

다중세션 공동저작 환경에서의 충돌 없는 객체 공유를 위한 대표의장 방식의 플로어 제어

신성운[†] · 오삼권^{††}

요약

많은 저작자들이 협력하여 작업을 수행할 수 있는 공동저작 시스템에서는 저작을 위해 객체들을 공유하는 일들이 참여 저작자들 간에 빈번히 발생한다. 따라서 객체 액세스를 위한 경쟁을 최소화하고 객체들의 배타적 사용을 보장해 주는 플로어 제어가 필요하다. 플로어(floor)란 공동저작 참여자에게 부여되는 한 공유자원에 대한 임시 사용 권한이다. 단일세션 환경을 위한 플로어 제어 기법들은 여러 문헌에서 찾아 볼 수 있으나 다중세션 환경을 위한 기법들은 거의 찾아 볼 수가 없다. 다중 세션 환경에서는 한 객체가 여러 세션에서 동시에 공유될 수 있다. 본 논문은 다중 세션 환경에서의 객체 공유로 인한 문제들의 해결을 위해 대표의장(chair-of-chair) 방식의 플로어 제어 기법을 제안한다. 기존의 의장형 방식을 확장한 이 방식에는 세션 의장들 중에서 선출된 대표의장이 존재한다.

키워드 : 공동저작 시스템, 다중세션, 플로어 제어, 공유객체, 대표의장

The Chair-of-Chair-Guided Floor Control for Sharing Objects without Collision in a Multi-Session Collaborative Authoring Environment

Shin Seong Woon[†] · Sam Kweon Oh^{††}

ABSTRACT

The sharing of objects for authoring frequently occurs among participating users in a cooperative authoring system where many users can collaborate. Floor control is therefore needed not only for minimizing the race condition during object accesses but also for ensuring the exclusive use of objects. A floor is a temporary access right that is given to a participating user who wants to use a shared object. Although the floor control methods for a single session environment can be found in various literature, those methods for a multi-session environment can hardly be found. In a multi-session environment, an object can be shared by multiple sessions. As a solution to the problems due to object sharing in such an environment, this paper presents a floor control method based on the chair-of-chair guidance. This method is an extension of those based on traditional chair guidance; the chair-of-chair elected among session chairs exists in this method.

Key Words : Cooperative Work System, Multi-session, Floor Control, Shared Object, Chair-of-chair

1. 서론

CSCW(Computer Supported Cooperative Work) 시스템은 망(network)으로 연결된 각자의 시스템에서 그래픽, 텍스트 등의 공유 데이터를 수정 또는 갱신하는 작업과 시청각적인 의사소통을 통해 편리하게 공동 작업을 수행하도록 지원해주는 시스템이다. CSCW의 대표적인 응용분야의 하나로 공동 저작(collaborative authoring)이 있다[1-5].

공동저작 시스템은 효율적인 공동 작업을 위해 저작의 결과물인 객체들을 공유한다. 이러한 공유객체(shared object)

들을 둘 이상의 저작자가 동시에 사용하면, 액세스 충돌 문제가 발생할 수 있다. 이러한 문제의 방지를 위해 플로어 제어(floor control) 기법이 적용된다. 플로어란 공유자원 액세스에 대한 경쟁 상태(race condition)를 최소화하고 배타적인 사용을 보장하기 위해 공동저작 참여자에게 부여되는 임시 권한이다[6, 12]. 단일세션 환경에서의 플로어 제어 연구는 많이 진행되어 왔으나, 다중세션 환경에서의 플로어 제어에 대한 연구는 거의 되어 있지 않다.

세션이란 공동저작 참여자들의 필요에 따라 생성, 초기화 또는 소멸될 수 있는 동적인 사용자 그룹을 말한다[10]. 여러 세션들이 동시에 존재 할 수 있는 분산 다중세션 환경에서는 하나의 공유객체가 여러 세션에 동시에 공유될 수 있다. 다중 공유 시의 충돌 문제를 해결하기 위해서는 세션들

※ 본 연구는 2005년도 호서대학교 교내연구비의 지원으로 수행되었습니다.

† 경희원 : SKY MobileMedia Korea 연구원

†† 종신회원 : 호서대학교 컴퓨터공학부 부교수

논문접수 : 2004년 2월 4일, 심사완료 : 2005년 8월 16일

간의 중재(coordination)뿐만 아니라 플로어 제어와 공유 객체 관리 기법간의 연동도 필요하다. 또한 효율적인 플로어 제어를 위해서는 잠금입자도(locking granularity)와 다중 세션 환경 지원 등의 플로어 제어 정책을 고려해야 한다. **잠금입자도**는 플로어 제어에서 제어권이 주어지는 대상의 범위이다.

본 논문은 효율적인 플로어 제어를 위해 공유객체의 플로어 제어를 위한 플로어 정보를 제시하고, 다중세션 공동 저작 환경에서의 충돌 없는 객체 공유를 위한 플로어 제어 기법을 제안한다. 본 논문의 구성은 다음과 같다. 2장은 기존의 플로어 제어 기법에 대해 설명한다. 3장은 공동저작 전체 시스템 구조를 제시하고, 4장은 플로어 제어 정책과 플로어 정보를 설명한다. 5장은 대표의장 방식의 플로어 제어를 제시한다. 6장에서는 구현 및 성능 평가를 설명하고, 마지막 7장에서 결론을 맺는다.

