

CW-MAN : 효율적인 멀티미디어 공동저작을 위한 혼합형 구조의 공동저작 관리 시스템

이 광 행[†] · 전 재 우^{††} · 오 삼 권^{†††}

요 약

분산환경에서의 효율적인 공동저작을 위해서는 분산된 자원들의 공유와 저작자들간의 상호 협력을 용이하게 해주는 공동 저작 관리 시스템이 요구된다. 본 논문은 공유객체(shared object) 및 공동저작 관리정보를 중앙 관리자 시스템에 집중시키고, 공동저작에 사용되는 공유객체들을 각 저작자 시스템에 분산시킨 혼합형 구조를 갖는 공동저작 관리 시스템인 CW-MAN을 제시한다. CW-MAN은 분산된 공유객체들을 저작자와 공유객체의 위치에 상관없이 액세스(access)할 수 있도록 하고, 공유객체에 대한 권한요청을 충돌 없이 처리되도록 하는 공유객체 관리 기능, 저작자들의 필요에 따라 생성, 초기화 또는 소멸시킬 수 있는 동적인 사용자 그룹인 세션(session) 관리 기능, 그리고 지역 포인터(local pointer)의 위치와 이벤트(event)를 공유윈도우(shared window)에 동기적으로 표현함으로써 동기적 공동저작을 지원하는 텔레포인팅(telepointing)을 제공한다.

CW-MAN : The Cooperative-Work Management System with Hybrid Architecture for Efficient Multimedia Collaboration

Kwang-Hang Lee[†] · Jae-Woo Jeon^{††} · Sam-Kweon Oh^{†††}

ABSTRACT

For the efficient co-authoring of multimedia documents in a distributed systems environment, the cooperative-work management systems that can not only allow the sharing of distributed resources but also facilitate the authors' interaction, are needed. In this paper, we present a *cooperative-work management system with hybrid architecture*, called *CW-MAN*, in which the information on shared objects and co-authoring management is stored in a dedicated computer system but the shared objects distributed all over the local computer systems. The main management functions provided by the CW-MAN are those for shared objects, session, and telepointing. The shared-object management allows authors not only to access the shared objects without knowing their physical location but also to use them exclusively. The session management allows authors to dynamically create, update, and destroy sessions. Lastly, the telepointing management allows local pointing events to be synchronously presented to remote systems.

* 본 연구는 '98년도 정보통신부 대학기초연구지원을 받아 행해
졌음

† 준 회 원 : (주) 디아이 연구소 연구원

†† 준 회 원 : 호서대학교 컴퓨터공학과

††† 총신회원 : 호서대학교 컴퓨터학부 교수

논문접수 : 1998년 9월 21일, 심사완료 : 1999년 3월 3일

1. 서 론

컴퓨팅 환경이 발전함에 따라 기존의 텍스트(text)와 이미지(image)를 위주로 하는 문서에서 동영상과 사운드(sound)가 추가된 멀티미디어 문서(multimedia document)의 저작이 가능해졌다. 이러한 멀티미디어 문서 저작의 경우 미디어의 종류에 따른 전문성이 요구되기 때문에 내용상 여러 분야의 전문가가 함께 작업해야 하는 경우가 많다. 이러한 이유로 지리적으로 분산된 저작자들이 망(network)을 통해 의사교환과 상호협력을 할 수 있고, 자원 및 작업 결과를 공유하여 저작에 사용할 수 있도록 지원하는 공동저작 시스템에 관한 연구가 진행되고 있다[1,2]. 공동저작 시스템의 주요 기능들로는 멀티미디어 문서의 편리한 저작을 위한 문서편집 기능, 저작자 간의 의사소통 기능, 분산된 자원과 작업결과를 공유하여 공동저작에 이용할 수 있도록 관리하고, 다른 사용자들의 작업위치와 작업상태를 알 수 있도록 함으로써 협력(cooperation)을 지원하는 공동저작 관리기능 등이 있다.

공동저작 시스템은 관리정보와 공유객체의 위치에 따라 집중형(centralized system)과 분산형(distributed system)으로 구분할 수 있다[3,4]. 집중형 구조는 관리정보와 공유객체가 하나의 서버에 집중되어 있고, 저작자들은 망으로 연결되어 있는 터미널을 통해 서버에 있는 공유객체를 이용하여 작업하는 구조이며, 분산형 구조는 관리 정보와 공유객체가 각 저작자들의 시스템에 분산되어 있어서 각 저작자가 자신의 관리정보, 또는 공유객체를 직접 이용하거나, 다른 저작자 시스템에 위치한 관리 정보, 또는 공유객체를 이용할 수 있는 구조이다. 집중형 구조의 공동저작 시스템으로는 MMConf[5]와 Rendezvous[6] 등이 있고, 분산형 구조의 공동저작 시스템으로는 MissCW[1] 등이 있다. 집중형 구조는 관리정보와 공유객체들이 하나의 시스템에 편중되어 있기 때문에 그 관리가 쉽지만 서버에 많은 부하(overhead)가 생길 수 있는 단점이 있다. 반면에 분산형 구조는 공유객체와 관리정보를 각 저작자 시스템으로 분산시켰기 때문에 집중형 시스템에서 발생할 수 있는 서버의 과부하 현상을 막을 수 있으나, 관리정보의 불일치와 같은 오류가 발생할 수 있으므로 공동저작 관리정보에 대한 일관된 관리가 어렵다[7].

