An Access Control Agent based on MHAP for Festival Site

Soongohn Kim and Eungnam Ko

1 Division of Computer and Game Science, Joongbu University, 101 Daehakro, Chubu-Meon, GumsanGun, Chungnam, 312-702, Korea sgkim@joongbu.ac.kr

2 Division of Information & Communication, Baekseok University, 115, Anseo-Dong, Dongnam-Gu, Cheonan, Chungnam, 330-704, Korea Corresponding Author: ssken@hanmail.net

Abstract. The focus of situation-aware ubiquitous computing has increased lately. This paper proposed a new model of access control based on situation-aware ubiquitous computing that offers a seamless view without interfering with concurrency control is also suggested. Currently, the regional local festivals, which are a part of community development projects that utilize the regional cultural resources of the local self-governing entities, contribute in the nurturing of local cultures. This paper describes a FACA_M (Festival Access Control Agent based on MHAP). It is a multi-agent based MHAP with function of an event access control for person’s moving line in festival site automatically. (Key words: situation-aware ubiquitous computing, regional local festivals, festival event access control, person’s moving line, MHAP, FACA_M)

1 Introduction

The development of middleware is closely related to the evolution of ubiquitous computing began in the mid of 1970s, when the PC first brought computers closer people. With the advent of networking, personal computing evolved into distributed computing [1, 2]. Context awareness (or context sensitivity) is an application software system’s ability to sense and analyze context from various sources; it lets application software take different actions adaptively in different contexts [3]. In a ubiquitous computing environment, computing anytime, anywhere, any devices, the concept of situation-aware middleware has played very important roles in matching user needs with available computing resources in transparent manner in dynamic environments [4, 5]. The smart festival management system is a management system that, for the various festivals that are operated by the local self-governing entities and agencies, enables a direct operation of all the process phases from the advance preparation phase to operation phase and the post management and the administrative tasks, etc. of the planning operational headquarters and agencies [6].

Thus, there is a great need for event access control agent in situation-aware middleware to provide dependable services in ubiquitous computing. This paper
proposes a new model of festival event access control agent running on situation-awareness ubiquitous computing such as MHAP.

2 MHAP Environment

As shown in figure 1, MHAP has four layered architecture [7]. The physical device and network layer consists of any network and physical device supporting any networking technology. The infrastructure layer introduces infrastructure to provide service management and deployment functions for MHAP services. The MHAP layer consists of MHAP services and provides functionalities constructing HA, which includes event notification, appliance control, HA rule configuration and device management. It uses MOM to support event-driven HA in heterogeneous environment. Facilitating Home Automation needs many different kinds of applications. There are DOORAE agent layer between application layer and MHAP service layer.

Fig.1. The organization of MHAP

3 Our Proposed Approach: FACA_M based on MHAP for Festival Sites

3.1 The Environment for Festival Event Access Control Agent: DOORAE

DOORAE services have many agents. They consist of AMA(Application Management Agent), MCA(Media Control Agent), ESA(Event Sharing Agent), SMA(Situation-Aware Session Management Agent), and ACA(Situation-Aware Access Control Agent).

As shown in Figure 2, when many festival users request for same festival media device at the same time, festival media device acquisition order is controlled.
Alternation of festival event input sequence, according to present load, processing capability of each system and network delay cause by participants’ use of different computers is another problem that occurs in festival distributed network environment.

3.2 The Algorithm for FACA_M

We attempted to solve such festival command serialization problem with centralized serialization server and distributed synchronization clock method. In order to guarantee festival synchronization control and command serialization, FACA_M maintains and manages festival command sequence history that is mutually exchanged. All festival input events are transmitted with creating time of event to the serialization server. Serialization server can be processed according to the order of occurrence of festival events. Command serialization occurs because of necessity to process inputs from several users of different locations using same festival application program and to see festival same view. Collaborative work system must have festival command serialization mechanism. In order to overcome the festival problem mentioned earlier, while the user carries out a festival serialization of the object that is created in a shared window, a local window that is in coincidence with the scope is bound to the object. This is the festival local image copy of a festival shared object that is to be created. By showing the user the abstract of a festival input command beforehand, the time that is needed to confirm whether the festival object is managed properly can be saved.

