Appl. Math. Inf. Sci. 6 No. 1S pp. 311S-321S (2012)

Applied Mathematics & Information Sciences
*An International Journal*



# Ubiquitous Robot and Its Realization

**Kang-Hee Lee[1] and Kyoungsu Oh*[2]**
[1]Global School of Media, Soongsil University, Republic of Korea
[2]Global School of Media, Soongsil University, Republic of Korea
*Email Address:* oks@ssu.ac.kr

Received: Received May 02, 2011; Revised July 25, 2011; Accepted September 12, 2011
Published online: 1 January 2012

**Abstract:** Objective: The standardized data may easily move within a network and connect to other systems without any time or geographical limitations. This concept is fundamental to the idea of the ubiquitous robot. Method: This paper discusses those definitions briefly including the approach of Samsung Electronics Co., Ltd (SEC). Here we present a ubiquitous robot S/W platform, AnyRobot Studio which covers all the aspects of this ubiquitous robot system. Based on the concepts of RUPI, URC, and Web 2.0, AnyRobot Studio aims to standardize the platforms and protocols, and to strongly encourage the participation of users and contents providers (CPs). Conclusions: AnyKids Service is made to test the feasibility of the AnyRobot based on the AnyRobot Studio platform. It proves the realization of the ubiquitous robot.

**Keywords:** Ubiquitous Robot, Sobot, AnyRobot Studio, AnyKids Service.

## 1 Introduction

The application-independent technology is defined as the common platform/infrastructure technology. Based on this concept, the main approach of the networked robot (NR) or the ubiquitous robot (UR) aims to distribute the robot's functional components such as sensing, processing, and acting through the network, and moreover, to fully utilize shared external sensors and external processing servers. Therefore, this paper proposes a ubiquitous robot S/W platform for the network-based robot system or the ubiquitous robot, AnyRobot Studio of Samsung Electronics Co., Ltd (SEC). Its aim is to reduce the costs and the risk of building network-based robot system and to reduce the amount of resources & time efforts that are put into developing the robot systems. AnyRobot Studio aims to standardize the platforms and protocols and to strongly encourage the participation of users and CPs.

This paper is organized as follows. A brief description of the underlying concepts and various definitions of the networked robot or the ubiquitous robot in Section 2 are presented. Section 3 presents the theoretical background for the AnyRobot Studio, which utilizes the concepts of URC, RUPI, and Web 2.0. In Section4 AnyRobot Studio is presented in detail, which is composed of Server platform, Robot Platform, and Service authoring Platforms such as AnyMap Studio, AnyAction Studio, and simulation tools. Section 5 proves the feasibility and the effectiveness of AnyKids service employing AnyRobot Studio. Finally, concluding remarks are presented in Section 6.

## 2 Ubiquitous Robot

The definitions of the networked robot in IEEE, EU, Japan, South Korea are described. Section 2.2 presents its concept in SEC which this paper is based on, and compares with the others.

[2] **Corresponding Author: Kyoungsu Oh, oks@ssu.ac.kr**

**2.1 Overview**

The IEEE RAS Technical Committee on Internet and Online Robots was founded in May 2001. It originally focused on Internet-based tele-operated robots, but expanded its scope to account for a broader set of problems and applications. The name “Networked Robot" was approved and adopted in May 2004. During that period, Korea, Japan, EU, IEEE etc. each established their own organization for the networked robot and organized their research frameworks as well.

**IEEE RAS Technical Committee on Networked Robots (IEEE TC-NR)** defined the networked robot as follows:

"A networked robot is a robotic device connected to a communication networked such as the Internet or LAN. The network could be wired or wireless, and based on any of a variety of protocols such as TCP, UDP, or 802.11 [1]. There are two subclasses of networked robots: 1) Tele-operated, where human supervisors send commands and receive feedback via the network, 2) Autonomous, where robots and sensors exchange data via the network."

**Europe: The Research Atelier on Network Robot Systems (NRS)** was created on December 15th, 2005 at the start of the EURON II (European Robotics Network), with three purposes: to generate a Roadmap of the NRS in Europe, to start a NRS community in Europe and to disseminate the results of the Research Atelier among research institutions and companies, through scientific and technological channels [2]. NRS defined the network robot system as follows:

"A network robot system is a group of artificial autonomous systems that are mobile and that makes important use of wireless communications among them or with the environment and living systems in order to fulfill their tasks."

**Japan: Network Robot Forum (NRF) and Ubiquitous Networking Forum (UNF)** are leading the research of the network robot in industry in Japan [3-4]. Under the national administration by the Ministry of Internal Affairs and Communication, Common Platform for Robots is being carried out as the standardization for this NRS. In Japan, the goal of a network robot system is the quantum leap in robotics technology by making use of ubiquitous network connecting three types of robots such as the virtual robot, the unconscious robot, and the visible robot in particular environment [5]. To achieve this goal, they are emphasizing on two core technologies as follows:

"One is the Robot Plug and Play (Robot PnP) network robot platform for coordination and cooperation among multiple types of robots as a connection unit. Based on the Robot PnP, three types of robots can communicate with each other through the network and support people in conjunction with each other. The other is HRI, which means the flexible interaction with people, taking into account their situation. It is the embodied interaction which is semi-autonomous and context-dependent."

