepts. It causes high cost, and thus an automatic creation function should be studied.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This research was supported by the Next-Generation Information Computing Development Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT & Future Planning (2012M3C4A7033346). And this research was supported by the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education, Science and Technology (no. 2011-0004911). The cocorresponding authors are Doo-Kwon Baik and Dongwon Jeong.

References

- [1] M. Botts, G. Percivall, C. Reed, and J. Davidson, "OGC sensor web enablement: overview and high level architecture," OpenGIS White Paper OGC 07-165, Open Geospatial Consortium, 2007.
- [2] A. Sheth, C. Henson, and S. S. Sahoo, "Semantic sensor web," *IEEE Internet Computing*, vol. 12, no. 4, pp. 78–83, 2008.
- [3] D. J. Russomanno, C. Kothari, and O. Thomas, "Sensor ontologies: from shallow to deep models," in *Proceedings of the 37th Southeastern Symposium on System Theory (SST'05)*, 2005.
- [4] M. Compton, P. Barnaghi, L. Bermudez et al., "The SSN ontology of the W3C semantic sensor network incubator group," *Journal of Web Semantics*, vol. 17, pp. 25–32, 2012.
- [5] R. Eastman, C. Schlenoff, S. Balakirsky, and T. Hong, "A sensor ontology literature review," NISTIR 7908, National Institute of Standards and Technology, 2013.
- [6] W3C Incubator Group, "Semantic sensor network XG final report," W3C Incubator Group Report, W3C, 2011.
- [7] D. Kim, S. Cha, and K. Cho, "Ontology-based methodology for managing heterogeneous wireless sensor networks," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 610684, 9 pages, 2013.
- [8] L. Lezcano, L. Santos, and E. García-Barriocanal, "Semantic integration of sensor data and disaster management systems: the emergency archetype approach," *International Journal of Distributed Sensor Networks*, vol. 2013, Article ID 424821, 11 pages, 2013.
- [9] Y. Shi, G. Li, X. Zhou, and X. Zhang, "Sensor ontology building in semantic sensor web," in *Internet of Things*, pp. 277–284, 2012.
- [10] ISO/IEC JTC1/SC32, *ISO/IEC 11179 Metadata Registries (MDR) Part 1: Framework*, 3rd edition, 2013.
- [11] ISO/IEC JTC1/SC32, *ISO/IEC 11179 Metadata Registries (MDR) Part 2: Classification*, 2nd edition, 2005.
- [12] ISO/IEC JTC1/SC32, *ISO/IEC 11179 Metadata Registries (MDR) Part 3: Registry Metamodel and Basic Attributes*, 3rd edition, 2013.
- [13] ISO/IEC JTC1/SC32, *ISO/IEC 11179 Metadata Registries (MDR) Part 4: Formulation of Data Definition*, 2nd edition, 2007.
- [14] ISO/IEC JTC1/SC32, *ISO/IEC 11179 Metadata Registries (MDR) Part 5: Naming and Identification Principles*, 3rd edition, 2013.

- [15] ISO/IEC JTC1/SC32, *ISO/IEC 11179 Metadata Registries (MDR) Part 6: Registration*, 3rd edition, 2013.
- [16] S. Castano, A. Ferrara, and S. Montanelli, "Ontology-based interoperability services for semantic collaboration in open networked systems," in *Interoperability of Enterprise Software and Applications*, pp. 135–146, 2006.
- [17] S. Castano, A. Ferrara, and S. Montanelli, "Matching ontologies in open networked systems: techniques and applications," in *Journal on Data Semantics V*, pp. 25–63, 2006.
- [18] G. Pirro, M. Ruffolo, and D. Talia, "SECCO: on building semantic links in Peer-to-Peer networks," in *Journal on Data Semantics XII*, pp. 1–36, 2009.
- [19] M. Bakillah and M. A. Mostafavi, "G-Map semantic mapping approach to improve semantic interoperability of distributed geospatial web services," in *Advances in Conceptual Modeling: Applications and Challenges*, pp. 12–22, 2010.
- [20] C. Matuszek, J. Cabrai, M. Witbrock, and J. DeOliveira, "An introduction to the syntax and content of Cyc," in *Proceedings of the Spring Symposium on Formalizing and Compiling Background Knowledge and Its Applications to Knowledge Representation and Question Answering*, pp. 44–49, March 2006.
- [21] B. Biebow and S. Szulman, "TERMINAE: a linguistics-based tool for the building of a domain ontology," in *Knowledge Acquisition, Modeling and Management*, pp. 49–66, 1999.
- [22] SensorOntology2009, "CSIRO sensor ontology," 2009, <http://www.w3.org/2005/Incubator/ssn/wiki/SensorOntology2009>.
- [23] "MMI, Marine Metadata Interoperability," 2009, <http://marine-metadata.org/community/teams/ontdevices>.
- [24] Defence Science and Technology Laboratory, "ISTAR and sensors," <https://www.dstl.gov.uk/istarandsensors>.
- [25] M. Compton, C. Henson, L. Lefort, H. Neuhaus, and A. Sheth, "A survey of the semantic specification of sensors," in *Proceedings of the 2nd International Workshop on Semantic Sensor Networks (SSN '09)*, pp. 17–32, 2009.
- [26] H. Neuhaus and M. Compton, "The semantic sensor network ontology: a generic language to describe sensor assets," in *Proceedings of the AGILE Pre-Conference Workshop Challenges in Geospatial Data Harmonisation*, 2009.
- [27] C. Rueda, L. Bermudez, and J. Fredericks, "The MMI ontology registry and repository: a portal for marine metadata interoperability," in *Proceedings of the MTS/IEEE Biloxi: Marine Technology for Our Future: Global and Local Challenges (OCEANS '09)*, pp. 1–6, 2009.
- [28] A. Preece, M. Gomez, G. de Mel et al., "Matching sensors to missions using a knowledge-based approach," in *Defense Transformation and Net-Centric Systems*, Proceedings of SPIE, 2008.

