웹데브 기반 협업시스템에서의 접근 제어 관리☆

Management of the Access Control for a WebDAV-based Collaborative System

김 성 훈* 이 홍 창** 이 명 준*** 박 양 수****
Seong-Hune Kim Hong-Chang Lee Myung-Joon Lee Yang-Su Park

요 약

웹데브는 분산 저작활동을 지원하는 IETF의 표준 프로토콜이다. 웹데브 접근 제어 프로토콜은 웹데브 서버에 의하여 관리되는 자원과 이들의 속성에 대한 접근을 임의적으로 제어할 수 있는 기능을 제공하며 높은 수준의 협업작업이 웹데브 서버를 통하여 수행될 수 있는 길을 열어주고 있다.

본 논문에서는 높은 수준의 협업을 제공하도록 웹데브 접근 제어 프로토콜을 통하여 웹데브 자원들에 대한 접근 제어를 관리하는 기법을 소개하고 이러한 기법을 CoSlide 협업시스템에 적용시킨 접근 제어 관리자의 개발에 대하여 기술한다. 웹데브 접근 제어 프로토콜에서 제공하는 표준 권한를 사용자에게 쉽게 이해할 수 있는 방법으로 접근 제어 기능을 제공하기 위하여 웹데브 메소드 기반으로 제시한다. 또한 이미 설정되어 있는 접근 제어와 새로 추가하려는 접근 제어 사이에 충돌을 탐지하고 이를 해결하는 기법을 제공한다. 이러한 웹데브 메소드 기반의 접근 제어 관리 기능을 CoSlide 협업시스템에 적용하였다. 개발된 접근 제어 관리자는 그룹작업장에 소속된 사용자들과 자원들에 대하여 유연한 접근 제어 관리 기능을 가진 그룹작업장을 생성하고 이를 관리하는 기능을 제공한다.

ABSTRACT

WebDAV is an IETF standard protocol which supports asynchronous collaborative authoring on the Web. The WebDAV Access Control Protocol provides various methods of controlling the resources on a WebDAV server and their properties, helping high-level group activities to be performed through the WebDAV server.

In this paper, to provide high level collaboration, we introduce a technique for managing access control over WebDAV resources through the WebDAV Acess Control Protocol and describe the development of an access control manager for the CoSlide Collaborative system based on the technique. To provide users with the access control features in an easily understandable manner, the developed technique presents the privileges for performing WebDAV methods instead of the standard privileges in the WebDAV Access Control Protocol. In addition, we present the facility for detecting conflicts between new access privileges on resources and old access privileges on them. We applied the method-based access control management technique to the CoSlide collaborative system. The developed access control manager enables us to create group workspaces with flexible access control strategies for group members and resources.

☞ KeyWords : WebDAV(웹데브), WebDAV Access Control Protocol(웹데브 접근 제어 프로토콜), CoSlide Server, Open workspaces, Group workspaces, CoSpace, Access Control Manager

1. 서 론

네트워크의 발달은 개인이나 집단이 원거리에 있는 사람들과 협업을 수행하는데 크게 기여하고 있다. 사람들은 메신저, 메일, 게시판 등을 이용하여 다른 사람들과 정보를 교환하며 협업을 수행할 수

* 정 회 원 : 경남점자정보도서관 웹관리자 재직
heinz@nate.com
** 정 회 원 : 울산대학교 대학원 컴퓨터정보통신공학부(박사과정)
myhyunii@mail.ulsan.ac.kr
*** 정 회 원 : 울산대학 컴퓨터정보통신공학부 교수
mjlee@ulsan.ac.kr
**** 정 회 원 : 울산대학교 컴퓨터정보통신공학부 교수
yspk56@ulsan.ac.kr(교신저자)

[2009/09/08 투고 - 2009/09/11 심사 - 2009/10/28 심사완료]
☆ 이 논문은 2009년도 울산대학교 연구비에 의하여 연구되었음.

있지만, 이러한 도구들은 사람들이 협업을 수행할 때 필요로 하는 자료와 정보를 비동기적으로 공유하기에 관련 자원의 집중화와 구조화가 이루어지지 않으며 효과적인 관리가 불가능하다. 따라서 협업을 수행하는 사람들은 효율적인 자원관리와 협업을 지원하는 협업지원도구 사용이 요구된다. 이러한 협업지원도구에는 특정 플랫폼과 응용 프로그램을 필요로 하는 PublicSpace[1]와 TeamRooms[2], 웹기반 도구인 GMD의 BSCW[3,4]과 울산대학교의 iPlace[5] 등이 있다. 이러한 도구들은 협업을 지원하기 위하여 전용 시스템을 요구하거나 그들만의 통신프로토콜을 정의하여 시스템을 구성하고 있는 관계로 상호운용성이 결여되어 있다.

웹데브(WebDAV: Web-based Distributed Authoring and Versioning)[6,7,8]는 IETF(Internet Engineering Task Force)에서 발표한 웹상의 분산 저작활동을 지원하기 위한 표준 프로토콜이다. 웹데브는 원거리에 있는 사용자간에 웹상의 자원을 편집하고 관리할 수 있는 기반 하부 구조를 제공하며, 이를 바탕으로 웹데브 기능이 구현된 서버와 클라이언트들이 활발하게 개발되고 있다. 웹데브 접근 제어 프로토콜(WebDAV Access Control Protocol)[9,10]은 웹데브 서버에 의하여 관리되는 자원과 이들의 속성에 대한 접근을 임의적으로 제어할 수 있는 기능을 표준적으로 제공하며 높은 수준의 협업이 웹데브 서버를 통하여 수행될 수 있는 길을 열어주고 있다.

CoSlide 협업 시스템은 CoSlide 서버[11,12,13]와 이 서버와 통신하는 CoSpace 클라이언트[14]로 구성되어 있다. 분산저작을 지원하는 웹데브 기반의 Jakarta Slide[15,16] 서버를 확장한 CoSlide 서버는 개인 작업장뿐만 아니라 그룹작업장과 공개작업장을 위한 서버환경을 제공한다. 이와 더불어 각 작업장들의 사용자 인터페이스를 제공하는 CoSpace 클라이언트는 웹데브 메서드를 이용하여 서버에 등록된 자원을 효과적으로 관리할 수 있는 도구이다. 하지만 CoSlide 협업 시스템은 작업장과 자원들에 대하여 접근 제어 관리기능이 제공되고 있지 않다. 높은 수준의 협업이 수행되려면 작업장의 사용자와 자원들에 대하여 접근 제어를 관리할 수 있는 기능이 요구된다.

본 논문에서는 협업시스템이 높은 수준의 협업을 제공할 수 있도록 웹데브 접근 제어 프로토콜을 이용하여 작업장과 자원들에 대한 접근 제어를 관리하는 기법과 이러한 기법을 CoSlide 협업시스템에 적용한 접근 제어 관리자에 대하여 기술한다. 접근 제어 관리를 지원하는 기존의 협업시스템은 웹데브 접근 제어 프로토콜의 표준 권한(standard privilege)을 통하여 특정 자원에 대하여 접근 제어 관리를 제공하고 있다. 하지만 웹데브 기반의 협업시스템은 자원의 읽기와 수정, 삭제 등 모든 기능을 웹데브 메소드 기반으로 제공하고 있기 때문에 기존에 설정되어 있는 접근 제어 권한을 잘 인지하지 못한 상황에서 새로운 접근 제어 설정은 생각지 못한 다른 접근 제어의 권한이 제약을 받는 상황이 예기치 않게 발생하게 된다. 현재 기존의 모든 웹데브 시스템들은 접근 권한을 제어하기 위하여 표준 권한만을 사용하고 있다.