이러한 집중형과 분산형 구조가 갖는 단점의 보완을

위해, [8]에서는 복합실행방식(hybrid execution approach)을 제안하였다. 복합실행방식은 공동저작에 사용되는 응용프로그램들이 각 저작자 시스템으로 복제되어 실행되도록 함으로써 집중형일 경우 발생할 수 있는 서버의 과부하 현상을 해결하고자 하였고, 관리정보, 공유객체들, 그리고 세션 책임자(session chair)를 서버에 집중시킴으로써 분산형일 경우 발생할 수 있는 관리정보 불일치 문제를 해결하고자 하였다. 그러나 이러한 실행방식의 경우에는 응용프로그램들의 실행에 따른 서버의 과부하 문제는 해결할 수 있으나, 공유객체가 서버에 집중되어 있으므로 공유객체들의 액세스에 따른 서버 과부하 문제는 해결할 수 없다. 복합실행방식은 웹(web)상에서의 공동저작을 고려한 것으로 웹서버로부터 각 클라이언트로 응용프로그램이 다운로드(download)됨에 따른 서버의 과부하는 무시하고 있다. 또한 세션 책임자가 한 서버에 존재하므로 각 저작자 시스템들의 관리정보 액세스와 응용프로그램들의 통신이 서버에 집중되는 문제가 있다. 응용프로그램들을 분산시키는 것만으로는 집중형 구조의 문제를 해결할 수 없으므로 공유객체들의 액세스와 각 응용프로그램들간의 통신이 서버에 집중되어 발생할 수 있는 과부하 문제도 고려해야 한다. 또한 공동저작에 사용되는 공유객체들이 각 저작자 시스템에 자연적으로 분산(naturally distributed)되어 있는 분산환경에서는 서버에 공유객체를 임의로 집중시키는 방식은 바람직하지 않을 수 있다.

공동저작 시스템은 크게 공동저작을 위한 공동저작 응용프로그램부와 공동저작 관리를 위한 공동저작 관리시스템부로 나뉘어 질 수 있다. 공동저작 응용프로그램부는 각 저작자들로 하여금 단일미디어 및 멀티미디어 객체 그리고 공동저작 문서를 편리하게 저작할 수 있도록 하는 저작도구(authoring tool)들과 저작자들 간의 원활한 의사소통을 위한 채팅(chatting), 화상회의(audio/video conference), 화이트 보드(white board), 전자게시판(bulletin board), 그리고 전자메일(e-mail) 등과 같은 응용프로그램들로 구성될 수 있다. 공동저작 관리 시스템부는 각 저작자 시스템에 분산되어 있는 공유객체들을 관리하여 공동저작에 사용할 수 있도록 해주는 관리 프로그램들과 관련정보들로 구성된다. 공동저작 관리 시스템은 공동저작에 참여하고 있는 저작자들의 정보와 각 저작자 시스템에 분산된 공유객체들의 대한 정보들을 관리하여 각 저작자 시스

템에 존재하는 공동저작 응용프로그램들이 상호 협력할 수 있도록 하는 환경을 제공한다. 따라서 효율적인 공동저작을 위해서는 공동저작 관리 시스템의 전체구조와 관리 시스템이 가져야하는 기능들에 대한 연구가 필요하다.

공동저작 관리 시스템은 효율적인 공동저작을 위해 분산된 공유객체들을 저작자 및 공유객체 위치에 상관없이 액세스하고, 공유객체들에 대한 권한 요청을 충돌없이 처리해주기 위한 공유객체 관리 기능, 동적인 저작자 그룹인 세션을 관리하여 세션 구성원들로 하여금 상호협력하여 작업할 수 있도록 지원하는 세션 관리 기능, 그리고 다른 저작자들의 작업위치 및 상태를 알 수 있도록 하여 동시적 공동저작을 가능하게 하는 **텔레포인팅** 기능을 지원한다. 여기서 세션이란 공동저작 참여자들의 필요에 따라 생성, 초기화 또는 소멸될 수 있는 동적인 사용자 그룹을 말하며, 텔레포인팅이란 지역 포인터의 위치 및 움직임과 같은 이벤트를 원격지 시스템(remote system)에 동기적으로 표현하는 기법을 말한다. 본 논문에서는 분산 멀티미디어 환경에서의 효율적인 공동저작 지원을 위한 공동저작 관리 시스템인 **CW-MAN**(Cooperative-Work Management System)을 제시한다. CW-MAN은 집중형과 분산형 구조의 단점을 보완하기 위해, 공동저작에 사용되는 응용프로그램과 공유객체들은 각 저작자 시스템에 분산시키고, 관리정보는 중앙의 관리 시스템에 집중시킨 혼합형 구조이다. 그리고 각 세션 내에서 선거(election)를 통해 세션참가자 중의 하나를 세션책임자로 정하여 세션참가자들의 관리정보 및 공유객체에 대한 액세스를 세션책임자를 통해 서버로 요청하도록 함으로써 중앙의 관리 시스템에 집중될 수 있는 관리정보 액세스와 응용프로그램들간의 통신을 각 세션 책임자 시스템으로 분산시켰다. 또한 CW-MAN은 효율적인 공동저작 지원을 위해 공유객체 관리 기능, 세션 관리 기능, 그리고 텔레포인팅 기능을 제공한다. 본 논문은 공동저작 관리를 위한 공동저작 관리 시스템의 전체구조와 지원기능들에 대한 개념적인 연구이므로 공동저작 관리 시스템의 성능평가 부분은 여기서 고려하지 않는다. 2장에서 CW-MAN의 전체 구조를 설명하고 3장에서 세션 관리 및 플로어 제어(floor control)에 대해 설명한다. 4장에서는 텔레포인팅 기법을 설명하고 5장에서는 공유객체 관리 기법을 설명하며, 마지막 6장에서 결론을 맺는다.

