효율적인 기관 정보 관리를 위한 유비쿼터스 서비스 구조에 관한 연구

정회원 황 광 일*, 종신회원 엄 두 섭*, 허 경**

A Ubiquitous Service Framework for Efficient Organizational Information Management

Kwang-il Hwang* *Regular Member*, Doo-seop Eom*, Kyeong Hur** *Lifelong Members*

요 약

본 논문에서 우리는 특정 지역에 익숙하지 않은 사용자들에게 그 지역에 대한 지능적이고 유용한 서비스를 제공해 줄 수 있는 구체적이고 실질적인 유비쿼터스 기관 정보(UOI : Ubiquitous Organizational Information) 서비스 구조를 제안한다. 제안하는 UOI 구조는 분산 센서 네트워크에 기반한 계층적 네트워크로 구성된다. 무엇보다 보다 양질의 서비스를 제공하기 위하여, UOI 서비스 소프트웨어와 기반 소프트웨어를 모듈화하여 개발하였다. 이 UOI 서비스 와 기반 소프트웨어는 자체 개발한 하드웨어 위에서 구현되었다. 우리는 세 종류의 대표적인 UOI 서비스를 정의하고, 제안된 UOI 네트워크에서 동작하는 각 서비스의 흐름을 묘사한다. 덧붙여, 개발된 UOI 테스트 베드를 통한 구현의 세부사항을 묘사하고 결과를 요약한다.

Key Words : Distributed Sensor Networks, Embedded Systems, Intelligent Environment, Ubiquitous Computing

ABSTRACT

In this paper, we introduce a concrete, practical Ubiquitous Organizational Information (UOI) service framework, providing novice users intelligent and useful services respecting the environment. The UOI framework consists of hierarchical network architecture and is based on distributed sensor networks. To provide a rich array of services, the modular software framework and foundation software is designed. The UOI framework and foundation software is implemented on our hardware prototype. We define three representative UOI services and illustrate each service flow operating in our UOI network. In addition, we describe some details in the implementation of a distributed UOI network on the UOI test-bed.

Ⅰ. Introduction

To coincide with the grand pervasive computing vision, everyday computing spaces will need to become a component of the user's normal background environment, gradually becoming more ubiquitous in nature. Mark Weiser first initiated the notion of Ubiquitous Computing at Xerox PARC[1], who envisioned in the upcoming future, ubiquitous interconnected computing devices that could be accessed from any location, used effortlessly, and operate unobtrusively, even without people's notice of them, just

* 고려대학교 전자컴퓨터공학과 미래정보모망연구실 (brightday@final.korea.ac.kr, eomds@korea.ac.kr)
** 경인교육대학교 컴퓨터교육과 (khur@ginue.ac.kr)
논문번호 : KICS2006-05-196, 접수일자 : 2006년 5월 3일, 최종논문접수일자 : 2006년 8월 24일

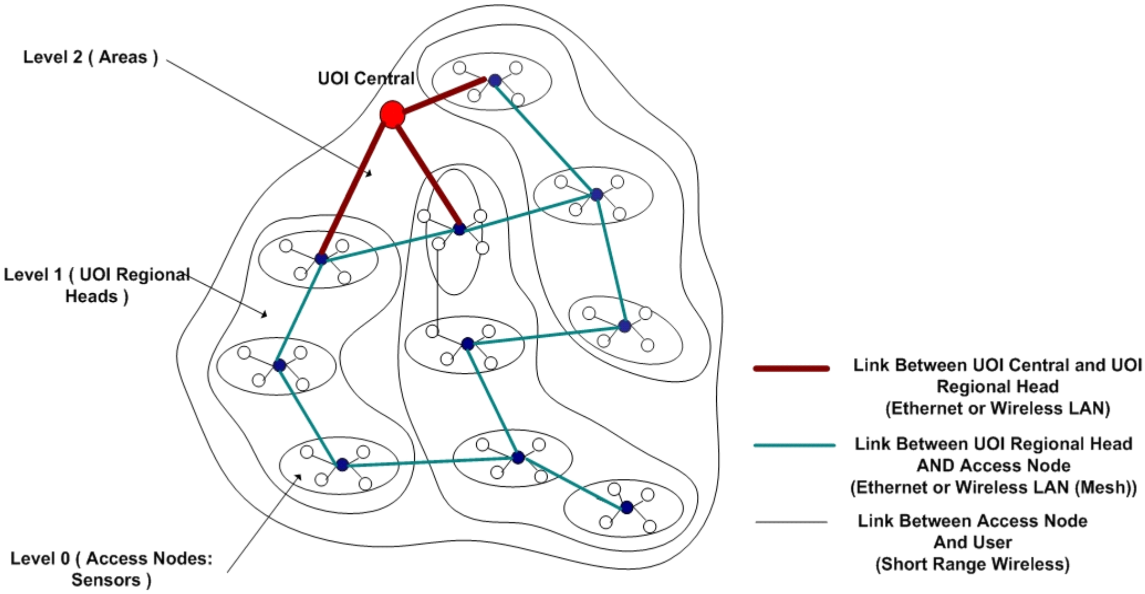
Level 2 (Areas)
UOI Central
Level 1 (UOI Regional Heads)
Level 0 (Access Nodes: Sensors)
Link Between UOI Central and UOI Regional Head (Ethernet or Wireless LAN)
Link Between UOI Regional Head AND Access Node (Ethernet or Wireless LAN (Mesh))
Link Between Access Node And User (Short Range Wireless)

