Hindawi Publishing Corporation
International Journal of Distributed Sensor Networks
Volume 2013, Article ID 610684, 9 pages
http://dx.doi.org/10.1155/2013/610684

Research Article

Ontology-Based Methodology for Managing Heterogeneous Wireless Sensor Networks

Dae-Young Kim,[1] Si-Ho Cha,[2] and Kuk-Hyun Cho[1]

[1] *Department of Computer Science, Kwangwoon University, Seoul 139-701, Republic of Korea*
[2] *Department of Multimedia Science, Chungwoon University, Incheon 402-803, Republic of Korea*

Correspondence should be addressed to Si-Ho Cha; shcha@chungwoon.ac.kr

Received 22 May 2013; Accepted 22 August 2013

Academic Editor: Tai-hoon Kim

With the growing interest in ubiquitous technologies recently, many studies have been conducted in order to manage wireless sensor network, which is the central component in ubiquitous environment. To manage a certain wireless sensor network, a variety of techniques can be applied and thereby the network can be managed. However, it is very complicated and time-consuming to incorporate and manage more than a heterogeneous wireless sensor network since each network has different data, meanings, and terms. Instead of conducting verification through examining the detailed technology or simulation in order to manage a variety of wireless sensor networks, we suggest a methodology which manages data coming from the various sensors mentioned above, a technique by which those data from various sensor networks are managed in each different network, and a methodology which manages sensor networks where management languages exist. This paper defined the network-related terminology using Protégé for the tools and definitions necessary for ontology-based methodology, thereby defined OWL and the related rules, and suggested a methodology for incorporated management. In addition, this study showed that the methodology presented in this study has expandability and it can incorporate and manage a variety of wireless sensor networks.

1. Introduction

Many researchers have paid much attention to sensor networks for the past several years. Thus, a variety of researches, such as a research for solving the problems occurring from sensor devices' restriction, a research for effectively placing sensors, and a research for effectively managing the data coming from sensors, have been conducted. USN (Ubiquitous Sensor Network), which is the core technology of ubiquitous environment, places a variety of sensors, collects the data coming from those sensors, and manages them. Although it is a hard work to manage the data collected from the various heterogeneous sensors, it is necessary for providing ubiquitous environment. To do this, several researches, such as the research on filtering sensor data [1], the research on interpreting and utilizing the meaning of sensor data [2], and the research on managing sensor networks through inquiry language [3], have been progressed. Moreover, sensor networks can be used for various areas due to their high level of usage. A variety of sensor networks are built, depending on their use purpose, and the data collected from those different sensor networks are used as proper information, depending on each network's circumstance. In ubiquitous environment, we may have to be able to incorporate and manage those sensor networks. However, it is very complicated and time-consuming to incorporate and manage different sensor networks, and it is also very difficult since the technique to manage each network differs from each other. In addition, most of the sensor network management models use their own scenarios and management languages. These languages may express different meanings from each other since each network has its own language.

To solve this problem, we may need to manage sensor networks from the higher level of viewpoint rather than just a simple sensor networks. Even if this paper did not verify the detailed technology or simulation necessary for managing a variety of sensor networks, this paper suggests a methodology for managing data coming from various sensors mentioned above, a management technique for different sensor networks, and a methodology for incorporating and managing sensor networks, in which many management languages exist. Furthermore, this study defined the network-related terminology using Protégé for the tools and definitions necessary for managing ontology-based heterogeneous sensor networks, thereby defined OWL (Web Ontology Language) and the related rules. The management methodology suggested by this paper is a network structure that has expandability, and it can incorporate and manage a variety of wireless sensor networks.

The rest of this paper is organized as follows. In Section 2, the existing ontology-related technologies and management methods of the heterogeneous sensor networks are described. Section 3 introduces the management structure of the proposed ontology-based heterogeneous sensor networks. Section 4 describes the implementation of the proposed management methodology. Finally, in Section 5, conclusions are made including the future research.

2. Related Work

2.1. Ontology-Related Technologies. If we classify ontology-related technologies, these are largely classified into ontology language, ontology tool, Reasoner, and ontology API, as shown in Figure 1.

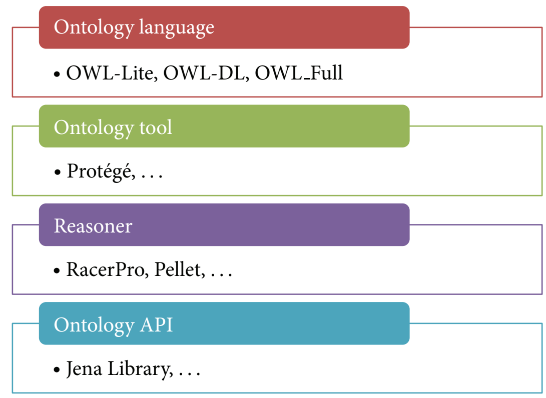

Figure 1: Ontology-related technologies.