2. 시스템 구조

앞장에서 설명한 바와 같이 공동저작 관리 시스템은 관리정보와 공유객체들의 위치에 따라 집중형(centralized system)과 분산형(distributed system)으로 구분할 수 있다. 이 두 가지 방식의 특징과 장단점은 <표 1>과 같다.

<표 1> 집중형과 분산형 구조의 비교

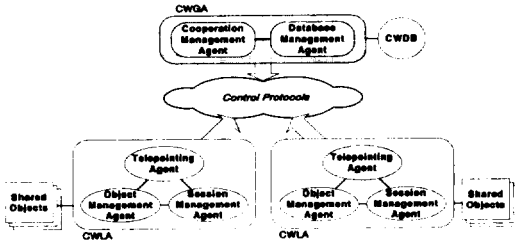
구분	특징	장점	단점
집중형	관리정보와 공유객체가 서버에 집중.	관리정보와 공유객체가 서버에 집중되어 관리가 용이.	공유객체가 서버에 집중되어 과부하 현상이 발생할 수 있음.
분산형	관리정보와 공유객체가 각 저작자시스템에 분산.	공유객체 액세스에 따른 과부하가 비교적 적음.	관리정보의 분산으로 인한 관리정보 불일치 현상이 발생할 수 있음.

본 논문에서 제시하는 공동저작 관리 시스템인 CW-MAN은 집중형과 분산형 구조의 단점을 보완하기 위해 공유객체들을 각 참여자 시스템으로 분산시키고, 관리정보를 중앙의 관리자 시스템으로 집중시킨 혼합형 구조(hybrid architecture)를 갖는다. 혼합형 공동저작 관리 시스템의 장점은 다음과 같다.

- 공유객체들이 각 저작자가 작업하는 지역 시스템(local system, 이하 저작자 시스템이라 칭함)에 분산되어 있으므로 집중형일 경우 발생할 수 있는 서버의 과부하가 각 저작자 시스템으로 분산된다.
- 공동저작 관리를 위한 관리정보가 중앙의 관리자 시스템에 집중되어 있으므로 분산형일 경우 있을 수 있는 관리정보 불일치 문제가 방지된다.
- 공유객체들이 각 저작자 시스템에 분산되어 있으므로 저작자 시스템 중 오류가 발생한 시스템이 있더라도 집중형의 경우보다 피해가 적다.

(그림 1)은 CW-MAN의 전체 구조를 나타내고 있다. CW-MAN은 관리자 시스템에 존재하는 CWGA(Cooperative-Work Global Agent)와 각 저작자 시스템에 존재하는 CWLA(Cooperative-Work Local Agent)

로 구성된다. CWGA는 관리자 시스템에 존재하는 에이전트로써 공동저작 관리 정보와 공유객체 정보를 관리한다. CWLA는 각 저작자 시스템에 존재하는 에이전트로써 CWGA와 상호협력을 통해 공동저작을 지원하며, 저작자 시스템에 존재하는 공유객체 관리를 담당한다.



(그림 1) CW-MAN의 전체 구조

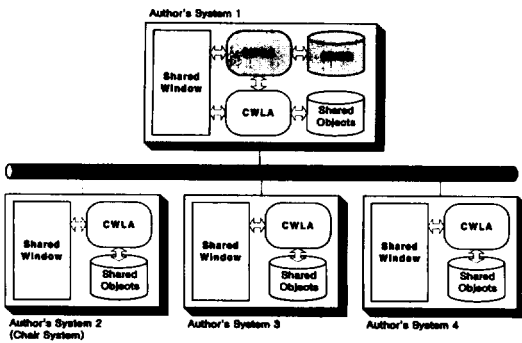
- CWGA** : CWGA는 관리자 시스템에서 동작하는 에이전트로써 DBMA(Database Management Agent)와 CMA(Cooperation Management Agent)로 구성된다. DBMA는 공동저작 참여자 및 공유객체들의 관리 정보와 각 세션의 작업 히스토리(history)와 같은 공동저작 관리를 위한 정보를 보관하는 데이터베이스인 CWDB(Cooperative-Work DataBase)를 관리하고, CWLA나 CMA로부터의 CWDB 액세스 요청을 처리하는 기능을 담당한다. 관리정보의 일관성 유지를 위해 CWDB에 대한 액세스는 항상 DBMA를 통해서만 이루어진다. CMA는 CWLA의 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. 또한 CMA는 **세션 책임자**(session chair)가 위치한 시스템의 갑작스런 오류나 기존 세션의 재시작 등의 경우에 대비하여 세션의 작업 히스토리를 각 세션 책임자로부터 전송받아 DBMA를 통해 CWDB에 저장하는 기능도 제공한다. 여기서 세션 책임자란 세션 참가자 중 한사람으로써 세션의 관리 정책을 결정하는 관리자이다.
- CWLA** : CWLA는 각 저작자 시스템에서 동작하는 에이전트로써 SMA(Session Management