본 논문에서는 이러한 문제점을 해결하기 위하여 협업시스템의 접근 제어 관리 기법을 웹데브 메소드 기반으로 관리하는 기법을 설계하였다. 또한 접근 제어 설정에 있어서 설정되어 있는 접근 제어와 새로 추가하려는 접근 제어 사이의 충돌을 탐지하여 이를 해결하는 기법을 개발하였다. 이러한 웹데브 메소드 기반의 접근 제어 관리 기법을 CoSlide 협업시스템에 적용하여 접근 제어 관리자를 개발하였다. 접근 제어 관리자는 자원에 대한 접근 제어 관리 기능을 제공하며 협업에 필요한 접근 제어를 적절히 설정하여 그룹작업장을 생성하고 그룹작업장에 소속된 사용자들과 자원들에 대하여 접근 제어 관리 기능을 제공한다.

본 논문의 구성은 다음과 같다. 서론에 이어 2장에서는 관련 연구로서 웹데브 프로토콜과 웹데브 접근 제어 프로토콜에 대한 소개와 CoSlide 협업시스템을 살펴본다. 3장에서는 웹데브 메소드 기반의 접근 제어 설계에 대하여 살펴보고 4장에

서는 설계된 웹데브 메소드 기반의 접근 제어 관리 기법을 CoSlide 협업시스템에 적용하여 개발된 접근 제어 관리자에 대하여 살펴본다. 5장에서는 접근 제어 관리자를 지원하는 CoSlide 협업시스템과 다른 협업시스템들과의 비교를 설명하고 끝으로 6장에서는 결론에 대하여 기술한다.

2. 관련 연구

2.1 웹데브

웹데브(WebDAV: Web-based Distributed Authoring and Versioning, RFC 2518)는 인터넷을 통하여 광범위하고 다양한 콘텐츠의 비동기적인 협업 저작을 지원하기 위한 프로토콜이다. 웹데브는 HTTP/1.1 프로토콜[17]의 확장을 통하여 사용자들에게 원거리 서버들의 파일들을 수정하고 관리할 수 있도록 한다. 웹데브의 주요 기능으로는 잠금 관리(Lock Management), 속성 관리(Property Management), 컬렉션(Collection), 이름 공간 관리(Namespace management) 등이 있다. 각각의 기능들은 표 1에서 보는 것과 같이 HTTP/1.1의 메소드를 사용하고 있으며 일부 메소드는 더 확장되었고 몇몇의 메소드는 추가 되었다.

(표 1) 웹데브 메소드

메소드		기능
HTTP	CONNECT	서버에 접속
	HEAD, TRACE	네트워크 행동을 찾고 추적하는 기능
	GET	문서를 서버에서 받음
	PUT, POST	문서를 서버에 전달
	DELETE	자원 삭제
	OPTIONS	서버가 지원하는 메소드 출력
	MKCOL	컬렉션 생성
웹데브	PROPFIND, PROPPATCH	자원의 속성을 검색하고 설정
	DELETE (for collections)	컬렉션 삭제
	PUT (for collections)	컬렉션을 서버에 전달
	COPY, MOVE	자원의 복사와 이동
	LOCK, UNLOCK	덮어 쓰기 방지 기능

2.2 웹데브 접근 제어 프로토콜

웹데브 접근 제어 프로토콜(WebDAV Access Control Protocol)은 웹데브 서버에 의하여 관리되는 자원과 이들에 대한 정보의 접근을 임의적으로 제어할 수 있는 기능을 표준적으로 제공한다. 표 2는 웹데브 접근 제어 프로토콜에서 사용되는 주요 용어들을 보여준다.

웹데브 접근 제어 명세는 특정 자원에 대한 접근 제어 권한을 정의하기 위하여 표준 권한을 정의하고 있다. 표 3은 웹데브 접근 제어 명세에서 제공하는 10개의 표준 권한을 보여 준다.

(표 2) 웹데브 접근 제어 프로토콜의 주요 용어

주요 용어	설 명
principal	사용자가 누구(UserID)이며 어떤 그룹(Group)에 속하여 있는지를 정의
group	공통된 권한을 갖는 사용자들을 대표할 수 있는 Principal을 의미
privilege	서버에서 자원을 관리하기 위한 접근 제어 권한을 정의
access control element (ACE)	특정 자원에 대한 권한을 부여(grant) 또는 거부(deny)여부를 정의
access control list (ACL)	특정 자원에 대한 접근 제어를 정의한 ACE의 리스트를 의미

(표 3) 표준 권한

privilege	설 명
read	파일 또는 컬렉션의 내용을 읽을 수 있는 권한
read-acl	ACL 속성을 읽을 수 있는 권한
read-current-user-privilege-set	현재 사용자에게 주어진 privilege들을 읽을 수 있는 권한
write	파일의 속성과 내용을 쓰거나 수정할 수 있는 권한
write-properties	파일의 속성을 변경할 수 있는 권한
write-content	파일의 내용을 수정할 수 있는 권한
write-acl	ACL 속성을 수정할 수 있는 권한
bind	컬렉션을 생성하거나 컬렉션의 내용을 추가, 수정할 수 있는 권한
unbind	컬렉션을 이동, 삭제할 수 있는 권한
unlock	lock 설정이 되어 있는 파일 또는 컬렉션을 unlock 시킬 수 있는 권한

2.3 협업 시스템

CoSlide 협업 시스템은 CoSlide 서버와 CoSlide 서버를 이용하는 CoSpace 클라이언트가 있다. CoSlide 서버는 협업 구성원들 간의 자원 공유를 위한 그룹작업장을 지원하며, 사용자는 CoSpace 클라이언트를 이용하여 분산저작 처리의 자동화, 드래그 앤 드롭을 이용한 파일이동 등의 고급 기능을 수행할 수 있다.

CoSlide 서버는 Apache 그룹의 Jakarta Slide 웹데브 서버를 확장하여 구현하였으며 웹데브 서버의 특징인 분산저작을 지원하며 가상의 작업공간을 제공함으로써 협업 환경을 지원하고 있다. 이러한 가상 작업공간으로는 협업시스템에 등록된 사용자가 자신의 자료를 보관, 관리할 수 있는 개인작업장, 협업시스템에 등록되지 않은 사용자들이 협업 활동을 할 수 있는 공개작업장, 그리고 공동의 과제를 수행하기 위한 작업그룹을 지원하는 그룹작업장으로 구성된다.

2.3.2 CoSpace 클라이언트

CoSpace 클라이언트는 CoSlide 서버의 효율적인 사용을 위하여 개발된 윈도우즈 응용 프로그램이다. CoSpace 클라이언트는 CoSlide 서버가 제공하는 각종 작업장에 대하여 웹데브 기본 메소드를 정의하여 손쉽게 사용할 수 있도록 제공하여 주며 협업을 위해 필요한 각 작업장에 대한 생성, 수정, 삭제, 그리고 사용자 초대와 그룹작업장 참가요청과 같은 작업장을 관리하는데 필요한 인터페이스를 제공하고 있다.

3. 웹데브 메소드 기반의 접근 제어 설계

협업시스템에서 높은 수준의 협업을 제공하기 위하여 작업장과 자원들에 대하여 접근 제어 관리 기능이 요구된다. 개별적인 작업장과 자원들에 대하여 특정 사용자에게만 읽기와 수정 등과 같은 권한을 허용하고, 다른 사용자들은 기본적인 권한을 불허하도록 하는 것과 같은 자원에 대한 사용자의 접근 제어 관리 기능이 협업시스템에 제공됨으로써 수준 높은 협업이 수행될 수 있다.