To build the management methodology for ontology-based heterogeneous sensor networks, we basically need ontology-expression language, ontology tool, Reasoner, and ontology API. Reasoner plays a role to find the answer in the knowledge base. Reasoner provides a methodology to reason information of the knowledge and formalize conclusions of them. Jena provides a Java framework for Semantic Web application development, and Ontology API is provided through the Jena Library.

Largely, there exist RDF (Resource Description Framework) and OWL, as a language that can express ontology. RDF is the XML-based framework that describes the metadata for specific resources, and it has a concept that it treats resource, attribute, and attribute value as one unit rather than the existing record unit, and each resource has its own identifier through URI (Uniform Resource Identifier). The relation setup between these resources can be made infinitely through attributes.

OWL is a simpler and easier language for securing more universal users, and since it includes rich vocabulary and formal semantics, it enables us to utilize the stronger mechanical interpretation ability than using XML, RDF, and RDF schema (RDF-S). OWL includes three types such as OWL-Lite, OWL-DL, and OWL-Full. Each language is determined by expression range. OWL-Lite has the lowest expression ability, while OWL-Full has the highest expression ability. OWL-DL has a little more expression ability than OWL-Lite. Protégé [4] is the system that was developed by Stanford University to design and acquire the knowledge-based model, and it allows users to build domain ontology [5]. Protégé is one of the most frequently used platforms, and it is the ontology package developed by SMI (Stanford Medical Informatics). To generate and edit ontology, Protégé uses the components such as Protégé-OWL Class, Protégé-Properties, Protégé-Forms, Protégé-Individuals, and Protégé-OWLViz. Protégé allows researchers to generate the ontology-based knowledge management system [6]. In addition, it supports the expandable plug-in structure, can use all the ontology languages, and it can be easily used to generate and edit the ontology language, OWL.

2.2. Management of Heterogeneous Sensor Networks. A variety of researches have been progressed to incorporate and manage the different sensing data coming from heterogeneous sensor networks, and give meaning to them through inference [7–13].

Reference [14] proposed the ontology-based sensor network prototype environment to infer between heterogeneous sensors and fuse them into one. This is the prototype that can process the ad hoc quality based on ontology, considering the dispersed sensor network environments. However, this study just made a simple suggestion, and it does not include the specific research results.

Reference [15] suggested the semantic data model and suggested the inquiry-processing method on multisensor networks, based on SSDL (Semantic Sensory Data Language). That is, we define the meaning tag of sensor data using SSDL, and based on this, we conduct aggregation operation for the data. But it has a problem that we have difficulty in processing the new meaning of tag since we can only statically manage the whole meaning of tag, which is defined by SSDL.

Reference [16] proposed the structure that can effectively manage metadata under the Federated Sensor Network. It automatically generates metadata by combining the dispersed metadata through the data stream management system called GSN (Global Sensor Networks). However, only when the system is equipped with GSN, the combination is possible, and this study did not provide the process of evaluating effectiveness for the automatically generated metadata.

3. Management Structure of the Ontology-Based Heterogeneous Sensor Networks

In this section, we suggest a variety of sensor information and the methodology that can manage it. To analyze the data

