유비쿼터스 서비스를 위한 서비스 지향 센서 네트워크 온톨로지

김정희* · 권 훈* · 김도현** · 곽호영**

Service-Oriented Wireless Sensor Networks Ontology for Ubiquitous Services

Jeong-Hee Kim* · Hoon Kwon* · Do-Hyeun Kim** · Ho-Young Kwak**

이 논문은 2007년도 산업자원부 연구비를 지원받았

요 약

본 논문은 미래의 유비쿼터스 컴퓨팅에서 지식 데이터베이스로 활용될 수 있는 서비스 지향 무선센서 네트워크 온톨로지 모델을 설계한다. 기존의 접근방식과 달리 본 논문은 사용자 중심의 서비스가 가능하도록 서비스 항목에 초점을 두고, 새로운 서비스 클래스(ServiceProperty, LocationProperty, PhysicalProperty)들과 이들에 대한 속성, 그리고 제한사항들을 정의하였다. 또한 기존 온톨로지(GML, SWE, SensorML, SUMO, OntoSensor)들과의 병합은 서비스 항목들만을 고려하여 OWL의 오브젝트 프로퍼티인 "equivalentClass"을 사용하여 처리하였다. 제안된 모델의 설계와 타당성 검증은 각각 Protege 3.3.1과 추론 도구인 Racer 1.9.0을 이용하였으며, SPARQL 질의어를 사용하여 기존 온톨로지내의 속성과 관계없이 새롭게 정의된 속성을 사용한 서비스 검색 결과를 제시하였다.

ABSTRACT

This paper designs a service-oriented wireless sensor network ontology model which can be used as a knowledge base in future ubiquitous computing. In contrast to legacy approaches, this paper defines the new service classes (ServiceProperty, LocationProperty, and PhysicalProperty), as well as their properties and constraints that enable the service-oriented service based on service items. The service item merging between the proposed model and the legacy ontology was processed using the "equivalentClass" object property of OWL. The Protégé 3.3.1 and RACER 1.9.0 inference tools were used for the validation and consistency check of the proposed ontology model, respectively, and the results of service query was applied to the newly defined property in SPARQL language without reference to the properties of legacy ontology.

키워드

Wireless Sensor Networks, WSNs, Ontology, Service-Oriented, USN, Ubiquitous Service, Semantics Web

* 제주대학교 유비쿼터스 컨버전스 사업단
** 제주대학교 통신컴퓨터공학부

접수일자 2008. 03. 25

Ⅰ. INTRODUCTION

Sensor networks are dense wired or wireless networks for collecting and disseminating environmental data. They consist of a large number of sensor nodes that are connected to central processing nodes called gateways. These networks are characterized by three main features.

First, they are highly dense so that hundreds or thousands nodes may be deployed in limited geographical areas. These nodes return huge amount of data that must be efficiently searched to answer user queries. Unfortunately, classical information retrieval techniques showed poor performance in searching sensor networks data as they return many false positives / negatives.

Second, many of the captured data are analogous in nature making the chance of finding a specific term quite good. Most sensors are characterized by similar calibration mechanisms that can be described using different terms. String matching search techniques may not retrieve all relevant data because different words / terms were used that did not match directly the term. This compromises the performance of the search engine. A big improvement in search engine performance could be achieved if these relationships are captured and utilized, and this is exactly what ontology can do. This was demonstrated in some recent work on the use of process ontologies [1, 2, 3, 4, 5] that showed an increase in the precision of service discovery queries when semantic representations were used over syntactic representations.

Third, high filtering must be required when either the user makes full use of this information or the provider offers the user this information because legacy sensor ontology has been designed to manage the sensor networks resources under focus of physical approach. However, first of all, the user will be expected that they much prefer property information (temperature, humidity, pressure, and so on) to physical information of sensor networks in future ubiquitous environments [6]. And the rest, they may need the location property joined with property information.

Therefore, this paper designs the service-oriented sensor ontology model which can be used as knowledge base in future ubiquitous computing. In different to legacy approaches, we defined the new service classes (ServiceProperty, LocationProperty and PhysicalProperty), its properties and constraints to enable the service-oriented service based on service items. Merging related to service item between proposed model and legacy ontology (Geography Markup Language (GML), Sensor Web Enablement (SWE), SensorML and Suggested Upper Merged Ontology (SUMO) and OntoSensor) also processed using the "equivalentClass" object property of OWL. Even if service-oriented service data focused on sensing data has differences with Open Geospatial Consortium (OGC), SUMO, OntoSensor, ISO 19115 and IEEE 1451, it has compatibility with them under semantic technologies. We also presented the validation and consistency check of the proposed ontology model using Protégé 3.3.1 and RACER 1.9.0 inference tool, respectively, and indicated the results of service query which was applied the new defined properties in SPARQL language without reference to properties of legacy ontology.