Figure 1. Hierarchical UOI Network Architecture Based on Distributed Sensor Networks

as that of electricity or telephones are used today.

Many researchers define an intelligent environment, as one of the most representative applications of Ubicomp,as an augmented spacious environment populated with many sensors, actuators and computing devices. These components are interwoven and integrated into a distributed computing system, capable of perceiving context through sensors, to execute intelligent logic on computing devices and serve occupants by actuators. This intelligent environment is extending its range from a users' personal room or classroom, to a large house or building. Let us suppose the following situation. When we visit an unfamiliar environment, which presents a wide area, such as an amusement park, university campus, or large building, confusion often arises when finding the location of something or where to travel next. Furthermore, the use of certain facilities may be desired or someone in the organization may need to be found. Such users'needs can be satisfied with an intelligent service involving information regarding the organization. Our UOI service framework is designed to provide such an intelligent service to users, especially for large scale environment. In this paper, the presented UOI service framework presents more concrete and practical way to create intelligent environment.

In the rest of this paper, we first outline several researches related to the intelligent environment. Then, the UOI framework is presented, which is composed of three major components. Subsequently, the UOI service flow through the distributed UOI network is illustrated. Our hardware prototype and UOI foundation software operated on the prototype is also introduced. Lastly, details in the implementation of a distributed UOI network are described. A conclusion is provided with a description of future work.

Ⅱ. Related Work

There has been substantial research relating to the construction of ubiquitous environments.

Cooltown[2] and the associated CoolBase infrastructure aim to give people, places, and things a Web presence. Although Web technology is proven and widely available, it has inherent complexity, since, to be connected to the web, a fully supported TCP/IP stack and system capable of running the relatively heavy software is required.

Projects, such as Gaia[8], Microsoft Easy Living[3], and CORTEX[9], aim to develop an infrastructure to support augmented environments in a fairly broad sense. They provide basic abstractions and mechanisms for coping with the dynamics and device heterogeneity of pervasive computing environments. There is quite a large difference between the projects and the framework presented in this paper. While they provide application models that are still rather generic, our work supports a rather specific application model.

In such a sense, PiNet[10] is the most similar to the presented model in that the final goal is to provide an organizational information service to users. However, the work in this paper is distinguished from PiNet primarily in the uses of sensor networks. In contrast with PiNet using a global cellular network as an infrastructure, the UOI adopts distributed sensor networks. Service network infrastructure and service framework based on sensor networks, are also emphasized, instead of focusing on user perception or virtual reality as in [4-7].

Ⅲ. Architecture of UOI Framework

The UOI service framework infrastructure is based on distributed sensor networks. It is assumed that the environment is covered with innumerable tiny sensor nodes, which are extremely limited in power, processing, and memory resources. The sensor nodes are called access nodes, since they become access points that users must use to connect to UOI services. Each node is aware of its own location information by manual planning or other localization algorithms, and possesses the ability to communicate with user’s devices via short range wireless communication. Using sensor networks, not global networks such as cellular networks or GPS, a UOI service infrastructure presents advantages as follows.