Agent), OMA(Shared Object Management Agent), 그리고 TPA(Telepointing Agent)로 구성된다. SMA는 각 CWLA의 SMA와 상호협력을 통해 세션 관리 기능을 수행한다. SMA는 세션 책임자 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 텔리포인터(telepointer)에 대한 입력권한을 관리하는 기능을 담당하며, 세션 참여자 시스템에서 동작할 경우, 지역 시스템의 공유객체를 관리하고 저작자, 또는 응용프로그램으로부터의 공유객체 액세스 요청을 받아 CWGA를 통해 처리하는 기능을 담당한다. 또한 세션 책임자 시스템에 오류가 발생하거나 기존 세션을 다시 시작하는 경우를 대비해 세션의 작업 히스토리를 CWGA로 전송하여 CWDB에 저장되도록 한다. OMA는 지역 시스템에 있는 공유객체들을 관리하고, 응용프로그램과 텔리포인터 에이전트로부터 공유객체들에 대한 액세스 요청을 받아 공유객체들을 그 위치에 상관없이 액세스 할 수 있도록 한다. TPA는 공유 윈도우 안에서의 실시간 텔리포인터 기능을 지원하는 에이전트로서 세션 책임자 시스템에서 동작할 경우에는 텔리포인터 시 텔리포인터 입력 권한을 가진 저작자의 입력을 모든 세션 참여자 시스템으로 전송하는 코디네이터(coordinator) 역할을 담당한다.

CWGA와 CWLA간, 또는 CWLA와 CWLA간에는 세션 관리, 공유객체 관리, 세션당 하나씩 존재하는 텔리포인터의 권한 관리, 그리고 텔리포인터를 위한 제어 프로토콜(control protocol)이 존재한다. 제어 프로토콜은 세션의 생성, 초기화, 또는 소멸을 위한 세션제어 프로토콜(session control protocol), 각 저작자 시스템에 분산된 공유객체를 액세스하기 위한 공유객체 액세스 프로토콜(shared object access protocol), 그리고 세션 단위로 발생하는 텔리포인터(telepointing protocol)을 위한 텔리포인터 프로토콜로 구성된다.

(그림 2)는 앞서 설명한 CW-MAN의 물리적 구조의 한 예이다. 한 세션이 생성되어 있고 네 명의 세션 참가자가 공동저작 중임을 나타내고 있다. 시스템 공

유객체 관리 및 공동작업 관리 정보를 갖는 CWGA는 저작자 시스템 1에 위치해 있고, 시스템 2가 세션 책임자이며, 시스템 3과 4는 세션 참여자 시스템이다. 여기서 CWGA는 저작자 시스템 1에 위치해 있는데, 이는 CWGA가 독립된 시스템에 존재할 수도 있고, 공동 저작에 참여하고 있는 저작자 시스템들 중의 하나에 존재할 수도 있음을 나타낸다. 공동저작 참가자는 CWGA가 존재하는 시스템의 위치만 알고 있으면 제어 프로토콜을 통해 기존 세션에 참가하거나, 저작에 필요한 공유객체를 액세스할 수 있다.



(그림 2) CW-MAN의 물리적 구조의 한 예

3. 세션 관리와 플로어 제어

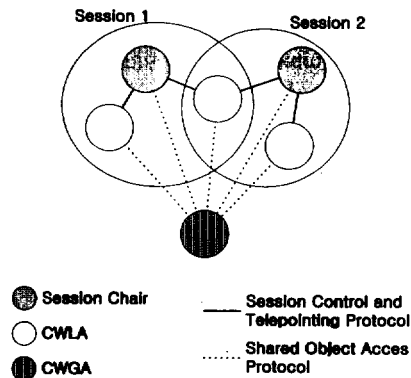
3.1 세션 관리

세션은 공동저작 참여자들의 필요에 따라 생성, 초기화, 또는 소멸될 수 있는 동적인 저작자 그룹이다. CW-MAN은 저작자들과 저작자 그룹들간의 상호동작 및 협동이 원활히 이루어 질 수 있도록 하는 세션 관리 기능을 지원한다. 세션은 하나의 공유 윈도우(shared window)를 가지며, 세션 참가자들은 공유 윈도우를 통해 다른 참가자들의 동작을 알 수 있고, 실시간 텔레포인팅을 통하여 동기적 공동저작을 수행할 수 있다. 각 공유 윈도우에 한 개씩 존재하는 텔레포인터는 세션 참가자들이 공유하는 중요한 자원중의 하나이다. CW-MAN은 텔레포인터에 대한 입력 권한을 한 순간에 한 저작자만 갖도록 해야 하므로 텔레포인터에 대한 플로어(floor) 제어 기능을 수행한다. 여기서 플로어란 공유자원에 대한 경쟁상태(race condition)를 최소화하고, 그 배타적인 사용을 보장하기 위해 공동저작 참여자에게 부여되는 임시 권한을 말한다[9].

각 세션에는 세션 책임자가 존재한다. 2장에서 설명한 바와 같이, 세션의 책임자는 세션 참가자 중 한 사람으로써 세션의 관리 정책을 결정하는 관리자이다. 세션 책임자는 다음과 같은 방법들 중의 한가지에 의해 결정된다.

- 세션을 처음 생성한 참가자가 책임자가 된다.
- 처음 세션을 초기화 한 참가자가 책임자가 된다.
- 선거(election)를 통해 선출된 참가자가 책임자가 된다.

세션이 처음 생성되었거나, 또는 초기화된 경우, 세션을 생성하거나 초기화된 참가자가 세션의 책임자가 된다. 세션 책임자 시스템의 CWLA는 세션의 작업 위치, 참가자 목록(list), 또는 작업 히스토리 등의 세션 작업 정보가 변경될 때마다 이를 CWGA로 전송하여 CWDB에 저장되도록 한다. 또한 세션 책임자는 작업의 진행 중에 세션 참가자들의 선출에 의해 변경될 수 있다. 이 경우 새로운 세션 책임자의 시스템에서 동작 중인 CWLA는 CWGA로부터 CWDB에 저장되어 있는 세션 정보와 히스토리를 전송 받는다. 세션 책임자 시스템에 오류가 발생했을 경우에는 세션 참가자중 한 사람이 다시 세션을 초기화할 수 있으며, CWGA로부터 CWDB에 저장되어 있는 세션 정보와 히스토리를 전송 받아 세션을 이전 상태로 복원하고 작업을 다시 시작할 수 있다.