2. 관련 연구

CSCW에서의 플로어 제어 방법은 선입선처리형(first-come first-serve), 시분할형(time slicing), 라운드-로빈형(round-robin), 방입형(free control), 선점형(preemptive), 그리고 의장형(chair guide) 등이 있다. <표 1>은 CSCW에서의 플로어 제어 방법의 장단점을 요약한 표이다[6, 12].

CSCW의 대표적인 응용분야인 공동저작 시스템에서의 플로어 제어에 관한 연구로는 JVTOS[7], SHARE[8], MASH[9]

<표 1> 기존 CSCW에서의 플로어 제어 장단점

구분	설명	장단점
선입선처리형 (first-come first-serve)	플로어 요청이 큐에 저장되고 현재 플로어를 가지고 있는 사용자가 플로어를 석방하였을 때 큐의 맨 앞에 있는 사용자가 플로어를 갖는다.	큐의 마지막에 있는 사용자가 플로어를 할당 받으면 장시간 기다려야 하는 단점이 있음.
시분할형 (time slicing)	각 사용자들은 정해진 순서에 의해 정해진 시간만큼씩 플로어를 가질 수 있다.	작업시간이 일정하지 않은 작업에서는 불리함
라운드-로빈형 (round-robin)	플로어를 정해진 순서에 따라서 플로어를 할당 받는다.	작업이 순서대로 진행되는 공동작업에 적합함.
방입형 (free control)	모든 사용자가 언제든지 플로어를 가질 수 있다.	비교적 작은 규모의 공동작업에 적합함.
선점형 (preemptive)	모든 사용자는 현재 플로어를 쥐고 있는 사용자로부터 선점식 권한을 받을 수 있다.	작업 선점의 영향을 받지 않은 공동작업 환경에 적합.
의장형 (chair-guided)	의장이 신청자 중에서 플로어를 부여한다. 현재 사용자가 플로어를 놓았을 때 플로어는 무조건 다시 의장에게 돌아가고 의장은 이 플로어를 다른 사용자에게 넘겨줄 수 있다.	규모가 비교적 큰 공동작업 환경에 적합.

<표 2> 공동작업 시스템에서의 플로어 제어

	[JVTOS]	[SHARE]	[MASH]
플로어 할당 매커니즘	토큰	순서 배정	수신자 기반
플로어 제어 (혼합형)	의장형 방식, 큐	선점형, 의장형 방식	중계형
세션 관리 구조	분산 관리	집중형	---
응용프로그램 구조	분산	집중	분산
다중 세션 지원	---	---	---
플로어 잠금입자도	---	---	---

등이 있다.

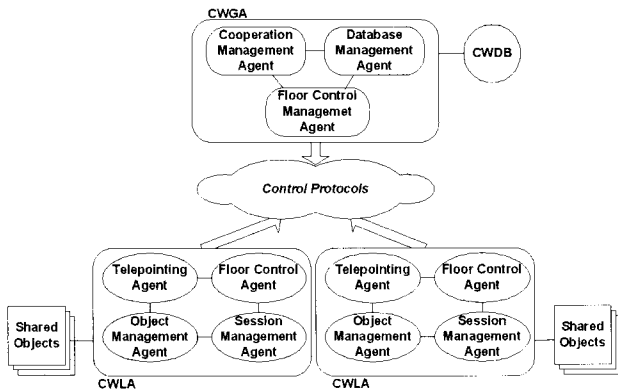
JVTOS는 이기종(heterogeneous)환경에서 범용으로 사용할 수 있는 공유윈도우 시스템을 제안하며, 하부시스템은 분산형 구조 기반이다. 플로어 제어는 공유된 응용프로그램에 대한 제어와 응용프로그램에 대한 입력권한 제어를 위한 토큰(token) 전달 방식의 관점에서 의장형 방식을 사용한다.

SHARE는 중앙집중형 구조의 공유윈도우 시스템에서 입력권한 제어를 위한 플로어 제어 기법을 제안한다. SHARE에서 플로어 제어는 의장이 순서를 할당하는 순서배정(turn taking) 방식을 사용한다. SHARE에서는 세션개념을 사용하지 않고 등록기(register)개념을 사용했다. 등록기는 사용자들의 회의 참여와 탈퇴, 종료를 관리한다. 플로어 제어는 선점형과 의장형 방식 등을 사용한다.

MASH는 버클리 대학에서 개발한 MASH 프로젝트의 한 모듈로서 개발됐다. MASH는 MBONE상에서 원격회의를 진행할 때 오디오, 비디오, 채널에 대한 중재(coordination), 화이트보드에 대한 입력권한 제어를 위한 플로어 제어 모듈을 제안한다. 플로어 제어는 세션 제어와 통합된 관리방식으로 수신자 기반(receiver based) 방식을 택한다. 수신자 기반 방식은 중계자(moderator) 시스템의 에이전트가 플로어 권한이 없는 사용자의 입력은 무시(mute)하고 플로어 권한이 있는 사용자의 입력만을 회의 참여자들에게 멀티캐스트 한다. 다음 <표 2>는 공동저작 시스템에서의 플로어 제어를 비교한 내용이다.

