

# U-캠퍼스 환경에서의 동적 접근 제어 메커니즘

조창제\*, 김영갑\*\*

\*고려대학교 컴퓨터정보통신대학원 소프트웨어공학과

\*\*고려대학교 컴퓨터학과 소프트웨어 시스템 연구실

e-mail : ccj612@korea.ac.kr, ygkim@software.korea.ac.kr

## Dynamic Access Control Mechanism in U-Campus Environment

Chang-Je Cho\*, Young-Gab Kim\*\*

\*Dept. of S/W Engineering, Graduate School of Computer & Information Technology, Korea University

\*\*Software System Lab., Computer Science & Engineering, Korea University

### 요 약

학교생활에서 학사업무의 생산성은 물론 구성원들의 편리하고 효율적인 학습활동 지원을 통하여 학습자의 교육 생산성을 높이기 위한 유비쿼터스 컴퓨팅 기술을 이용한 U-캠퍼스 구축이 요구된다. 따라서 본 논문에서는 U-캠퍼스 환경에서 여러 가지 상황 정보를 고려한 동적 접근 제어 메커니즘을 제안한다. 본 논문에서 제시한 동적 접근 제어 메커니즘은 기존의 역할 기반의 접근 제어(RBAC) 모델을 확장하여 사용자에게 부여된 역할 활성화 상태에 따라 접근 권한을 부여하는 방법으로, 상태검사 에이전트(agent)와 상황인식 에이전트, 상황인식 매트릭스(CAM)에 의한 위치와 시간 및 시스템 자원과 같은 여러 가지 동적인 상황 정보를 고려하여 효율적으로 권한을 부여 할 수 있다.

### 1. 서론

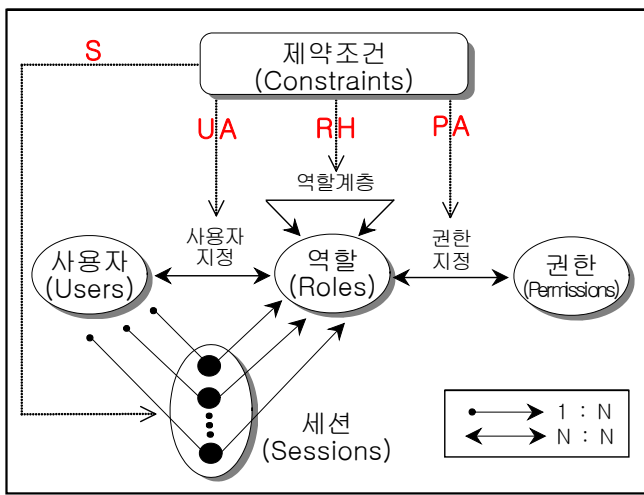
가정과 사회에서 정보통신의 발달과 더불어 사용자들은 PDA나 인터넷 모바일 폰과 같은 휴대용 개인 정보 단말기를 이용하여 언제 어느 곳이라도 쉽게 정보를 접할 수 있고 유비쿼터스 컴퓨팅(ubiquitous computing) 기술도 그것들과 더불어 급속히 발전하고 있다. 이러한 유비쿼터스 컴퓨팅 환경에서는 일상생활 곳곳에 편재되어 있는 컴퓨팅 자원과 이것들 사이의 유기적인 연결을 통해서 사용자의 의도와 상황 정보를 파악하여 개별화된 서비스를 사용자에게 제공할 수 있다[1]. 또한 개별화된 서비스를 제공 받기 위하여 주변 환경으로부터 생성된 자료와 이것을 수집하여 제공된 상황 정보(context information)가 중요한 역할을 한다[2]. 이러한 것을 기반으로 유비쿼터스 컴퓨팅 기술을 교육환경에 접목 하게 되면 교육현장이 진화 되어 나갈 수 있을 것이다. 이러한 유비쿼터스 컴퓨팅 기술은 학교생활에서 컴퓨터와 네트워크 기술을 이용하여 학사업무의 생산성과 구성원들의 편리하고 안전한 활동을 지원하며 학습자