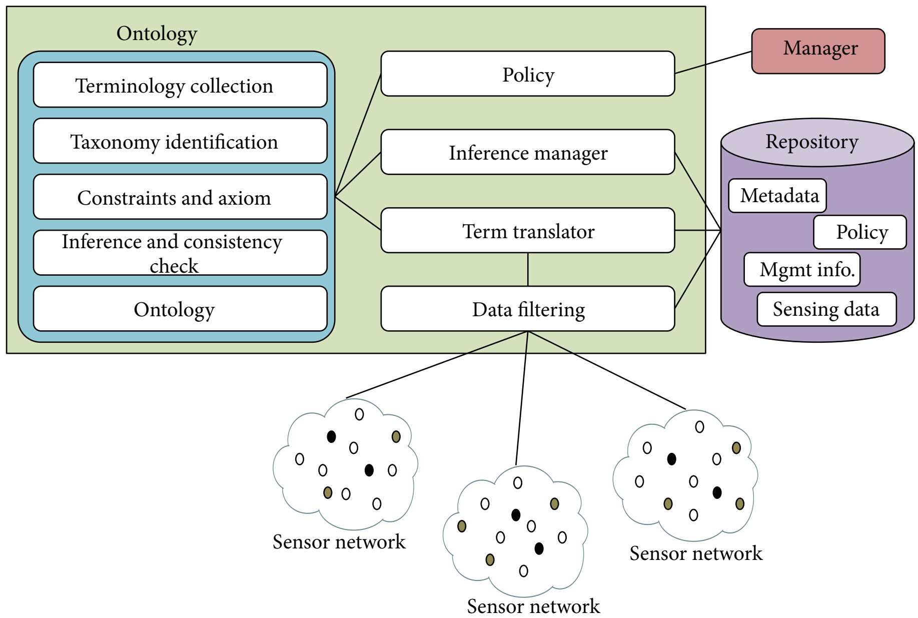

Figure 2: Management structure of wireless sensor networks.

collected from a variety of sensors and effectively manage the different sensor networks, we suggest the sensor network management architecture as shown in Figure 2.

The proposed management methodology for ontology-based wireless sensor networks first builds ontology to manage wireless sensor networks and then extracts only the necessary data from the data collected from sensors through filtering process, depending on administrator's management policy. The data extracted like this are transformed through Term Translator, and the transformed information is stored at Repository. And then, it manages the whole sensor networks through the various inferences with inference engine, and in the event that administrator directs another different policy, it manages both the existing ontology and the policy through inference engine. Each feature in Figure 2 is as follows.

(i) Ontology: it builds the basic ontology necessary for managing wireless sensor networks.

(ii) Policy: it receives the policy from administrator, in order to manage sensor networks, and then makes it ontology.

(iii) Inference manager: it makes inference, depending on the sensed information coming from subsystem and the policy directed by upper system.

(iv) Term Translator: it serves the feature of providing consistent terms because the management information and data coming from different sensor networks are different from each other. For this, it uses the Repository, already stored, and Ontology.

(v) Data filtering: it serves the feature of extracting only the necessary information among the data coming from a number of sensors.

3.1. Ontology Building Process. First of all, in order to manage a variety of sensor networks, we have to build the ontology, which collects and manages the terms used at each sensor network. Therefore, we build the ontologies through the process shown in Figure 3. The proposed ontology-based wireless sensor network management methodology collects the terms from subnetworks, incorporates the management technologies carried out by each sensor network, and uses the technology in which subnetworks are managed by upper ontology as shown in Figure 3.

Above all, it adds the terms and axioms defined by subnetworks. Of course, though there are subnetworks' rules and inferences, these are the ones for managing the networks of subnetworks, and this kind of information is not needed because later the network administrator can decide the desired policies and then add rules and inferences. And then, it collects the administration terms for incorporating and managing. There are a considerable number of terms coming from a variety of management models. Though these management terms have the same meaning, it is the case that they have different meanings. Accordingly, it should collect these terms and classify them. They do not need to be modified and have easy expandability since they can be applied as they are, in the event that other networks are added. In the stage of inference and consistency examination, inference and consistency examination are made to manage subnetworks. If ontology is built through this process, the basic structure for managing the whole sensor networks is made.

3.2. Term Translator. Term Translator plays a role in solving inconsistency between the ontology terms defined above and the terms defined in the subsensor networks. The terms, which were classified in the stage of recognition, are defined