- Guarantee of freshness with respect to the dynamics of information
- Security
- Service charge issues regarding information use

Firstly, the update of information with respect to the change of an organizational structure or service will be achieved faster and easier through a scalable UOI framework. Secondly, organizational information will be safer from outside networks. Lastly, users are allowed to use the service without any extra communication charge. In addition to these advantages, compared to only WLAN networks, the UOI network architecture provides more elaborate location information and increases efficiency in the use of facilities through sensors and actuators. The UOI network architecture also enables localized information processing and fusion, by clustering regionally adjacent access nodes.

3.1 Key Features of UOI Framework

The presented framework is designed to efficiently provide UOI services. For achieving the design goal, the UOI framework includes the following array of features.

Hierarchical architecture: As shown in Fig. 1, the UOI framework is composed of a three-tier hierarchical architecture, more specifically, access node level, regional head level, and UOI central. Regionally adjacent nodes form a cluster in which a cluster head node is responsible for managing its own cluster member nodes. In addition, the clusters form a network of tree-based or mesh-based topology for communication between each other. The clusters are also connected to the UOI central, which operates similar to a central server. This hierarchical architecture for UOI framework makes it possible to localize various information in a cluster, such as the service request by the user or organizational information, and reduce the network traffic by aggregation and fusion within a cluster head. This is also useful in terms of the regional environmental management.

Property-based naming and information-centric routing: The UOI platform uses property-based naming, similar in concept to naming in the Directed Diffusion [11], not global ID such as IP address or MAC address, as a node identifier. Each node has an inherent name related to its property such as location or sensing ability. For example, ’East2 Floor1’ means the node is the 2nd node from UOI regional head of the 1st floor to east. In addition, ’Tb21 KoreanRest1’will be the 21th table number of a Korean restaurant in a huge amusement park. This property-based naming is enabled by information-centric routing, not address-centric routing. In the UOI service network, end-to-end data transfer is not effective. Instead, some information is obtained from the set of specific location based properties, through querying and tasking.

Distributed querying and tasking: In the UOI framework, user's service requests are translated into a query to be flooded to the UOI network. The query is injected in each access node and the UOI regional head via the UOI hierarchy. In each node, the query generates a corresponding task, executing the task on the UOI foundation software. This distributed tasking demonstrates some results by querying and only the nodes, which have matching data by the query, report matching information to UOI central. This feature reinforces the distributed information processing ability in the UOI network, in contrast to other global or centralized networks.

Transparent services: The UOI framework provides transparent services to users. Users only request a service with their device, and specific actions for configuration or registration are not required. When entering an area covered by a UOI network,the user is expected to turn on the device and be automatically connected. No configuration changes are necessary as the user moves from one site to another. The network needs no pre-knowledge regarding the device attempting to connect to it.

3.2 Components for UOI Service Framework

The UOI network infrastructure is composed of three distinguished components, Access node, UOI regional head, and UOI central, as shown in Fig. 2. These independent components play an important role in building a UOI service framework with a hierarchical architecture.

Access node is the most basic of components, allowing users to access the UOI infrastructure. This component is also composed of UOI foundation software, Query Translation Unit, Task Manager Unit, User Interaction Manager, and Location Management Unit.

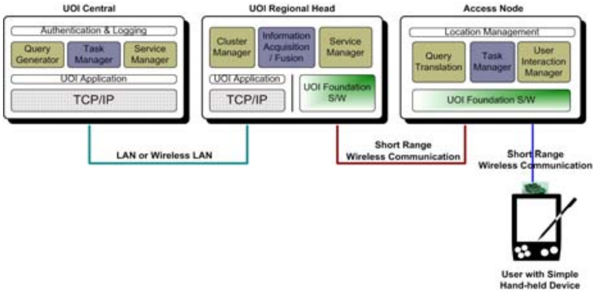

Figure 2. Components for UOI Service Framework

UOI Regional Headas a cluster head is responsible for managing access nodes in its own cluster and performing a gateway role between access nodes and UOI central. To efficiently perform their own roles, it has independent network software: TCP/IP for connecting to UOI central by LAN or WLAN, and UOI foundation software for managing cluster members. In terms of network software, the UOI head operates with UOI application, Cluster Manager Unit, Information Acquisition and Fusion Unit, and Service Manger Unit.

UOI Central represents a highest level component in the UOI service framework. It plays an important role in generating queries with respect to user's service request, managing tasks and services and authenticating users and logging. The work is performed by the Query Generation Unit, Task Manager, Service Manager, and Authentication Unit.