의 교육 생산성을 높일 수 있다. 그러나 학교 내의 사용자들은 많은 시간 동일한 서비스와 자원을 사용하게 되는데 조직 내의 역할 및 상황 변화에 따라 개별화된 서비스와 자원을 제공받지 못하고 있다. 다시 말해서, U-캠퍼스 환경에서 사용자의 상태, 즉 사용자의 위치 및 시간 또는 시스템 자원에 따라 사용하려는 서비스 또는 자원과 접근할 수 있는 권한을 상황변화에 따라 동적으로 부여하는 연구가 미비한 상태이다. 따라서 본 논문에서는 이러한 문제점을 해결하기 위하여 U-캠퍼스 환경에서 상황 정보를 고려한 동적 접근 제어 메커니즘을 제안하고자 한다.

### 2. 연구배경 및 관련연구

일상생활에서 발생하는 많은 상황들을 컴퓨팅 환경에서 정보로 활용하기 위하여 많은 노력을 하여왔다. 항상 변화되는 상황 정보를 능동적으로 얻어내기 위하여 상황인식(Context-Awareness)을 최초로 시도한 것은 Olivetti Research사의 액티브 배지(Active Badge)일 것이다[3]. 액티브 배지를 착용함으로써

사용자의 위치를 탐지하고, 사용자의 상황을 파악하여 정보로 활용할 수 있다. 이러한 방식으로 제공되어진 사용자나 사용자 주위의 상황에 대한 모든 정보의 값을 상황 정보라 하고 이러한 상황 정보를 사용자나 사용자 주위의 환경으로부터 얻어 내는 과정을 상황인식이라고 한다. 상황인식 어플리케이션은 사용자의 입력정보에 대하여 사용자의 상태, 물리적 환경, 컴퓨팅 자원, 기존 정보를 통한 분석 등 다양한 상황 정보를 통해 사용자의 상황에 대한 결과를 제공한다[4]. 현재의 U-캠퍼스 환경에서는 구성원들의 PDA나 휴대용 단말기 등 모바일 도구나 대화형 스마트카드 등이 활용하기 쉬운 자원들이다. 이러한 어플리케이션을 사용함으로써 구성원들의 동적인 상황 정보가 발생하게 되고, 발생한 상황 정보를 이용하여 적절한 권한을 부여할 수 있어야 한다. 현재 접근 제어 메커니즘은 접근 제어 리스트(Access Control List, ACL)나 역할 기반 접근 제어(Role-based Access Control, RBAC)에 기반을 두고 있다[5]. RBAC 모델은 조직 내에서 구성원들이 가지고 있는 직무나 행위에 관련된 역할을 기반으로 접근권한을 결정하는 방법으로 접근 권한은 역할에 의해 결정되고 조직의 구성원인 사용자는 여러 가지 역할을 부여 받을 수 있다. 그러나 U-캠퍼스 환경에서 응용하고자 할 경우 위치와 시간 및 시스템의 자원 등과 같은 상황 정보를 고려해야 하는데 기존의 접근 제어 메커니즘은 이러한 것을 적용하기 위한 방법을 제시하지 못하고 있다. (그림 1)은 기존의 RBAC의 모델이다.



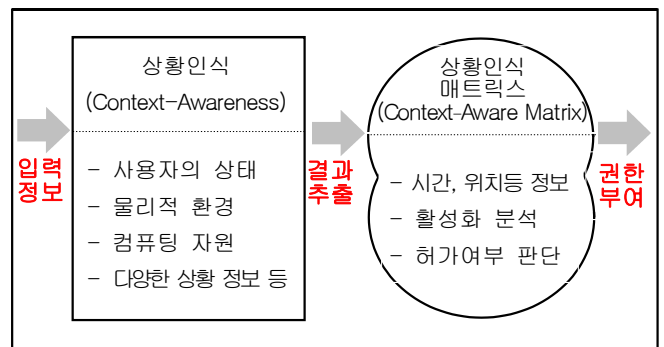
(그림 1) RBAC 모델

RBAC 모델의 중요한 네 가지 구성요소로 사용자(Users), 역할(Roles), 권한(Permissions), 세션(Sessions)을 가지고 있다[6]. 구성요소를 보면 사용자는 조직