3. 시스템 모델

CW-MAN(Cooperative-Work Management System)은 집중형 구조와 분산형 구조의 단점을 보완한 혼합형 구조의 공동저작 관리시스템이다[10]. CW-MAN은 관리자 시스템에 존재하는 CWGA(Cooperative-Work Global Agent)와 각 저작자 시스템에 존재하는 CWLA(Cooperative-Work Local Agent)로 구성된다. (그림 1)은 CW-MAN 기반에서 플로어 제어 모듈을 추가한 전체구조를 보여주고 있다.



(그림 1) 전체 시스템 구조

- CWGA** : CWGA는 관리자 시스템에서 동작하는 에이전트로써 DBMA(Database Management Agent), CMA(Cooperation Management Agent), 그리고 FCMA(Floor Control Management Agent)로 구성된다. DBMA는 공동저작 참여자 및 공유객체들의 관리 정보와 각 세션의 작업 히스토리와 같은 공동저작 관리를 위한 정보를 보관하는 데이터베이스인 CWDB(Cooperative-Work DataBase)를 관리하고, CWLA나 CMA로부터의 CWDB 액세스 요청을 처리하는 기능을 담당한다. 관리정보의 일관성 유지를 위해 CWDB에 대한 액세스는 항상 DBMA를 통해서만 이루어진다. CMA는 CWLA로부터의 공유객체 액세스 요청을 받아 공유객체가 위치한 시스템의 CWLA에 연결하도록 하는 기능과 세션의 생성, 초기화, 참가 등의 요청들을 받아 새로운 세션의 생성 및 기존 세션으로의 참여를 할 수 있도록 지원하는 역할을 담당한다. FCMA는 효율적인 플로어 제어를 위해 각 세션 의장들을 통합 관리하는 역할을 담당한다. FCMA에서 대표의장 역할을 수행하며, 각 CWLA FCA를 통합 관리한다.
- CWLA** : CWLA는 각 저작자 시스템에서 동작하는 에이전트로써 SMA(Session Management Agent), OMA(Shared Object Management Agent), TPA(Telepointing Agent), 그리고 FCA(Floor Control Agent)로 구성된다. SMA는 각 CWLA의 SMA와 상호협력력을 통해 세션 관리 기능을 수행한다. SMA는 세션 의장 시스템에서 동작할 경우, 각 저작자로부터의 세션 참가, 탈퇴 또는 새로운 세션 생성 등의 요청을 세션 관리 정책(session management policy)에 따라 처리하는 세션 관리와 텔레포인터(telepointer)에 대한 입력권한을 관리하는 기능을 담당하며, 세션 참여자 시스템에서 동작할 경우, 지역 시스템의 공유객체를 관리하고 저작자, 또는 응용프로그램으로부터의 공유객체 액세스 요청을 FCA에 전달하는 기능을 담당한다. OMA는 지역 시스템에 있는 공유객체들을 관리하고, 응용프로그램과 텔레포인터 에이전트로부터 공유객체에 대한 액세스 요청을 받아 공유객체들을 그 위치에 상관없이 액세스

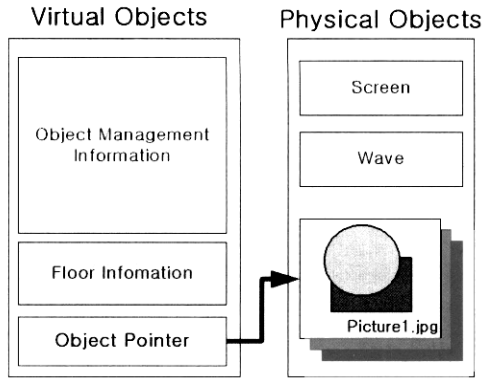
할 수 있도록 한다. TPA는 공유 윈도우 안에서의 실시간 텔레포인터 기능을 지원하는 에이전트로서 세션 의장 시스템에서 동작할 경우에는 텔레포인터 시 텔레포인터 입력 권한을 가진 저작자의 입력을 모든 세션 참여자 시스템으로 전송하는 코디네이터역할을 담당한다. FCA는 세션 의장 시스템에서 동작할 경우 플로어 제어 정책에 따라 플로어를 할당하여, 해당 세션 내 공유객체의 플로어를 제어한다. 세션 참여자 시스템에서 동작 할 경우 공유객체의 플로어 제어 정보를 관리하고, 공유객체 액세스를 위한 플로어 요청과 반환을 담당한다.

4. 플로어 제어 정책과 플로어 정보

공동저작 환경에서 효율적인 플로어 제어를 위해서는 다양한 공유객체의 잠금입자도를 고려해야 한다. 잠금입자도는 멀티미디어 통신에서 발언권으로, 문서에서는 쪽, 단락, 심지어 한 글자로 정할 수 있으며, 화이트보드에서는 그림 객체 등으로 정의 할 수 있다. 일반적으로 잠금입자도에 대한 플로어 권한은 단일사람에게만 주어지는 것이 일반적이다. 하지만, 상황에 따라 두 사용자 이상에게 주어져야 하는 경우도 있다. 예를 들어 토론식 회의의 경우 특정 한 사람에게만 발언권을 주기 보다는 일부에게 권한을 주어야 하며, 회의 상황에 따라 참가자 모두에게 발언권을 부여하거나, 제한할 수 있어야 한다. 이를 위해 공유객체의 사용 권한은 사용자 수를 제한하거나 수정할 수 있어야 하며, 관리가 용이해야 한다. 또한 플로어가 할당된 이후 플로어의 독점을 막고, 효율적인 플로어 반환을 위해 플로어의 휴면 시간을 계수하여야 한다. 플로어 제어를 위해서는 플로어를 참가자들에게 어떻게 할당할 것인가를 결정하는 **플로어 할당 정책**과 이를 수행하기 위한 구체적인 방법인 **플로어 제어 메커니즘**이 필요하다[11]. 다음은 본 연구에서 가정한 플로어 할당 정책이다.