The rest of this paper is organized as follows. Section 2 introduces the background of ontology design and section 3 highlights related work in semantic sensors data. Section 4 describes the initial taxonomy for sensors data and details the development stages. Section 5 presents the validation and consistency check of the proposed ontology. Finally, we conclude in section 6 by summarizing the preliminary validation of the proposed ontology and recommending directions for future work.

Ⅱ. BACKGROUND

The term ontology can be defined as "an explicit formal specification of a shared conceptualization"[7]. An ontology comprises three components: first, classes or concepts that may have subclasses to represent more specific concepts than in super-classes, second, properties or relationships that describe various features and properties of the concepts, also named slots or roles, third, restrictions on slots (facets) that are superimposed on the defined classes and / or properties to define allowed values (domain and range).

Individuals can be defined simply as instances of the classes and properties. The ontology together with a set of instances of classes and slots constitute the knowledge base. Reference [3] presents a detailed description of the development stages of ontologies. Also, many advantages of ontology design are explained in [8], including: sharing common understanding of the structure of information among people or software agents, enabling reuse of domain knowledge, making domain assumptions explicit, separating the domain knowledge from the operational knowledge, and analyzing domain knowledge.

On the other hand, there exist several arguments and challenges, among which are the lack of an agreed-upon taxonomy and quantitative evaluation procedures.

Ⅲ. RELATED WORK

Despite the amount of research devoted to ontology design and development, very little attention has been paid to semantic representation of sensor networks data. The idea of using ontology-driven information system for sensor networks is not entirely new. The work in [3] presents an attempt to capture the most important features of a sensor node that describes its functionality and its current state. The ontology describes the main components of a sensor node such as processor CPU and memory, power supply, and radio and sensor modules.

A step further in ontology-based sensor nodes is presented in [9] and [10]. The researchers in [9] define an ontology that integrates high level features that characterizes sensor networks for customizing routing behavior. The proposed ontology describes the network topology and settings, sensor description, and data flow.

Again, there is no mention of sensor data. Subsequent work like [11] is an effort in the direction of facilitating semantic-service oriented sensor information systems. The notion of ontology used in this research is to capture the information about physical entities that sensors sense and their relationships.

The IEEE 1451 is a family of proposed standards that provide a single generic interface between a transducer and external network protocol in use [12]. The IEEE 1451 standard family uses Transducer Electronic Data Sheet (TEDS) to capture sensor characteristics, such as transducer identification, calibration, correction data, and manufacturer related information. Consequently, much of the knowledge captured by the ontology describes the widely accepted IEEE 1451 TEDS templates.

On the other hand, SensorML imports the OGC's GML and SWE, and has various terms, but it is not ontology based but rather XML-based. So inference based on ontology is impossible [13].

More than these, relation researches that take advantage of semantic web technology are developing network Ontology at wireless sensor network on focus of hardware resource or resource management. But the future of this paper, presented in a ubiquitous service-oriented environment, the ontology of the building is required.

Ⅳ. THE PROPOSED ONTOLOGY

4.1 Design concept

It is possible for user participation in networking at anytime and anywhere through portable devices under ubiquitous environments like Figure 1. Therefore, the user is able to get the information immediately which is provided from sensor nodes in WSNs, and we can think the following questions at this time.

- Is providing information from sensor nodes what either sensor node ID like "0001" or value of temperature like "15"?
- And also is any additional available? If so, Is it physical information?
- Where does the user utilize the physical information?

Generally, user expects the value of temperature like "15" for above questions and hopes that it will be displayed on portable device in addition. Therefore, sensor ontology of this paper focused on service-oriented approach like Figure 1.

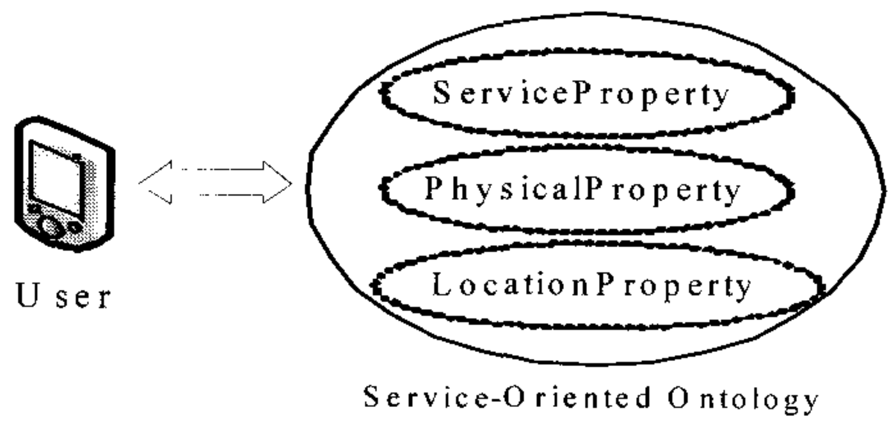


그림 1. Legacy 기반 서비스 지향 온톨로지
Fig. 1 Service-oriented ontology based on legacy sensor ontology

4.2 Service-oriented properties

Sensing data is different to sensor data. The meaning of the service-oriented property is sensing data which is expected by the user.