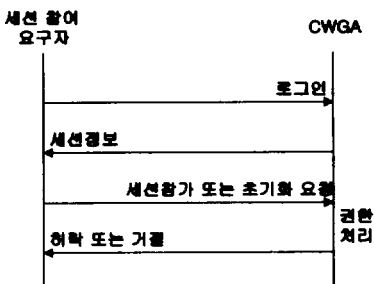
(그림 3) 세션 관리 구조

(그림 3)은 세션 관리 구조를 나타내고 있다. 세션 책임자 시스템의 CWLA는 텔레포인터에 대한 입력권

한을 가진 저작자의 입력이 세션에 참가 중인 사용자들의 시스템에 전송되어 동기적으로 적용되도록 하는 코디네이터(coordinator) 기능을 수행한다. 텔리포인팅은 지역시스템의 위치, 움직임, 이벤트 등을 원격지 시스템에 동기적으로 적용해야 하므로 그 프로토콜에 따른 과부하가 생길 수 있다. 본 논문에서 제시하는 세션 관리 구조는 각 세션의 책임자 시스템이 텔리포인팅 이벤트의 정렬 및 동기화를 담당하는 코디네이터 역할을 하도록 함으로써 텔리포인팅 프로토콜로 인한 부하를 각 세션의 책임자 시스템들로 분산시키는 장점이 있다.

세션은 그 작업 성격에 따라서 단기간에 끝이 나는 **임시 세션**과 장기적으로 지속해야 하는 **지속적 세션**으로 나눌 수 있다. 임시 세션은 저작자들의 필요에 의해 일시적으로 생성되었다가 종료함과 동시에 CWDB에 기록되어 있던 세션 히스토리 정보도 함께 제거되므로 한번 세션을 종료하면 다시는 세션을 재개할 수 없다. 그러나 지속적 세션은 세션 작업 정보 및 히스토리가 계속 CWDB에 기록되어 보관되므로 세션 참가자들 중의 한 사람이 세션을 초기화하고 작업을 재개할 수 있다. CWDB에는 세션의 작업 히스토리가 저장되므로 지속적 세션을 재개하였을 경우 세션 작업 종료 시의 상태로 세션을 초기화할 수 있다. 세션 책임자는 세션 참가 제한 정도에 따라 다음과 같은 세 가지 모드(mode)로 세션을 관리 할 수 있다[3].

- **열림(open) 모드** : 공동저작 참여자라면 누구나 참가할 수 있다.
- **참여가능(joinable) 모드** : 세션 책임자의 허락을 받아 참가할 수 있다.
- **닫힘(closed) 모드** : 미리 지정된 특정 저작자만이 참가할 수 있다.



(그림 4) 저작자의 세션 참가 과정

(그림 4)는 기존의 세션에 참가하거나 새로운 세션을 초기화 하고자 하는 저작자의 공동저작 참가 과정을 나타내고 있다. 먼저 저작자는 공동저작에 참가하기 위해 CWGA에게 로그인(login)을 요청한다. 저작자로부터 로그인 요청을 받은 CWGA는 모든 임시 세션과 지속적 세션들의 정보를 알려준다. CWGA로부터 세션들의 정보를 수신한 저작자는 기존 세션의 참가, 또는 새로운 세션의 초기화 등과 같은 요청을 CWGA에게 송신하며, 이때 CWGA는 저작자의 권한 레벨과 세션의 관리 모드에 따라서 저작자의 요청을 허락, 또는 거절할 수 있다. CWGA는 요청을 허락한 저작자에 한하여 참가하고자 하는 세션의 책임자 시스템과 저작자를 연결하여 기존의 세션에 참가하도록 하거나, CWDB에 세션의 정보를 생성하여 새로운 세션을 초기화 할 수 있도록 한다. 이러한 세션 참가 과정을 통해 기존의 세션에 참가하거나, 새로운 세션을 초기화 하고자 하는 저작자는 현재 진행중인 세션들에 대한 별도의 정보가 없어도 CWGA만을 통해 세션에 참가할 수 있다.

3.2 플로어(floor) 제어

세션 관리가 참여자 그룹을 관리하고 저작자간의 협력을 지원하기 위한 것이라면 플로어 제어는 하나의 세션 내에서 저작자들의 공유자원에 대한 권한을 관리하기 위한 것이다. 텔리포인터는 세션 참가자들이 공유하는 중요한 공유자원 중 하나로 세션 당 하나씩만 존재하기 때문에 공동저작 관리 시스템은 텔리포인터에 대한 입력 권한을 한 순간에 한 저작자만 가지도록 보장해야 하며, 세션 참가자들의 텔리포인터 입력권한에 대한 경쟁상태(race condition)를 최소화하기 위한 플로어(floor) 제어 기능을 수행해야 한다. 앞 절에서 설명한 바와 같이, 플로어란 공유자원에 대한 경쟁상태를 최소화하고 그 배타적인 사용을 보장하기 위해 공동저작 참여자에게 부여되는 임시 권한이다. 플로어 제어를 위해서는 플로어를 세션 참가자들에게 어떻게 할당할 것인가를 결정하는 **플로어 할당 정책(floor allocation policy)**과 이를 수행하기 위한 구체적인 방법인 **플로어 제어 메커니즘(floor control mechanism)**이 필요하다. 다음은 플로어 제어 정책들의 예이다.