웹데브 접근 제어 프로토콜은 웹데브 서버에 의하여 관리되는 자원과 이들과 관련된 정보의 접근을 임의적으로 제어할 수 있는 기능을 표준 권한으로서 제공한다. 웹데브 기반 협업시스템은 자원의 읽기와 수정, 삭제 등 모든 기능을 웹데브 메소드를 이용하여 제공하기 때문에 표준 권한을 접근 제어 관리를 위하여 사용하기에는 무리가 있다. 협업시스템의 자원에 대하여 표준 권한을 이용하여 접근 제어를 설정하면 특정 하나의 권한의 접근 제어 설정에 대하여 다수의 웹데브 메소드의 기능이 제약되기 때문이다. 웹데브 기반 협업시스템의 효과적인 접근 제어 관리를 위하여 이러한 표준 권한을 웹데브 메소드 기반으로 설계하였다. 또한 새로운 접근 제어 설정은 서버에 저장되는 과정에서 표준 권한들의 충돌이 발생할 수 있어서 이를 탐지하고 해결방안을 제시하는 것이 바람직하다. 본 장에서는 웹데브 메소드와 표준 권한의 관계를 설명하고, 웹데브 메소드 기반의 접근 제어를 지원하기 위하여 접근 제어 충돌을 탐지하는 기법에 대하여 설명한다.

3.1 웹데브 메소드와 웹데브 접근 제어 표준 권한

(표 4) 웹데브 메소드와 표준 권한

웹데브 메소드		privilege
ALL		all
GET		read
PUT	파일	write-content
	컬렉션	bind
PROPPATCH		write-properties
ACL		write-acl
PROPFIND		read, read-acl, read-current-user-privilege-set
COPY	파일	read, write-content, write-properties
	컬렉션	read, bind

MOVE		unbind, bind
DELETE	파일/컬렉션	unbind
MKCOL	컬렉션	bind
LOCK	파일	write-content
	컬렉션	bind
UNLOCK		unlock

웹데브 기반 협업시스템에서 사용자는 웹데브 메소드를 이용하여 자원들의 다운로드와 업로드, 복사, 이동, 삭제 등의 기능을 사용할 수 있다. 웹데브 접근 제어 프로토콜은 서버에 존재하는 자원들의 접근 제어를 표준 권한을 통하여 제공한다. 표 4는 웹데브 메소드들과 이와 관련되는 웹데브 접근 제어 프로토콜의 권한들을 표로 정리하였다.

3.2 접근 제어 설정에 있어서의 충돌 탐지 기법

3.1절에서 웹데브 메소드와 관련되는 웹데브 접근 제어 표준 권한에 대하여 살펴보았다. 웹데브 메소드 기반의 접근 제어 설정에 있어서 특정 메소드의 접근 제어를 설정하게 되면 이 메소드에 해당하는 표준 권한들이 서버에 저장되게 된다. 여기서 동일한 접근 정의(principal)에 대한 접근 제어에 있어 웹데브 메소드와 표준 권한 사이에서 중복이 발생하기 때문에 접근 제어간의 충돌을 야기한다. 예를 들어 MOVE 메소드의 기능을 허용하기 위하여 접근 제어를 허용(grant)으로 설정하게 되면 unbind, bind 권한이 grant로 설정되고 이에 더하여 MKCOL 메소드의 기능을 불허하기 위하여 접근 제어를 거부(deny)로 설정하게 되면 bind 권한이 deny로 설정되어 MOVE 메소드의 기능이 제공되지 않게 됨으로써 메소드들 간에 충돌이 발생하게 된다. 협업시스템의 자원들에 대하여 접근 제어를 원활하게 관리하려면 이러한 충돌을 탐지하고 해결해야 한다.

충돌 탐지에 있어 가장 먼저 해야 할 일은 동일한 접근 정의에 설정된 접근 제어 설정을 찾는 것이다. 웹데브 기반 협업시스템에서는 접근 정의를 표현할 때 다른 접근 정의를 포함하는 것이 가능하기 때문에 이러한 경우도 찾아서 충돌 탐지를 해야 한다. 충돌 탐지를 위하여 동일한 접근 정의를 찾을 때에는 다음과 같은 사항들을 고려해야 한다.

① 설정하려는 접근 제어의 접근 정의와 동일한 접근 정의가 존재하는 경우
② 설정하려는 접근 제어의 접근 정의이 다른 접근 정의에 포함되어 있는 경우
③ 설정하려는 접근 제어의 접근 정의가 다른 접근 정의를 포함하는 경우
④ 관계없는 경우

④의 관계없는 경우에는 충돌을 탐지할 필요가 없다. 따라서 ①~③의 경우를 찾는 것이 충돌 탐지를 위하여 가장 먼저 해야 할 일이다. ①~③의 경우에 해당하는 동일한 접근 정의에 설정된 접근 제어 설정을 찾게 되면 이제 설정하려는 접근 제어와 설정되어 있는 접근 제어를 비교하여 충돌을 탐지해야 한다.

동일한 접근 정의에 설정하려는 특정 메소드의 접근 제어는 관련된 표준 권한들이 허용 여부에 따라 서버에 grant/deny로 저장되게 된다. 예를 들어 PUT 메소드의 접근 제어 설정을 위해서는 write, write-properties, bind 권한을 설정해야 한다. 따라서 PUT 메소드의 기능을 허용하기 위하여 접근 제어 설정을 허용으로 설정하려 할 때 write, write-properties, bind 권한들 중에 하나의 권한이라도 포함해야 기능이 제공되는 PUT, COPY, MOVE, MKCOL, LOCK 메소드들중 한 개 이상의 메소드 접근 제어 설정이 deny로 설정되어 있다면 충돌이 발생하게 된다. 이와는 반대로 PUT 메소드의 기능을 불허하기 위하여 접근 제어 설정을 deny로 설정할 때 PUT, COPY, MOVE, MKCOL, LOCK 메소드들중 한 개 이상의 메소드 접근 제어 설정이 허용으로 설정되어 있다면 충돌이 발생하게 된다. 각 메소드의 접근 제어 설정에 있어 충돌 관계를 표 5로 정리하였다. 표 5에

서 설정하려는 접근 제어와 설정되어 있는 접근 제어와 비교해서 충돌이 발생하는 접근 제어를 C로 표시하였다.

(표 5) 웹데브 메소드의 충돌 관계

C : 충돌(Conflict)

설정되어 있는 접근 제어 / 설정 하려는 접근 제어	ALL:G	ALL:D	GET:G	GET:D	PUT:G	PUT:D	PRP:G	PRP:D	ACL:G	ACL:D	PRP:G	PRP:D	COP:G	COP:D	MOV:G	MOV:D	DEL:G	DEL:D	MKC:G	MKC:D	LOC:G	LOC:D	ULK:G	ULK:D
ALL:G		C		C		C		C		C		C		C		C		C		C		C		C
ALL:D	C		C		C		C		C		C		C		C		C		C		C		C	
GET:G		C		C								C		C										
GET:D	C		C								C		C											
PUT:G		C				C								C		C				C		C		
PUT:D	C				C								C		C				C		C			
PRP:G		C						C						C										
PRP:D	C						C						C											
ACL:G		C								C														
ACL:D	C								C															
PRP:G		C		C								C		C										
PRP:D	C		C								C		C											
COP:G		C		C		C		C				C		C		C				C		C		
COP:D	C		C		C		C				C		C		C				C		C			
MOV:G		C				C							C		C		C		C		C			
MOV:D	C				C								C		C		C		C		C			
DEL:G		C													C		C							
DEL:D	C														C		C							
MKC:G		C				C							C		C					C		C		
MKC:D	C				C								C		C				C		C			
LOC:G		C				C							C		C				C		C			
LOC:D	C				C								C		C				C		C			
ULK:G		C																						C
ULK:D	C																						C	

PRP : PROPPATCH
PRP : PROPPIND
COP : COPY
MOV : MOVE
DEL : DELETE
MKC : MKCOL
LOC : LOCK
ULK : UNLOCK

웹데브 기반의 협업시스템의 접근 제어는 상위 자원인 컬렉션이 존재한다면 이 컬렉션에 설정되어 있는 접근 제어를 상속받게 된다. 접근 제어 설정에 있어 동일한 접근 정의을 발견하는 과정에서 다수의 접근 제어가 발견될 수 있으며, 이는 상위 컬렉션에서도 발견될 수 있다. 설정하려는 접근 제어의 충돌 탐지는 자원에 설정되어 있는 접근 제어와의 충돌 탐지뿐만 아니라 상속받은 모든 접근 제어와도 충돌 탐지를 해야 한다. 이러한 과정을 "/slide/GroupWorkSpace/TempWork"에 존재하고 있는 "sample.txt" 자원을 예로 들어 설명하겠다.