As shown in Figure 3, access control inevitably occurs in a multimedia distance education environment where many users perform collaborative work at the same time. There may be a case where processing cannot be done in order of arrival due to variation according to present load, processing capability of each system and network.
delay caused by many participants putting in commands. Access control solves such problems.

### Access Control based on MHAP

<table>
<thead>
<tr>
<th>INITIATOR</th>
<th>PARTICIPANT</th>
</tr>
</thead>
<tbody>
<tr>
<td>(1) Issue commands with its session id, timestamp, sequence number and event type</td>
<td>(2) Broadcast serialized commands</td>
</tr>
<tr>
<td>(3) Start processing the command and awareness</td>
<td></td>
</tr>
</tbody>
</table>

*Fig. 3. Access Control based on MHAP*

### 4 Simulation Results

To evaluate the performance of the proposed system, an error detection method was used to compare the performance of the proposed model against the conventional model by using DEVS formalism. In DEVS, a system has a time base, inputs, states, outputs based on the current states and inputs. DEVS (Discrete Event System Specification) is a formalism of being developed by Bernard P. Zeigler. The structure of atomic model is as follows [8 - 13].

\[
M = \langle X, S, Y, \delta_{int}, \delta_{ext}, \lambda, t_a \rangle 
\]

- X: a set of input events.
- S: a set of sequential states.
- Y: a set of output events.
- \(\delta_{int}\): internal transition function
- \(\delta_{ext}\): external transition function
- \(\lambda\): output function
- \(t_a\): time advance function

Because this mechanism is basically structured in a way that it is easy to manage the object and window’s private window into shared workspace conference environment, and this also means that it supports the late comer is easier way. Since the mechanism that is suggested in this paper basically uses two windows, it is advisable to accept this as metaphor. The user can use local window as if its own local window image is being mapped or projected to multiple user’s shared space or screen. The method adopted by the DOORAE makes network delay look like the time that is
needed to project the image. Different from the mechanism that makes use of two windows, the unified window can be employed with a timer for the automatic destruction of failed inputs.

As shown in Table 1, you can see the characteristic function of each system function for access control based on MHAP. A proposed main structure is distributed architecture but for application program sharing, centralized architecture is used. Basically, there are two architectures to implement such collaborative applications; the centralized architecture and replicated architecture, which are in the opposite side of performance spectrum.

### Table 1. Comparison for Software Architecture

<table>
<thead>
<tr>
<th></th>
<th>Centralized</th>
<th>Replicated</th>
<th>Hybrid (proposed)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set initial State</td>
<td>easy</td>
<td>hard</td>
<td>Medium</td>
</tr>
<tr>
<td>Command Serialization</td>
<td>easy</td>
<td>hard</td>
<td>Easy</td>
</tr>
<tr>
<td>Communication Overhead</td>
<td>high</td>
<td>low</td>
<td>Low</td>
</tr>
<tr>
<td>Performance</td>
<td>bad</td>
<td>good</td>
<td>Good</td>
</tr>
<tr>
<td>Control Complexity</td>
<td>Low</td>
<td>high</td>
<td>High</td>
</tr>
</tbody>
</table>

5 Conclusions

The development of multimedia computers and communication techniques has made it possible for a mind to be transmitted from one festival site to another festival site in distance environment. This paper proposed a new model of festival event access control agent running on situation-awareness ubiquitous computing such as MHAP. The focus of situation-aware ubiquitous computing has increased lately. An example of situation-aware applications is a multimedia person’s moving line for festival sites. FACA_M is a system that is capable of accessing software event control for distributed multimedia environment running on MHAP for festival sites. We have given a detailed discussion of FACA_M, a suit of an event access control system that ensures the festival continuous applications. In this paper, we have discussed a method for enhancing performance through a quick festival event access control for distributed multimedia environment running on MHAP for festival sites.
References