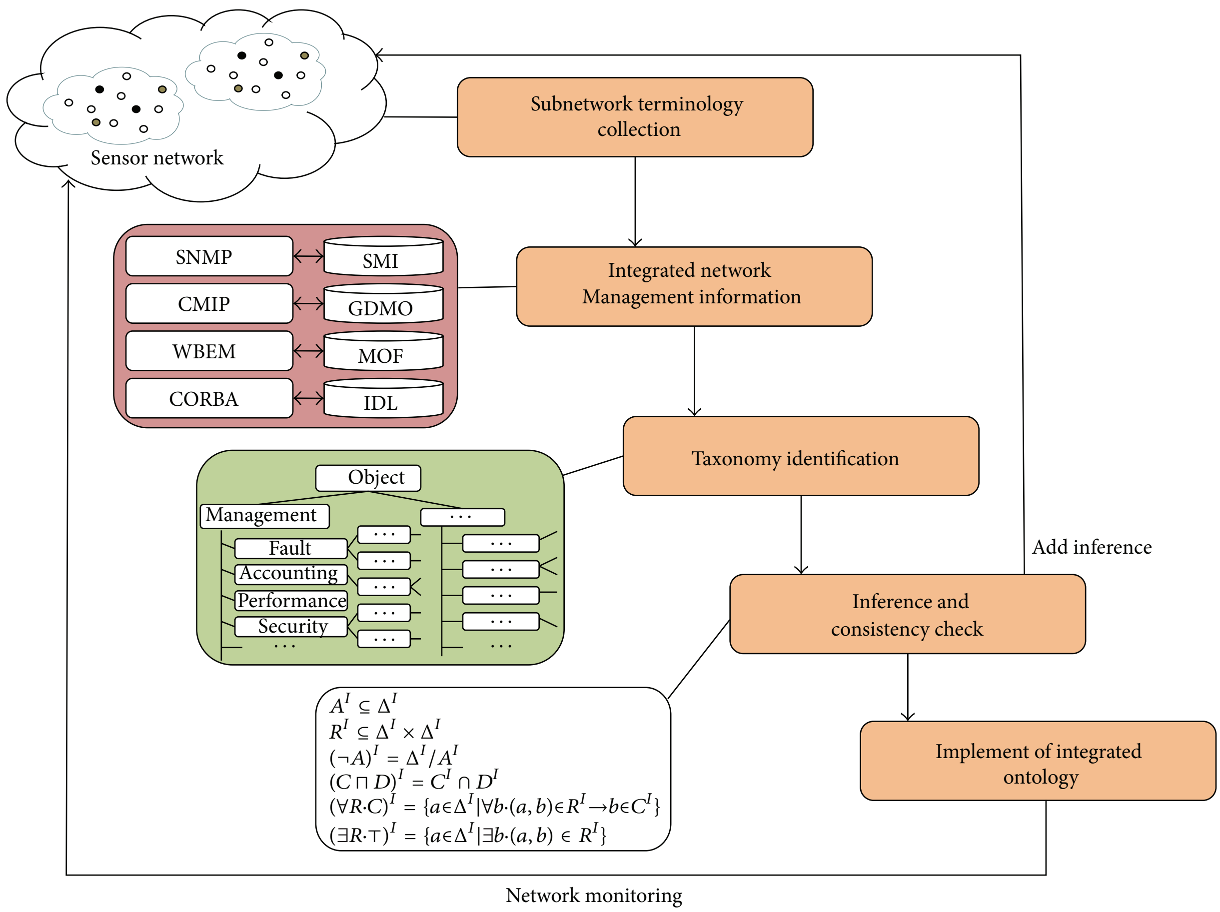

Figure 3: Ontology building process.

as class and subclass, and in the subsensor networks as shown in Figure 4, the instances for the class defined through Term Translator are generated.

The role of Term Translator is to transform the terms since each different network uses different terms. Through these instances that are generated, the integrated network administrator can manage the management information related to a variety of sensor networks.

3.3. Inference Manager. For the various sensor networks, each sensor network has its own management information, and accordingly the management information exists. The management information of each network is stored at Repository, and the integrated management for each network is possible through inference engine using this kind of stored information. Figure 5 shows this inference process. The sensed data request the process of inference while passing through inference engine, and the results pass through inference process again and then are stored at Repository. In addition, inference is carried out to manage sensor networks, depending on the obtained inference results.

4. Implementation of the Management Methodology for Ontology-Based Heterogeneous Sensor Networks

In this section, we describe the actual realization method in accordance with the methodology presented above. Figure 6 shows an example that collects and classifies the terms necessary for managing sensor networks. Even if there are many other terms in addition to the terms shown in Figure 6, here we showed just sensor network and sensor node-related terms for managing wireless sensor networks. Since the heterogeneous wireless sensor networks consist of various sensor networks, the management information, which manages these sensor networks, should include the items such as sensor network id and administrator's information, security, location, channel, and topology, and as for sensor node, the management information can be largely classified into system, device, connection, performance, and event.

Figure 7 shows the results, which were obtained by actually using Protégé tool and carrying out the classification of the terms collected above. The classes, which are defined in this way, are described as DL (Description Logic) type