(표 6) "/slide/GroupWorkSpace/TempWork /sample.txt" 자원에 설정되어 있는 접근 제어

Principal	Method	Grnat/Deny	Inherited Form	레벨
ALL	ALL	deny	/slide	0
…	…	…	…	I
그룹K	GET	grant	/slide/GroupWorkSpace	1
그룹L	GET	grant	/slide/GroupWorkSpace	1
사용자D	GET	grant	/slide/GroupWorkSpace	1
사용자E	GET	grant	/slide/GroupWorkSpace	1
…	…	…	…	I
그룹K	PUT	grant	/slide/GroupWorkSpace/TempWork	2
사용자A	ACL	deny	/slide/GroupWorkSpace/TempWork	2
…	…	…	…	I
그룹K	COPY	grant		3
사용자A	UNLOCK	deny		3
사용자A	MOVE	grant		3

표 6은 협업시스템의 "/slide/GroupWorkSpace /TempWork"에 존재하고 있는 "sample.txt" 자원의 접근 제어를 보여주고 있다. 접근 제어는 자원의 상위 컬렉션들의 접근 제어를 상속 받고 있기 때문에 "Inherited Form"을 통하여 상위 컬렉션에 설정되어 있는 접근 제어를 확인할 수 있다. "Inherited Form" 내용이 없는 부분은 "sample.txt" 자원에 설정되어 있는 접근 제어이다. 최상위 컬렉션인 "/slide"를 레벨0이라 가정하면 "/slide/Group WorkSpace/TempWork"에 존재하는 "sample.txt" 자원은 레벨3이 되게 되며 i번째 레벨에 있는 자원은 레벨i라고 할 수 있다. "그룹K"라는 접근 정의는 "사용자A", "사용자B", "사용자C" 접근 정의를 포함하고 있다고 가정하면 "그룹K" 접근 정의에 설정되어 있는 접근 제어는 "사용자A", "사용자B", "사용자C" 접근 정의들에게도 적용된다. 표 6에서 협업시스템의 최상위 컬렉션인 "/slide"에는 모든 사용자들에게 모든 권한을 불허하기 위하여 ALL 접근 정의에 메소드 ALL이 deny로 설정되어 있다. "/slide" 컬렉션의 하위 컬렉션인 "/slide/GroupWorkSpace"에는 "그룹K", "그룹L", "사용자D", "사용자E" 접근 정의에 대하여 GET 메소드를 허용하고 있으며, "/slide/GroupWorkSpace" 컬렉션의 하위 컬렉션인 "/slide/GroupWorkSpace/TempWork" 컬렉션에는 "그룹K" 접근 정의에 대하여 PUT 메소드가 허용되었으며, "사용자A" 접근 정의에 대하여 ACL 메소드가 불허되어 있다. 그리고 "sample.txt" 자원에는 "그룹K" 접근 정의에 대하여 COPY 메소드가 허용되었으며 "사용자A" 접근 정의에 대하여 UNLOCK 메소드는 불허하고, MOVE 메소드는 허용하고 있다.

표 6의 "/slide/GroupWorkSpace/TempWork/sample.txt" 자원에 새로운 접근 제어를 추가하려고 할 때 접근 제어 충돌 탐지를 위하여 앞에서 설명하였듯이 가장 먼저 해야 할 일은 동일한 접근 정의를 찾는 것이다. 새로 추가하려는 접근 제어의 접근 정의가 "사용자A"라고 한다면 표 6의 접근 제어에서 "사용자A"와 동일한 접근 정의를 가진 접근

제어를 찾아서 충돌을 탐지해야 한다. ALL 접근 정의는 모든 접근 정의를 포함하는 접근 정의이며 “사용자A” 접근 정의는 “그룹K” 접근 정의에 포함되어 있으므로 ALL, “그룹K” 접근 정의도 “사용자A” 접근 정의과 동일한 접근 정의이다. 이러한 “사용자A” 접근 정의과 동일한 접근 정의의 접근 제어를 모두 찾아서 표 7로 보여주고 있다.

(표 7) “사용자A” 접근 정의와 동일한 접근 정의에 설정되어 있는 접근 제어

Principal	Method	Grnat/Deny	Inherited Form	레벨
ALL	ALL	deny	/slide	0
그룹K	GET	grant	/slide/GroupWorkSpace	1
그룹K	PUT	grant	/slide/GroupWorkSpace/TempWork	2
사용자A	ACL	deny	/slide/GroupWorkSpace/TempWork	2
그룹K	COPY	grant		3
사용자A	UNLOCK	deny		3
사용자A	MOVE	grant		3

설정된 접근 제어는 레벨별로 적용되며 상위 레벨에서 설정된 접근 제어는 하위 레벨에서 상속받게 되며, 하위 레벨의 접근 제어가 상위 레벨의 접근 제어에서 충돌이 발생하더라도 하위 레벨의 접근 제어가 상위 레벨의 접근 제어를 덮어 쓰게 된다. 따라서 접근 제어 설정에서 충돌 탐지는 레벨별로 이루어져야 할 필요가 있다. 레벨 i에 있는 자원에 대하여 접근 제어를 설정하려 할 때 레벨 0에서 레벨 i까지 설정하려는 접근 정의와 동일한 접근 정의의 접근 제어를 찾아서 리스트를 생성한다. 그리고 새로 추가하려는 접근 제어의 충돌 탐지를 위하여 리스트를 레벨별로 정리할 필요가 있다. 이를 위하여 추가하려는 접근 제어의 접근 정의와 동일한 접근 정의의 접근 제어를 savedACListnumber[] 배열을 두어 레벨별로 저장한다. savedACListnumber[] 배열의 크기는 i가 되며 i레벨에 설정되는 접근 제어 savedACListnumber[i]의 값은 “XXXXXXXXXXXXXXXXXXXXXX”로 초기화 되어 있으며, i레벨에 저장되어 있는 접근 제어들은 표 8을 참조하여 savedACListnumber[i]에 저장된다. 표 8은 접근 제어 충돌을 탐지하기 위하여 i레벨에 설정되는 접근 제어 savedACListnumber[i]의 값을 저장하기 위하여 설계된 표이다.

(표 8) i레벨에 저장되는 접근 제어

	all	r	r-a	r-c	w	w-p	w-c	w-a	b	ub	ul	GET	PUT	PROPPATCH	ACL	PROPFIND	COPY	MOVE	DELETE	MKCOL	LOCK	UNLOCK
i레벨에 설정되는 접근 제어 savedACListnumber[i]	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X	X
ALL	O											O										
GET		O																				
PUT					O		O		O				O									
PROPPATCH						O								O								
ACL								O							O							
PROPFIND		O	O	O												O						
COPY		O				O	O		O								O					
MOVE									O	O								O				
DELETE										O									O			
MKCOL									O											O		
LOCK							O		O												O	
UNLOCK											O											O

r : read
r-a : read-acl
r-c : read-current-user-privilege-set
w : write
w-p : write-properties
w-c : write-content
w-a : write-acl
b : bind
ub : unbind
ul : unlock
X : 미설정
O : 접근 제어 설정 (grant - G, deny - D 으로 변환되어 설정)

표 7에서 보여주는 “사용자A”의 접근 제어를 각 레벨별로 표 8을 이용하여 savedACListnumber []의 값이 저장되는 과정을 살펴보도록 하자.