For example, the temperature value "15" which is captured by a temperature sensor in the field, is service-oriented property, and CPU type, memory size, sensing mechanism, battery power, actuator and transducer are sensor data as legacy ontology in Figure 1. As recent researches, the approach of sensor ontology has designed to focus on physical sensor information like Figure 2 [5].

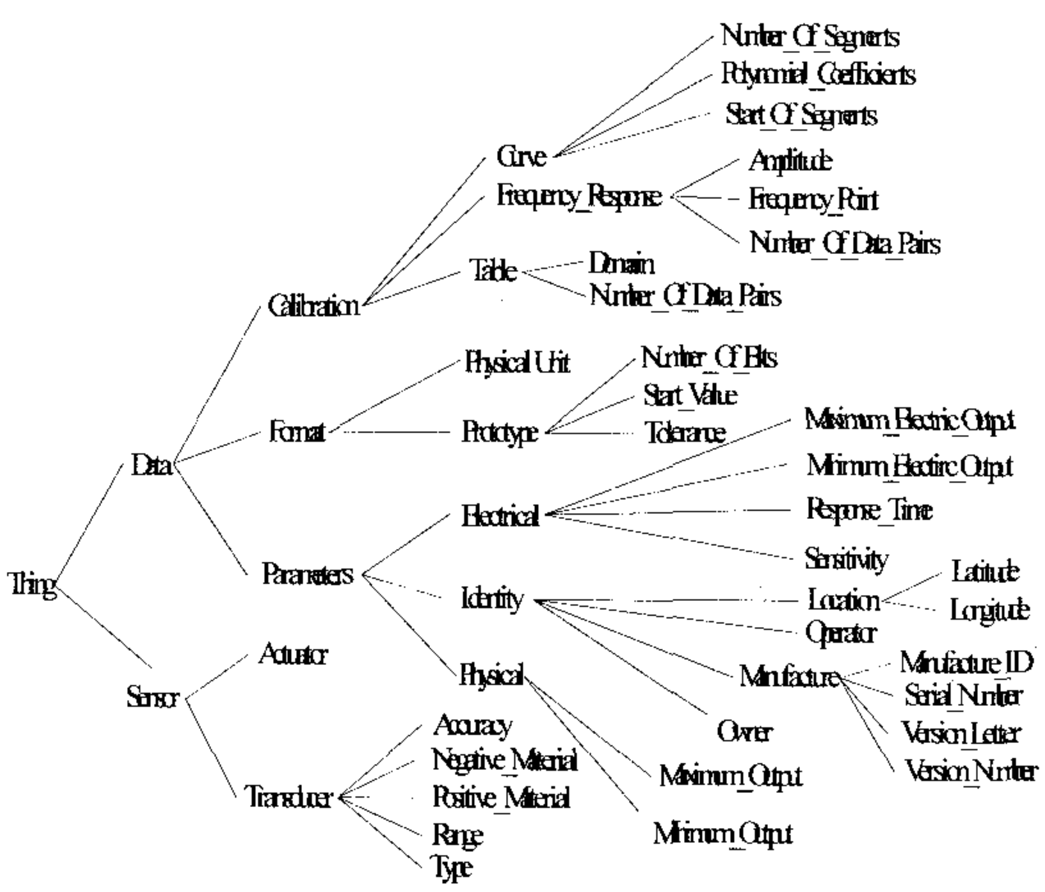


그림 2. Sensor hierarchy ontology 분류
Fig. 2 Taxonomy for the sensor hierarchy ontology

Therefore, we defines service-oriented ontology (ServiceProperty, LocationProperty, and PhysicalProperty) using service-oriented properties like Figure 1. In addition, equivalent classes about the service-oriented classes are derived from legacy sensor ontology including great many sensor and property data, provides the compatibility with service-oriented classes for service-oriented service.

4.3 Ontology design

The ontology development follows an evolving prototype life cycle rather than a waterfall or an iterative one. This implies that one can go back from one stage to another stage in the development process as long as the ontology does not satisfy or meet all the desired requirements. The usually accepted stages through which ontology is built are: collecting vocabulary commonly used, identifying an initial taxonomy, adding restrictions and axioms, consistency checking, incremental modifications, and evaluation [14].