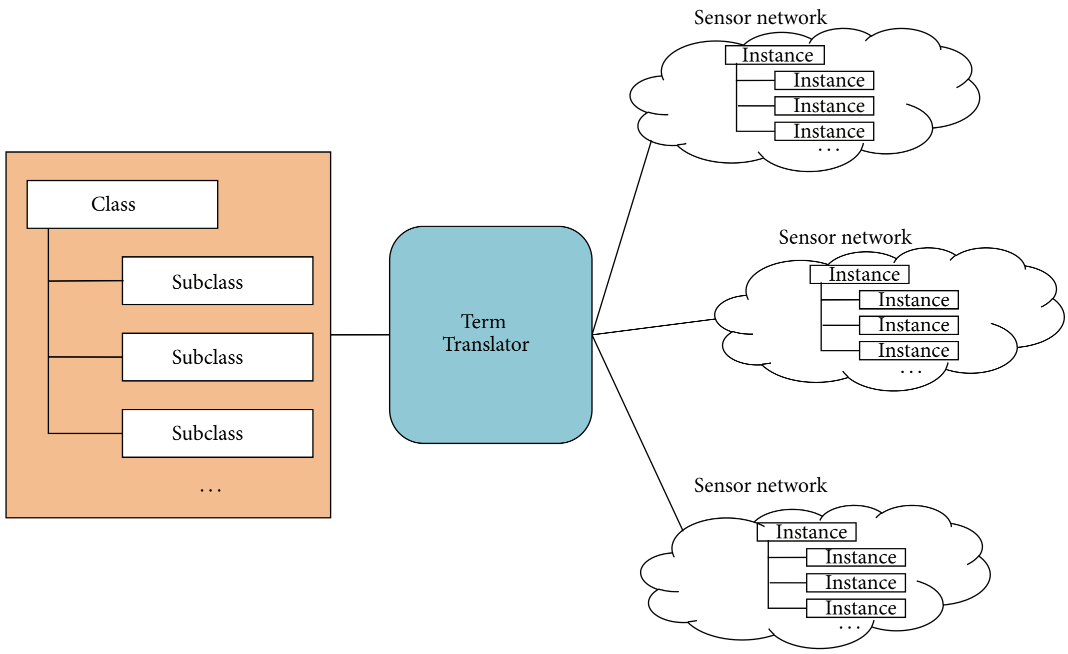

Figure 4: Term transformation through Term Translator.

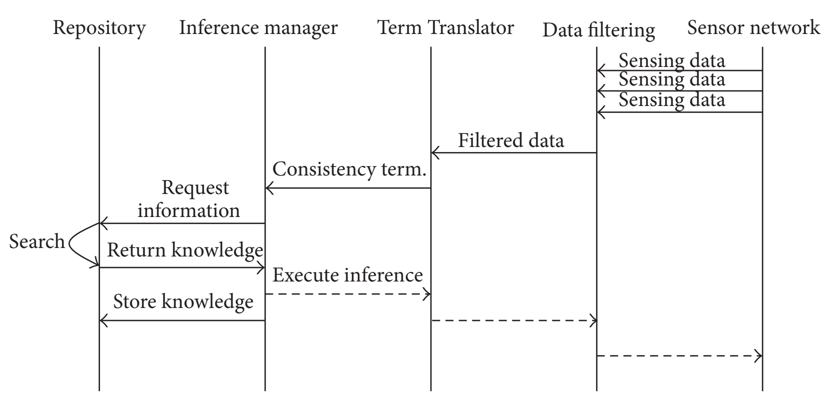

Figure 5: Inference process.

Target		Management information
Network		Net_ID, manager info, security, location, channel, topology, number of nodes, deployment time, Etc.
Node	System	MAC_addr, OS type and version, application info, boot-up time, synchronization, Etc.
	Device	CPU: type, clock, memory, ... RF: type, power, range, ... Battery: type, capacity, energy, ... Sensor: type, unit, value, cycle, ... Etc.
	Connection	Neighborlist, routing policy, role, Etc.
	Performance	Sent/recv packets, reliability, Etc.
	Event	Threshold, method, Etc.

Figure 6: Example of collecting terms for managing sensor networks.

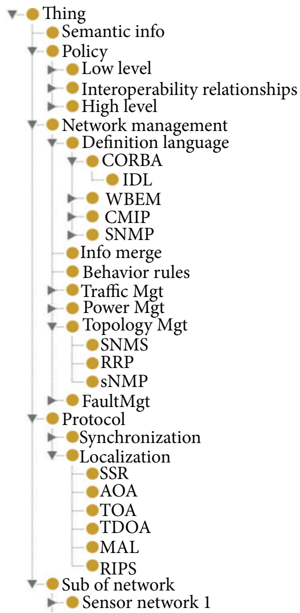

Figure 7: Term classification—Protégé Expression.

at Protégé, and Taxonomy inference and Classification is possible through ontology inference engine, using this kind of description. Since this study aims at integrating and managing sensor networks, we can see that the ontologies related to this are defined. As for the items that are required or may be needed on each subnetwork, if we add the items to Semantic Info, the request is reflected on Policy and thus automatically managed.

