## Table of Contents

### Keynote Speech
- Sensor Networks Specification: an Introduction to Active Sensor Processes  
  [Jin Song Dong, Jun Sun] / 13

### Ad hoc, RFID, and Sensor Networks
- Mesh Routing Schemes for Wireless Sensor Networks  
- A load distribution scheme based on the black classification for ad–hoc sensor networks  
- A Localized Algorithm for Reducing the Size of Dominating Set in Mobile Ad Hoc Networks  
  [Yamin Li, Shiliung Peng, Waming Chu] / 39
- Code structure study for optimization in RFID applications  
  [JungJung Na, HyounDeock Yim, KItaek Cho, Weon Kim, Kwan–Ho Song] / 47

### Wireless Communication and Networks
- An Improved TDOA Positioning Method for CDMA Networks  
  [Dishod Akbarov, Heedong Choi, Yongwan Park, Wooyoung Jung, Sangdong Kim] / 57
- Comparison of Address Allocation Schemes for Scale-Free uPAN Networks  
- MAC Layer Technique for Cognitive Radio : A Survey  
- Wireless Communication and Networks  
  [Khulanazarova N,Sh.] / 75

### Mobile and Embedded Software Engineering
- A Virtual Development Environment for Mobile Software Development  
  [Sang–Young Cho, Yoojin Chung, Jeong–Bae Lee] / 81
- Mobile Devices in Requirements Elicitation Activities  
  [Syed Ahsan Fahmi, Ho–Jin Choi] / 86
- Design and Implementation of a DM Adapter for a Mobile Device Management Agent  
  [Ju Geon Pak, Kae Hyun Park, Dae Jin Jang, Myung Sook Jang, Jong Jung Woo] / 90
- Roundoff Errors at a Parallelizing of Evaluations by Systems of Basic Splines  
Mobile Computing Technology, Services, and Applications

- Enhanced Asynchronous UWB Indoor Positioning Algorithm
  [Jaesuk Cho, Junghwan You, Yongwan Park, Wooyoung Jang, Jouhun Lee] / 101

- Location Estimation based Personalization using Support Vector Machine and Signal Strength of Mobile Phone
  [Seong Jin Cho, Sungyoung Lee] / 108

- Service Adaptation Middleware based on Cognitive Lattice for Mobile Computing
  [Svetlana Kim, Yong-ik Yoon] / 113

- Study in implementation of CoN Number Portability System based on ENUM DNS
  [Jongyoon Ra, Sungwoo Shin, Kwan-Ho Song] / 119

Mobile Platforms, Standards, and Internet Computing

- User and Contents through Single-Sign-On Certification platform Design
  [Jinhyung Kim, Jun Hwang, Taemin Kim, Byungwook Lee] / 127

- Related Technology and Standardization Tendency of Mobile 2.0 and Service Plan
  [Hyunsang Park, Sungsup Kang, Imchul Kang, Chanjung Park] / 133

- Web Anomaly Traffic Detection for Mobile Clients using a Web Traffic Flow Analysis Model
  [Sung-Min Jang, Yoo-Hun Won] / 139

- Survey on the Terms that Relating to the Internet Address Resources

Mobile Security

- Security Key Management Mechanism for Group Distribution
  [Zonghua Liu, Guangqing Liu, Bohyun Wang, Byungwook Lee, Myung-Mook Han] / 153

- Threats and Assumptions of Embedded Software Development Site
  [Sook-soo Kim, Song-soo Yoo, Taihoon Kim, Gil-cheol Park] / 158

- The Adaptive Anomaly Detection for the Reduction of False Alarm Rate on the NIDS
  [Mi-Sun Kim, Min-Seok Kim, Kyung-Woo Park, Jae-Hyun Seo] / 163

- A Modification of Role Hierarchy for Flexible Database Security System
Poster Session

- A Location Independent Coordination Scheme Based on Teamwork for Mobile Agent System [YunHee Kang] / 177
- Decision Policy for Adaptation level of Graphic contents [Hyukran Kim, Yongik Yoon, Hwajin Park] / 188
- Experiments of Security Vulnerabilities for Extension Headers on Mobile IPv6 [Jeong-Wook Kim, Gil-Jong Mun, Jae-Hyun Seo, Yong-Min Kim, Bong-Nam Non] / 199
- A Study for Better Understanding of Senior’s Daily Activity Based on a Photo-sensor Network [Seung Ho Cho, Bonghee Moon, Li Zhengxi] / 208
- Design of Security Policy Model for Privacy in Personal Information System [Ik-Su Park, Jo Yong Seon, Eeyeong-Kyun Oh] / 210
- A Link-Layer Broadcasting Scheme for Scalable Ubiquitous Personal Area Networks (UPANS) [Sungroo Cho, Wonsuk Choi, Yun-Jae Wo, Seung-Ok Lim, Jin-Woong Cho] / 214
- Challenges for Web Annotation Systems on Mobile Web [Ahmad Ibrahim, Ho-Jin Choi] / 219
- Addressing the Privacy issue in Pervasive Environment through the Use of Composite Approach [Enkhbold Nyiamsuren, Ho-Jin Choi] / 223
- Secure Execution Guarantee Scheme for Agent Privacy from Traitors [Chang-Ryul Jung, Yo-Han Kim, Jin-Wang Koh, Sung-Keun Lee, Hyung-Dae Koh, Chang-Seok Kim, Pyoung-Kee Kim, Soo-Cheol Ha] / 228
- Application of DDC and Bluetooth for Mobile IT Devices [Joohoon Park, Ryumduck Oh] / 238
- Technology of Personalized Service for a Digital Frame [Sung-Han Bae] / 243
An Extension of Executable UML for Modeling u-Home Network: Mapping HCI with SE