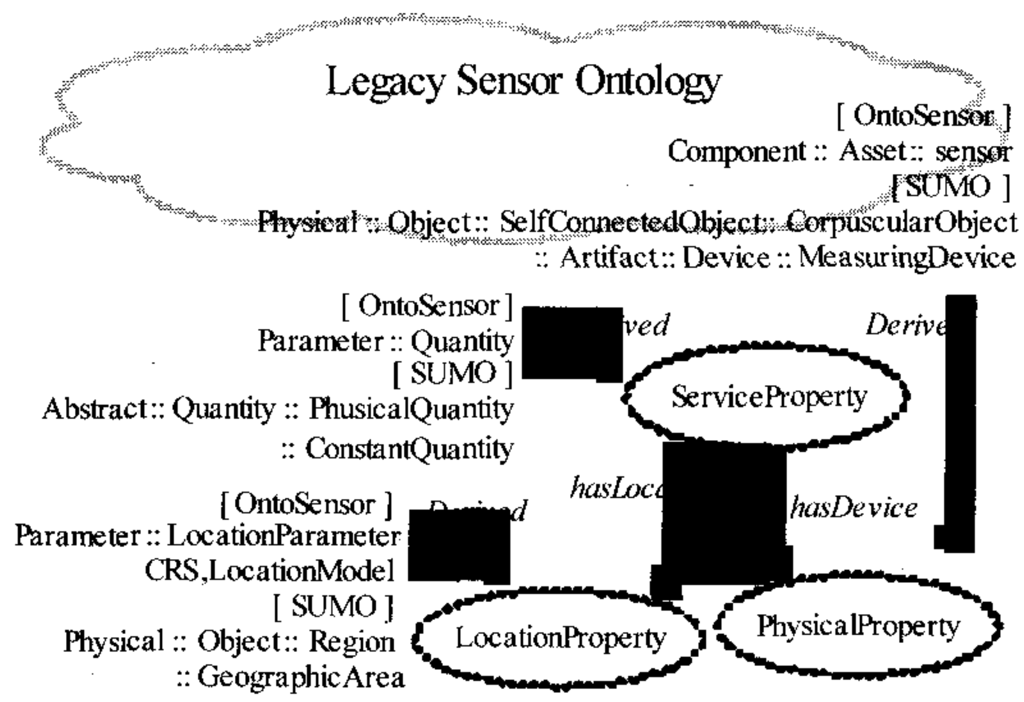


그림 3. 제안된 온톨로지 모델의 용어들
Fig. 3 Terms for proposed ontology model

Our main sources for collecting commonly used terms in service domain are the GML, SWO, SensorML, SUMO and OntoSensor. We also designed the serviced-oriented sensor ontology based on subsection 4.1 and subsection 4.2 like Figure 3.

We imported the GML, SWO, SensorML and OntoSensor terms based on GOC and SUMO, respectively, defined the serviceProperty, LocationProperty and PhysicalProperty classes and linked the relationships with legacy ontology using "owl:equivalentClass" property for

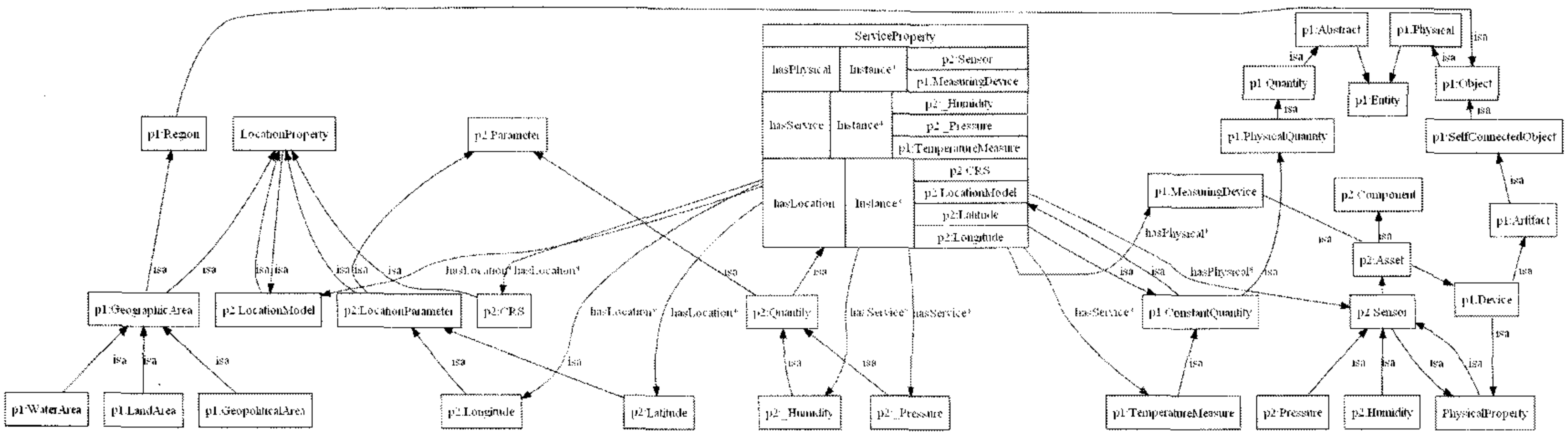


그림 4. 제안된 온톨로지 모델
Fig. 4 Proposed ontology model

service-oriented service. "owl:equivalentClass" is a built-in property that links a class description to another class description. The meaning of such a class axiom is that the two class descriptions involved have the same class extension (i.e., both class extensions contain exactly the same set of individuals). And "Parameter::Quantity" implies that "Quantity" class is the sub class of the "Parameter" class in Figure 3.