내 구성원으로서 사용자의 집합으로 시스템 내에서 정보를 사용하는 사람 혹은 지능형 프로세스나 자율적인 대리인을 말하고, 역할은 조직 내 업무의 유기적인 관계로 직무와 관련된 권한과 책임이 사용자에게 부여 되는 것을 말한다. 권한은 하나 이상의 객체에 대해 수행이 가능하도록 접근을 승인하는 것이며, 세션은 시스템 사용시 사용자가 역할을 수행하기 위하여 시스템을 활성화 시키는 상태이다. 따라서 사용자는 역할을 부분적으로 활성화하는 동안 세션을 수립한다. 그 외 제약조건은 직무의 분리 및 통합과 역할에 따른 사용자의 조정 등이 해당되며, 사용자를 배정하거나 직무에 따른 역할 및 권한 지정 등 모든 요소에 제약조건이 관장하게 된다. RBAC의 기본적인 개념은 어떠한 주체에 대한 행위나 역할에 의해 접근 권한이 결정되는 접근 제어 방식으로 사용자들의 역할을 지정하고, 사용자는 구성원으로서 역할에 따라 권한을 요구하는 것이다. RBAC 모델에서 UA(User Assignments) 와 PA(Permission Assignments) 지정은 직무의 기능에 따라 결정 될 것이다. RBAC에 기반을 둔 권한 부여 메커니즘을 U-캠퍼스 환경에서 적용하고자 할 경우 위치와 시간 및 시스템의 자원 등과 같은 동적인 정보를 고려하여야 한다. 본 논문에서는 이러한 문제를 해결하고자 기존의 RBAC 모델을 U-캠퍼스 환경에 맞도록 확장하였다.

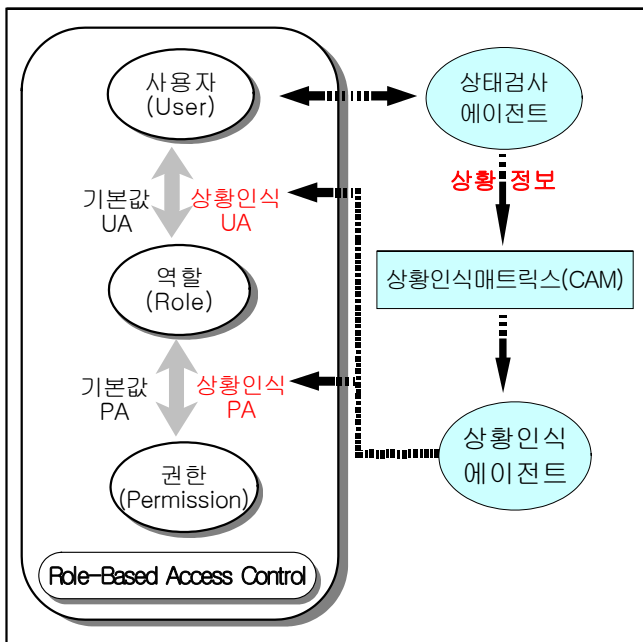
### 3. U-캠퍼스환경에서 접근 제어 메커니즘

본 논문에서 제시하는 접근 제어 메커니즘은 U-캠퍼스 환경에서 사용자의 위치, 시간 정보와 같은 상황정보 등을 가지고 활성화 분석을 한 후 상황에 맞는 상태를 확인하여 상황인식 매트릭스(Context-Aware Matrix, CAM)에 적용하여 접근 권한을 부여하게 된다. (그림 2)는 상황인식을 이용하여 권한을 부여하는 것에 대한 개념을 보여주고 있다.



(그림 2) 상황인식 권한 부여 개념도

상황인식 권한 부여를 U-캠퍼스 환경에서 응용할 경우 사용자의 상황 정보가 변화 될 때 사용자에게 부여된 접근 권한도 같이 변경되어야 한다. 기존 RBAC 모델에서 기본적으로 사용자는 역할을 부여 받고 역할은 RBAC 모델에서 권한을 할당받아 사용자는 역할을 통해서 권한을 획득한다. 변화된 접근 제어를 결정하기 위하여, 상황인식 매트릭스 와 상황인식 UA 및 상황인식 PA를 이용한다. (그림 3)는 U-캠퍼스 환경에서 상황정보를 고려하여 권한을 부여하는 메커니즘의 전체적인 구조이다.



