Ontology Context Model for Context-Aware Learning Service in Ubiquitous Learning Environments

Myoung-woo Hong and Dae-jea Cho

***Abstract*—**In the vision of ubiquitous computing, computer systems will be seamlessly integrated into our everyday life, anticipating our needs and providing relevant services and information to us in an anytime any where fashion. The vision of ubiquitous computing makes an impact on the appearance of ubiquitous learning and presents an advanced direction of learning system. This paper describes the conceptual architecture and ontology-based context model for providing context-aware learning services in ubiquitous learning environments. If ubiquitous computing environments have been realized, it can lead to ubiquitous learning. In ubiquitous learning environments, various embedded computational devices will be pervasive and interoperable across the network for supporting the learning, so users may utilize these devices anytime anywhere. An important next step for ubiquitous learning is the introduction of context-aware learning service that employing knowledge and reasoning to understand the local context and share this information in support of intelligent learning services. In this paper, we assume that a school was equipped with ubiquitous computing environments. We present context-aware manager based architecture to support user-centric ubiquitous learning services, and describe ontology based context model for intelligent school spaces.

***Keywords*—**Context aware learning, Ontology context model, Rule based reasoning, Ubiquitous learning.

I. INTRODUCTION

WE describe the conceptual architecture and ontology-based context model for providing context-aware learning services in ubiquitous learning environments. The majority of existing learning systems are implemented either with client-server architecture or are centralized server based. The client-server and centralized server approaches are metaphors of student-teacher and repository centric which reflect real world learning scenarios in which teachers act as the content producers while students act as the content consumers [1], [2]. Recently, the need of a mobile environment where the contents are accessed through various mobile devices has been increased in the learning systems. Especially, the moving from the internet environments to ubiquitous environments is to be a natural trend in the educational environments. The vision of ubiquitous computing technology means to provide invisible computing environments so that a user can utilize services at any time and place [3]. Characteristic of this vision is the use of wireless networking, sensor rich environments, mobile and wearable computing, and intelligent human0computer interfaces [4]. If the vision was to be realized, users could utilize learning services while moving because, in ubiquitous learning environments, various embedded computers, to support learning activities, are provided everywhere and these tools are connected to one another through the network. Also, by introducing context-aware technology, intelligent learning services can be provided on the basis of the context [5]. In order to construct user-centric, ubiquitous, computing environments to provide intelligent learning services like this, it is necessary to design both architecture which can support context-aware services and a context model which can describe context information. Thus, on the assumption that ubiquitous computing environments are constructed at every school, this study suggests conceptual architecture led by a context-aware manager that provides learning services applying context-aware technology. Besides, this study designs an ontology-based context model which describes context information through OWL-DL (Web Ontology Language-Description Logic).

The rest of this paper is organized as follows. The second chapter introduces studies about context-aware computing and context models based on ontology as a related work. The third chapter suggests a conceptual model of ubiquitous learning architecture to support context-aware learning services. The fourth chapter discusses an ontology model design concerning ubiquitous computing environments and the possibility to apply this model. The final chapter shows conclusions and future work.

Manuscript received March 28, 2008: Revised version received July 26, 2008.

II. RELATED WORKS

A. Context and Context-Aware Computing

Context is defined as information used for identifying the status of an entity. An entity can become a person, a place, or physical or computing object [6]. This context includes a user and application, and reflects the relationship of interactions between a user and an application. Based on this, entity in ubiquitous learning environments means users, position, activity, computing device of the environments which a student belongs to and information figures of the entity while changes of information are referred to as context.

Schilit defines that context-aware computing is not only adaptable according to a place, a person and a group of objects, but also it is a software which can accept changes of objects as time goes by [7]. Sometimes, when relevant information or services related with a user's work is provided for a user, context is used. A system with these capabilities can examine the computing environment and react to changes to the environment. Recent studies define these cases as a context aware system. As the feasibility of context-aware computing in ubiquitous computing environments has been getting higher, the related studies have progressed. The Active Badge Project which is about phone switch applications developed a system which locates people through Active Badges attached to people and connects to the nearest phones to where they are [8]. This project also examines alternative location techniques, system design issues and applications, particularly relating to telephone call routing, Cooltown Project of HP aimed to construct a web which has people, things and spaces online and suggested a web-based context model in which each object has the corresponding web description [9]. Enabling the automatic discovery of URIs from our physical surroundings, and using localized web servers for directories, they create location-aware but ubiquitous systems. On top of this infrastructure they leverage device connectivity to support communication services. ContextToolkit, by using an object-oriented method, has been providing framework and many reusable components in order to support quick manufacture of prototypes of sensor-based context aware applications [10]. They illustrated the usefulness of the conceptual framework by describing a number of context-aware applications that have been prototyped using the Context Toolkit. They also demonstrated how such a framework can support the investigation of important research challenges in the area of context-aware computing.