표 1. 속성 리스트
Table. 1 Properties list

Domain	Property Name	Range
Service Property	hasLocation	LocationProperty
	hasDevice	PhysicalProperty
	hasService	_Humidity, _Pressure, _Temperature, etc.
Location Property	hasLongitude	p2:Longitude
	hasLatitude	p2:Latitude
	hasCRS	p2:CRS
	hasGeopoliticalArea	p1:GeopoliticalArea
	hasWaterArea	p1:WaterArea
	hasLandArea	p1:LandArea
Physical Property	hasSensor	p2:Pressure, p2:Humidity
	hasMeasuringDevice	p1:MeasuringDevice

The next step is to take the list of concepts as described by the identified terms and form the initial class taxonomy. This implies looking at whether a concept is a sub-concept of another one or not. Figure 4 shows our initial taxonomy after adding design concepts. Concepts were added one at a time, structuring the taxonomy as needed to accommodate each concept. Notice that the links from classes to their sub-classes represent properties that are listed in Table 1. For instance, the link from LocationProperty class to p2:Longitude class represents the property "hasLongitude".

4.4 Properties and Constraints

Relationships among classes are usually referred to as properties (for further description of properties classifications, refer to [14]). A property links an individual from its domain to an individual of its range. For example the has-Service property links the ServiceProperty class to either Humidity or Pressure, or Temperature classes.

V. IMPLEMENTATION AND VALIDATION

5.1 Ontology for preparation

For in this Section, we present our technical judge of the designed ontology by performing the tests mentioned in Section 1. The experimental evaluation is validating the ontology (checking for logical inconsistencies) and querying the services.

Imported ontology (SUMO and OntoSensor) for our service-oriented ontology showed in Figure 5.

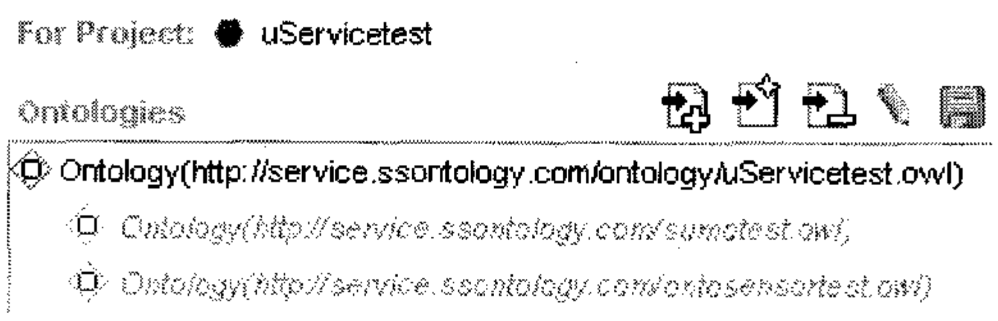


그림 5. 포함된 온톨로지
Fig. 5 Imported ontology

5.2 Protégé and RACER

To implement the constructed taxonomy an ontology development tool, called Protégé [11] is used to build and edit the ontology. The knowledge representation language for modeling the various data types of sensor data is OWL-DL. We manually add classes to the ontology by creating ServiceProperty, LocationProperty and Physical Property classes and all their sub-classes, and also linking relationships with legacy ontology using created object properties and "owl:equivalentClass" property. The constructed class hierarchy is called the manually created classification hierarchy and is shown in Figure 6.