3.3 Organizational Information Services through Distributed UOI Networks

Users entering an unfamiliar environment need to get some organizational information and be available to freely use various facilities in the organization. Furthermore, users do not want to manually configure or register themselves to use organization services. In the UOI service framework fulfilling such user's requirements, we divide the UOI services into the following services. Location guide service: First of all, the lowest level service offered to novice users is location guidance. Generally, guiding services using GPS are the most common. However, the service cannot guarantee service of high resolution. In addition, GPS is difficult to be used for indoor locations, such as large buildings.

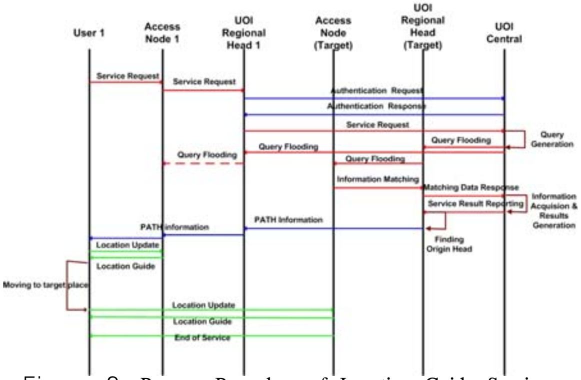

Figure 3. Process Procedure of Location Guide Service in UOI framework

Conversely, our UOI framework provides a reliable location guide service with higher resolution, dependent on the density of deployment, through distributed UOI networks. Fig. 3 shows the process procedure of location guide services in the UOI framework. As shown in Fig. 3, the nearest access node listens to user's service request and then sends the request message to the UOI regional head. After successful authentication between the head and central, the user's service request at UOI centralis transformed into a corresponding query and the query is flooded over the network. As soon as receiving the query, each node executes a task from the query. The nodes with matching data send the response upwardly instantly. UOI central performs aggregation, fusion, and result generation during the available period. The result is sent to the UOI regional head of the targeted node.

The targeted head finds the original head to establish the communication link between the two regional heads. After discovering, the head sends PATH Information to the origin head, including all path information from the target to source. Note that this logical path is regarded as a reliable real path, since we assume the environment is densely covered with sensor nodes. At thattime, all access nodes, included in all regional heads on the path, execute the location guide task and are ready to receive location update messages from the user. A user can now view the location where the user wants to go, which will be displayed on his or her device. The user's device sends location update message periodically and each access node responds with a location guided message. Eventually, the user will be guided to a target location through the UOI service over a distributed network.

Use of facilities in the organization: One of the representative intelligent services, allows users to use various facilities without prior knowledge in the organization. For example, in an amusement park, we want to find a specific amusement facility. However, generally we must wait for a long time just for the short instance actually using the service. This delay may annoy some users. However, they will usually still use the service.

However, inthe UOI service framework, users do not have to wait for a long line. Instead of waiting until the user's order comes, users can simply reserve the use of the facility that the user wants while enjoying a different amusement facility. After the reservation, when the user's service is available, the user is alarmed through the UOI network. Figure 4 shows the detailed flow for user to use facilities in the UOI framework. The service is divided into four phases. The first phase consists of service request andauthentication, and second phase is the discovery of service access points and a status result report. These two phases are almost similar in a location guide service. The third phase is user reservation to the service access point. The final is service reservation and result reporting. All these processes are achieved through distributed UOI networks and the actual service is completely transparent to users.

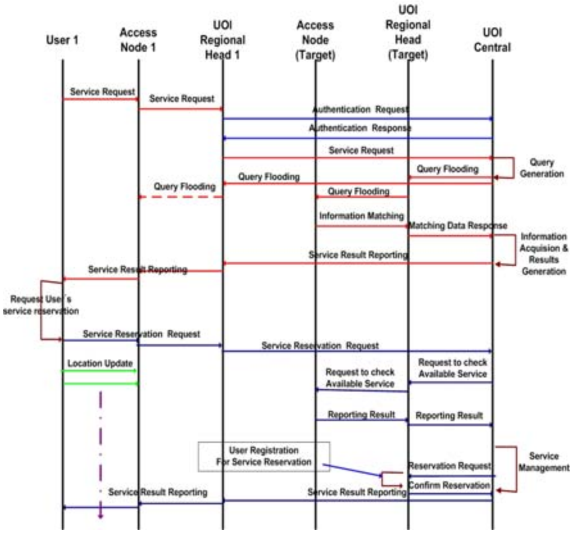

Figure 4 Process Procedure for Use of Facility in UOI framework

People search: Another required service involves finding people within an organization. This is useful to find out a missing child's location or the location of someone who is not available to use a phone in the organization. In addition, this service is used to find the corresponding user, who is reserved to use a facility, to notice that the serviceis now available. For this service, general steps are similar to others but it is outstanding that the logical path between the user, who requests the service, and target user is established. Note that for concurrent mobility support for two users, location information is delivered for both nodes as shown in Fig. 5.