B. *Ontology and Ontology-based Context Model*

According to Semantic Web led by W3C (World Wide Web Consortium), ontology is a way to describe knowledge systematically; a typical and explicit specification about concepts and conceptualization, that is, it also defines concepts and relations required to describe meaning and information [11], [12]. The term ontology has a long history in philosophy, in which it refers to the subject of existence. In the context of the knowledge management, ontology is referred as the shared understanding of some domains, which is often conceived as a set of entities, relations, functions, axioms and instances [13]. Through this ontology, the vocabularies of a specific domain can be defined in a common way, and thus knowledge can be shared [14]. Typical languages to describe ontology are as follows: XML (eXtensible Markup Language), XML Schema [15], RDF (Resource Description Framework) [16], RDF Schema [17], DAML+OIL (DARPA Agent Markup Language + Ontology Interface Language) [18], OWL [19], etc. Among these languages, OWL which has been developed most recently is an ontology language which defines classes and properties and also their relationships more clearly, acquiring consistency of concepts based on DAML+OIL. Further OWL is divided into OWL-Lite, OWL-DL and OWL-Full according to its ability to describe. In particular, OWL-DL is based on description logics, which can be judged as true or false by first order predicate logic and also can have inference ability in itself. In our paper, we present a context model based on ontology using OWL-DL to support various tasks in our context-aware architecture.

Research on ontology-based context models, which are able to share context information and reason context by defining contexts using these ontology languages, have been conducted. Ranganathan, et al. suggested the infrastructure that supports collection of context information from other sensors and supports delivery of appropriate context information through ubiquitous computing application [20]. To describe different possible contexts and support the complicated reasoning of context, they described context with the first order predicate logic made by DAML + OIL. This context model makes it possible to draw new context from the sensed context. Wang, et al. created context ontology which is known as CONON [21]. It supports reasoning to find and correct inconsistent context information and reasoning as a means to draw farther context information. Gu, et al. suggested an ontology-based model in intelligent environments as well as service-oriented context aware, middleware and architectures through OWL [22]. In this architecture, they deal with semantic context descriptions, context reasoning, knowledge sharing, context classifications, context dependency, and so on. This context model has an advantage that it makes possible the reasoning of different contexts by using the formal analysis of domain knowledge such as the first order logic and temporal logic. CoBrA project suggested the agent-based infrastructure for context representation, knowledge sharing, user privacy control and developed COBRA-ONT, the ontology of pervasive computing environments supporting context awareness [23]. Especially, it suggested the methods to find the position of a person and support a context aware service in the smart room. CoBrA employs reasoning for detecting and resolving inconsistent context information, evaluating privacy policies and inferring additional context information based on properties such as temporal and spatial relations.

III. Context Aware-Based Architecture in Ubiquitous Learning Environments

In ubiquitous learning environments, a conceptual ubiquitous learning architecture like figure 1 to support services considering different contexts is designed and this is called CALA (Context Aware Learning Architecture). CALA architecture consists of a personal agent, a computing entity, a physical sensor and an activity agent centering on a context-aware manager in the school space. This architecture supposes that a learning space is an intelligent ubiquitous school space where a ubiquitous computing environment is realized. Accordingly, all components of this architecture are connected to enable both wired and unwired networking, and exchange and share sensing information as well as context information through WiFi, Bluetooth, and Ethernet. Also teachers, students and administrative staff acting in a ubiquitous learning space wear a smart tag and a mobile device to be sensed by a physical sensor.

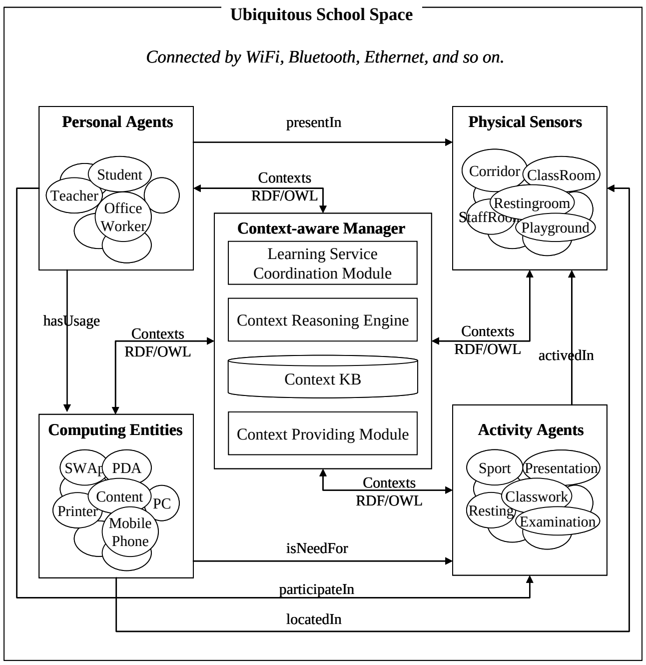

Fig. 1 an outline of CALA, the architecture to support a context aware-based learning service

A personal agent is software to be operated in an individually owned-device and participates in learning activities and routine activities representative of students, teachers and administrative staff. A personal agent updates and manages owner's ID, name, e-mail, address, and so on. A personal agent also provides a context-aware manager with this information, describes it and keeps it as context information, and enables an activity agent, a computing entity and other facilities to share context information.