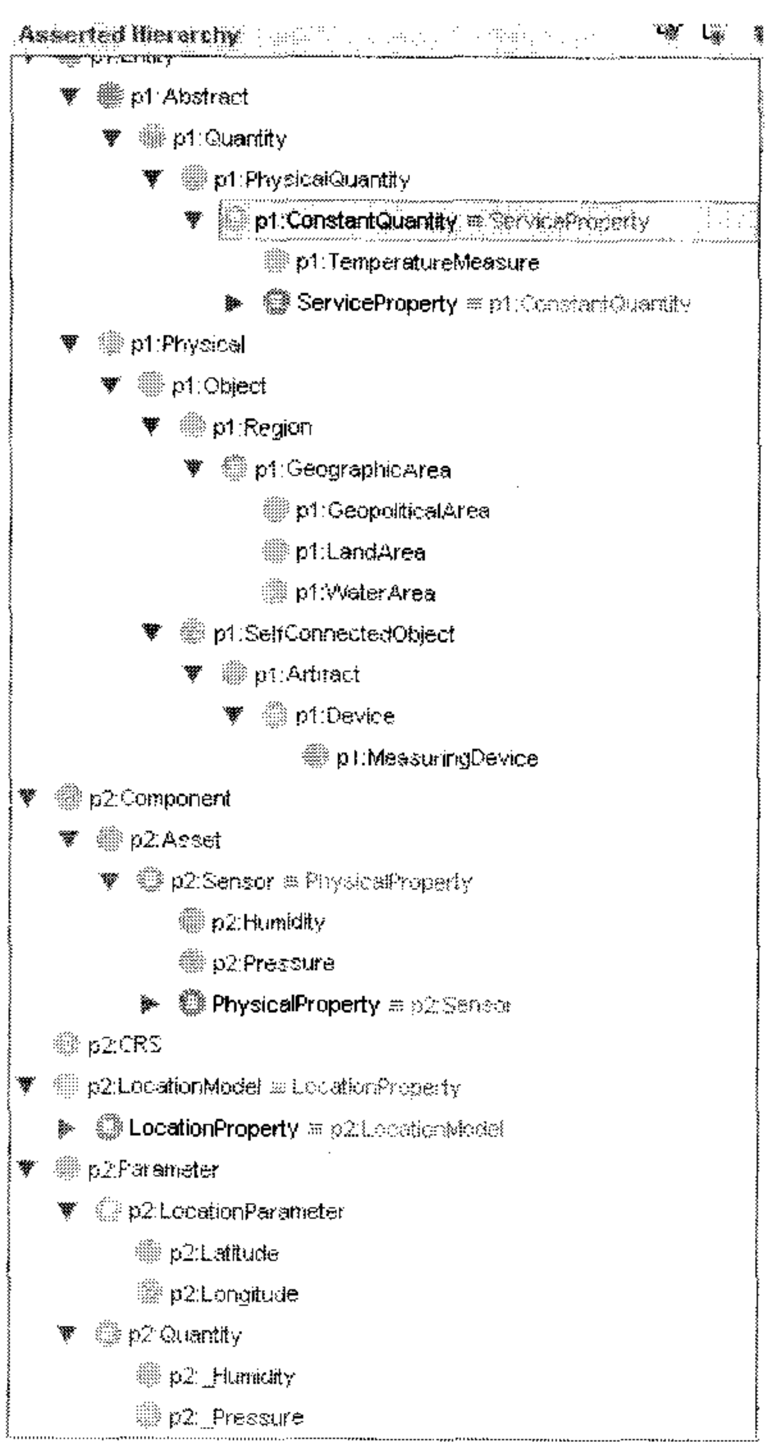


그림 6. 제안된 온톨로지 계층구조
Fig. 6 Hierarchy of the proposed ontology

As a validation tool, we used RacerPro because of its strong reasoning capabilities and interoperability with Protégé. The manually created class hierarchy is fed to RacerPro whose main responsibility is to automatically compute the inferred class hierarchy (called asserted ontology) based on the description of classes and relationships. To perform the subsumption test, both Protégé and RacerPro should be up and running.

Having started RACER, the ontology now can be sent to the reasoner to automatically compute the classification hierarchy (called taxonomy classification), and also to check the logical consistency of the ontology. We should distinguish two ontologies: the manually constructed class hierarchy (developed according to previous section) and the Figure 7. Logical consistency checking automatically computed one, both must be identical if the subsumption classification is error free.

On the other hand, if the ontology has inconsistencies, the logical consistency check test must be able to detect them. A snapshot of Protégé is shown in Figure 7 where the inconsistent class does not appear. If there is the inconsistent class, then the bullet of the class is marked in red.