The Policy is divided into LowLevel and HighLevel in order to define the Policy for upper-network and the Policy for subnetwork, and we defined Interoperability Relationship in order to operate them together. The Network Management allowed to add the terms necessary for managing sensor networks and made it possible to effectively manage networks. We defined DefinitionLanguage since the management language on each network is different from each other, defined InfoMerge for the information to be shared, and added BehaviorRules in order to effectively manage, which can regulate the rules for behaviors. Other things are the terms basically needed to manage networks, and SubOfNetwork means the networks that are actually under management. In addition, since various protocols can be used, we classified protocols by use purpose, and we realized the structure to which additional protocol can be added, if necessary, depending on network expansion.

Algorithm 1 shows the example of building ontology through OWL Language, to be mapped for the terms defined above. This example is the one that expresses the information about address and attribute by OWL, with ontology for sending management information to manage sensor networks. It shows that the terms and information necessary for the networks are hierarchically expressed by class and subclass, through OWL.

Algorithm 2 shows the rule that, if DeliveryAddress is 128.60.X.X and SourceOntology is KwOntology, it represents the same network. As for the policies to be managed in this way, we can express them by IF-THEN structure, generate rules, and then apply them. In addition, if we intend to manage wireless sensor networks through cluster header, it is important to select cluster header. In this study, we made the rule for selecting basic cluster header and thereby defined it. Algorithm 2 shows the rule for how to find another header when cluster header node's energy is Low after selecting cluster header.

Algorithm 3 is the example of expressing by SWRL syntax, and in case of temperature sensor, a value detecting unusual movements varies, depending on the temperature of the area where temperature sensor is installed. It continues to seek for the mean value of the sensed values after the sensor is installed, and if the value is over the mean value and higher than administrator's policy value, this temperature sensor considers it as emergency. In addition, if the sensor value is between −5 and +5, compared to the mean value, the sensor's state is considered normal, and if the sensor value is under the mean value and lower than the fixed minimum value, the sensor's state is considered abnormal.

Figure 8 shows the example that is mapping the subsensor networks' terms. It analyzes and classifies the various terms collected above, and then manages similar terms by incorporating them into upper ontology, and if subnetworks use the different terms from each other, they are transmitted through the transformation process via Term Translator.

In case of Figure 8, it is difficult to separately manage the terms added later, due to the classification and recognition on the previously collected terms. In this case, we can transform and manage the terms through the following inference process. When the terms added later are compared to the previously defined terms in the aspect of similarity, if they have similar meaning, they can be expressed and managed as shown in Algorithm 4.

Figure 9 is one example of the management methodologies suggested by this paper, and it shows the wireless sensor network that is composed of cluster. Sensor shape means its type. Circular points and green dots represent temperature sensors and vibration sensors, respectively. And blue dots represent cluster headers. Since it has to select again another cluster header if header's energy is Low, it comes to select cluster, following the rule defined above. If cluster header's energy is Low, it lets us know the cluster header of which sensor network is Low, and makes us find another cluster. In case of Figure 9, it is progressed through warning window, but it can automatically select cluster header.

Figure 10 is the screen that monitors the sensed values by temperature sensor. Temperature sensors are sensing the values on different networks (WSN1, WSN2). Temperature of WSN1 is a little low, and that of WSN2 is a little high. Even if they are the same temperature sensors, they come to have different values, depending on the type of sensor network and the location of placement. The sensed data are mostly filtered, and if Warning occurs as shown in the figure, it comes to manage the network in accordance with administrator's policy.

```
<owl:Ontology rdf:about="">
<rdfs:comment> Ontology for Sensor Network Management </rdfs:comment>
<rdfs:label> Sensor Network Management Ontology </rdfs:label>
</owl:Ontology>
<owl:Class rdf:ID="&network:deliveryAddress">
<rdfs:subClassOf> <owl:Class rdf:ID="&network:preference"/></rdfs:subClassOf>
<owl: equivalentClass rdf:resource = "&KwNetwork:destination"/>
<owl: equivalentClass rdf:resource = "&deliveryVan;deliveryLoc"/>
</owl:Class>
```

Algorithm 1: Example of OWL language for management information to manage sensor networks.