A computing entity includes mobile devices such as a PDA and a cellular phone, devices occupying a space such as a printer or a projector, software (for example, power points, editors, moving picture players, mp3 players, and so on.) as well as contents providing users with services. These are used by activity agents and people and are able to provide services through a characteristic function of devices and software. In addition, a computing entity provides a context-aware manager with not only the fixed information of specifications, functions, formats and spaces but also with the changed information related to an individual or an activity agent currently active.

An activity agent is software which controls learning activities such as lessons and examinations, and daily activities such as interviews, presentations and sports. An activity agent manages the information such as a schedule, a place, and a person and a computing entity as well as providing a context-aware manager with it. For example, a presentation agent is a sub-class of an activity agent, managing presentation activities, defining information such as time, place, a presenter, an audience, a related content and a facility of a presentation as well as transmitting it to a context-aware manager.

A physical sensor is one arranged in the ubiquitous environment such as a classroom, a corridor or a lounge. It provides a context-aware manager with the information about its location, present temperature, lighting, and noise level. Mostly, a physical sensor can be used in perceiving a person through the existence of their smart tag.

A context-aware manger consists of a context providing module, a context knowledge base, a context reasoning engine and a learning service coordination module. A context-aware manager controls a context module based on ontological reasoning, transforming a new context into a semantic place and updating a context module. In addition, it makes context information shared and provides a proper context-aware learning service by in advance applying a changing context into a rule set. Summaries about the role of the constituents as follows:

- *Context providing module: It obtains context information from various sensors, a person, an activity agent, and computing entity*
- *Context knowledge base: It preserves and shares context knowledge on behalf of the personal agent, restricted by resource, and a computing entity.*
- *Context reasoning engine: It reasons out the context, interpreting context information.*
- *Learning service coordination module: It coordinates and provides learning services based on context information through a user defined learning support rule.*

IV. Ontology Context Model about Ubiquitous Learning Environments

A. Context Information and OWL-DL

In our ontology context model, context information is expressed in first order predicate logic and context model is defined by ontology written in OWL-DL.

Context information has the form of *(subject, predicate, value)*. In the form, the subject is a subject of context, e.g. a person, a place, a computing entity or an activity. The predicate represents a property of subject, e.g., personName, locatedIn or participatedIn, etc. The value represents all values of subject or subjects, e.g., open, closed, empty, an activity, etc. Context information also can be extended to form a complex context by combining the predicate and Boolean algebra (union, intersection and complement). For example, *(ClassRoom_1_1, curtainStatus, close)* represents context information that status of ClassRoom_1_1's curtain is closed. This context information can be converted into the form of OWL-DL as a figure 2.

```
<ClassRoom rdf:ID="ClassRoom_1_1">
    <curtainStaus rdf:datatype="&xsd;string">close</curtainStaus>
</ClassRoom>
```

Fig. 2 OWL-DL representation of context information

In our ontology context model, ontology consists of Individuals, Properties, and Classes. Individuals represent objects in the domain that we are interested in. For example, figure 3 represents an individual HongYunSang written in OWL-DL. This means that HongYunSang is an individual of class Student.

```
<Student rdf:ID="HongYunSang">
```

Fig. 3 representation of context information

Properties are binary relations on individuals – i.e. properties link two individuals together. There are two main types of properties, Object properties and Datatype properties. Object properties link an individual to an individual. Datatype properties link an individual to an XML Schema Datatype value or an rdf literal. Figure 4 and 5 show examples of each type of property.

```
<owl:ObjectProperty rdf:ID="hasActivity">
   <rdfs:domain rdf:resource="#Place"/>
   <rdfs:range rdf:resource="#Activity"/>
   <owl:inverseOf rdf:resource="#activedIn"/>
</owl:ObjectProperty>
```

Fig. 4 en example of an Object property hasActivity

```
<owl:DatatypeProperty rdf:ID="placeName">
   <rdfs:domain rdf:resource="#Place"/>
   <rdfs:range rdf:resource="&xsd;string"/>
 </owl:DatatypeProperty>
```

Fig. 5 en example of a Datatype property placeName

Classes are interpreted as sets that contain individuals. They are described using formal descriptions that precisely the requirements for membership of the class. Classes may be organized into a superclass-subclass hierarchy. For example, figure 6 shows that class IndoorCampus is a subclass of class Place.

```
<owl:Class rdf:ID="IndoorCampus">
   <rdfs:subClassOf rdf:resource="#Place"/>
</owl:Class>
```

Fig. 6 Datatype property placeName

B. Proposed Ontology Context Model

We design an ontology context model, which is called CALA-ONT (Context Aware Learning Architecture ONTology). Our CALA-ONT consists of four top-level classes and sub-classes, and contains twelve main properties which describe the relations between individuals in top level class and its sub properties. An Individual of a class, a real instance of ontology, is defined only in parts needed in explaining our ontology context model.