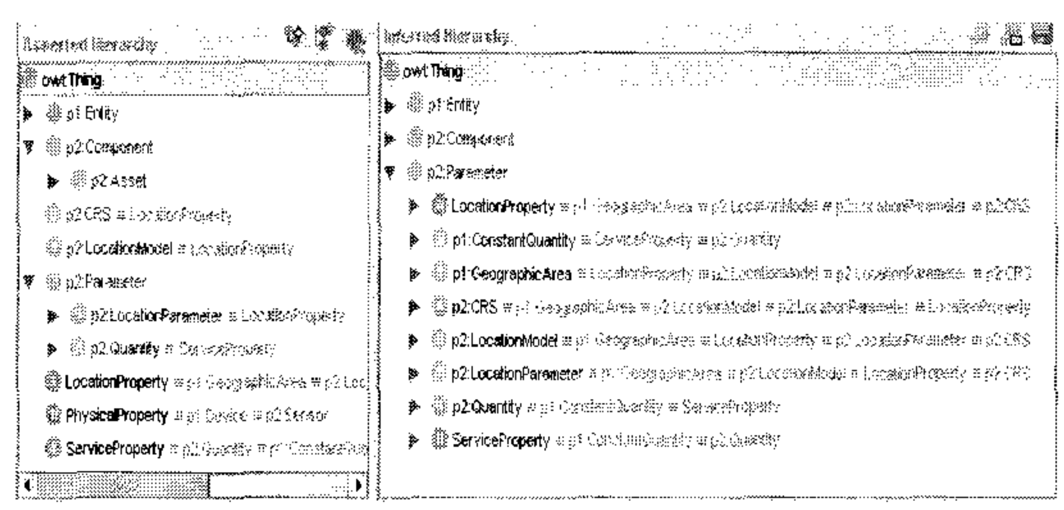


그림 7. 논리적인 일관성(모순성) 검사
Fig. 7 Logical consistency checking

5.3 Query results

As a query language, we used to SPARQL query language which is provided and plugged-in to Protégé. SPARQL can be used to express queries across diverse data sources, whether the data is stored natively as RDF or viewed as RDF via middleware. It also contains capabilities for querying required and optional graph patterns along with their conjunctions and disjunctions, and also supports extensible value testing and constraining queries by source

RDF graph. The results of SPARQL queries can be results sets or RDF graphs.

Figure 8 showed the query results about temporary service query related with equivalent class. And we have following question: "Is there anything service at current location?" when we are somewhere.

표 2. SPARQL 질의문
Table. 2 SPARQL query statements

```
prefix
sn:http://service.sensorontology.com/ontology/uServicetest.owl#
//…..using object properties ….
Select ?service ?property ?individuals ?location ?value
Where { ?service ?property ?individuals .
        ?service sn:hasService ?individuals .
        ?individuals ?type ?value .
        filter (!(?type = rdf:type))    }
```

In here, current location, based on longitude and latitude, is the same location as its own portable device. Therefore, service query can be makes as Table 2, and its results were showed like Figure 8.

Results

service	property	individuals	location	value
service_1	hasService	p2:Humidity_3	p1:CRS_4	Humidity
service_1	hasService	p2:Humidity_3	p1:CRS_4	80
service_1	hasService	p2:Humidity_3	p1:CRS_4	%
service_4	hasService	p2:Humidity_4	p1:CRS_5	Humidity
service_4	hasService	p2:Humidity_4	p1:CRS_5	70
service_4	hasService	p2:Humidity_4	p1:CRS_5	%
service_3	hasService	p2:Temperature_6	p1:CRS_6	Temperature
service_3	hasService	p2:Temperature_6	p1:CRS_6	20
service_3	hasService	p2:Temperature_6	p1:CRS_6	centigrade
service_6	hasService	p2:Temperature_7	p1:CRS_4	Temperature
service_6	hasService	p2:Temperature_7	p1:CRS_4	25
service_6	hasService	p2:Temperature_7	p1:CRS_4	centigrade

그림 8. 서비스지향 기반 질의 결과
Fig. 8 Temporary query results based on service-oriented

Ⅵ. CONCLUSION AND FUTURE WORK

The semantic representation of wireless sensor networks data is an exciting vision that enables structured information to be interpreted unambiguously. Precise interpretation is a necessary prerequisite for automatic search, retrieval, and processing of sensor data and sensing data. For ubiquitous computing, this paper is the first attempt to define ontology for describing concepts and relationships of the wireless sensor networks based on service-oriented services.

The benefits of our work are to classify the property and improve the approach about the sensor ontology for service-oriented service in the future ubiquitous computing.

As for future work, we are considering extending the ontology so that it describes the entire available property base on user preference; including URL and UFID location property. Moreover, we plan to test the effectiveness of the ontology approach by quantitatively measuring the improvements in the property and recall rates of a search engine when utilizing the ontology against traditional string-based searching approaches. This effort will be a further step in the direction towards enabling ubiquitous services to access and process sensing and sensors data.

ACKNOWLEDGMENT

This research was supported by the regional technology innovation program of the Ministry of Commerce, Industry and Energy, Republic of Korea.

REFERENCE

[1] M. Klien and A. Bernstein, "Searching for services on the Semantic Web using process ontologies," 1st Semantic Web Working Symposium (SWWS-1), Stanford, CA, 2001.

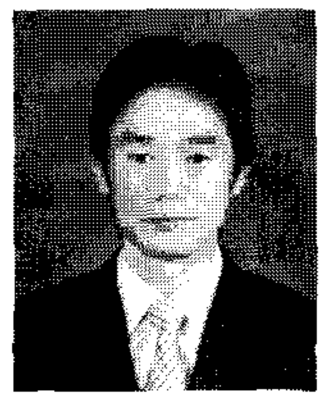
[2] D.J. Russomanno, C. Kothari and O. Thomas, "Building a Sensor Ontology: A Practical Approach Leveraging ISO and OGC Models," The 2005 International Conference on Artificial Intelligence, Las Vegas, NV, 2005, pp. 637-643.