**Korea: Ubiquitous Robotics Companion (URC)** focuses on the network computing (server-based sensing and planning) and is defined as follows:

"Generally, service robots need three major function; sensing, planning, and action. The basic concept of URC is to distribute those three functions through the network, instead of cramming them into a single machine [6]. It enables robots to reduce costs and improve service qualities by taking full advantage of IT infrastructure including internet and wireless network."

The user-oriented definition of URC is as follows: "a network-based robot system which provides necessary services to me in anywhere at any time [7]." In reality, the term "ubiquitous robot" is more widely used than the term "networked robot" in Korea.

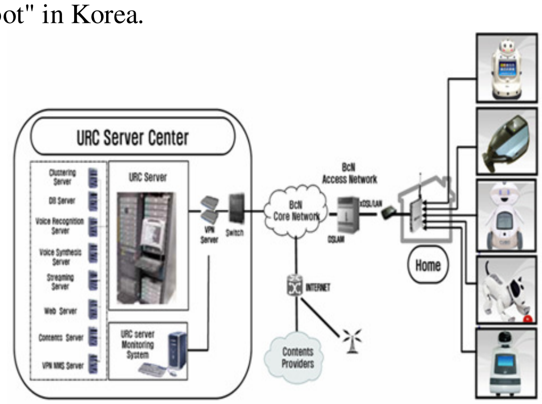


**Fig.1** System structure of the URC field test

Korean Ministry of Information and Communication has taken the lead in related research since 2002, 2003 and ETRI, KAIST, SAMSUNG, and KT are competitively defining and developing the ubiquitous robotics framework. Based on these two concepts, the URC Field Test was conducted in 64 households during 2 months from October to December in 2005 and currently five different mobile robots are being placed in about 900 households and kindergartens in Seoul and its vicinity in the year 2006. Figure 1 shows the system structure of the URC field test for the practical service of the URC server center. In the course of the URC field test, it was proven that the structure of a robot can be simplified by distributing some robot's functionalities to the external server. And it was also found that it is possible to provide various kinds of services which are necessary for daily lives more effectively under the URC concept [8].

### 2.2 Ubiquitous Robot of Samsung Electronics Co., Ltd

The Ubiquitous Robot (UR) is proposed based on the concepts of Ubiquitous Computing (UC) and robot technology [9-12]. UR consists of System Robot, Software Robot, Embedded Robot, and Mobile Robot. UR system is technically defined as the networked robot system incorporating Sobot, Embot, and Mobot via Sysbot in a u-space (refer to Fig.3.1). Each robot has specific individual intelligence and roles, and communicates information via networks. UR will provide users with various services using any device through any network at anytime and anywhere in a u-space. The operation of UR will be seamless, calm, context-aware and networked. For the detail description, refer to [13-14].

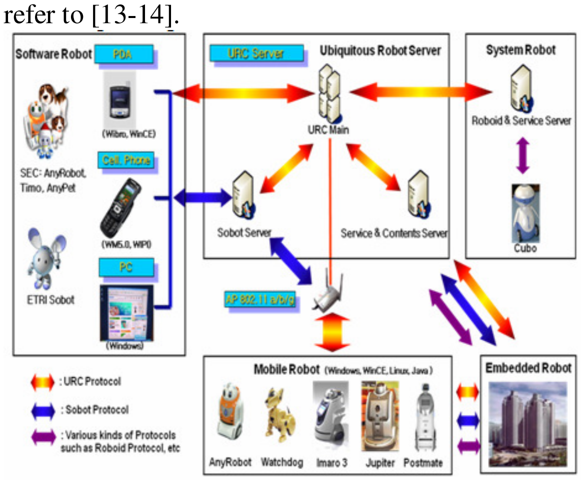


**Fig.2** Ubiquitous robot system

## 3 Motivation and Background

Based on the definition of UR in Section 2.2, the UR of this paper aims for URC, RUPI, and Web 2.0 in the following.

### 3.1 Ubiquitous Robotic Companion: URC

As shown in Figures 1 and 2, the robot can be connected to the communication network to receive some useful information for achieving tasks from the high-performance computer or the sensor-network. If we have a robot with limited capabilities, then that robot is only able to achieve simple tasks or perform basic algorithms. But if we use high performance computer to perform a novel algorithms, then the robot can receive this information from the network. So, we can expect to make a robot more intelligent by using the network server. Conceptually, the network server can be regarded as the robot's remote 'Big Brain' in this network based robot system [15].

The URC requires not only the hardware infrastructure such as ubiquitous networks or sensor networks and high performance computing servers, but also the software infrastructure which resides above the hardware infrastructure. The software infrastructure plays an important role to expand the robot functions and services, to improve the context awareness in the external environment, and to enhance the robot intelligence.