Figure 7 shows that we comply with XML, RDF Schema and OWL as a part of the CALA-ONT model and give a definition of four top level classes.

```
<rdf:RDFxmlns="http://www.owl-ontologies.com/Ontology1197601859.owl#"
xml:base="http://www.owl-ontologies.com/Ontology1197601859.owl"
xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:owl="http://www.w3.org/2002/07/owl#">
  <owl:Class rdf:ID="Person"/>
  <owl:Class rdf:ID="Place"/>
  <owl:Class rdf:ID="ComEntity"/>
  <owl:Class rdf:ID="Activity"/>
  <owl:Class rdf:ID="Person">
    <owl:disjointWith rdf:resource="#Place"/>
    <owl:disjointWith rdf:resource="#ComEntity"/>
    <owl:disjointWith rdf:resource="#Activity"/>
  </owl:Class>
```

Fig. 7 a part of ontology expressions in CALA-ONT

The CALA-ONT model defines person (Person), places (Place), activities (Activity), and computational entities (ComEntity) in a top-level class. Figure 8 shows the relations between the classification of the CALA-ONT model and major properties through the graph.

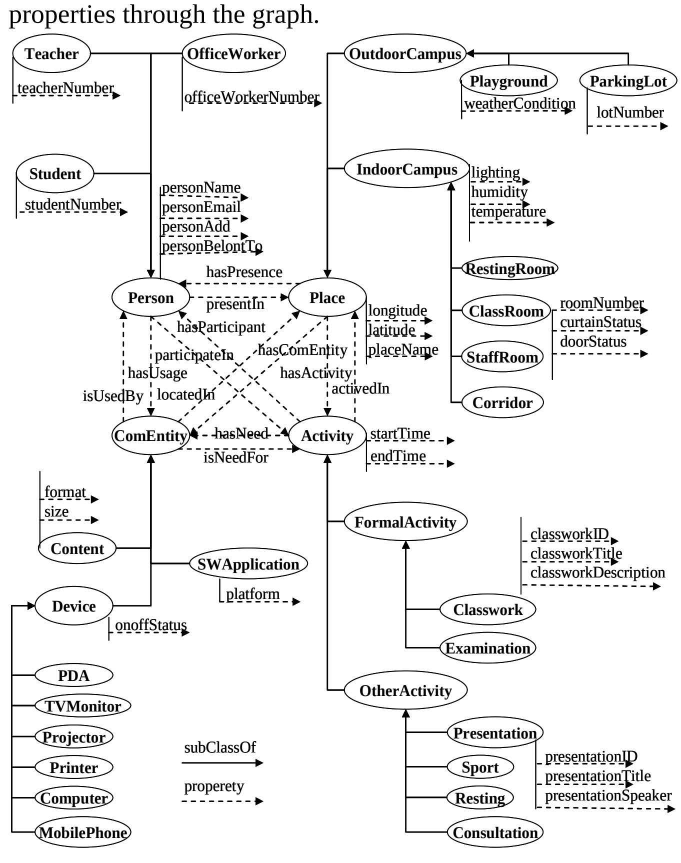

Fig. 8 graphic presentation about the classification of CALA-ONT model and its property

The class of a person, which defines the most general feature about a person, presents a student (Student), a teacher (Teacher) and an office worker (OfficeWorker) as its sub-classes.

The class of a place, which defines the general properties about place, presents an outdoor campus (OutdoorCampus) and an indoor campus (IndoorCampus) as its sub-classes. The class of an outdoor campus contains a playground (Playground) and a parking lot (ParkingLot) in its sub-class. The class of an indoor campus contains a classroom (ClassRoom), a staff room (StaffRoom), a resting room (RestingRoom) and a corridor (Corridor) in its sub-class.

The class of an activity (Activity), which defines the general activities in the school, contains a formal activity (FormalActivity) and other activities (OtherActivity) in its sub-classes. The class of a formal activity also contains classwork (Classwork) and an examination (Examination). The class of other activities contains a presentation (Presentation), a sport (Sport), a resting place (Resting) and a consultation (Consultation) in its sub-classes.

The class of a computing entity (ComEntity) defines the general computing entities and contains a device (Device), SW application (SWApplication) and contents (Content) in its sub-classes.