(1) 레벨0 - “/slide” 레벨에 설정되어 있는 접근 제어

레벨0에 설정되어 있는 접근 제어는 ALL 메소드의 deny이다. 따라서 savedACListnumber[0]은 “DXXXXXXXXXXXXXXXXXXXXX”로 저장된다.

(2) 레벨1 - “/slide/GroupWorkSpace” 레벨에 설정되어 있는 접근 제어

레벨1에 설정되어 있는 접근 제어는 GET 메소드의 grant이다. 따라서 savedACListnumber[1]은 “XGXXXXXXXXXGXXXXXXXXXX”로 저장된다.

(3) 레벨2 - “/slide/GroupWorkSpace/TempWork” 레벨에 설정되어 있는 접근 제어

레벨2에 설정되어 있는 접근 제어는 PUT 메소드의 grant와 ACL 메소드의 deny이다. 따라서 savedACListnumber[2]는 우선 PUT 메소드의 grant를 적용하여 “XXXXGXGXGXXXGXXXXXXXXX”로 설정되고, 여기서 ACL 메소드의 deny를 적용하여

savedACListnumber[2]는 “XXXXGXGDGXXXGXDXXXXXXX”으로 저장된다.

(4) 레벨3 - “/slide/GroupWorkSpace/TempWork/sample.txt” 레벨에 설정되어 있는 접근 제어

레벨3에 설정되어 있는 접근 제어는 COPY 메소드의 grant, UNLOCK 메소드의 deny, MOVE 메소드의 grant이다. 따라서 savedACListnumber[3]는 우선 COPY 메소드의 grant를 적용하여 “XGXXXGGXGXXXXXXXGXXXXXX”로 설정되고, 여기서 UNLOCK 메소드의 deny를 적용하여 savedACListnumber[3]는 “XGXXXGGXGXDXXXXXGXXXXD”으로 설정되고, 마지막으로 MOVE 메소드의 grant를 적용하여 “XGXXXGGXGGDXXXXXGGXXXD”으로 저장된다.

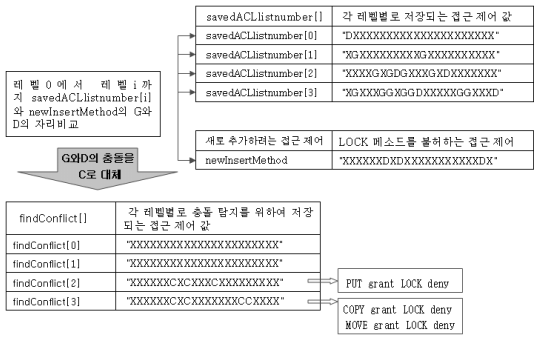
(표 9) 각 레벨별로 저장된 접근 제어

savedACListnumber[]	각 레벨별로 저장된 접근 제어
savedACListnumber[0]	DXXXXXXXXXXXXXXXXXXX
savedACListnumber[1]	XGXXXXXXXXGXXXXXXXXX
savedACListnumber[2]	XXXXGXGDGXXXGXDXXXXXX
savedACListnumber[3]	XGXXXGGXGGDXXXXXGXXXD

(1)~(4)의 과정에서 각 레벨별로 설정되는 savedACListnumber[]의 값을 표 9로 정리하여 보여주고 있다.

새로 추가하려는 접근 제어를 newInsertMethod에 저장하려고 한다. 새로 추가하려는 접근 제어와 생성한 savedACListnumber[]를 각 레벨별로 비교하기 위하여 newInsertMethod 역시 “XXXXXXXXXXXXXXXXXX” 형태로 생성해야 한다. LOCK 메소드를 불허하는 접근 제어를 추가하려는 경우 newInsertMethod의 값은 LOCK메소드의 deny에 해당하는 “XXXXXXDXDXXXXXXXXXXXXDX”이 된다.

이제 레벨0에서 레벨i까지 savedACListnumber[i]와 newInsertMethod를 비교하여 레벨별로 충돌을 탐지해야 한다. 탐지된 충돌 결과를 레벨별로 저장하기 위하여 findConflict[] 배열을 두어 레벨0에서 레벨i까지 충돌 탐지 결과를 저장한다. 충돌은 각 레벨의 savedACListnumber[i]값과 newInsertMethod 값의 G와 D의 자리비교로 탐지할 수 있으며 G와 D의 충돌이 탐지되는 자리를 C로 대체하고 해당되는 메소드 자리 역시 C로 대체하여 저장한다. 각 레벨별로 탐지된 충돌이 저장된 findConflict[]를 해석하면 어느 레벨에서 무슨 메소드와 충돌이 탐지되었는지 알 수 있게 된다. 그림 1은 이제까지 살펴본 레벨0에서 레벨i까지 savedACListnumber[i]와 newInsertMethod를 비교하여 레벨별로 충돌을 탐지하고 그 결과를 findConflict[] 배열에 저장하고 이를 해석하여 새로 추가하려는 접근 제어와 각 레벨별로 저장되어 있는 접근 제어에 있어 어떤 레벨에서 어떤 메소드들에게서 충돌이 탐지되었는지를 보여주고 있다.

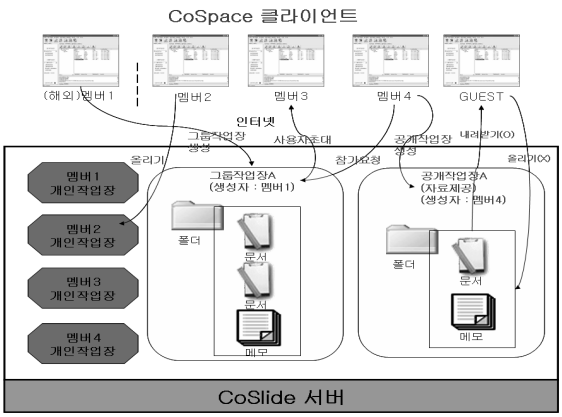


(그림 1) 충돌 탐지 프로세스

4. CoSlide 협업시스템에서의 접근 제어 관리자

CoSlide 서버는 소속되지 않은 사용자들이 협업에 참가하여 공동의 작업을 수행할 수 있는 공개작업장과 공동의 과제를 수행하는 작업그룹을 위한 공간인 그룹작업장을 지원하고, CoSpace 클라이언트는 개인작업장을 위한 사용자 인터페이스와 더불어 그룹작업장과 공개작업장의 사용자 인터페이스를 지원한다. 사용자는 CoSpace 클라이언트를 이용하여 분산저작 처리의 자동화, 드래

그 앤 드랍을 이용한 파일이동 등의 고급 기능을 수행할 수 있다. 그림 2는 CoSlide 협업시스템의 시스템 구조도를 보여 준다.



(그림 2) CoSlide 협업시스템의 시스템 구조도

4.1 그룹작업장 접근 제어 관리자 설계

그룹작업장은 그룹 구성원들이 효과적으로 협업을 수행할 수 있도록 지원하며 그룹작업에 참여하고 있는 모든 구성원들은 그룹작업에 필요한 자원을 생성, 공유할 수 있을 뿐만 아니라, 자원에 대한 추가, 변경, 삭제 등 여러 가지 작업을 수행할 수 있다. 현재의 그룹작업장들은 모두 동등한 접근 제어를 부과하여 생성되며, 그룹작업장의 접근 제어를 관리하는 기능은 제공되지 않고 있다.