- 세션 책임자가 플로어 권한자를 지정한다.
- 참가자들간의 선거에 의해 플로어 권한자를 선출한다.

- 미리 지정된 방법에 의해 플로어 권한자를 결정한다.
- 시간적 제한을 두어 플로어 권한을 일정 시간 뒤에 다음 요구자에게 넘긴다.

공동저작 시스템에서는 이러한 플로어 제어 정책이 모두 적용 가능해야 하며, 플로어 제어 정책은 작업의 종류와 상태에 따라서 변경될 수 있어야 한다. 세션 책임자는 세션의 작업 성격에 맞게 정책을 바꾸어 설정할 수 있다. 플로어 제어 매커니즘은 각 정책을 수행하는데 필요한 구체적인 방법들로 요구(request), 할당(assign), 반환(release), 취소(revoke) 등으로 나눌 수 있다[9].

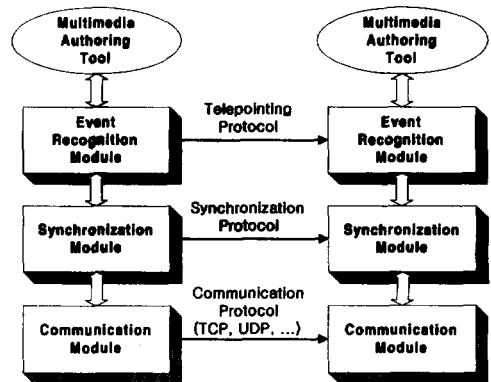
4. 텔리포인팅

1장에서 설명한 바와 같이, 텔리포인팅은 지역 포인터(local pointer)의 위치 및 움직임과 같은 이벤트를 원격지 시스템(remote system)에 동기적으로 표현하는 기법이다. 이러한 텔리포인팅의 기능은 다음과 같이 세 가지로 나누어 진다.

- **의사소통 기능** : 텔리포인팅은 저작자들 간의 의사교환 시 설명하고자 하는 내용을 원격지에서 직접 포인팅 할 수 있도록 함으로써 효율적인 의사소통을 가능하게 한다. 텔리포인터에 대한 입력 권한을 가진 저작자가 설명하고자 하는 내용을 텔리포인터로 포인팅하면 다른 저작자는 공유 윈도우를 통해 포인팅된 내용을 보며 의사교환을 할 수 있다.
- **동기적 공동저작 기능** : 저작자가 텔리포인터를 이용하여 해당 객체를 이동, 삭제 및 변경하는 등의 작업을 행하면 그 상황이 다른 저작자의 시스템에 텔리포인팅 이벤트로 변환되어 전송된다. 이러한 텔리포인팅 이벤트는 세션 참가자들의 시스템에 동기적으로 적용되어 동기적인 공동저작을 수행할 수 있도록 한다.
- **동기적 프리젠테이션 기능** : 공동저작 시 저작자는 텔리포인터를 이용하여 동영상이나 사운드 등과 같은 멀티미디어 저작물의 재생, 일시정지, 또는 전 페이지로 이동 등의 이벤트를 발생시킬 수 있다.

CW-MAN에서의 텔리포인팅 기능은 CWLA를 구성하는 에이전트 중 하나인 텔리포인팅 에이전트를 통해 제공된다. 텔리포인팅 에이전트는 공유 윈도우에서 발생한 지역 포인팅 이벤트를 인식하여 텔리포인팅 이벤트를 발생시키기 위한 **인식(recognition)**과 원격지 시스템에서 발생한 텔리포인팅 이벤트를 공유 윈도우에 알리는 **통보(notification)** 기능을 제공한다. 또한 동기적 공동저작 및 동기적 프리젠테이션을 위해 텔리포인팅 이벤트를 발생 순서대로 정렬(**ordering**)하고 동기화(**synchronization**)하는 기능을 수행한다. 이를 위해 세션 책임자 시스템의 텔리포인팅 에이전트가 텔리포인팅 이벤트를 조정(**coordinating**)하는 **조정 에이전트(coordination agent)**의 역할을 담당한다. 텔리포인팅 에이전트는 **이벤트 인식 모듈(event recognition module)**, **동기화 모듈(synchronization module)**, 그리고 **통신 모듈(communication module)**로 구성된다. 각 모듈들의 기능은 다음과 같다.

- **이벤트 인식 모듈** : 공유 윈도우에서 발생하는 지역 포인팅 이벤트를 인식하여 텔리포인팅 이벤트를 발생시키고, 원격지 시스템으로부터의 텔리포인팅 이벤트를 공유 윈도우에 알린다.
- **동기화 모듈** : 텔리포인팅 이벤트는 각 저작자의 공유 윈도우에 동기적 뷰(view)를 제공하기 위해 발생 순서대로 정렬되어야 하며, 동기화 되어야 한다. 동기화 모듈은 텔리포인팅 이벤트의 정렬(**ordering**) 및 동기화 기능을 수행한다.
- **통신 모듈** : 다른 텔리포인팅 에이전트와의 통신 기능을 수행한다.