Table I show twelve main object properties related to the top-level class. Each property presents the binary relationship linking an individual in the domain to an individual in the range. For example, the property of presentIn links an individual in the class of a person to a location in the class of place, which means someone is somewhere.

Table I object property and its meaning

Property	Domain	Range	Meaning of the property
presentIn	Person	Place	someone is somewhere
participateIn	Person	Activity	someone is participating in some activity
hasUsage	Person	ComEntity	someone is using some computing entity
hasPresence	Place	Person	some place has one's presence
hasActivity	Place	Activity	some place concludes some activities
hasComEntity	Place	ComEntity	some place concludes some computing entity
hasParticipant	Activity	Person	some activity concludes someone as a participant
activedIn	Activity	Place	some activity happens in some where
hasNeed	Activity	ComEntity	some activity needs some computing entity
isUsedBy	ComEntity	Person	some computing entity is used by someone
locatedIn	ComEntity	Place	some computing entity is some where
isNeedFor	ComEntity	Activity	some computing entity is needed in some activity

C. Application of the CALA-ONT Model

In order to provide a learning service, according to the context, the context-aware manager of the CALA architecture can acquire the context information and reason new contexts based on the CALA-ONT model.

C.1 Expression of the Context Information about a Person's Existence

The context providing module of the context-aware manager acquires context information directly from sensors, and expresses it by ontology. For example, the context providing module can acquire the context information that a certain person is one of those individuals that constitutes the class and the express the context information by ontology. As a scenario, a student named HongYunSang makes his RFID badge perceived by the RFID sensor attached to the door, and enters the classroom. Then the RFID sensor makes his presence known to the context providing module, which expresses the student's presence in Class Room-1-1 through the property of present-in. Figure 9 shows the flow of context providing module which is acquired according to the scenario, and figure 10 shows the ontology expression corresponding to the context information.

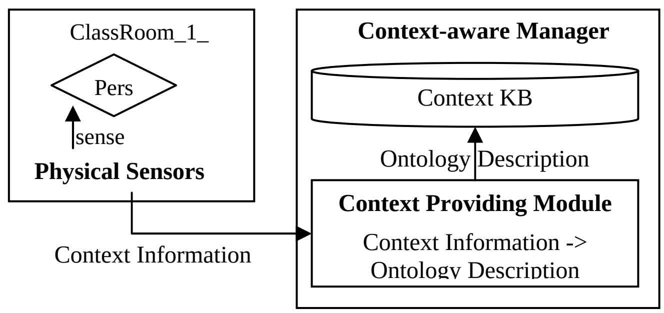

Fig. 9 context providing module for a person's existence

Context Information
(HongYunSang, presentIn, ClassRoom_1_1)
Ontology Expression
<Student rdf:ID="HongYunSang">
<presentIn rdf:resource="#ClassRoom_1_1"/>
</Student>

Fig. 10 ontology expression of a person's existence

C.2 Context Reasoning

The context reasoning engine has functions to interpret context information and reason new contexts: ontology reasoning and rule-based reasoning.

C.2.1 Ontology Reasoning

Ontology reasoning takes advantage of the class relationship, property characteristics, and property limitations. Ontology reasoning of the context reasoning engine can be expressed in the first order predicate logic as shown in Figure 11.

subClassOf
(?A rdfs:subClassOf ?B), (?B rdfs:subClassOf ?C)
-> (?A rdfs:subClassOf ?C)
inverseOf
(?P owl:inverseOf ?Q), (?X ?P ?Y)
-> (?Y ?Q ?X)
finctionalProperty
(?P rdf:type owl:FunctionalProperty), (?A ?P ?B), (?A ?P ?C)
-> (?B = ?C)
transitiveProperty
(?P rdf:type owl:TransitiveProperty),(?A ?P ?B), (?B ?P ?C)
-> (?A ?P ?C)

Fig. 11 expression of ontology reasoning in the first order predicate logic

subClassOf. It represents a hierarchical relationship between classes. Consider the classes A, B and C. If A is a subclass of B and B is a subclass of C, then A can be a subclass of C. For example, class ClassRoom is a subclass of class IndoorCampus and also IndoorCampus is a subclass of class Place, then we can infer that ClassRoom is a subclass of the Place. This context reasoning is expressed in the OWL-DL in figure 12.

```
Context 1
  <owl:Class rdf:ID="ClassRoom">
       <rdfs:subClassOf rdf:resource="#IndoorCampus"/>
  </owl:Class>
Context 2
  <owl:Class rdf:ID="IndoorCampus">
       <rdfs:subClassOf rdf:resource="#Place"/>
  </owl:Class>
New context
  <owl:Class rdf:ID="ClassRoom">
     <rdfs:subClassOf rdf:resource="#Place"/>
  </owl:Class>
```

Fig. 12 ontology expression of subClasssOf

inverseOf. Each object property may have a corresponding inverse property. If some property links individual X to individual Y then inverse property will link individual Y to individual X. For example, figure 13 shows the property hasPresence and its inverse property presentIn – if *ClassRoom_1_1 hasPresence HongYunHo* (Context 2 in figure 14), then because of the inverse property (Context 1 in figure 14) we can infer that *HongYunHo presentIn ClassRoom_1_1* (New context in figure 14). This context reasoning is expressed in the OWL-DL in figure 14.