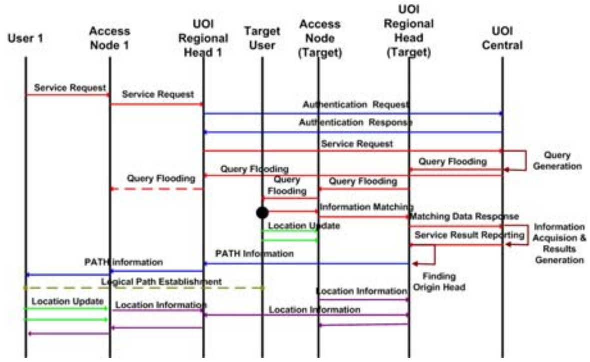

Figure 5. Process Procedure for People Search

Ⅳ. PROTOTYPE SYSTEM

We designed a prototype system to implement and verify the UOI network framework. In this Section, we introduce our prototype system and illustrate the foundation software architecture operated under a UOI service framework.

4.1 Hardware Prototype for UOI Connectivity

A prototype having a very compact size of a square inch, powered from two 1.5V AA was developed. As shown in Fig.6, this hardware prototype is composed of four individual boards, Main board, RF boards, I/O boards and Interface board for interface with a PC or PDA.

The main board in the system plays the most important role of driving the RF board and I/O board. An ATMEL AT89c51ED2 MCU is used as a main MCU with a 12MHz clock. It includes 64kbytes of internal flash memory for code, and 2048 bytes of internal RAM. The MCU allows power management to be performed in software by supporting three power mode levels, ACTIVE, IDLE, and POWER DOWN. For efficient I/O connection, the prototype uses address decoded I/O methods. This method has the advantage of being more flexible, possessing scalable I/O connections compared to general purpose direct I/O ports.

The RF board is composed of a RF modem and additional RF circuits, such as an impedance matching circuit and antenna, for short-range wireless communication. An outstanding feature of our prototype relating to RF communication, is the support of four kinds of modems with minor differences in characteristics, these consist of: RF102 (RFwaves), nRF2401 (Nordic), CC2400 (Chipcon), and CC2420 (Chipcon, 802.15.4 compatible). The common interface between the independent RF modems was also devised, connecting the RF board and Main board. This common interface uses a 3-wired Serial Peripheral Interface (SPI) and 4 general I/O ports, providing simple but useful usage with RF modems for the software developer.

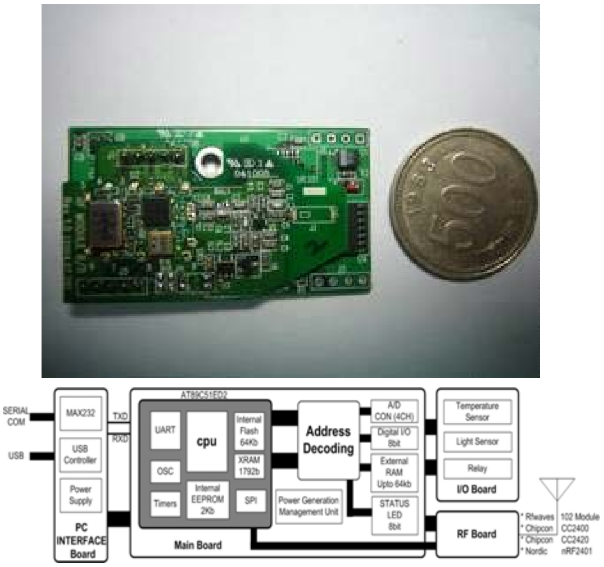

Figure 6. Prototype for UOI Access Nodes and User Interface

To support various sensor and actuator applications, the I/O board is separated from the Main board. The I/O board includes a Temperature sensor, a CDS cell, Ultrasonic sensors, and a relay circuit to control external illumination or another actual system. The board can also accommodate additional I/O devices. Generally, most nodes are battery powered. However, some specific nodes may be connected to high level computing devices, such as a PC or PDA. For such nodes, the Interface board in the prototype provides twokinds of communication interfaces, RS-232C and USB, powered from an AC adaptor. The interface board is also used as an interface to load a program or to perform serial debugging.