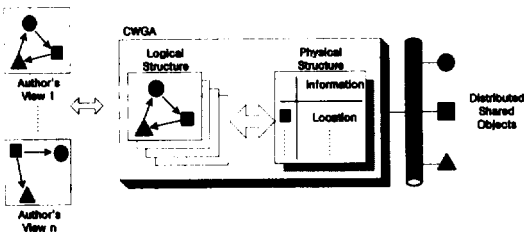


(그림 5) 텔리포인팅 구조

(그림 5)는 텔리포인팅 구조를 나타낸 것이다. 각 저작자의 지역 시스템 공동저작 도구에서 발생한 지역 이벤트는 텔리포인팅 에이전트의 이벤트 인식 모듈에 의해 인식된다. 이벤트 인식 모듈을 통해 인식된 이벤트는 텔리포인팅 이벤트로 변환되어 이벤트 정렬 및 동기화를 위해 동기화 모듈로 전달된다. 텔리포인팅 이벤트를 전달받은 동기화 모듈은 통신 모듈을 통해 텔리포인팅 조정 에이전트 및 세션 내의 다른 텔리포인팅 에이전트들과 통신하며, 이벤트를 발생순서대로 정렬하고 동기화한다. 동기화 모듈을 통해 정렬되고 동기화된 텔리포인팅 이벤트는 이벤트 인식 모듈을 거쳐 공동저작 도구로 전달되며, 공동저작 도구는 이벤트 인식 모듈로부터 텔리포인팅 이벤트를 전달받아 CWLA의 공유객체 관리 에이전트에게 텔리포인팅 이벤트 표현을 위한 공유객체를 요청한 후 공동저작 윈도우 내에 텔리포인팅된 내용을 표현한다.

5. 공유객체 관리

CW-MAN은 저작자나 응용프로그램이 분산되어 있는 공유객체를 작업 위치나 공유객체의 위치에 상관없이 액세스 할 수 있도록 하는 위치 투명성(location transparency)과 공유객체를 액세스하는 과정에 대해 알지 못해도 액세스할 수 있도록 하는 액세스 투명성(access transparency)을 제공한다. CW-MAN은 멀티미디어 문서 구조에 따른 공유객체들의 논리적 구조(logical structure)와 실제 분산되어 있는 객체들의 물리적 위치 정보에 따른 물리적 구조(physical structure)를 분리하여 관리함으로써 이러한 위치 투명성과 액세스 투명성을 제공한다.

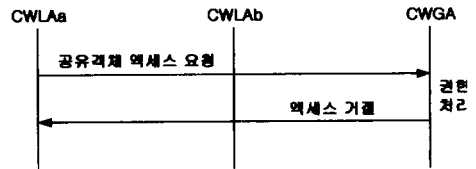


(그림 6) 공유객체의 논리적 구조와 물리적 구조의 매핑

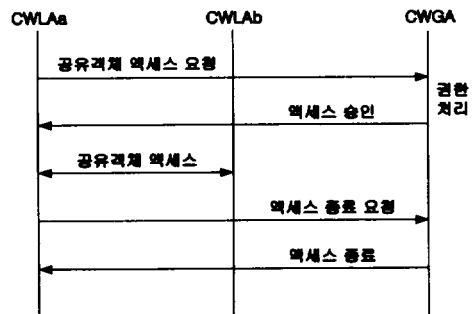
(그림 6)은 공유객체의 논리적 구조와 물리적 구조를 분리하여 관리하고, 공유객체 액세스 시에 CWGA

에서 논리적 구조 정보를 물리적 구조 정보로 매핑(mapping)하는 매핑 구조를 나타내고 있다. 멀티미디어 문서는 다양한 형태의 미디어 객체들이 논리적으로 연결된 구조를 가진다. 공유객체의 논리적 구조 정보는 멀티미디어 문서에 어떠한 객체들이 어떤 구조로 구성되어 있는가에 대한 정보를 말한다. 물리적 구조 정보는 저작자들의 시스템에 분산된 공유객체들의 망 주소와 저작자 시스템 내에서의 위치 정보를 말한다. 저작자가 멀티미디어 문서에 포함되어 있는 공유객체들에 대한 액세스 요청을 하면, CWLA는 이를 CWGA로 전송한다. CWGA는 공유객체의 논리적 구조 정보를 물리적 구조로 매핑하여 물리적 위치 정보를 얻어낸다. 이렇게 함으로써 얻어지는 장점은 다음과 같다.

- 저작자의 작업 위치나 공유객체의 물리적 위치에 상관없이 공유객체를 액세스 할 수 있다.
- 공유객체들의 물리적 위치를 작업 중에 변경해야 할 경우 CWDB에 있는 공유객체의 등록 정보만 수정하면 되므로 그 변경이 용이하다.



(a) 액세스 요청이 거절되는 경우



(b) 액세스 요청이 승인되는 경우

(그림 7) 공유객체 액세스 추적도

(그림 7)은 공유객체 액세스 과정을 나타내고 있다. CWLAa와 CWLab는 각각 공유 객체 액세스를 요청한

시스템과 공유객체가 실제로 위치한 시스템에서 동작중인 CWLA이다. 저작자나 응용프로그램으로부터의 공유객체 액세스 요청이 발생하면 CWLAa는 이를 CWGA로 전송한다. CWGA는 공유객체의 현재 사용 상태와, 액세스 요청을 한 저작자가 해당 객체에 대해 액세스 권한이 있는지를 검사하고, CWDB에 저장되어있는 공유객체 등록정보를 참조하여 CWLab와 CWLAa를 연결한다. 실제 공유객체의 액세스는 공유객체를 요청한 CWLA와 실제 공유객체가 위치한 시스템의 CWLA간에 연결이 이루어지고 난 후에 두 CWLA간에서 행해지며, CWGA는 관여하지 않는다. 공유객체 액세스가 완료되고 나면, CWLAa는 액세스 종료료를 CWGA에게 요청하여 공유객체를 반환(release)하게 된다.