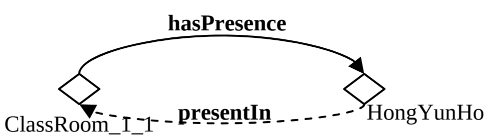

Fig. 13 an example of an inverse property

```
Context 1
  <owl:ObjectProperty rdf:ID=" hasPresence">
      <owl:inverseOf rdf:resource="# presentIn"/>
  </owl:ObjectProperty>
Context 2
  <ClassRoom rdf:ID="ClassRoom_1_1">
      <hasPresence rdf:resource="#HongYunHo"/>
  </ClassRoom>
New context
  <Student rdf:ID="HongYunHo">
      <presentIn rdf:resource="#ClassRoom_1_1"/>
  </Student>
```

Fig. 14 ontology expression of an inverse property

functionalProperty. If a property is functional, for a given individual, there can be at most one individual that is related to the individual via the property. Figure 15 shows an example of a functional property locatedIn. If we say that the individual Projector_1 locatedIn ClassRoom_1_1 and we also say that the individual Projector_1 locatredIn RoomNumber_405, then because locatedIn is a functional property, we can infer that ClassRoom_1_1 and RoomNumber_405 must be the same individual.

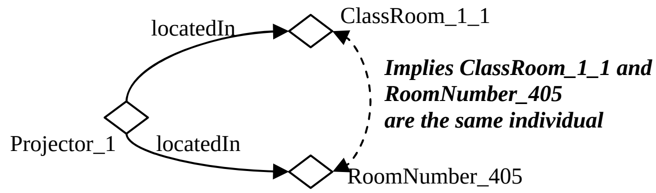

Fig. 15 an example of a functional property

transitiveProperty. If a property is transitive property, and the property relates individual A to individual B, and also individual B to individual C, then we can infer that individual A is related to individual C via the property.

C.2.2 Rule-based Reasoning

The rule-based reasoning is reasoning new contexts based on information about various other contexts. This rule is to combine much information about various contexts using Boolean algebra, and if the rule is true, to define the information about the new contexts. For example, if a person is using a computing entity, and the computing entity is in a certain place, it can be reasoned that the person is also in that place. Figure 16 shows the rule that one new context is reasoned from the information of two other contexts. The information of two contexts connected using AND on the left of -> of the rule corresponds to the ontology expression of Context 1 and Context 2, and the reasoned context information on the right of -> of the rule corresponds to the ontology expression of the New context.

```
Rule
  (?Person, hasUsage, ?ComEntity)
  AND (?ComEntity, locatedIn, ?Place)
  -> (?Person, presentIn, ?Place)
Context 1
  <Student rdf:ID="HongYunHo">
     <hasUsage rdf:resource="#PDA_1004"/>
  </Student>
Context 2
  <PDA rdf:ID="PDA_1004">
    <onoffStatus rdf:datatype="&xsd;boolean">true</onoffStatus>
    <comEntityNumber rdf:datatype="&xsd;int">31004</comEntityNumber>
    <locatedIn rdf:resource="#ClassRoom_1_1"/>
  </PDA>
New context
  <Student rdf:ID="HongYunHo">
     <presentIn rdf:resource="#ClassRoom_1_1"/>
  </Student>
```

Fig.16 example of rule-based reasoning

C.3 Context-Aware Learning Service

The context-aware learning service is similar to the rule-based reasoning. When context information changes, the learning service coordination module of the context-aware manager provides service according to the user defined learning support rules. Context information can be collected from the context knowledge base of the context-aware manager or by acquiring new context information from the context providing module. The user defined learning support rule can be regarded as a set of various contexts combined in the first order predicate logic. The learning service coordination module checks for the observance of the rules accordingly as contexts change. If the result is true, the module gets "learning service performed." The learning service performed at this time can be by an agent, an application program or a service which operates according to the current context.

As an example of the context-aware learning service, figure 17 diagrams a classroom lighting service of three steps. When all the people in the classroom leave, the learning service coordination module is provided with the changed context information that there is no one in the room (First Step). Next the module collects all the context information about the classroom light status (Second Step), and if the light is on, the module performs the task of turning off the light in the classroom (Third Step). These processes are expressed in the first order predicate logic in figure 18.