IF deliveryAddress IS 128.60.x.x AND sourceOntology IS KwOntology
THEN equivalnetwork
IF deliveryAddress IS 128.60.x.x AND sourceOntology IS SeoilOntology
THEN differentnetwork and KwToSeoil
IF clusterHeaderEnegy IS Low THEN sendBroadcasting
IF sensor IS notClusterHeader AND IsSameNetwork THEN clusterCandidate
IF sensor IS clusterCandidate AND sensorEnegy IS Highest THEN sensor = clusterHeader

Algorithm 2: Rule defined by IF-THEN structure.

Temperature Sensor Rule
(i) hasTemperatureSensor (?value) ∧ swrlb:greaterThan(?value, average) ∧ hasTemperatureSensor(?policy) ∧ swrlb:greateThan(?value, policyValue) → hasType(emergency)
(ii) hasTemperatureSensor(?value) ∧ swrlb:lessThanOrEqual(?value,average) ∧ hasTemperatureSensor(?minValue) ∧ swrlb:lessThan(?value, minValue) → hasType(trouble)
ClusterHeader
(i) hasClusterHeaderEnegy (?value) ∧ swrlb:lessThan (?value, policyLow) → hasType(low)
(ii) hasClusterHeaderEnegy (?value) ∧ swrlb:lessThan (?value, policyHigh) → hasType(high)
(iii) hasSensorEnegy(?value) ∧ swrlb:greateThan(?value, policyValue) → hasDescription(custerHeaderCandidate)

Algorithm 3: Rule defined by SWRL syntax.

Mapping Terms
(i) (SensorID rdf:type OID) & (OID rdf:type SensorOID) → SensorID rdf:type SensorOID
(ii) (hertz rdf:type HERTZ) & (HERTZ rdf:type Hz) → hertz rdf:type Hz
(iii) (Fault rdf:type Fault_Management) & (Fault_Management rdf:type FaultMgt) → Fault rdf:type FaultMgt
Selecting Cluster Header Candidates
(i) (clusterHeaderEnegy rdf:type Low) → sendBroadcasting
(ii) (sensor rdf:type notClusterHeader) & (sensor rdf:location isSameNetwork) → clusterCandidate

Algorithm 4: Terms transformed by Term Translator.

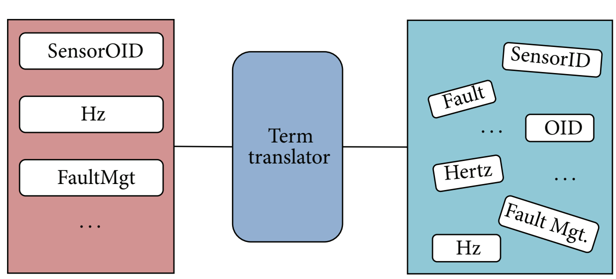

Figure 8: Example of mapping terms.

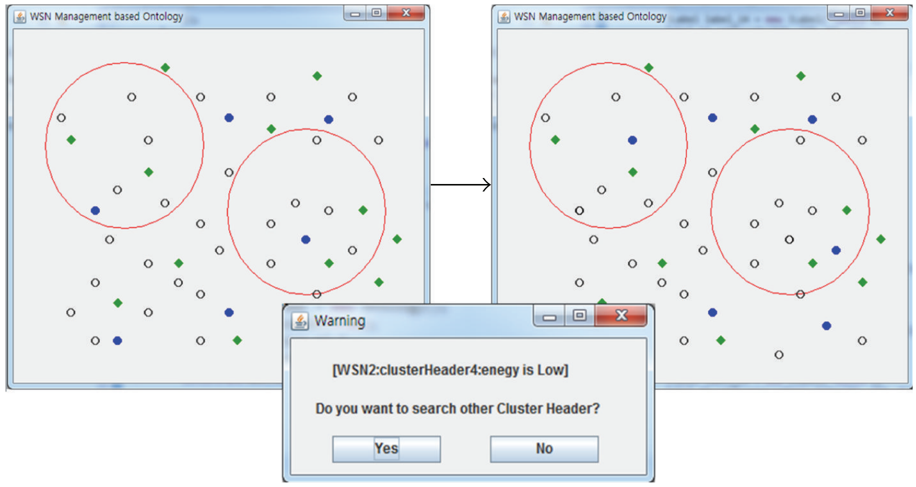

Figure 9: Screen for selecting cluster header.

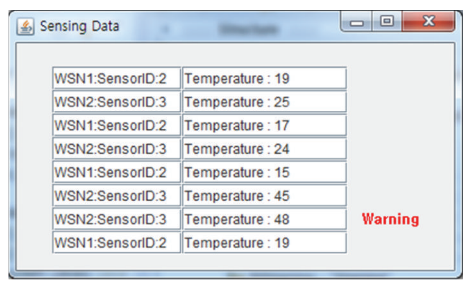

Figure 10: The values sensed by temperature sensor.