[3] S. Avancha, C. Patel, and A. Joshi, "Ontology-driven Adaptive Sensor Networks,"First Annual International Conference on Mobile and Ubiquitous Systems, Networking and Services, 2004, pp. 194-202.

[4] Mohamad Eid, Ramiro Liscano, Abdulmotaleb El Saddik, "A Novel Ontology for Sensor Networks Data," IEEE International Conference on Computational Intelligence for Measurement Systems and Applications La Coruna - Spain, 12-14 July 2006

[5] Mohamad Eid, Ramiro Liscano, Abdulmotaleb El Saddik, "A Universal Ontology for Sensor Networks Data,"IEEE International Conference on Computational Intelligence for Measurement Systems and Applications Ostuni- Italy, 27-29 June 2007
[6] Mark Wiser, "The computer of the 21st century," Scientific American, 1991.
[7] Nicola Guarino, Massimiliano Carrara, Pierdaniele Giaretta, "An Ontology of Meta-Level Categories," Proceedings of the Fourth International Conference (KR94), Morgan Kaufmann, San Mateo, CA, 1994.
[8] Natalya F. Noy and Deborah L. McGuinness, "Ontology Development 101: A Guide to Creating Your First Ontology," Stanford Knowledge Systems Laboratory Technical Report KSL-01-05 and Stanford Medical Informatics Technical Report SMI-2001-0880, March 2001.
[9] R. Jurdak, C. V. Lopes, and P. Baldi, "A Framework for Modeling Sensor Networks,"Proceedings of the Building Software for Pervasive Computing Workshop at OOPSLAï04, Vancouver, Canada 2004.
[10] George V. Cybenko, Guofei Jiang, and Wayne Chung, "Semantic Agent Technologies for Tactical Sensor Networks," Proceedings of the SPIE Conference on Unattended Ground Sensor Technologies and Applications V, pages 311-320, Orlando, FL, September, 2003. SPIE.
[11] Jie Liu and Feng Zhao, "Towards Semantic Services for Sensor-Rich Information Systems,"Proceedings of the 2nd IEEE/CreateNet International Workshop on Broadband Advanced Sensor Networks (Basenets 2005), Boston, MA, Oct. 3, 2005.
[12] IEEE P1451Project official web site, http://ieee1451.nist.gov, accessed in 27, Sep., 2007.
[13] M. Botts, A. Robin, "OpenGIS Sensor Model Language (sensorML) Implementation Specification," http://www.opengeospatial.org, accessed Oct. 12, 2007.
[14] M. Horridge, H. Knublauch, A. Rector, R. Stevens, and C. Wroe, "A Practical Guide to Building OWL Ontologies Using the Protégé-OWL, Plugin and COODE Tools, Edition 1.0," Tutorial, http://www.coode.org, accessed Oct. 10, 2007.

저자소개



김정희(Jeong-Hee Kim)

1994년 제주대학교 정보공학과 (학사)
1997년 제주대학교 대학원 정보공학과 (석사)
2005년 제주대학교 대학원 정보공학과 (박사)
2000～현재 제주대학교 시간강사
2007～현재 유비쿼터스컨버전스사업단연구원
※관심분야 : Internet Application, WSNs, Semantic Web 등

권 훈(Hoon Kwon)

2003년 제주대학교 해양생물공학 (학사)
2005년 제주대학교 대학원 컴퓨터공학과 (석사)
2006～ 현재 제주대학교 대학원 컴퓨터공학과 (박사과정)
2004～현재 제주한라대학 시간강사
2007～현재 유비쿼터스컨버전스사업단연구원
※관심분야 : XML, WSNs, RFID 등

김도현(Do-Hyeun Kim)

1988년 경북대학교 전자공학과 (학사)
1990년 경북대학교 대학원 전자공학과 (석사)
2000년 경북대학교 대학원 전자공학과 (박사)
1990～1995 국방과학연구소 연구원
1999～2004 천안대학교 정보통신학부 조교수
2004～현재 제주대학교 통신컴퓨터공학부 조교수
※관심분야 : Sensor Networks, 이동성 관리, 유비쿼터스 서비스 등

곽호영(Ho-Young Kwak)

1983년 홍익대학교 전자계산학과 (학사)
1985년 홍익대학교 대학원 전자계산학과 (석사)
1991년 홍익대학교 대학원 전자계산학과 (박사)
1990～현재 제주대학교 통신컴퓨터공학부 교수
2007～현재 유비쿼터스컨버전스사업단장
※관심분야 : 객체지향 프로그래밍, 프로그래밍 언어론, Internet Application 등