The URC system has several benefits such as sensing capability, processing capability, and network connectivity. For sensing capability, through the network, a robot can recognize environment user's circumstance, because a robot can use information from the external sensor devices which are also connected with network. For processing capability, a robot's functional capability intelligence can be enhanced with the remote 'Big Brain', the network server. And also, because the server can manage the connected robots, robot services can be available in anywhere, at anytime if the network connection is available [15].

### 3.2 Robot Unified Platform Initiative: RUPI

The RUPI is a Robot Unified Platform Initiative and open standard for the network-based robots in South Korea. It was based on the URC project.

In the network-based robot systems such as URC, many robots can be connected with robot server and communicate with the server to request something or response to the server's requests. In order to reduce the errors in communication and to enhance the system's reliability and S/W reusability, it is necessary to standardize the robot's S/Ws such as S/W components, or API, or even communication protocols, etc. So, we want to develop a unified S/W framework for network-based robot system. By doing so, we can reduce the costs and risks associated with building the network-based robot system and also reduce resources & time efforts that are put into the robot system development.

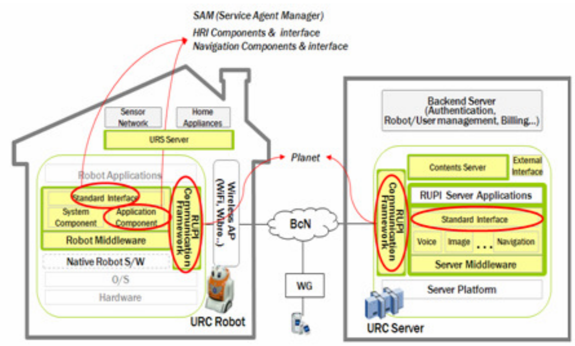


**Fig.3** System RUPI framework concept

Figure 3 shows the overall RUPI Framework concept. Here, we can find some standardization items for the network-based robot system. First item is the communication protocol & interface between URC server and robot. The second item is the interface for the robot S/W Components such as HRI, robot navigation, etc.

### 3.3 Web 2.0

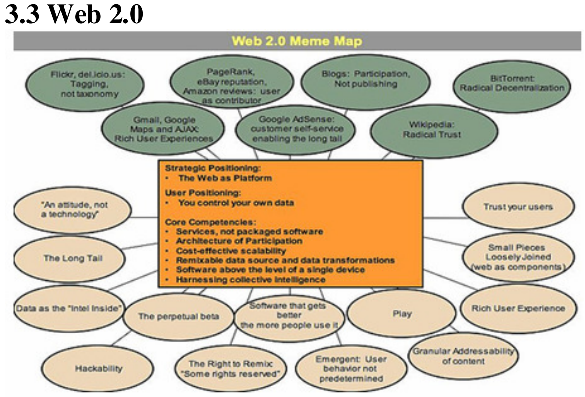


**Fig.4** Web 2.0 Meme map

A new Web 2.0 era is just around the corner. Its motto is the "openness to everyone, the participation of users, and the common ownership of contents." As shown in Figure 4, Web 2.0 is a term often referred to the perceived ongoing transition of the World Wide Web from a collection of websites to a full-fledged computing platform serving web applications to the end users. Ultimately Web 2.0 services are expected to replace desktop computing applications for many purposes [16]. Web 2.0 does not refer to a technology but the web as a platform. Every aspect of Web 2.0 is driven by the user participation. The world of Web 2.0 is only possible through a spirit of openness where developers and companies provide open, transparent access to their applications and contents. A primary goal of Web 2.0 is for the users to control the content they create, the data captured about their web activities, and their identity. This powerful trend is driven by the clear desires of participants [17-21]. In this paper, this Web 2.0 trend is reflected in the Ubiquitous Robot Platform, AnyRobot Studio 1.0 made by SEC.

## 4 Ubiquitous Robot S/W Platform: AnyRobot Studio

AnyRobot Studio is the Robot S/W Package for the network-based robot system, especially for URC system. It can also accommodate Sobot's need for attracting users with more contents and functionalities.

AnyRobot studio supports various robot H/W platforms and provides development environment & tools for building various robot services using resources of the network server. And it will be compliant with RUPI framework. For these purposes, we have developed AnyRobot Studio which consists of Server platform, Robot Platform, and Service authoring Platforms such as AnyMap Studio, AnyAction Studio, and simulator tools.