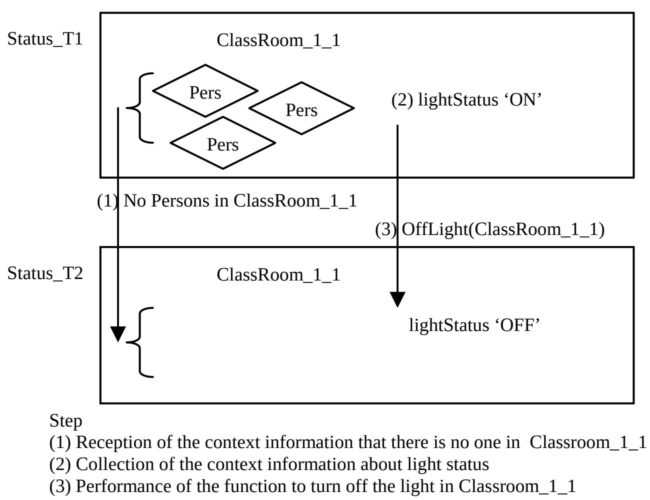

Fig.17 processes of performing classroom lighting service

NOT(?ClassRoom, hasPresence, ?Person)
AND(?ClassRoom, lightStatus, 'ON')
-> OffLight(?ClassRoom)

Fig.18 classroom lighting service expressed in the first order predicate logic

As another example of the context aware learning service, we can provide a contents forwarding service as shown in Figure 19. Suppose that a person, P1 is using certain contents, C1 through a device, D1 in a certain place Classroom_1_1. At the time, that information about the person's existence is found in another place, the learning service coordination module is provided with the context information that the person has moved from the original place (Classroom_1_1) to another place (Classroom_1_2) (Step 1). Then the module collects the previous context information that the person was using C1 (Step 2), and transfers the contents that the person was using to the device of the person's new location (Step 3). These processes are expressed in the first order predicate logic as shown in figure 20.

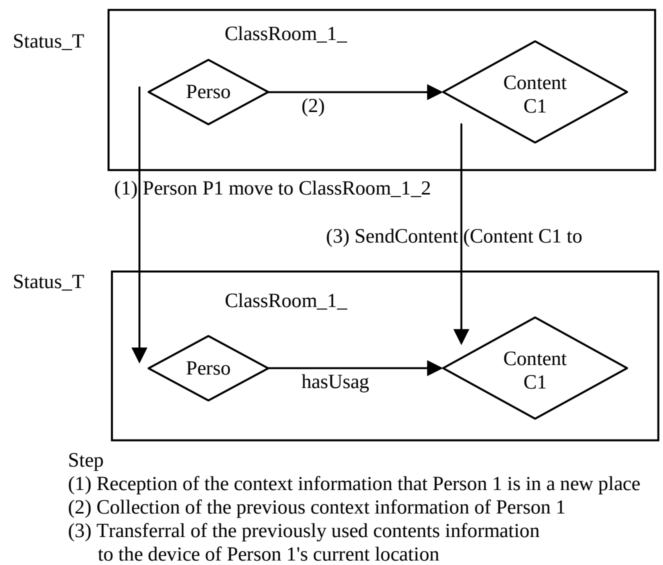

Fig.19 processes of performing content forwarding service

(?Person, present In, ?another Place)
AND (?Person, has Usage, ? Device)
AND (?Person, has Usage, ?Contents)
-> Forward Contents (?Contents, ?another Device)

Fig.20 content forwarding service expressed in the first order predicate logic

V. Conclusion

This study supposes schools are ubiquitous learning environments, and has proposed CALA architecture and the CALA-ONT model. The first proposal focuses on a context-aware manager to provide learning service according to the context changes of learners' environments, and the second expresses school environments by ontology. The study also suggests possibilities of providing context reasoning and context-aware learning service by using architecture and ontology. In these two aspects, this study is distinguished from the preceding studies which concentrated on learning contents and designed ontology. First, the subject of the study is school in which practical learning activities are taking place. Second, this study has designed context models based on ontology using OWL-DL and has expanded them making them reusable. Though the ontology models suggested by the study didn't express all the objects and contexts, the models could be modified and expanded given the possibilities of reuse. This study has yet to apply varied and standardized methods to context reasoning and context-aware learning service, which will be a task of a future study along with verification of many other relevant things.

REFERENCES

[1] S. Yang, "Context Aware Ubiquitous Learning Environments for Peer-to-Peer Collaborative Learning," *Educational Technology & Society*, 9 (1), 2006, pp. 188-201.

[2] S. Nansaarng, S. Kaewkuekool, and S. Siripattanakunkajorn, "Improve learning achievement in Engineering Metallurgy Course by Online Asynchronous learning," *Proceedings of the 3rd WSEAS/IASME International Conference on ENGINEERING EDUCATION*, Vouliagmeni, Greece, July 11-13, 2006, pp.1-5.