Woo Yeol Kim, Hyun Seung Son, R. Young Chul Kim
Dept. of Computer & Information Comm., Hongik University, Jochiwon, 339-701, Korea
{join, sonhs, bobj}@selab.hongik.ac.kr

Abstract

Exactly to develop the customer’s needs with the new coming ubiquitous mechanism, it might need the new development method, which is focused on user centered analysis instead of system or developer centered development. Therefore, this paper attempts to graft SE (software engineering) into HCI (Human Computer Interface) for developing the future ubiquitous related systems to be needed new paradigm against the current existing system developments.

To do this, we suggest the user behavior analysis to develop products of new paradigm, and extend the standardized software modeling language, UML (Unified Modeling Language), to model based on the user centered behavioral analysis for the future systems on Ubiquitous environment.

To illustrate the proposed approach we use a modeling example of u-Home Control System with our extended UML.

1. Introduction

In near future's ubiquitous environment [1], it will be very important to estimate user demands and to develop a new product through modeling based on the user behavior analysis. Just even within one small domain, it will be happened a lot of diverse observable user data with which we be hard to analyze user behaviors. We cannot model with these behavior data. So, if just collect user behaviors with satisfying the particular goal, we get the user behaviors limited, but satisfied. To analyze behaviors which satisfy the user’s particular goal on these data, it will protect to occur errors during developing new ubiquitous paradigm’s product [2]. Then it will be possible to model U-home control system with the goal oriented user behaviors. Also it may be applied with adding or improving new functions of new appliances.

To analyze and model user behavior data, it will be necessary to have the modeling language on the view of HCI. Also many researches are in progress to reduce a gap between SE (Software Engineering) and HCI (Human Computer Interaction).

So, it is necessary to unify a standardized common language to deal with comprehend representation for these two fields [3]. There is more focused on modeling the system to represent the system structure in the view of SE, while it makes clues of the application to disclose the behaviors of the system in the view of HCI. Paula [4] suggests new modeling language to deal with HCI-SE model with Interaction modeling language. Kim [3,5,9] also suggests the user behavior analysis methodology to develop products of new paradigm at the coming ubiquitous age. Through this methodology, Kim [3,5,9] mentions to analyze the user behaviors to achieve the same purpose within a particular system (or an environment), then extracting common / uncommon behaviors, and identify objects on user behavior scenarios. With this information, Kim [3,5,9] may model the system of new paradigm which will come.

In this paper, section 2 describes related work. Section 3 suggests the extension of UML for modeling the user behavior: Mapping HCI and SE. Section 4 shows one example to model u-Home Control System with our proposed
2. Related Works

It tries to have the modeling language and tool on the view of HCI to analyze and model user behavior data. There is more focused on modeling the system to represent the system structure in view of SE, while it makes clues of the application to disclose the behaviors of the system in view of HCI. Especially the User Process Based Product Architecture (UPPA) is focused on evaluating to represent the functional relationship between the user and the system [11,14].

Also many research are in progress to reduce a gap between SE (Software Engineering) and HCI (Human Computer Interaction).

So, it is necessary to unify a standardized common language to deal with comprehension of representation for these two fields.


2.1. The original UML (Unified Modeling Languages)

Booch and Rambaugh combined the concepts from the OMT and the Booch method at Rational Software Corporation in 1994. Jacobson also joined to work together. Their jointed work was called the Unified Modeling Language (UML). The Unified Modeling Language (UML) is a general-purpose visual modeling language that is used to specify, visualize, construct, and document the artifacts of software system [6,7,8]. The UML [10,12,15] captures information about the static structure and dynamic behavior of a system. The static structure defines the kinds of objects important to its implementation and the relationships among the objects. The dynamic behavior defines the history of objects over times the communications among objects.

In this time, it is very popular modeling language UML on diverse fields, especially software engineering.