- 세션을 처음 생성하거나, 세션을 처음 초기화한 참가자가 세션 의장이며, 플로어 제어 권한자이다.
- 일반적으로, 다양한 공유객체를 고려하기 위해 잠금입자도는 물리적 파일(txt, bmp, jpg.....)과 화이트보드의 객체, 발언권, 영상권 등을 지원한다.
- 공유객체의 종류에 따라 플로어를 단일 사용자 또는 다중 사용자에게 할당이 가능하며, 사용자 수를 제한한다.
- 분산 다중세션 환경에서 두 세션 이상 동시에 공유된 공유객체를 지원한다.
- 시간적 제한을 두어 일정 시간 작업이 이루어지지 않으면 다음 사용자를 위해 플로어를 자동 반환한다.
- 시스템 결함 및 네트워크 단절 시 작업의 진행을 보장한다.

공동저작 시스템에서는 이러한 정책을 모두 적용해야 하며, 플로어 제어 정책은 작업의 종류와 상태에 따라서 변경



(그림 2) 가상객체와 물리 객체

될 수 있다. 플로어 제어 메커니즘은 각 정책을 수행하는데 구체적인 방법들로 요구(request), 할당(assign), 반환(release), 취소(revoke) 등으로 나눌 수 있다[11].

본 논문에서는 공동저작 시스템에서의 대표의장형 방식의 플로어 제어를 적용하기 위해 기존 CW-MAN의 공유객체 관리를 보완하여 공유객체를 가상객체와 물리적 객체로 나누어 저장 및 관리하였다. 가상객체는 객체의 정보, 저작자, 물리적 위치, 공유 윈도우에서의 위치, 세션 정보, 버전정보, 그룹정보와 플로어 제어를 위한 플로어 정보 그리고, 물리적 객체의 지시자로 구성하였으며, 물리 객체는 공유객체의 저작된 결과 내용만을 파일형태로 저장하였다. (그림 2)는 플로어 제어 관점에서의 가상객체와 물리 객체의 관계를 보여주고 있다.

이러한 객체의 분류는 일반적인 공동저작 시스템과는 달리 기능의 추가, 확장이 용이하며, 물리 객체의 독립성을 보장하기 때문에, 다른 응용프로그램을 이용한 작업과 OLE(Object Linking and Embedding) 작업을 가능하게 한다. <표 3>은 다중세션 환경을 지원하는 대표의장 방식의 플로어 제어를 위해 관리되어야 할 플로어 정보를 설명한다.

플로어 상태는 공유객체 액세스의 가능 여부 판별에 사용되며, 사용자 제한은 멀티미디어 회의에서 발언권과 영상권

에서의 사용자 수를 제한한다. 마지막으로 플로어 레벨과 플로어 모드 정보는 다중 세션 환경을 지원하는 대표의장 방식의 플로어 제어를 위해 구성했다.

5. 대표의장 방식의 플로어 제어

공동저작 시스템의 플로어 제어 방법으로는 비교적 규모가 큰 공동작업 환경에 적합하며 의장이 모든 플로어를 할당함으로써 충돌 없는 플로어 제어가 용이한 의장형 방식이 적합하다. 하지만, 기존의 의장형 방식은 의장이 모든 플로어를 관리하기 때문에 다중세션 공동저작 환경에 적용될 경우 몇 가지 문제를 유발한다. 다음은 다중세션 환경에서의 기존 의장형 방식의 문제점이다.

- 하나의 공유객체가 두 세션 이상에 동시 공유되었을 경우 공유객체의 플로어 제어 권한은 각 세션에 있는 여러 의장들이 가지기 때문에 해당 공유객체 액세스 시 충돌 문제를 야기 할 수 있다.
- 한 시스템에서 전체 플로어를 제어 할 경우 시스템 및 네트워크의 과부하가 발생할 수 있다.
- 세션 의장 시스템의 결함 시 작업의 진행이 불가능하다.

이의 해결을 위해 대표의장(chair-of-chair)형 방식의 플로어 제어를 제시한다. 대표 의장형 방식은 기존의 의장형 방식을 보완하여, 각각의 참가자들과 의장들 그리고 여러 의장들을 통합 제어하는 대표의장으로 구성된다. 그리고 플로어 제어 방법으로 다음과 같은 네 가지 모드(mode)로 동작한다.