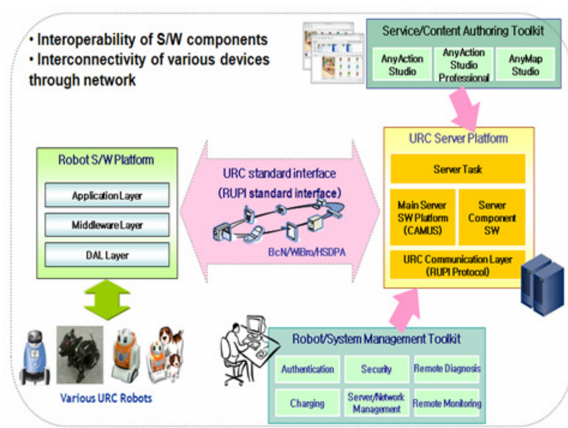


**Fig.5** Configuration of AnyRobot Studio

Figure 5 shows the AnyRobot studio's configuration. Because AnyRobot studio has been inspired by RUPI framework concept, it can support interoperability of each S/W components and interconnectivity of various devices through network, robots, pc and portable device, and so on. AnyRobot studio consists of two Platform, Server platform & Robot platform. And some tools for authoring robotic services and contents, and robot/system management tools. URC Server platform consists of Main Server S/W Platform CAMUS, Server Component S/W, URC Communication Layer and Server tasks. And the CAMUS is developed by ETRI. Robot S/W Platform, GRIS consists of DAL Layer, Middleware Layer, Application Layer, and is developed by SEC.

And these two Platforms are connected with each other through the RUPI framework interfaces.

### 4.1 Context-Aware Middleware for URC Servers (CAMUS)

CAMUS is the Context-Aware Middleware for the URC servers. It is used to develop a context-aware URC application. And through CAMUS, a robot can acquire information from various external sensors and devices, understand the contextual situation, and perform the appropriate tasks.

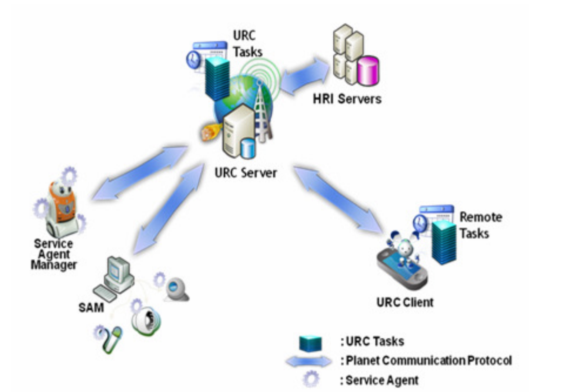


**Fig.6** CAMUS deployment

Figure 6 shows the deployment of CAMUS for URC system. In the center, there is URC Server. Then the clients such as robots, PCs, and several devices are connected to the server, and they must have SAM (Service Agent Manager) on board. Each device connected to the URC Server has its own service agents. For example, robot may have wheel service agent, and the PC may have web cam service agent. Then URC Client can control the robot or devices connected to the URC server to take advantage of their services via the URC server. The URC server delivers URC client's service request to the robot, and then robot performs that service. Here, the communication middleware of CAMUS is PLANET [8].

### 4.2 General Robot Infra Structure (GRIS)

CAMUS is the Context-Aware Middleware for the URC servers. It is used to develop a context-aware URC application. And through CAMUS, a robot can acquire information from various external sensors and devices, understand the contextual situation, and perform the appropriate tasks.

GRIS refers to the General Robot Infra Structure. It is a robot S/W platform developed by SEC. We wanted the S/W platform for developing robot S/W which can satisfy these requirements: it should expedite the development process of the robot application, support various robot H/W platforms, and be easy and independent of OS to eliminate porting issues. Therefore, GRIS should have following features: common robot interface, H/W Independency and multi OS platforms. And GRIS consists of Virtual Message Queue which is the robot middleware, Robot Factory Class (RFC) and GDA which are common robot interfaces, and finally GRIS Remote Interface (GRI) and Behavior Interpreter (BI) which are tools for RFC and GDA. Furthermore, Multi-OS Wrapper is supported.

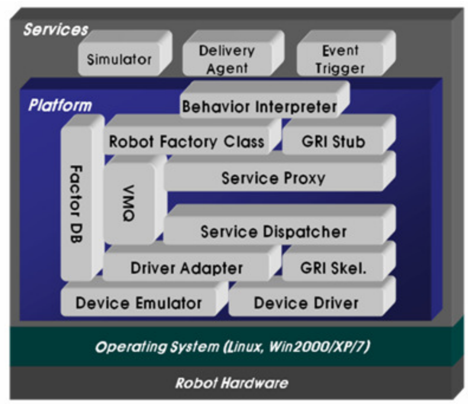


**Fig.7** GRIS architecture

Figure 7 shows the GRIS architecture. The main part of GRIS is VMQ. VMQ is the communication middleware for the purpose of separating robot application layer from the robot H/W device driver layer. To do so, we need two interfaces. One is RFC, and the other is GDA. RFC is the application interface, and GDA is the device driver interface. This way, we don't have to make any modifications to the robot application even when the robot H/W configurations are changed (such as adding some devices or removing them). And modification of robot S/W can be made independently.