그룹작업장의 접근 제어 관리 기능은 누구에게나 제공되어져서는 안 된다. 누군가가 접근 제어 관리 기능을 악용한다면 그룹작업장의 기능이 상실된다. 따라서 시스템 관리자나 그룹작업장을 관리하는 관리자와 이에 준하는 권한을 가진 사용자에게만 허용이 되어져야 한다. 이를 위하여 그룹작업장 생성 시 접근 제어 관리를 사용하기 위한 웹데브 메소드에 해당되는 웹데브 접근 제어 프로토콜의 권한이 설정되어져야 한다. 그림 3은 그룹작업장 저장소인 "../GroupWorkSpace"에 설정되어 있는 접근 제어를 보여준다. <permissions>에 접근 제어가 설정되게 되는데 "root"와 그룹작업장 관리자인 "owner"에 "actions/write-acl" 권한을 허용함으로서 이들은 그룹작업장의 접근 제어를 관리하는 권한을 가지게 된다.

```
<?xml version="1.0" encoding="UTF-8"?>
<data>
 <objectnode classname="org.apache.slide.structure.SubjectNode" uri="/GroupWorkSpace">
  <children>
                    ...
   <child name="test_heinz" uuri="/GroupWorkSpace/test_heinz" />
   <child name="test_jinop" uuri="/GroupWorkSpace/test_jinop" />
   <child name="test_bouncer" uuri="/GroupWorkSpace/test_bouncer" />
   <child name="test_bouncer2" uuri="/GroupWorkSpace/test_bouncer2" />
                    ...
  </children>
  <parents>
   <parent name="GroupWorkSpace" uuri="/" />
  </parents>
 </objectnode>
 <permissions>
                    ...
  <permission subjectUri="root" actionUri="/actions/write-acl" inheritable="true" negative="false" />
  <permission subjectUri="owner" actionUri="/actions/write-acl" inheritable="true" negative="false" />
                    ...
</permissions>
```

(그림 3) 그룹작업장 접근 제어

접근 제어 관리자가 그룹작업장의 접근 제어를 관리하기 위하여 다음과 같은 기능들이 요구된다.

(1) 협업이 요구하는 접근 제어를 설정하여 그룹작업장 생성 기능

협업의 종류에 따라 요구하는 접근 제어는 다양해질 수 있다. 따라서 협업에서 요구하는 접근 제어를 미리 설정하여 그룹작업장을 생성하는 기능을 제공해야 한다.

(2) 그룹작업장의 접근 제어 관리

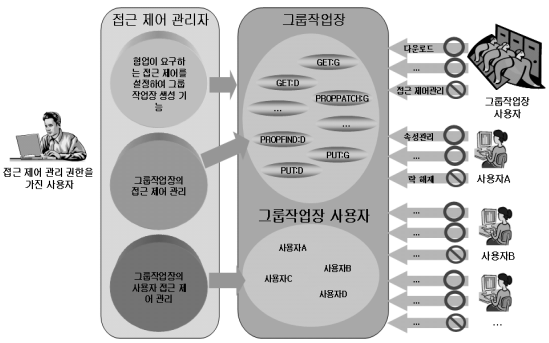
협업이 요구하는 접근 제어는 협업이 진행되면서 변화할 수 있다. 처음에 접근 제어를 설정한 그룹작업장에서 필요에 의하여 설정된 접근 제어를 수정하는 기능을 제공해야 한다.

(3) 그룹작업장의 사용자 접근 제어 관리

협업의 종류와 협업의 진행과정에 따라 각 사용자에게 그룹작업장을 사용하는 권한은 변경될 수 있다. 따라서 각 사용자들의 접근 권한을 관리할 수 있는 사용자 접근 제어 관리 기능을 제공해야 한다.

이러한 기능들을 그림 4를 통하여 보여주고 있다. 접근 제어 관리 권한을 가진 사용자가 접근

제어 관리자를 이용하여 협업이 요구하는 접근 제어를 설정하여 그룹작업장을 생성할 수 있으며 또 한 그룹작업장의 접근 제어를 관리할 수 있다. 그리고 그룹작업장의 사용자 접근 제어를 관리할 수 있으며 그룹작업장 사용자들을 자신에게 허용된 접근 제어를 통하여 그룹작업장에서 협업을 수행할 수 있다.



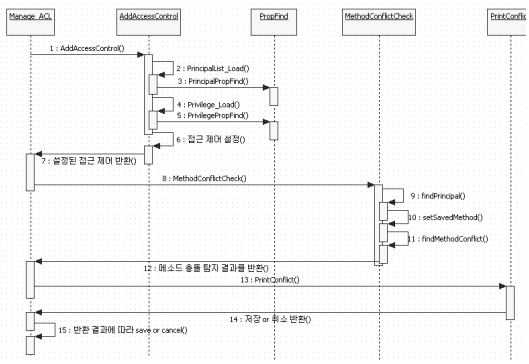
(그림 4) 접근 제어 관리자의 기능

4.2 자원들에 대하여 접근 제어 관리

그룹작업장의 자원들은 기본적으로 그룹작업장에 설정된 접근 제어를 따른다. 그리고 특정한 자원에 대하여 임의의 접근 제어를 추가할 수 있다. 그림 5는 자원들에 대하여 접근 제어를 관리하는 인터페이스를 보여주고 있다. 그룹작업장의 임의의 자원에 대하여 접근 제어 설정 메뉴를 선택하면 접근 제어 관리창이 뜨게 된다. 접근 제어 관리창은 상속받은 모든 접근 제어를 보여주며 새로운 접근 제어를 추가하거나 삭제하는 기능을 제공한다. 접근 제어 추가 버튼을 실행하면 접근 제어 추가창이 뜨게 되고 여기에서 새로운 접근 제어 추가를 위하여 접근 정의와 메소드, 허용/거부 여부를 선택하여 새로운 접근 제어를 추가할 수 있게 된다.

(그림 5) 자원들에 대하여 접근 제어 관리

새로운 접근 제어 추가는 먼저 접근 제어 충돌 탐지를 위하여 동일한 접근 정의를 탐지하고 여기에 설정되어 있는 접근 제어를 탐지해서 충돌 탐지 검사를 진행해야 한다. 그림 6은 새로운 접근 제어를 추가하는 과정을 보여준다. AddAccessControl 클래스는 그룹작업장과 관련되는 접근 정의와 추가 가능한 권한들을 가져와서 새로운 접근 제어를 추가하는 인터페이스를 제공하며, 새로운 접근 제어를 추가할 경우에는 접근 제어 메소드 간의 충돌 탐지를 위하여 동일한 접근 정의를 탐지하고 설정되어 있는 접근 제어 리스트를 저장하여 추가하려는 접근 제어와 충돌 탐지를 거쳐 그 결과를 출력한다. 차 후 최종적으로 접근 제어 추가 여부를 결정하여 저장하거나 취소한다.



(그림 6) 새로운 접근 제어의 추가

4.3 접근 제어 관리자를 이용한 그룹작업장 생성

기존의 CoSlide 협업시스템은 미리 정의된 접근 제어를 적용하여 그룹작업장을 생성하고 이를 관리하는 기능은 제공되지 않고 있다. 접근 제어 관리자는 협업이 효과적으로 수행될 수 있도록 필요한 접근 제어를 임의로 설정하여 그룹작업장을 생성할 수 있는 기능과 그룹작업장에 사용자를 등록하는 기능을 제공한다. 또한 접근 제어 관리자는 협업의 목적에 맞도록 접근 제어를 설정하여 그룹작업장을 생성하는 기능과 간편하게 그룹작업장을 생성할 수 있는 기능을 제공한다. 본 절에서는 접근 제어 관리자를 이용하여 그룹작업장을 생성하고 사용자를 등록하는 기능을 살펴보겠다.