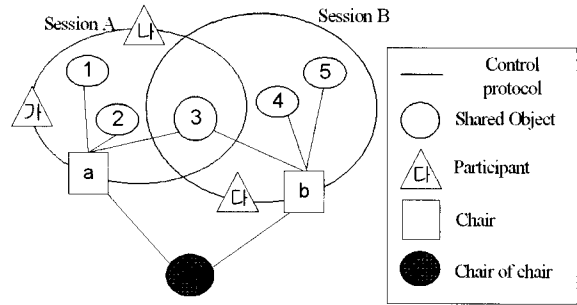
- **의장 자치 모드**: 공유객체가 단일 세션에 공유되는 경우로, 각 세션 의장이 플로어를 제어한다.
- **의장 대표의장 모드**: 공유객체가 두 세션 이상 동시에 공유된 경우로, 대표의장만이 최종적으로 해당 객체의 플로어를 할당 또는 취소할 수 있는 권한자이며, 해당 의장들은 대표의장의 대리인이 되어 플로어를 관리

<표 3> 대표의장 방식을 위한 플로어 정보

구분	내용	입력 형태	비고
잠금 입자도 (locking granularity)	플로어 제어권이 주어지는 대상	발언권, 영상권, text, bitmap등	공유객체에 따른 사용자 정의.
플로어 상태 (floor status)	공유객체의 플로어 상태를 나타낸다.	free, busy,	
사용자 제한 (user limit)	공유자원의 사용자 개수를 제한 또는 관리 한다.	NULL, (1/ 0), (2/ 1)...	
휴면 시간 (idle time)	플로어를 할당받은 후 작업이 이루어지지 않은 휴면 시간을 계수한다.	0,1,2,3...	일정시간 경과 시 플로어를 자동 반환한다.
다중 세션 지원	플로어 레벨	공유객체의 계급으로 몇 개의 세션에 속해 있는지를 계수한다.	0,1,2,3... 0: 단일 저작(세션에 속하지 않음) 1: 하나의 세션에 공유된 경우. 2: 두 세션에 공유된 경우.
	플로어 모드	플로어 제어의 동작 모드를 나타낸다.	단일 저작 모드, 의장자치 모드, 의장 대표의장 모드, 대표의장 모드

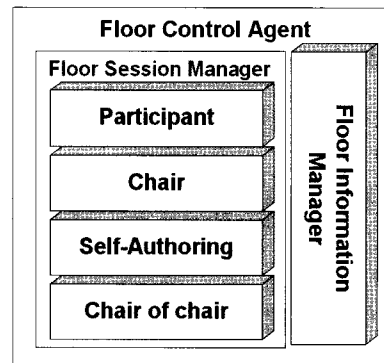
및 제어한다.

- **대표의장 모드**: 세션 의장 시스템의 결합 및 의장과의 네트워크 단절 발생 시 플로어를 직접 대표의장에게 요청하여 대표의장이 플로어를 제어한다.
- **단일 저작 모드**: 단일 시스템에서의 혼자 저작하는 경우거나 의장 및 대표의장과 네트워크가 단절된 경우로 공유객체가 비 공유 상태로 변환되며, 시스템 내부에서 자체적으로 플로어를 제어하게 한다.



(그림 3) 대표의장형 방식의 플로어 제어 구조

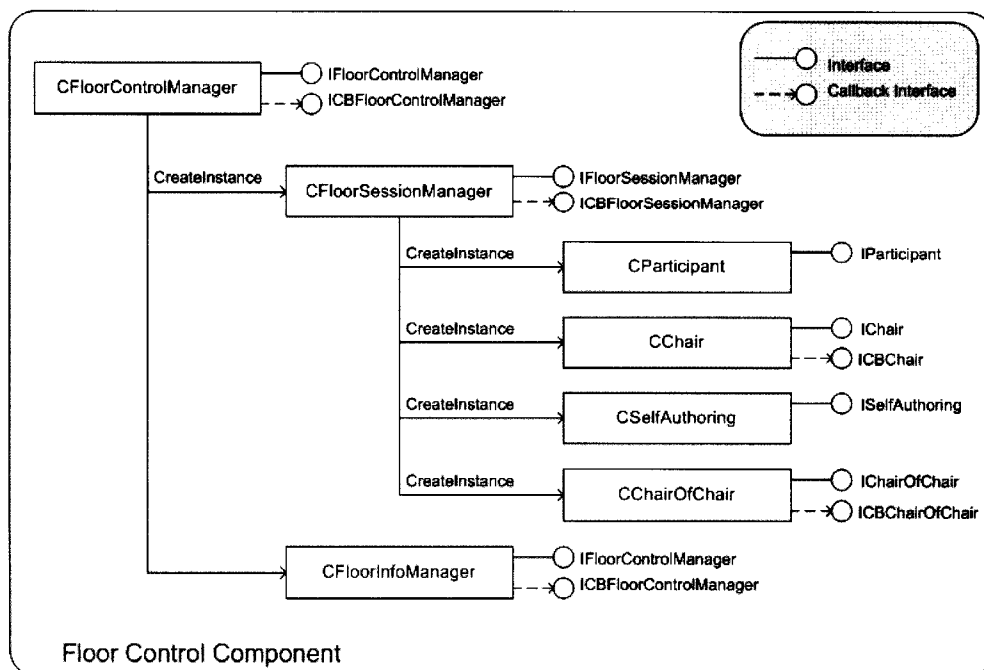
(그림 3)은 대표의장형 방식의 플로어 제어 구조이다. 세션 A의 참가자 <가, 나>의 경우 해당 세션 내 특정 공유객체의 플로어를 얻기 위해서는 세션 A의 의장 a에게 플로어를 요청하게 된다. 공유객체 1, 2,와 같은 단일 세션에 속한 경우 의장 자치 모드로 동작하여 의장 a가 플로어를 할당 또는 거절한다. 하지만, 공유객체 3의 경우 세션 A와 세션 B에 동시에 공유되어 있기 때문에, 각각의 해당 의장들이 플로어를 할당할 경우, 충돌 문제가 발생할 수 있다. 이의 해결을 위해, 의장 a, b는 플로어를 대표의장에게 요청하게 되며, 최종적으로 대표의장이 플로어를 할당 또는 거절한다. 이때 의장 a, b는 대표의장의 대리인이 되어 플로어 할당 여부를 전달하게 된다. 또한, 세션 A의 참가자 (가)의 경우, 작업 중 해당 세션의장 a의 종료 메시지가 전달되거나, 플로어 요청 시, 응답이 없으면, 플로어 제어 모드가 대표의장 모드로 변경되어 전체의장에게 직접 플로어를 요청 및 할당 받게 된다. 하지만, 대표의장과 통신도 두절되게 되면, 플로어 제어 없이 모든 공유객체의 수정이 가능한 단일 저작모드로 동작하게 된다. 다음 (그림 4)는 공동저작 시스템에서의 플로어 제어 모델을 보여주고 있다.