### 4.3 AnyRobot Simulator

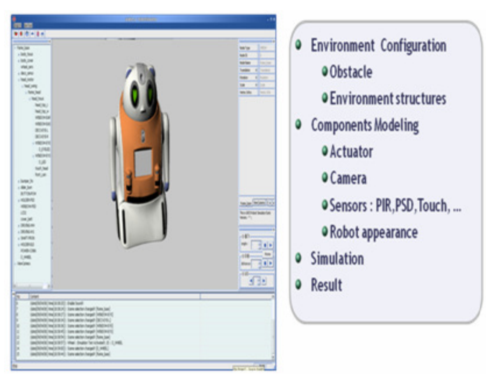


**Fig.8** AnyRobot simulator

AnyRobot simulator is developed for testing real robot program in the absence of the real robot platform as shown in Figure 8. For this purpose, AnyRobot Simulator can simulate robot motion control and sensor data, support behavior scripts, and provide virtual device driver interface (virtual camera, distance sensor, touch sensor, PIR sensor, sound, actuator, etc.) so that the operation of the robot application can be tested beforehand with this simulator and it can later be applied to the real robot without making any modification. In other words, AnyRobot Simulator can simulate how the real robot would operate under the same circumstances and settings. Virtual device driver has been implemented exactly in the same manner as the real device driver interface of GDA.

### 4.4 AnyMap Studio

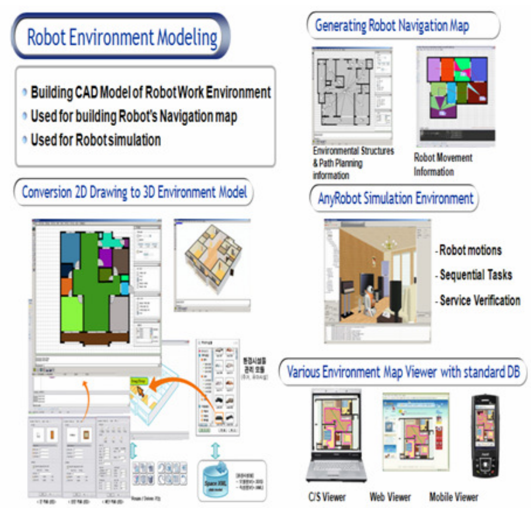


**Fig.9** AnyMap Studio

We have developed several tools for developing robot services and contents, or for making robot map. The first is AnyMap studio. There was a need to develop a tool which can build Robot Working Environment Model. And it should be used for building Robot's Navigation map, and for Robot Simulation. Our AnyMap Studio can convert 2D drawing to 3D Environment Model automatically. And because this 3D Environment Model contains Environment structures & path planning information, we can use this model for Robot Navigation Map information. And also, it can be applicable to AnyRobot Simulator. Furthermore, AnyMap studio has standard DB for Robot Environment model, which can be used in other platforms.

### 4.5 AnyAction Studio

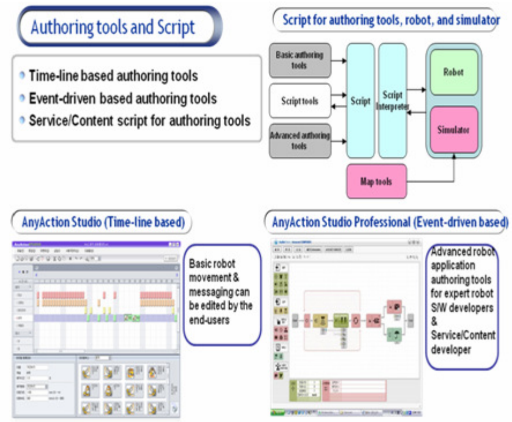


**Fig.10** AnyAction Studio

AnyAction Studio is an authoring tool as shown in Figures 10 and 11. There are two types of AnyAction Studio. The first is AnyAction Studio. It is time-line based authoring tool, and mainly it is used by the end-users. The second type is AnyAction Studio Professional. This is event-driven authoring tool. And because of its complexity and comprehensiveness, it is used by robot application developers and Service/Contents developers.


**Fig.11** Synchronization of AnyRobot Simulator and AnyRobot using AnyAction Studio

### 4.6 AnyAction Studio

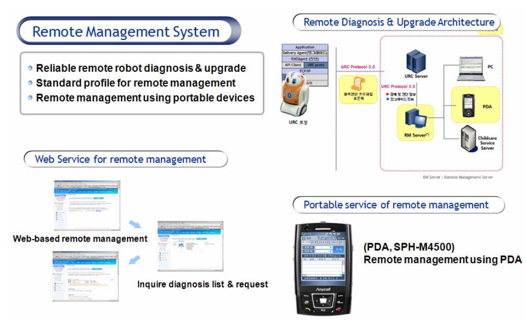


**Fig.12** Remote Management System

We have developed remote diagnosis & software update system. As shown in Figure 12, this remote Management system can provide its remote management function as one of the web services or one of portable services (refer to Figure 13).