4.3.1 간편한 설정을 이용한 그룹작업장 생성

접근 제어 관리자는 접근 제어를 세부적으로 설정하지 않고, 미리 정의되어진 보편화된 그룹작업장을 선택하여 간편하게 그룹작업장을 생성할 수 있다. 간편한 설정은 다음 4가지를 지원하고 있다.

(1) 기본 그룹작업장

그룹작업장에 등록된 사용자들은 협업을 위하여 모든 권한을 이용할 수 있지만, 그룹작업장을 삭제할 수 없으며, 접근 제어 기능을 이용할 수 없도록 접근 제어를 설정하여 그룹작업장을 생성한다. 가장 널리 사용되는 일반적인 그룹작업장의 역할을 수행하게 된다.

(2) 다운로드 전용 그룹작업장

그룹작업장에 등록된 사용자들은 그룹작업장의 자원들을 다운로드하는 기능만을 이용할 수 있다. 관리자가 정보를 제공하는 작업장으로서의 역할을 수행하게 된다.

(3) 업로드 전용 그룹작업장

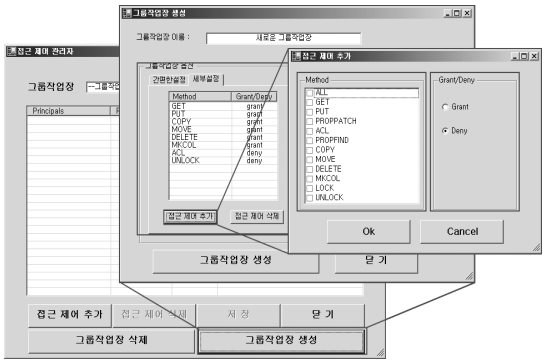
그룹작업장에 등록된 사용자들은 새로운 자원을 업로드하는 기능만을 이용할 수 있으며, 자원의 다운로드 기능은 제공되지 않는다. 사용자들이 과제물을 제출하는 작업장으로서의 역할이 수행가능하다.

(4) 모든 권한을 제공하는 그룹작업장

그룹작업장에 등록된 사용자 누구나 모든 권한을 이용할 수 있다.

4.3.2 세부 설정을 이용한 그룹작업장 생성

접근 제어 관리자는 접근 제어를 세부적으로 설정하여 그룹작업장을 생성하는 기능을 제공한다. 협업에 요구되는 웹데브 메소드의 접근 제어 허용 여부를 세부적으로 추가할 수 있다.



(그림 7) 세부 설정을 이용한 그룹작업장 생성

그림 7은 세부 설정을 이용하여 그룹작업장을 생성하는 인터페이스를 보여주고 있다. 접근 제어 추가 버튼을 이용하여 웹데브 메소드 기반의 접근 제어를 추가할 수 있으며, 추가된 메소드는 이미 설정되어 있는 메소드와 충돌 검사 후 충돌 탐지 시 적용 여부를 제시하여 최종적으로 충돌 해결된 웹데브 접근 제어 프로토콜의 권한을 설정하게 된다.

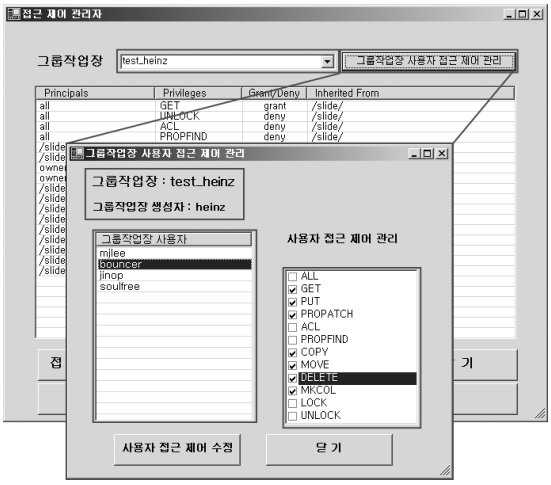
4.3.3 그룹작업장 사용자 추가

접근 제어 관리자는 그룹작업장 생성 시 사용

자 추가 기능을 제공한다. 그룹작업장 사용자 등록란의 사용자 추가 버튼을 이용하여 그룹작업장의 구성원을 추가할 수 있다.

4.4 그룹작업장 사용자 접근 제어 관리

접근 제어 관리자는 사용자의 접근 제어 관리 기능을 제공한다. 접근 제어 관리자에서 그룹작업장을 선택하면 그룹작업장 사용자 접근 제어 관리창이 실행된다. 그룹작업장 사용자 접근 제어 관리창에는 그룹작업장의 이름과 그룹작업장의 생성자, 그룹작업장에 등록된 사용자들을 보여준다. 그룹작업장 사용자를 선택하면 사용자에게 허용된 웹데브 메소드가 나타나게 되고 이를 수정하면 이미 설정되어 있는 메소드와 충돌 검사 후 충돌 탐지 시 적용 여부를 제시하여 최종적으로 충돌 해결된 웹데브 접근 제어 프로토콜의 권한을 설정하게 된다. 그림 8은 그룹작업장 사용자 접근 제어 관리 인터페이스를 보여주고 있다.



(그림 8 사용자 접근 제어 관리 인터페이스

5. 타 시스템과의 비교

접근 제어 관리자는 CoSlide 협업시스템에서 웹데브 접근 제어 프로토콜을 이용하여 그룹작업장의 접근 제어와, 그룹작업장 사용자, 그룹작업장 자원들에 대한 세부적인 접근 제어 설정을 지원한다. 표 10은 BSCW, iPlace, CoSlide 협업시스템들의 특징을 보여준다.

(표 10) 각 협업시스템들의 특징

협업시스템 / 특징	BSCW	iPlace	CoSlide
기반 프로토콜	http	http	WebDAV
개인작업장	O	O	O
그룹작업장	△	O	O
공개작업장	X	O	O
작업장 접근제어 관리 지원	X	X	O
자원들의 접근 제어 관리 지원	X	X	O

○ : 지원 △ : 부분적 지원 X : 미지원

BSCW는 대표적인 웹 기반의 협업시스템으로서 사용자를 위한 개인작업장과 그룹 사용자를 위한 그룹작업장을 부분적으로 지원한다. 하지만 다양한 작업 환경을 고려한 클라이언트와 그룹작업에 필요한 기능들을 제공하지 못한다. iPlace는 웹 기반의 협업시스템으로서 다양한 가상공간을 지원하여 효과적인 협업 수행을 지원한다. 하지만 http 프로토콜의 제약에 따라 가상공간 관리나 작업장과 자원들에 대한 접근 제어 관리 기능을 제공하지 못한다. CoSlide 협업시스템은 다양한 작업장을 제고할 뿐만 아니라 작업장과 자원들에 대하여 접근 제어 관리 기능을 제공한다.

기존의 협업 클라이언트들은 웹데브 서버에 접속하여 자원 관리, 잠금 관리, 속성 관리 등의 여러 가지 웹데브 기능을 수행할 수는 있지만 그룹작업장의 생성 및 삭제, 그룹작업장의 사용자와 자원들에 대한 접근 제어를 설정하는 기능이 미흡하다. 표 11은 이러한 접근 제어 관리자를 이용한 CoSlide 협업시스템의 특징을 여러 부분으로 나누어 웹데브 기반의 타 협업시스템 클라이언트와의 비교를 보여주고 있다.