(그림 4) 플로어 제어 모델

플로어 제어 모델은 CW-MAN기반에서의 플로어 제어 에이전트(floor control agent)이다. 플로어 제어 에이전트는

플로어 정보 관리자와 플로어 세션 관리자로 구성된다. 플로어 정보 관리자는 공유객체 에이전트를 통해 가상객체의 플로어 정보를 액세스 하며, 플로어 세션 관리자는 공유객체의 플로어 정보에 따라 4가지 모드(참가자, 의장, 단일저작 그리고 대표의장) 중 하나의 모드로 동작한다. (그림 5)는 COM 기반에서의 플로어 제어 에이전트의 객체 구조와 객체들의 생성 모습을 보여주고 있다.



(그림 5) 플로어 제어 에이전트의 객체 구조 및 인터페이스

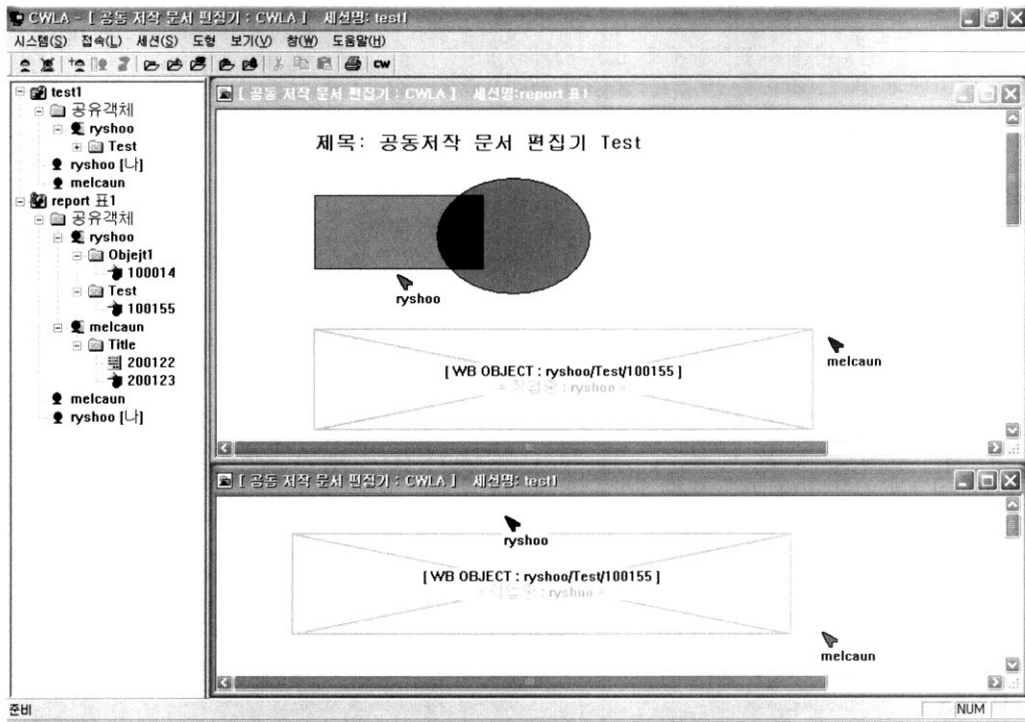
플로어 에이전트의 구성은 객체와 인터페이스 그리고 필요여부에 따른 Callback 인터페이스를 제공한다. CFloorControlManager는 공동저작에서 세션의 생성 및 참가 시 자동으로 생성된다. CFloorControlManager는 플로어를 관리하기 위한 CFloorSessionManager와 공유객체의 플로어 정보에 액세스하기 위한 CFloorInfoManager를 생성한다. CFloorControlManager는 특정 공유객체의 플로어를 제어하기 위해서는 공유객체의 플로어 제어 정보를 요청하고, 정보와 함께 CFloorSessionManager에게 필요한 모드의 인터페이스를 받아 플로어를 요구, 할당, 거절 또는 취소한다.