There are three prominent parts of a system's model:

- Functional Model
  - the functionality of the system from the user's point of view.
  - Includes Use case diagrams.
- Object Model
  - the structure and substructure of the system using objects, attributes, operations, and associations.
  - Includes class diagrams.
- Dynamic Model
  - the internal behavior of the system.
  - Includes sequence diagrams, activity diagrams and state machine diagrams.


We suggest extending UML for modeling the user behavior centered system. Extended UML [13] has the three extended diagrams, such as Class Diagram, Concurrent Message Diagram, and Concurrent State Diagram, for modeling the user behaviors.

Table 1. Notation of Actor, Class, Object, Role, and Rule

<table>
<thead>
<tr>
<th>Element</th>
<th>Notation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Actor</td>
<td></td>
</tr>
<tr>
<td>Object</td>
<td></td>
</tr>
<tr>
<td>Interface</td>
<td></td>
</tr>
<tr>
<td>Role</td>
<td></td>
</tr>
<tr>
<td>Event</td>
<td></td>
</tr>
<tr>
<td>Condition</td>
<td></td>
</tr>
<tr>
<td>Action</td>
<td></td>
</tr>
</tbody>
</table>

3.1. Class Diagram (CD)

CD describes the static structure of a system with role and rule mechanism. Its basic class may have the template types of role such as recognizer, decision, communication, and transaction. Also rule concept in the class is followed by ECA (Event/Condition/Action) in table 1.

3.2. Concurrent Message Diagram (CMD)

We also extend the basic sequence diagram with concurrent mechanisms (such as fork-join, reverse fork-join, etc) for handling real things. Table 2 describes the CMD notations of Extended UML.
In Extended UML, the object in concurrent message diagram also is followed by Ivar Jacobson's stereotype such as interface, control, and service object. Interface object (or boundary object) just transfers a message to control object without any state. Control object makes a decision and mediates (or controls) between interface and service object. Service object transacts with data needed. Like class's role, its object may or may not have one or more roles of recognizer/ decision/ communication/ transaction.

### Table 2. The CMD notation of the Extended UML

<table>
<thead>
<tr>
<th>Element</th>
<th>Notation</th>
<th>Notation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Incoming</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Choice</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ForkJoin</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Communication</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Timeout</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

In other words, the object has role(s) based on which ECA (Event/ Recognition/ Communication/ Decision/ Transaction). Each object may or may not have zero or more role(s). Also the object is included with ECA (Event/Condition/action) rule. For example, one object comes in an event, checks the condition, and then acts some service. Basically our approach is adopted synchronized message passing mechanism in CMD. And we include time delay that the message is received in time.

### 3.3. Concurrent State Diagram (CSD)

Still on our research, CSD will be basically followed by nested mechanism with OCL(Object Constraint Language). This diagram will contain some mechanisms for deterministic/stochastic system. Next time, we will extend automatically to check and convert from nondeterministic to deterministic state diagram.

### 4. Case Study

We shows one modeling example of “u-Home control System” with our extended UML.

#### 4.1. Static Modeling using Class Diagram

Figure 1 shows the static modeling of u-Home Control System based on 'Fire prevention' use case scenario. In this class diagram, the system controller is important to control the u-Home environment. The Controller has the association relationship with Beeper, GasRange, PowerSupply, and Sensor class. The controller class controls all devices in the system.

![Fig. 1. Class diagram of u-Home](image)

#### 4.2. Behavior Modeling using CMD

Figure 2 shows the dynamic modeling of u-Home Control System based on 'Fire prevention' use case scenario. In this step, we use CMD (Concurrent Message Diagram). For example, when happen a fire in the u-Home, HeatSensor and/or SmokeSensor will sense the symptom of a fire, and then send this information to the controller. In figure 2, $\textcircled{1}$ means to receive one message from either one sensor or other one, that is, the mechanism of OR gate. Then send this message to the controller. $\textcircled{2}$ means that the controller makes a decision with message received, and sends the asynchronous broadcasting control messages (such as alarm and display message) to the beeper. $\textcircled{3}$ and $\textcircled{4}$ means that the controller concurrently send messages to GasRange and PowerSupply for extinguishing the fire, that is, the mechanism of AND gate.
4.3. Behavior Modeling using CSD

Figure 3 shows the behavioral change of the controller in time with Concurrent State Diagram (CSD) of u-Home.

6. Conclusion

This paper attempts to map SE (software engineering) with HCI (Human Computer Interface) for developing the future ubiquitous related systems to be needed new paradigm against the current existing system developments. We also extend the standardized software modeling language, UML (Unified Modeling Language), to model based on the user behavioral analysis for the systems of the new coming Ubiquitous environment.

We model the ‘Fire prevention’ use case scenario of u-Home Control System with our extended UML. In near future, we will research about extracting the rules and/or the services based on the user behavior analysis.

7. References
[14] Hong-Ju Bae, A study on comparison of the method for structuring the user behavior - Focus on the user behavior in home, MS degree, 2006.