5. Conclusion

With the ontology-based heterogeneous wireless network management methodology presented in this paper, it is easy to incorporate, manage, and expand wireless sensor networks. The reason is because there are many terms, of which the meaning is very similar to each other even if they use different management information. Therefore, the method of managing subsensor networks on the upper ontology is a very effective one, and even the process of adding subnetworks can be easily made through the terms and classification defined by the upper ontology. In this paper, we defined the related terms necessary for networks using Protégé, thereby defined OWL and the related rules, and explained several examples among them. To expand networks and manage multinetworks, the process like this study is necessarily needed. For the future work, the detailed verification of the methodology and the realized results, which are suggested by this paper, is needed, and we have a future plan to study it with the flexible, expandable structure.

Acknowledgment

This work was supported by an Academic Research Fund of Chungwoon University in 2013.

References

[1] I. Mohomed, A. Misra, M. Ebling, and W. Jerome, "HARMONI: context-aware filtering of sensor data for continuous remote health monitoring," in *Proceedings of the 6th Annual IEEE International Conference on Pervasive Computing and Communications (PerCom '08)*, pp. 248–251, March 2008.

[2] I. K. Ibrahim, R. Kronsteiner, and G. Kotsis, "A semantic solution for data integration in mixed sensor networks," *Computer Communications*, vol. 28, no. 13, pp. 1564–1574, 2005.

[3] G. Xue, Q. Pan, and M. Li, "A new semantic-based query processing architecture," in *Proceedings of the International Conference on Parallel Processing Workshops (ICPPW '07)*, September 2007.

[4] Protégé, Stanford University Protégé Teaching website, http://protege.stanford.edu/.

[5] H. Knublauch and M. A. Musen, *Editing Description Logic Ontologies With the Protege-OWL Plugin*, Stanford Medical Informatics, Stanford University, Stanford, Calif, USA.

[6] S.-Y. Yang and Y.-Y. Chang, "A new network management system with ontology-supported multi-agent techniques," in *Proceedings of the International Symposium on Parallel and Distributed Processing with Applications (ISPA '10)*, pp. 275–282, September 2010.

[7] H. Jeung, S. Sarni, I. Paparrizos et al., "Effective metadata management in federated sensor networks," in *Proceedings of the*

IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, SUTC 2010, 2010 IEEE International Workshop on Ubiquitous and Mobile Computing, pp. 107–114, June 2010.

[8] M. Horridge, H. Knublauch, A. Rector, and R. Stevens, "A practical guide to building OWL ontologies using the protégé-OWL plugin and CO-ODE tools," The University of Manchester, http://owl.cs.manchester.ac.uk/tutorials/protegeowltutorial/ .

[9] Y. Wang, X. Liu, and R. Ye, "Ontology evolution issues in adaptable information management systems," in *Proceedings of the International Conference on e-Business Engineering (ICEBE '08)*, pp. 753–758, October 2008.

[10] J. E. Lóopez de Vergara, A. Guerrero, V. A. Villagráa, and J. Berrocal, "Ontology based network management: study cases and lessons learned," *Computer Science Journal of Network and Systems Management*, vol. 17, no. 3, pp. 234–254, 2009.

[11] G. Yoo and E. Lee, "Self-Healing Methodology in Ubiquitous Sensor Network," *International Journal of Advanced Science and Technology*, vol. 3, pp. 9–18, 2009.

[12] N. Ahmad, N. Riaz, and M. Hussain, "Ad hoc wireless sensor network architecture for disaster survivor detection," *International Journal of Advanced Science and Technology*, vol. 34, pp. 9–16, 2011.

[13] R. Sheikhpour and S. Jabbehdari, "A two-level cluster based routing protocol for wireless sensor networks," *International Journal of Advanced Science and Technology*, vol. 45, pp. 19–30, 2012.

[14] Z. Xu, Y. Yin, J. Wang J, and U. - Kim, "An energy-efficient clustering algorithm in wireless sensor networks with multiple sinks," *International Journal of Control and Automation*, vol. 5, no. 4, pp. 131–142, 2012.

[15] C. Goodwind and D. J. Russomanno, "An ontology-based sensor network prototype environment," in *Proceedings of the 5th International Conference on Information Processing in Sensor Network*, pp. 1–2, April 2006.

[16] H. Jeung, S. Sarni, I. Paparrizos et al., "Effective metadata management in federated sensor networks," in *Proceedings of the IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing*, pp. 107–114, June 2010.