[3] M. Weiser, "The Computer for the 21st Century," *Scientific American*, 265 (30), 1991, pp. 94-104.

[4] H. Chen and T. Finin, "An Ontology for Context Aware Pervasive Computing Environment", *IJCAI Workshop on Ontologies and Distributed Systems*, Acapulco MX, August 2003.

[5] M. Roushdy, A. Badeeh, and M. Salem, "Case-Based and Ontology Learning Approaches for Developing e-Learning Systems," *WSEAS TRANSACTIONS on INFORMATION SCIENCE and APPLICATIONS* Issue 6, Volume 2, June 2005, pp. 795-799.

[6] A. Dey, G. Abowd, and D. Salber, "A Context-Based Infrastructure for Smart Environments," *Proceedings of the 1st International Workshop on Managing Interactions in Smart Environments (MANSE '99)*, 1999, pp. 14-128.

[7] B. Schilit, N. Adams, and R. Want, "Context-aware computing applications," *Proceedings of the Workshop on Mobile Computing Systems and Applications (Santa Cruz, CA, USA, December 1994), IEEE Computer Society*, pp. 85-90.

[8] R. Want, A. Hopper, V. Falcao, and J. Gibbons, "The active badge location system," *ACM Transactions on Information Systems 10 (1)*, 1992, pp. 91-102.

[9] T. Kindberg and J. Barton, "A Web-Based Nomadic Computing System," *Computer Networks, 35(4)*, 2001, pp. 443-456.

[10] A. Dey, D. Salber, and G. Abowd, "A Conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications," *Human-Computer Interaction*, *16*, 2001.

[11] T. Berners-Lee, J. Hendler, and O. Lassila, "The Semantic Web," *Scientific American*, May 17 2001, pp. 28-37, Available: http://www.sciam.com/article.cfm?id=00048144-10D2-1C70-4A9809EC588EF21

[12] T. Gruber, "Toward Principals for the Design of Ontologies Used for Knowledge Sharing," *International Journal of Human-Computer Studies*, Vol. 43, 1995.

[13] K. Seta, T. Okazawa, M. Umano, and M. Ikeda, "Ontology Based Organizational Risk Knowledge Creation Support Based on Incident Reports," *Proceedings of the 11th WSEAS International Conference on SYSTEMS*, Agios Nikolaos, Crete Island, Greece, July 23-25, 2007, pp. 337-342.

[14] C. Huang, C. Sun, F. Shih, and J. Hsieh, "Mining Domain Ontological Information from Online Publications," *Proceedings of the 6th WSEAS International Conference on E-ACTIVITIES*, Tenerife, Spain, December 14-16, 2007, pp. 155-160.

[15] http://www.w3.org/TR/xmlschema-0/

[16] http://www.dlib.org/dlib/may98/miller/05miller.html

[17] http://www.w3.org/TR/rdf-primer/

[18] http://www.w3.org/TR/daml+oil-model

[19] http://www.w3.org/TR/owl-guide

[20] A. Ranganathan and R. Campbell, "An infrastructure for context-awareness based on first order logic," *Personal and Ubiquitous Computing 7(6)*, 2003, pp. 353-364.

[21] X. Wang, T. Gu, D. Zhang, and H. Pung, "Ontology Based Context Modeling and Reasoning using OWL," *Workshop on Context Modeling and Reasoning at IEEE International Conference on Pervasive Computing and Communication. Orlando, Florida*, 2004.

[22] T. Gu, H. Pung, and D. Zhang, "An ontology-based context model in intelligent environments," *Proceedings of Communication Networks and Distributed Systems Modeling and Simulation Conference, San Diego, CA, USA, 2*004.

[23] H. Chen, T. Finin, and A. Joshi, "An Ontology for Context Aware Pervasive Computing Environments," *Knowledge Engineering Review-Special Issue on Ontologies for Distributed Systems, Cambridge University Press*, 2004

Myoung-woo Hong

Received a B.S. degree in computer engineering from Chungnam National University, Korea, 1993, and M.S. degree in computer engineering from Chungnam National University, Korea, 1995. In 1999 he joined the faculty of Woosong information College, Korea, where he is currently a professor in Division of Computer Information. His research interests include Image Processing, Ubiquitous Computing and E-learning System. He is a member of KISS and KCA.

Dae-jea Cho Received a B.S. degree in computer engineering from Kyungpook National University, KOREA, 1984, and M.S. degree in computer engineering from Kyungpook National University, 1986 and Ph D. degree in computer engineering from Kyungpook National University, 2001. In 2002 he joined the faculty of Andong National University, Korea where he is currently a professor in Department of Multimedia Engineering. His research interests include Multimedia systems, Multimedia Security, Digital Watermark. He is a member of KISS, KMMS, KIPS and KCA.