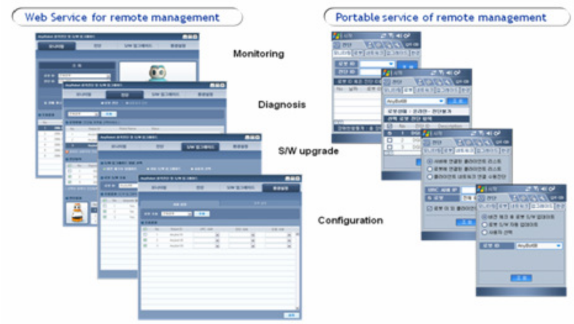


**Fig.13** Ubiquitous methodology of Remote Management System

## 5 Experiment: Ubiquitous Child-Care Robot Service, AnyKids Service

AnyRobot is a child-care robot helping teachers at the educational facilities such as nurseries, kindergartens, or elementary schools[22-25]. We have two types of AnyRobots: AnyRobot I and AnyRobot II, which are developed at Samsung Electronics Co.,Ltd. (SEC) in 2007. Their target users are both teachers and parents, who can have access to and control AnyRobots remotely through AnyDevices. In this paper, AnyDevices mean the GUIs of PCs (web) and PDAs, and behavior motion scripts. The behavior motion scripts including multimedia contents are created from AnyAction Studio for end-users such as parents and teachers, and AnyAction Studio professional for developers and contents providers as mentioned earlier in Section 4.5.

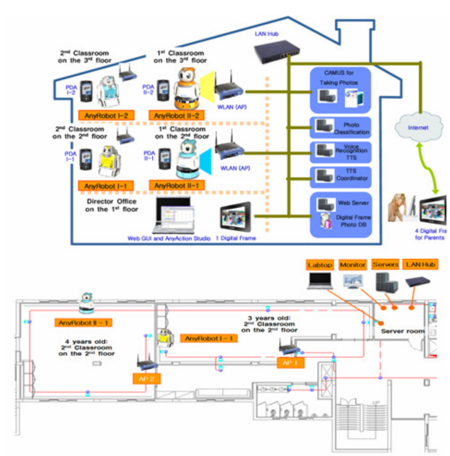

**Fig.14** Structure of AnyKids Service at Samsung nursery

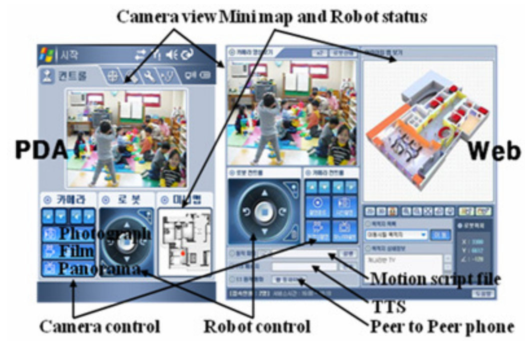


**Fig.15** GUIs of PDAs and PCs for AnyKids service

Figures 14 and 15 shows the structure of the AnyKids service. There are 9 main real-time services using AnyRobots in AnyKids service as follows:

(1) Photograph service: AnyRobots can take photographs of natural appearance or panorama of children and teachers automatically or manually as shown in Figures 16 and 17.

**Fig.16** Automatic photograph service

**Fig.17** Manual photograph service by a teacher with the PDA

(2) Film service: AnyRobots can make a film of natural appearance of children like Photograph service. As shown in Figures 16 and 17, the Film service is similar.


**Fig.18** Remote monitoring service by using Wibro phone

(3) Remote monitoring service: Parents can have access to AnyRobots from outside and monitor their children. Parents can call their children through AnyRobot with their direct voice or AnyRobots' indirect voice (Text-To-Speech; TTS). Users can control the movement of AnyRobots remotely through AnyDevices. Parents can play with children using AnyRobot as their avatar remotely. Refer to Figure 18.

**Fig.19** Event helper service

(4) Schedule manager and event helper service: If parents register their own children schedules such as taking medicine and making them attend piano lessons through AnyDevices, AnyRobots inform the schedule to the teachers. AnyRobots manage the event schedules such as birthday parties or graduation ceremonies and help the teachers by playing UCC or behavior motion scripts. Teachers and Parents can easily create their own interesting contents and motions by using AnyAction Studios. If teachers input the text-messages through PDAs, AnyRobots' TTS system with attractive voice and interesting sound effect can make children pay attention to teachers easily. Figures 15 and 19 show the implementation view.

(5) Voice command service: AnyRobots can understand the voice commands which are registered through the Web GUI. AnyRobots can play the multiplication table game with children by using dialog system.

(6) Personality service: AnyRobots can be an emotional creature with its own personality, which interacts with users like SONY AIBO. Several touch sensors are mainly used as inputs [26-28].

(7) Digital frame service: AnyRobots can upload children's photographs to parents' digital frames. By using the technologies of face recognition and automatic classification of digital contents management (DCM), parents can watch their own children's photographs periodically. Refer to Figure 20.