6. 구현 및 성능 평가

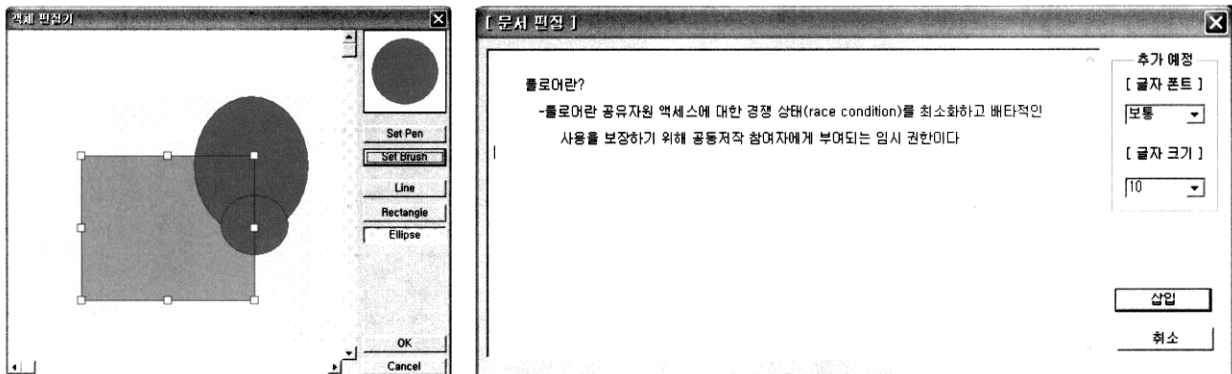
본 논문에서 제안한 다중세션 공동저작환경에서의 충돌

없는 플로어 제어가 잘 정의됐음을 보이기 위해 공동저작 문서편집기를 구현했다. 공동저작 문서편집기는 Microsoft Windows XP 환경에서 Visual Studio C++ 6.0을 사용하여 구현했으며, Windows 2000이하에서도 문제없이 실행됨을 확인했다.

(그림 6)은 다중 세션환경을 지원하는 공동저작 문서 편집기의 실행모습이다. 공동저작 문서 편집기는 글 그리고 도형 두 가지 잠금입자도를 지원하며, 실시간으로 텔리포인팅 및 작업 내용이 공유되는 공유윈도우와 객체 및 사용자 정보를 관리하는 작업 관리 윈도우로 구성 된다. 그리고, 공유객체의 수정 및 작업이 이루어지는 작업 어플리케이션이 있다. 작업 어플리케이션은 공유객체의 플로어가 할당되면, 자동으로 실행되며, 공동저작 시스템과는 독립적으로 동작하는 보조 어플리케이션이다.



(그림 6) 공동저작 문서편집기를 실행한 모습



(a)도형 편집기

(b) 문서 편집기

(그림 7) 작업 어플리케이션의 실행모습

(그림 7)은 작업 어플리케이션으로 도형편집기와 문서 편집기를 실행 모습이다. 이는 공유객체 관리 정보와 실제객체를 파일로, 분류하여 저장하기 때문에 다른 어플리케이션과의 호환과 특정 공유객체 편집기의 추가 적용이 용이하다. 문서 편집기로 저장된 글의 경우, 실제 객체의 내용이 TXT 파일로 저장되어, 메모장등 다른 어플리케이션을 이용한 편집이 가능했다. <표 4>는 본 논문에서 구현한 대표의장 방식의 공동저작 문서 편집기의 기능 비교를 위해 NetMeeting에서 제공하는 화이트보드와 기능을 비교한 것이다.

여기서, 텔리포인터 표현기능은 참여자들의 이름 및 텔리포인터의 움직임을 다른 모든 참여자들의 공유윈도우에 나타내는 것을 말한다.

대표의장형 방식 플로어 제어의 성능 평가로는 하나의 대표의장 시스템과 8명의 작업자가 한 세션에 4명씩 두 세션을 구성 그리고 각 세션은 한명의 의장과 3명의 참가자로 구성하였다. 한 세션에 공유된 공유객체와 두 세션에 공유된 공유객체를 번갈아가며 플로어를 할당 및 반환하는 방법으로, 시스템의 리소스 사용량을 수집했다. <표 5>는 대표의장 공동저작 시스템에서의 플로어 제어에 대한 리소스 사용률이다. 리소스 사용률 측정을 위해 같은 성능을 가지는 인텔CPU 기반의 PC 9대를 사용 했으며, Windows XP 환경에서 100Mbps 로컬LAN을 이용하여 공동저작 시스템을 운용했다. 또한 효율적인 사용률 측정을 위해, 텔리포인터와 공유객체 정보 전송 등은 제외했다.

위 실험결과 다중 세션 환경에서의 대표의장 방식의 플로어 제어는 기존의 단일 세션을 확장하여 안정적인 다중세션

환경에서의 충돌 없는 공유객체 공유를 지원하였다. 또한, 분산된 리소스 사용으로, 기존 의장형 방식의 네트워크 과부하를 크게 줄일 수 있었으며, 네트워크 단절 시 단일 저작 변경이 가능했다. 하지만, 단절에서 복귀 되었을 때의 버전관리 및 재공유등의 문제점 해결을 위한 연구가 필요하다.