4.2 UOI Foundation Software Architecture

The design of ubiquitous system software is different from that of traditional general purpose embedded systems in that the system should process application specific operations cooperatively in the network, as well as manage the efficient utilization of limited resources. Figure 7 describes our UOI foundation soft

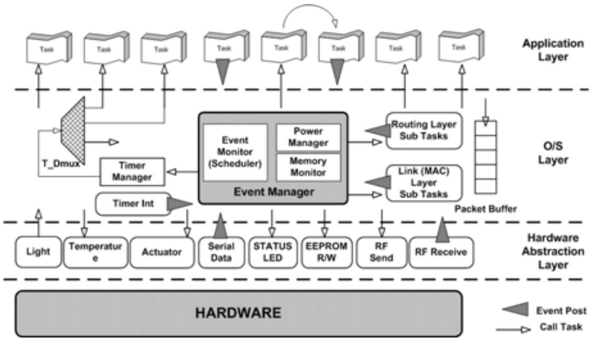

Figure 7. UOI Foundation Software Architecture

ware architecture for ubiquitous networks. The software architecture is divided largely into three important layers: hardware abstraction, operating system, and application. This layered software concept provides convenient usage to efficiently use each resource and provide the modularity of each software components.

In the hardware abstraction layer, control codes for each hardware component, such as manipulations of registers in the MCU, RF modem control, sensor and actuator control, are translated into an abstract layer, with each module consisting of different I/O functions. Using the abstraction, some hardware changes would not affect all software, but require just a minimum of code modification.

The UOI foundation software performs event-driven task management as in [12]. This event-driven method is suitable for application specific characteristics. However, it is distinguished from TinyOS in [12] in that the proposed system supports two kinds of events: sporadic events and periodic events. The former represent events that are not real-time, triggered from hardware or software, the latter are time-critical events triggered from timer interrupts or software. When registering these two separate events, time-critical tasks are protected by the scheduler from other sporadically occurring tasks.

The UOI foundation software basically includes CSMA/CA, as a Multiple Access Control (MAC) protocol. In addition, the system provides several routing protocols modified for more efficient routing on the basis of traditional routing protocols, a flooding, tree-based routing protocol, and mesh supported routing protocol. Based on this foundation software, UOI framework software is performed as an application.

V. UOI Network Implementation

In implementing the proposed UOI service framework, a minimized UOI test-bed as shown in Fig. 8, was developed. The constructed Test-bed was composed of our prototype nodes as access nodes, built-in UOI foundation and framework software, regional heads for higher performance and UOI central. For the experiment, an embedded system with Intel's Xscale MCU (PXA255) was used as a UOI regional head based on sensor gateway software and UOI central software was executed on a desktop PC. The deployment of each component is as shown in Fig. 8, considering two users and two service points.

In the test-bed, three kinds of UOI service that we described in former section were experimented with: in the first experiment, user A is guided to service point 2 through the UOI service. Secondly, user A and B reserved an available arbitrary seat in service point (room) 1 and 2, respectively, and then were guided to the corresponding service point. In the final experiment, mobile user A tracked mobile user B.

Initial architecture design was revised in a more practical sense through the experiment. After experiencing innumerable trials and errors in the services, measurable improvements were made to the complete UOI service framework architecture.

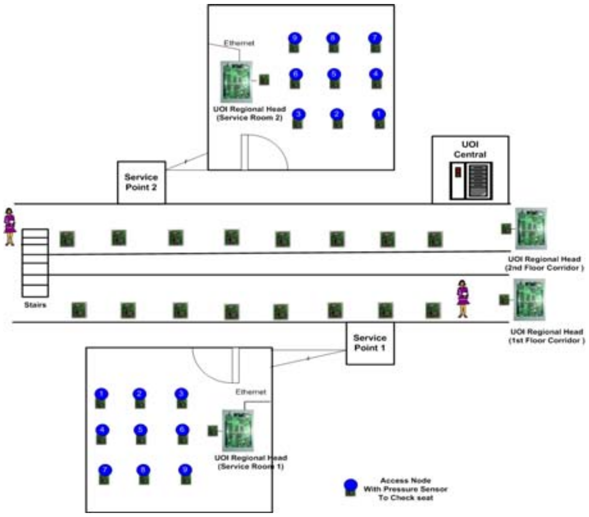
Ethernet
UOI Regional Head (Service Room 2)
9
8
7
6
5
4
3
2
1
UOI Central
Service Point 2
UOI Regional Head (2nd Floor Corridor)
Stairs
UOI Regional Head (1st Floor Corridor)
Service Point 1
Ethernet
1
2
3
4
5
6
7
8
9
UOI Regional Head (Service Room 1)
Access Node With Pressure Sensor To Check seat