(표 11) 타 시스템 클라이언트와의 비교

주요기능 \ 클라이언트	Internet Explorer	DAV-Explorer	Group-Explorer	접근 제어 관리자를 제공하는 CoSlide
자원 관리	△	○	○	○
그룹작업장 생성 및 삭제	X	X	○	○
그룹작업장 접근 제어 관리	X	X	○	○
그룹작업장 사용자 관리	X	X	○	○
그룹작업장 사용자 접근 제어 관리	X	X	X	○
그룹작업장 자원 접근 제어 관리	X	△	△	○
접근 제어 충돌 탐지	X	X	X	○

○ : 지원 △ : 부분적 지원 X : 미지원

Internet Explorer는 HTTP/1.1 프로토콜 전용 클라이언트로서 웹데브 기능 중 자원관리 기능을 부분적으로 수행할 수 있다. DAVExplorer는 웹데브 서버에 대한 트리 형태 보기와 잠금관리, 속성관리 등의 기능을 제공하고 자원들의 접근 제어 관리 기능을 부분적으로 제공한다. GroupExplorer는 그룹작업장의 생성과 삭제, 그룹작업 사용자 관리와 접근 제어 관리를 부분적으로 제공한다. 접근 제어 관리자를 제공하는 CoSlide 협업시스템은 협업에 요구되는 접근 제어를 설정하여 그룹작업장을 생성할 수 있으며, 그룹작업장 사용자와 자원들에 대하여 세부적은 접근 제어 관리 기능을 제공함으로서 높은 수준의 협업을 제공한다.

6. 결 론

본 논문에서는 협업시스템이 효과적인 협업을 제공할 수 있도록 웹데브 접근 제어 프로토콜을 이용하여 작업장과 자원들에 대한 접근 제어를 관리하는 기법과 이러한 기법을 CoSlide 협업시스템에 적용시킨 접근 제어 관리자에 대하여 기술하였다.

웹데브 접근 제어 프로토콜은 웹데브 서버에 의하여 관리되는 자원과 이들에 대한 접근을 임의적으로 제어할 수 있는 기능을 표준적으로 제공한다. 이러한 웹데브 접근 제어 프로토콜에서 제공하는 표준 권한을 웹데브 기반의 협업시스템에서 사용하기 위하여 웹데브 메소드 기반으로 재설계하였으며, 접근 제어 설정에 있어서 저장되어 있는 접근 제어와 새로 추가하려는 접근 제어와의 충돌을 탐지하고 이를 해결하는 기법을 설계하였다. 이러한 웹데브 메소드 기반의 접근 관리를 CoSlide 협업시스템에 적용하여 개발된 접근 제어 관리자는 협업에 필요한 접근 제어를 임의로 설정하여 그룹작업장을 생성할 수 있으며, 그룹작업장에 소속된 사용자들과 자원들에 대한 접근 제어 관리를 위하여 편리한 인터페이스를 제공함으로써 높은 수준의 협업을 수행할 수 있도록 지원한다.

참 고 문 헌

[1] F. Reiff, "PublicSpace:A Flexible Shared Workspace System," ECSCW'97, 1997.

[2] M. Roseman and S. Greenberg, "TeamRooms: Groupware for Shared Electronic Spaces," in the Proceedings of CHI'96, British Columbia, Canada, 1996.

[3] R. Bentley, W. Appelt, U. Busbach, E. Hinrichs, D. Kerr, K. Sikkel, J. Trevor, and G. Woetzel, "Basic support for cooperative work on the World Wide Web," International Journal of Human-Computer Studies, Vol.46, No.6 pp.827-846, 1997.

[4] W. Appelt, "WWW based collaboration with the BSCW system," In Proceedings of SOFSEM'99, Lecture Notes in Computer Science, Vol.1725, pp.66-78, Milovy, Czech Republic,

Springer-Verlag.

[5] 안건태, 정명희, 이근웅, 문남두, 이명준, “iPlace: EJB 기술을 이용한 웹 기반 협업시스템,” 정보처리학회논문지, 제8-D권 제6호, pp.735-746, 2001.

[6] Y. Goland, E. Whitehead, A. Faizi, S. Carter, D. Jensen, ″HTTP Extensions for Distributed Authoring - WEBDAV,″ RFC 2518, Standards Track, February, 1999.

[7] C. Kaler, J. Amsden, G. Celmm, B. Cragen, D. Durand, B. Sergeant, E. Whitehead, ″Versioning extensions to WebDAV,″ IETF Internet Draft, January, 1999.

[8] E. James Whitehead, Jr., Meredith Wiggings, ″WEBDAV: IETF Standard for Collaborative Authoring on the Web,″ IEEE Internet Computing, pp.34-40, September/October 1998.

[9] Geoffrey Clemm, ″WebDAV Access Control Protocol,“ IETF WebDAV Working Group, October, 2003.

[10] G. Clemm, E. Sedlar, J. Whitehead, ″Web Distributed Authoring and Versioning (WebDAV) Access Control Protocol,″ RFC 3744, Standards Track, May, 2004.

[11] 김동호, 박진호, 신원준, 이명준, ″웹데브 기반의 효과적인 협업 작업 지원,″ 2006년도 한국정보과학회 가을 학술 발표논문집 Vol. 33, No.2(D) pp.566-570, 2006년 8월.

[12] 김동호, 신원준, 박진호, 이명준, “웹데브 기반의 그룹 작업공간 지원,” 한국정보처리학회논문지, 제13권-C권, pp.521-532, 2006년 8월.

[13] 박희종, 김동호, 안건태, 이명준, “WebDAV 기반의 효과적인 공개 작업장 지원,” 한국정보처리학회논문지, 제 13권-C권, pp.249-258, 2006년 4월.

[14] Dong-Ho Kim; Won-Joon Shin; Jin-Ho Park; Myung-Joon Lee, ″Supporting Effective Collaborative Works Based on WebDAV,″ in Proceedings of the 1st International Forum on Strategic Technologies(IFOST 2006), pp.235-238, Oct, 2006.

[15] Oliver Zeigermann, “Jakarta Slide‘s Transcational Storage System,” http://www.theserverside.com/articles/article.tss?l=JakartaSlide, March, 2004.

[16] http://jakarta.apache.org/slide/, Jakarta Slide

[17] R. Fielding, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, “Hypertext Transfer Protocol -- HTTP/1.1,” RFC 2616, Standards Track, June, 1999.

◐ 저 자 소 개 ◑

김 성 훈
2007년 울산대학교 컴퓨터정보통신공학부 졸업(학사)
2009년 울산대학교 대학원 컴퓨터정보통신공학부 졸업(석사)
2009년~현재 경남점자정보도서관 웹관리자 재직
관심분야 : Java, C#, ASP.Net, 데이터베이스, 웹프로그래밍 등
E-mail : heinz@nate.com

이 홍 창
2006년 울산대학교 컴퓨터정보통신공학부 졸업(학사)
2008년 울산대학교대학교 대학원 컴퓨터정보통신공학부 졸업(석사)
2008년~현재 울산대학교 대학원 컴퓨터정보통신공학부 (박사과정)
관심분야 : 메시징 시스템, 웹 포탈 시스템, 웹기반 협업시스템,
E-mail : myhyunii@mail.ulsan.ac.kr

이 명 준
1980 서울대학교 수학과 졸업(학사)
1982년 한국과학기술원 전산학과 졸업(석사)
1991년 한국과학기술원 전산학과 졸업(박사)
1982~현재 울산대학 컴퓨터정보통신공학부 교수
관심분야 :웹기반 정보시스템, 프로그래밍언어, 생물정보학, 센서네트워크 프로그래밍환경
E-mail : mjlee@ulsan.ac.kr

박 양 수
1978년 울산대학교 전산학과 졸업(학사)
1981년 서울대학교 대학원 계산통계학과 졸업(석사)
1985년 서울대학교 대학원 계산통계학과 수료(박사)
1980~현재 울산대학 컴퓨터정보통신공학부 교수
관심분야 : 컴퓨터기반의 협업작업, 분산객체프로그래밍, 생물정보학
E-mail : yspk56@ulsan.ac.kr