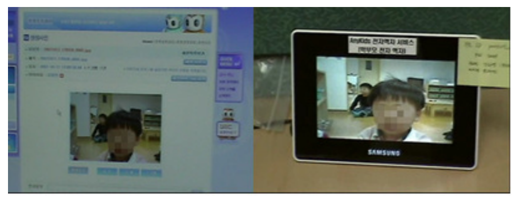

**Fig.20** Digital frame service

(8) AnyKids LAND web portal service: This service provides parents and teachers with the communication channels between them, which mean the web community and the private blogs for children and parents. Refer to Figure 21.

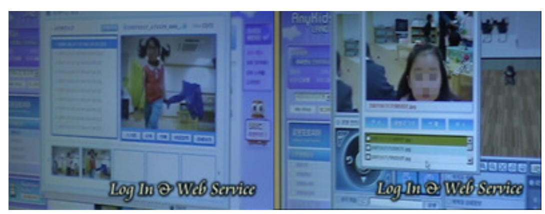


**Fig.21** AnyKids web portal service

(9) Projector service: AnyRobot II can play educational multimedia contents such as nursery songs, flash contents, movie-clips. Teachers can control AnyRobot II using PDA or the control panel on the back directly.


**Fig.22** AnyKids Projector service

The field test for AnyKids service was experimented at Samsung nursery, South Korea, in 4Q. 2007. As shown in Figures 14 and 15, total 120 hours' test was conducted for 2 months. Total 100 children of 3, 4, and 5 years old participated in this experiment. In consequence, most of teachers mainly preferred services 4, 5, 8 and 9 because AnyRobots could save a lot of their effort to take care of loud and inattentive children effectively. In contrast, services 1, 3, 7, 8 were preferred by parents and service 6 was preferred by children. Teachers' average satisfaction rate was generally more than 70 percent but they wanted more intuitive GUI of PDA. Nursery director evaluated the AnyRobot that if the price of AnyRobot is under 2000 dollars, it is worth buying this robot absolutely. Most of parents were very satisfied with telepresence function of AnyKids service and wanted the English tutoring as the most important function among future works...

## 6 Conclusion

This paper introduced AnyRobot Studio which incorporates all the aspects and traits of the URC system as a ubiquitous robot s/w platform. For 4th quarter of 2007, AnyRobot Studio version one was launched and successfully applied to AnyKids service as its application.

As mentioned earlier, we have worked on network-based system, URC, in past several years and shown its feasibility for success. So we hope to standardize some items from our past implementations such as the following: communication protocols & interfaces between URC server and robot interfaces for robot S/W components (HRI, navigation, etc), abstracted robot API for robot programming (Robot Factory Class), 3D robot work environment map DB, robot behavior script, remote diagnosis & software update system API, DB schema, and diagnosis items. We plan to apply AnyRobot Studio for several robots developed by SEC and it will certainly be compliant with the OPRoS (Open Platform for Robotic Services). Because both RUPI and SPIRE (Software Platform Initiative for Robotics Engineering) specifications have been consolidated into the OPRoS [29] since December 2007. OPRoS has developed an operating platform and many development tools for robot systems. The Platform is designed for reusability and fast development of robot software. Its framework consists of an execution engine, component manager, communication manager, and middleware adapter. Additionally, a robot plug-in module, task engine, and task interpreter are included. In addition, for fast development, many kinds of tools are released with their source code. They are the Component Editor, Component Composer, Task Editor, Test Tool, and Simulator. For hardware control programming, the Control Block Editor has also been developed. The convergence of AnyRobot Studio and OPRoS will serve as a momentum to speed up the realization of the omnipotent ubiquitous robot.

## Acknowledgements

This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(grant number: 2011-0015064)..

## References

[1] Robotics & Automation Society Technical Activities Database on Networked Robots: Committee Information, available at http://thth. berkeley. edu/tab-db/committeeinfo. php?tcid=15 and http://faculty.cs. tamu.edu/dzsong/ tc /index. html

[2] The Research Atelier on Network Robot System (NRS), available at http://haydn.upc.es/groups/nrs/ , Barcelona, Spain, April 1, (2005).

[3] Network Robot Forum (NRF), available at http://www.scat.or.jp/nrf/English/ , Sep. 30, (2003).

[4] Ubiquitous Networking Forum, available at http://www.ubiquitous-forum.jp , Nov. 20, (2002).

[5] Norihiro Hagita, Introduction to Network Robot Systems - Research Trends and Perspective, in *IEEE ICRA Workshop on Network Robot System: Ubiquitous, Cooperative, Interactive robots for Human Robot Symbiosis*, Roma, Italy, April 14, (2007).

[6] Ubiquitous Robotics Companion Forum, available at http://www.urcforum.or.kr, Sep. 17, (2004).

[7] Ubiquitous Robotics Companion, available at

http://www.urckorea.com/

[8] Hyun Kim, Young-Jo, and Sang-Rok Oh, Implementation and application of URC server framework, in *IEEE ICRA Workshop on Network Robot System: Ubiquitous, Cooperative, Interactive robots for Human Robot Symbiosis*, Roma, Italy, April 14, (2007).