Figure 8 Test-bed Environment for Implementation of UOI Service Framework

Ⅵ. Conclusion and Future Work

A concrete, practical Ubiquitous Organizational Information (UOI) service framework is proposed, providing novice users intelligent anduseful services respecting the environment. The UOI framework consists of hierarchical network architecture, based on distributed sensor networks. To provide a rich array of services, the designed UOI framework and foundation software are implemented on our hardware prototype. In addition, representative UOI services were tested on the UOI test-bed.

The current framework does not provide fully satisfactory services to users in an intelligent environment. Currently, we are trying to extend the kind of services and improve the quality of services. For a more intelligent environment, it is clear that more convenient user identification is required. Therefore, we are applying a RFID system for user identification in the UOI service framework. In addition, intelligent services through an increased variety sensors and actuators as well as communication are being developed.

References

〔1〕 M. Weiser, "The Computer for the 21st Century," Scientific American, September, 1991.

〔2〕 HP Cooltown. http://www.cooltown.com/cooltown/

〔3〕 Microsoft Easy Living. http://research.microsoft.com/easyliving/

〔4〕 MIT AI Lab AIRI Project, 2000. http://aire.csail.mit.edu/

〔5〕 Georgia Tech, Aware Home esearch Initiative (AHRI) Project, http://www.cc.gatech.edu/fce/ahri/

〔6〕 CMU Project AURA. http://www-2.cs.cmu.edu/~aura/

〔7〕 Sony Augmented Surfaces. http://www.csl.sony.co.jp/person/rekimoto/as/

〔8〕 M. Roman, C. Hess and R. Campbell, Gaia: An OO middleware infrastructure for ubiquitous computing environments, ECOOPOOOSWS 2002, 2002.

〔9〕 P. Verissimo, V. Cahill, A. Casimiro, K. Cheverst, A. Friday and J. Kaiser, CORTEX: Towards supporting autonomous and cooperating sentient entities, European Wireless 2002, Florence, Italy (February2002).

〔10〕 Carmel, B. et al., "PiNet: Wireless Connectivity for Organizational Information Access Using Light-weight Handheld Devices," IEEE Personal Communications, Aug. 2001.

〔11〕 C. Intanagonwiwat, R. Govindan, and D. Estrin, "Directed diffusion: A scalable and robust communication paradigm for sensor networks," 6th ACM/IEEE Mobicom, 2000.

〔12〕 Jason Hill, Robert Szewczyk, Alec Woo, Seth Hollar, David Culler, Kristofer Pister, "System Architecture Directions for Networked Sensors," ASPLOS 2000, 2000.

황 광 일 (Kwang-il Hwang) 정회원

2002년 8월 홍익대학교 전자전기공학부 졸업

2004년 8월 고려대학교 전자컴퓨터공학과 석사

2004년 9월~현재 고려대학교 전자컴퓨터공학과 박사과정

<관심분야> AD HOC networks, RFID systems, Sensor networks, Ubiquitous networks

엄 두 섭 (Doo-seop Eom) 종신회원

1987년 2월 고려대학교 전자전자공학과 졸업

1989년 2월 고려대학교 전자공학과 석사

1999년 2월 일본 오사카대학 정보컴퓨터과학과 박사

1996년~1999년 ETRI 선임연구원

1999년 9월~2000년 원광대학교 전임강사

2000년 9월~현재 고려대학교 전자공학과 교수

<관심분야> AD HOC networks, RFID systems, Sensor networks, Ubiquitous networks, Internet QoS.

허 경 (Kyeong Hur) 종신회원

1998년 2월 고려대 전자공학과 학사

2000년 2월 고려대 전자공학과 석사

2004년 8월 고려대 전자공학과 박사

2004년 9월~2005년8월 삼성종합기술원 전문연구원

2005년 9월~현재 경인교육대학교 컴퓨터교육과 전임강사

<관심분야> 컴퓨터 네트워크, All-IP 네트워크, Wireless MAC 시스템