(그림 3) 상황정보를 고려한 권한 부여 모델

본 논문에서 제안하는 모델의 세 가지 중요한 요소로 상태검사 에이전트(State Check Agent), 상황인식 매트릭스, 상황인식 에이전트(Context-Aware Agent)가 있다. 상태검사 에이전트는 사용자의 환경 상태를 확인하고 상황 정보를 수집하는 기능을 한다. 상황인식 매트릭스는 상태검사 에이전트에서 수집된 위치와 시간과 같은 상황 정보를 판단하여 사용자의 역할을 활성화하거나 비활성화 하는 아주 중요한 부분을 차지한다. 상황인식 에이전트는 상황인식 매트릭스의 변화를 확인하고 역할 변화에 따라 사용자와 권한을 지정한다. 다시 말해서 RBAC에 의해 설정되었던 기본값 UA와 PA는 상황인식 매트릭스에 의해 변화된 상황인식 UA, PA로 변경되고 사용자를 위한 역할 활성화 및 권한이 결정된다.

앞서 정의하였듯이, 상황인식 매트릭스는 상황정보 권한 부여 모델에서 사용자의 접근 권한을 결정하는데

가장 중요한 요소이다. 이것은 사용자의 상황 정보가 변화할 때, 위치와 시간과 같은 사용자의 동적인 상황을 통해 사용자를 위한 역할 활성화를 결정한다. RBAC 모델을 기반으로 확장하여 접근 제어 결정을 하기 위한 상황인식 매트릭스는 사용자 위치, 시간, 시스템 자원 상황에 대한 상황인식 매트릭스의 실질적인 예가 아래 표들과 같이 구성되어 있다.

<표 1> 위치 상황에 대한 CAM

	Location1	Location2	Location3
Role1	Active	Inactive	Inactive
Role2	Active	Active	Active
Role3	Active	Active	Inactive

<표 2> 시간 상황에 대한 CAM

	Time1	Time2	Time3
Role1	Active	Inactive	Inactive
Role2	Active	Active	Active
Role3	Active	Active	Active

<표 3> 시스템 자원에 대한 CAM

	Resource1	Resource2	Resource3
Role1	Active	Active	Inactive
Role2	Active	Active	Active
Role3	Active	Active	Active

위의 표들에서 Role은 사용자의 역할을 의미하고 접근 권한 결정은 상황 정보의 요소와 사용자의 역할에 따라 결정된다. 역할 활성화는 아래와 같이 나타낼 수 있다.

$$\begin{aligned}
 \text{역활의 활성화} &= \text{상황}(\text{상황1}, \text{상황2}, \dots, \text{상황n}) \\
 &= \text{상황}(\text{활성화}, \text{활성화}, \dots, \text{활성화}) \\
 &= \text{활성화}
 \end{aligned}$$

즉, 역할의 활성화는 상황 정보의 모든 요소가 활성화될 때만 활성화 된다. 위치와 시간, 활용할 자원에 관한 상황 정보의 예에서, 아래와 같은 경우에만 역할이 활성화 되며 그 이외의 경우에는 비활성화 된다.

$$\begin{aligned}
 \text{역활의 활성화} &= \text{상황}(\text{Location}, \text{Time}, \text{Resource}) \\
 &= \text{상황}(\text{Active}, \text{Active}, \text{Active}) \\
 &= \text{Active}
 \end{aligned}$$

동적인 권한 부여를 위한 역할의 활성화, 비활성을 보여주기 위하여 다음과 같은 시나리오를 구성하였다.

## 동적인 권한부여를 위한 역할 시나리오

### ① 가정:

- U-캠퍼스에서 센서가 모든 사용자의 활동에 관한 다양한 정보를 읽기, 저장, 수정을 실행한다.
- 모든 사용자는 위치와 시간, 사용할 자원 등의 상황에 따라 접근 권한이 동적으로 변경된다.
- 위치는 Location1이 교실이고, Location2가 교무실, Location3이 회의실 이라고 가정하고, 활용 자원으로 Resource1은 TV, Resource2는 컴퓨터, Resource3은 프린터라고 가정한다.