6. 결론 및 향후과제

본 논문은 분산환경에서의 효율적인 멀티미디어 공동저작을 위한 공동저작 관리 시스템인 CW-MAN을 제시했다. 이 CW-MAN은 분산형 시스템과 집중형 구조의 단점을 보완하기 위한 혼합형 구조를 가지고 있으며, 공유객체들을 저작자의 위치나 공유객체의 위치에 상관없이 액세스 할 수 있도록 하는 공유객체 관리 기능과 저작 그룹 구성원들 간의 상호동작 및 협동을 지원하기 위한 세션관리 기능을 지원한다. 또한 지역 포인터의 위치와 이벤트를 공유 윈도우에 동기적으로 표현함으로써 다른 저작자들의 작업위치와 상태를 알 수 있도록 하고, 동기적 공동저작을 가능하게 하는 텔레포인팅 기능을 지원한다.

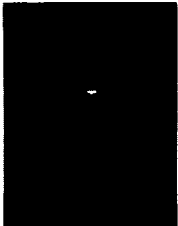
현재, 멀티미디어 공동저작에 사용되는 공유객체들을 일관된 방법으로 액세스 할 수 있도록 하는 공유객체 계층 구조 및 관리 기법에 대한 연구와 저작자들의 필요에 따라 생성, 또는 소멸될 수 있는 저작자 그룹인 세션 관리 기법에 관한 보다 세부적인 연구가 진행되고 있으며, 공동저작 시의 효율적인 의사소통, 동기적 공동저작, 그리고 동기적 프리젠테이션을 위한 텔레포인팅 구조 및 프로토콜에 관한 연구가 진행중이다.

참 고 문 헌

- [1] 성미영, "MissCW : 다중 사용자 동기적 공동 저작 시스템", 정보처리학회 논문지, pp.1697-1706, Dec., 1996.
- [2] 이만재, "멀티미디어 저작시스템", 정보과학회지, 제10권 제5호, Oct., 1992.
- [3] W. Reinhard *et al.*, "CSCW Tools: Concepts and Architecture," IEEE Computer, Vol.27, No.5, pp. 28-36, May, 1994.
- [4] R. Bentley *et al.*, "Architectural Support for Cooperative Multiuser Interfaces," IEEE Computer, Vol.27, No.5, pp.37-46, May, 1994.
- [5] T. Crowley *et al.*, "MMConf : An Infrastructure for Building Shared Multimedia Application," Proc. of CSCW '90, pp.329-342, 1990.
- [6] J. F. Patterson *et al.*, "Rendezvous : An Architecture for Synchronous Multiuser Applications," Proc. of CSCW '90, pp.317-328, 1990.
- [7] 김동성, 이광행, 오삼권, "분산형 멀티미디어 공동작업 관리 시스템의 설계", 한국정보처리학회 춘계 학술발표논문집, 제3권 제1호, pp.682-686, 1996.
- [8] 정의현, 박용진, "WWW상에서의 공동작업 시스템의 설계 및 구현", 정보과학회논문지(C), 제3권 제4호, pp.384-396, Aug., 1997.
- [9] 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.
- [10] K. Watabe *et al.*, "Distributed Multiparty Desktop Conferencing System : MERMAID," Proc. of CSCW '90, pp.27-38, 1990.
- [11] S. R. Ahuja *et al.*, "The Rapport Multimedia Conferencing System : A Software Overview," Proc. of 2nd IEEE Conference on Computer Workstations, pp.52-58, March, 1988.
- [12] 김명호, 이윤준, "멀티미디어 개념 및 응용", 홍릉과학출판사, Aug., 1997.
- [13] John A. Mariani and Tom Rodden, "Cooperative Information Sharing : Developing a Shared Object Service," The Computer Journal, Vol.39, No.6, pp. 455-470, Sep, 1996.
- [14] Ralf Steinmetz and Klara Nahrstedt, "Multimedia : Computing Communications and Application," Prentice Hall, 1995.
- [15] Jonathan Grudin, "Computer-Supported Cooper-

ative Work : History and Focus," IEEE Computer, Vol.27, No.5, pp.19-25, May, 1994.

- [16] 이광행, 김동성, 오삼권, "분산형 공동작업 시스템에서의 공유객체 관리", 한국통신학회 춘계학술 발표대회, pp.969-972, Nov., 1997.
- [17] Simon J. Gibbs, "Multimedia Programming," Addison-Wesley, 1995.
- [18] C. A. Ellis and S. J. Gibbs, "Concurrency Control in Groupware Systems," Proc. of Int'l Conf. on Management of Data, pp.399-407, 1989.



이 광 행

e-mail : khlee4@di.co.kr
 1996년 호서대학교 컴퓨터공학과 졸업(학사)
 1999년 호서대학교 컴퓨터공학과 (석사)
 1999년~현재 (주) 디아이 연구소 연구원

관심분야 : CSCW, Distributed systems, Multimedia Systems



전 재 우

e-mail : jean1225@nownuri.net
 1998년 호서대학교 컴퓨터공학과 졸업(학사)
 1998년~현재 호서대학교 컴퓨터공학과 석사과정
 관심분야 : CSCW, Distributed systems, Communication protocol



오 삼 권

e-mail : ohsk@dogsuri.hoseo.ac.kr
 1980년 한국항공대학교 항공전자공학과 졸업(학사)
 1980년~1984년 삼성전자 통신연구소
 1986년 University of South Florida, U.S.A.(석사)

1994년 Queen's University, Canada(박사)
 1994년~1995년 한국전자통신연구소
 1995년~현재 호서대학교 컴퓨터학부 조교수
 관심분야 : Real-time distributed systems, Communication protocol, Fault tolerance.