[9] Jong-Hwan Kim, IT-based UbiBot, in *the Issue of the 13th of May, 2003, The Korea Electronics Times, Special Theme Lecture Article*, Seoul, Korea, May (2003).

[10] Kang-Hee Lee, Yong-Duk Kim, Jun-Su Jang, Jong-Hwan Kim, Network-based software robot for implementation of ubiquitous robot, in *Proceedings of the ITRC Forum*, Jun. (2004) 204.

[11] Hyun-Sik Shim, Jongchang Lee, Tae-Hoon Kim, Hyun Kim, and Kang Woo Lee, Communication protocol for the URC robot and server, in *1st Japan-Korea Joint Symposium on Network Robot Systems*, Nov. (2005).

[12] Kang-Hee Lee, Ubiquitous Robotic Game incorporating Hardware and Software, *Journal of Future Game Technology, 1, 1(2011) 35.*

[13] Kang-Hee Lee, Kwang-Choon Kim, Hyun-Sik Shim, Joon-Koo Kim, and Jong-Chang Lee, Ubiquitous Robot S/W Platform: AnyRobot Studio 1.0, in *IEEE IROS Workshop on Ubiquitous Robotic Space Design and Applications*, San Diego, USA, Nov. 2, (2007).

[14] Kang-Hee Lee and Jong-Hwan Kim, Ubiquitous network-based software robot system with own Internet Protocol address, *patent no. KR 2006-0003255, application no. 10-2004-0052073, Korean patent*, July (2004).

[15] Hyun Kim, Young-Jo Cho, and Sang-Rok Oh, CAMUS: A middleware supporting context-aware services for network-based robots, *IEEE Workshop on Advanced Robotics and its Social Impacts* (2005).

[16] Web 2.0, available at http://en.wikipedia.org/wiki/Web 2.0

[17] Web 2.0 Framework, available at http://www. Rossdaw sonblog .com/ Web2 Framework.pdf

[18] What Is Web 2.0: Design Patterns and Business Models for the Next Generation of Software, available at http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html?page=1

[19] Young-Kwon Bae, Exploring of the Direction of Design on Educational Contents of Smartphone Based, *Journal of T he Korea Knowledge Information Technology Society*, 5, 5 ( 2010) 43.

[20] Kang-Hee Lee, Development of Icon-based Educational Robot Kit, Journal of The Korea Knowledge Information Technology Society, 5, 5 (2010) 71.

[21] Shin Jin Kang, Soo Kyun Kim, Sung Hyun Cho, Procedural Game Level Generation with Interactive Evolutionary Computation, Journal of Future Game Technology, 1, 1(2011) 9-20.

[22] PaPeRo, available at http://www.nec.co.jp/products/robot/en/index.html

[23] IRobi, available at http://www.edurobot.net

[24] Susan Karlin, Caregiver Robots: A scientist creates robots that help children, *IEEE spectrum,* Feb. (2010).

[25] Hee-Ryong Park, Implementation of Home network Monitering System Using a Vacuum Robot, Journal of The Korea Knowledge Information Technology Society, 4, 2 (2009) 7.

[26] Jong-Hwan Kim, Chi-Ho Lee, and Kang-Hee Lee, Evolutionary Generative Process for an Artificial Creature's Personality, *IEEE Transactions on Systems, Man, and Cybernetics – Part C*, 39, (2009) 331.

[27] Kang-Hee Lee, Evolutionary Generative Algorithm for Genetic Robot's Personality, *Ph,D. Dissertation, KAIST,* (2006).

[28] Kang-Hee Lee, Genetic Robot Platform and Genetic Robot Behavior Manifestation Method, *patent no. KR 10-0866212, Korean patent*, Oct. (2008).

[29] OPRoS, available at http://www.opros.or.kr.

Kang-Hee Lee received the B.S., M.S., and Ph.D degrees in electrical engineering and computer science from Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 1999, 2001, and 2006, respectively. Since 2006, he has been a Senior Engineer in Digital Media & Communication Research Center, Samsung Electronics Company, Ltd., Korea. He has been a dispatched researcher in the Robotics Institute, Carnegie Mellon University in 2008. Since moving to Soongsil University in 2009, he is with Global School of Media, Soongsil University, Seoul, Korea. His current research interests include the areas of ubiquitous robotics, evolutionary robotics, emotional robotics, media

robotics, cognitive task planning system, and knowledge-based reasoning system. Contact him at kanghee.lee@ssu.ac.kr. .

Kyoungsu Oh received the B.S., M.S., and Ph.D degrees in computer science from Seoul National University, Seoul, Korea, in 1994, 1996, and 2001, respectively. From 2001 to 2002, he worked at ZOIMENT where he developed several games. Since 2009, he joined Global School of Media, Soongsil University, Seoul, Korea. His current research interests include robotics, computer games, computer graphics. Contact him at oks@ssu.ac.kr .