### ② 기본조건:

- 모든 구성원은 기존 RBAC에 의해 기본역할이 결정된다.
- 역할의 활성화는 모든 상황(위치, 시간, 자원)이 활성화 되어야만 권한이 부여된다.

### ③ 시나리오:

- **시나리오1:** 교사가 기존 RBAC에 의해 할당된 역할 Role1, Role2, Role3을 가지고 있고 프린터 자원을 사용하기 위하여 Time1에서 교무실로 이동한다면,

Role1의 활성화 = (Location2, Time1, Resource3)  
= (Inactive, Active, Inactive)  
= Inactive

Role2의 활성화 = (Location2, Time1, Resource3)  
= (Active, Active, Active)  
= Active

Role3의 활성화 = (Location2, Time1, Resource3)  
= (Active, Active, Active)  
= Active

교사의 역할은 Role1의 역할이 비활성화 되기 때문에 Role2와 Role3이 될 것이다.

- **시나리오2:** 학생이 RBAC에 의해 할당된 역할 Role1, Role2, Role3을 가지고 있고 TV자원을 사용하기 위하여 Time3에서 회의실로 이동한다면,

Role1의 활성화 = (Location3, Time3, Resource1)  
= (Inactive, Inactive, Active)  
= Inactive

Role2의 활성화 = (Location3, Time3, Resource1)  
= (Active, Active, Active)  
= Active

Role3의 활성화 = (Location3, Time3, Resource1)  
= (Inactive, Active, Active)  
= Inactive

학생의 역할은 Role1과 Role3의 역할이 비활성화 되기 때문에 Role2가 될 것이다.

이와 같은 실질적인 예에서 상황 정보의 세 가지 요소(위치, 시간, 자원)를 적용하였지만, 특별 이벤트, 네트워크 상태, 네트워크 보안규정등과 같은 획득된 상황 정보를 상황인식 매트릭스에 적용함으로써 쉽게 상황 정보의 규모를 확대하여 적용 할 수 있다.

## 4. 결론

본 논문에서 U-캠퍼스 환경에서 상황 정보를 고려한 동적 접근 제어 모델을 제안했다. 제안된 동적 접근 제어 모델은 RBAC 모델을 확장하여 동적인 사용자의 상황 정보를 기반으로 역할에 대한 권한 부여를 관장한다. 본 논문은 U-캠퍼스 환경에서 변화하는 동적인 상황 정보에 따라 권한을 부여하고, 특히 역할의 활성화 및 비활성에 초점을 맞추었다. 상황정보에 의한 권한 부여 모델에서 상황인식 매트릭스는 상황 정보를 다루기 위하여 사용되고, 사용자의 상황 정보가 변화할 때 위치와 시간, 시스템 자원과 같은 상황 정보를 상황인식 매트릭스에 적용하여 사용자에게 부여된 역할의 활성화 상태에 따라 접근 권한을 부여하는 방법을 제안하였다.

## 참고문헌

- [1] Mark Weiser, "The Computer for the 21st Century", Scientific American pp. 94-104, Sep. 1991.
- [2] Anind K. Dey, "Understanding and using context", Personal and Ubiquitous Computing, Vol5, No. 1, pp. 4-7, 2001.
- [3] Roy Want, Andy Hopper, Veronica Falcao, Jonathan Gibbons, "The Active Badge Location System", ACM Transactions on Information System. Vol. 10. No. 1, pp. 91-102, 1992
- [4] Anind. K. Dey, "Context-Aware Computing: The CyberDesk Project", Proc. of the AAAI 1998 Spring Symposium on Intelligent Environments (AAAI Technical Report SS-98-02), pp. 51-54, Mar. 1998
- [5] M.A. Harrison, W.L. Ruzzo, and J.D. Ullman, "Protection in operation systems", Communications of the ACM, 19(8), pp. 461-471, August 1976
- [6] Ravi S. Sandhu, Edward J. Coyne, Hal L. Feinstein and Chales E. Youman, "Role-Based Access Control Models", IEEE Computer, Volume 29, number 2, pp. 38-47, February 1996