7. 결론 및 향후 과제

본 논문은 다양한 목적의 공동 저작에서 응용될 수 있는 다중세션 환경에서 충돌 없는 객체 공유를 위한 플로어 제어 기법과 공유객체의 플로어 관리를 위한 플로어 제어 정보를 제시했다. 플로어 제어 정보는 물리적 파일과 화이트보드의 객체, 발언권, 영상권 등의 잠금입자도를 지원하며, 멀티미디어 회의에서의 경우 플로어를 단일 사용자가 아닌 다중 사용자에게 할당할 수 있도록 사용자 수를 관리 및 제한한다. 또한 다중세션 환경을 위한 플로어 레벨과 플로어 제어 모드에 대한 정보를 포함한다. 플로어 제어 기법으로는 대표의장 방식을 제안한다. 이는 공유객체의 다중 세션 공유 여부에 따라 의장 자치 모드, 대표의장 모드, 의장-대표의장 모드 그리고 단일저작 모드 이렇게 네 가지 모드로 동작하여 기존의 하나의 시스템에서의 플로어 제어를 분산시켜 부하를 줄이며 효율성을 높였으며, 하나의 공유객체가 여러 세션에 동시에 공유 되는 다중 공유 시 충돌 문제를 해결했다. 보다 효율적인 플로어 제어를 위해서는 공유객체의 사용권한, 버전관리 그리고 공유객체 그룹화 관리기법에 대한 연구가 필요하다. 이를 위해 향후 연구에서는 범용성 있는 공동저작 시스템을 위한 공유객체 관리 기법과 공동저작의 응용으로 하나로서, 공동저작 XML 편집기(Editor)에 대해 연구를 진행할 계획이다.

참 고 문 헌

- [1] 성미영, "MissCW: 다중 사용자 동기적 공동 저작 시스템", 정보처리학회 논문지, pp.1697-1706, Dec., 1996.
- [2] Jonathan Grudin, "Computer-Supported Cooperative Work: History and Focus," IEEE Computer, Vol.27, NO.5, pp.19-25, May, 1994.
- [3] Walter ReinHard, et al, "CSCW Tools: Concepts and Architectures," IEEE Computer, Vol.27, No.5, pp.28-36, May, 1994.
- [4] F. Pacall, A. Sandoz, A. Schiper, "Duplex: A Distributed Collaborative Editing Environment in Large Scale," Proceedings on CSCW'94, ACM Press, pp.65-173, October, 22-26, 1994.
- [5] 김동성, 이광행, 오삼권, "분산형 멀티미디어 공동작업 관리 시스템의 설계", 한국정보처리학회 춘계학술발표논문집, 제3권, 제1호, pp.682-686, 1996.
- [6] H. P. Dommel and J. J. Aceves, "Floor Control for Multi-

<표 4> 대표의장형 문서 편집기의 기능 비교

구 분	Microsoft NetMeeting의 화이트 보드	대표의장 문서 편집기	비고
다중 세션 지원	제공하지 않음	제공됨	세션 생성 및 연결 제한 없음
사용자 제한	제한 없음	제한 없음	
텔리포인터 표현	제공하지 않음	제공됨	필요에 따라 텔리포인터 표현 해제 가능
공유 자원의 플로어 표시	제공하지 않음	제공됨	다른 사용자의 사용 여부 표시(이름, 작업 중 등)
플로어 자동 반환	제공하지 않음	제공됨	일정 시간 사용이 이루어 지지 않으면 플로어 자동 반환

<표 5> 대표의장형 문서 편집기의 성능 평가

구 분(평균)	대표의장	의장	참가자
CPU 사용률(%)	3%	28%	14%
초당 송수신 데이터(바이트)	34	198	115

media Conferencing and Collaboration,” ACM Multimedia Systems, Vol.5, No.1, pp.23-38, 1997.

- [7] Thomas Gutekunst, Thomas Schmidt, Gnter Schulze, Jean Schweitzer, Michael Weber: “A Distributed Multimedia Joint Viewing and Tele-Operation Service for Heterogeneous Workstation Environments.” GI/ITG Workshop on Distributed Multimedia Systems. Stuttgart, 1993.
- [8] S. Greenberg. “Sharing Views and Interactions with Single-User Applications.” In Proceedings of the Conference on Office Information Systems, pp.227-237. ACM, April, 1990.
- [9] CHAWATHE, Y., AND WANG, H. J. “A Coordination Architecture for Mbone Conferences.” U.C. Berkeley CS268 “Computer Networks” Course Project Report. Paper online 4 , May, 1997.
- [10] 이광행, 전제우, 오삼권, “CW-MAN: 효율적인 멀티미디어 공동저작을 위한 혼합형 구조의 공동저작 관리 시스템”, 정보처리논문지, 제6권, 제5호, pp.1253-1262, 5월, 1999년.
- [11] Hans-Peter Dommel and J.J. Garcia-Luna-Aceves, “Floor Control for Multimedia Conferencing and Collaboration,” Multimedia Systems(ACM/Springer Verlag), Vol.5, No.1, pp.23-38, Jan., 1997.
- [12] 신성운, 오삼권 “분산 다중세션 환경에서의 공동 저작을 위한 플로어 제어”, 한국정보처리학회 춘계학술발표논문집, 제10권, 제1호, pp.1089-1092, 2003.

신성운



e-mail : melcaun.shin@skymobilemedia.com
 2003년 호서대학교 컴퓨터공학과(학사)
 2005년 호서대학교 컴퓨터응용기술
 (공학석사)
 2004년~현재 SKY MobileMedia Korea
 연구원

관심분야: CSCW, Embedded Systems, Computer Games

오삼권



e-mail : ohsk@office.hoseo.ac.kr
 1980년 한국항공대학교 항공전자공학과
 (학사)
 1986년 University of South Florida
 컴퓨터과학 및 공학(공학석사)
 1994년 Queen's University, 컴퓨터 및
 정보과학(공학박사)

1980년~1984년 삼성전자 통신연구소
 1994년~1995년 한국전자통신연구원
 1995년~현재 호서대학교 컴퓨터공학부 부교수

관심분야: Ubiquitous Computing, Embedded Systems, Computer Games