

# 멀티미디어 공동 작업 공간에서의 화이트보드 및 오류 제어

고응남\*, 황대준\*\*

\*천안대학교 정보통신학부,

\*\*성균관대학교 전기 전자 및 컴퓨터공학부

## A Whiteboard and An Error Control Based Multimedia Collaboration Work Space

Eung-Nam Ko\*, Dae-Joon Hwang\*\*

\*Division of Information & Communication Engineering, Cheonan University

\*\*School of Electrical, Electronics & Computer Engineering, Sungkyunkwan University

### 요 약

본 논문에서는 멀티미디어 공동 작업 환경에서의 화이트보드 및 오류 제어에 대해서 기술한다. 기존 화이트보드 모델이 가지고 있던 문제점들을 해결하기 위해 화이트보드 모델은 기존 공동 작업 공간이 기능적으로 분화되어 각기 다른 도구로써 사용되었던 점을 개선하여 공동 작업 공간에 대한 단일 사용자 인터페이스를 제공한다. 화이트보드 자체를 하나의 미디어로 취급하여 여기서 발생하는 오류 및 제어에 대하여 기술한다.

### 1. 서론

전통적인 협동 작업은 일정한 시간에 일정한 장소에서 함께 만나 자료를 보고 서로 의견을 말하면서 진행되었다. 오늘날에는 컴퓨터와 통신 기술의 발달로 시간과 공간의 제약 없이 공유된 가상 공간에서 상호 작용을 하면서 효율적인 작업을 하는 새로운 시스템이 대두되고 있다[1,2]. 최근에 있었던 공동작업(collaborative work) 분야의 많은 연구와 발전에 힘입어 의료 및 교육을 포함하는 다양한 분야에서 컴퓨터를 이용하는 공동 작업에 대한 요구가 날로 커지고 있다. 공동작업을 위해 보편적으로 이용되고 있는 지원도구는 화이트보드, 화상회의, 채팅 등이다[3].

이러한 현재의 방향에도 불구하고 화이트보드 등에서 상호 작용하는 충분한 신뢰성(reliability)을 항상 보장하는 것은 아니다. 따라서, 본 논문에서는 화이트보드 자체를 하나의 미디어로 취급하여 기존 화이트보드 모델이 가지고 있던 문제점들을 해결하기 위해 화이트보드 모델은 기존 공동 작업 공간이 기능적으로 분화되어 각기 다른 도구로써 사용되었던 점을 개선하여 공동 작업 공간에 대한 단일 사용자 인터페이스를 제공한다. 여기서 발생하는 오류 및 제어에 대하여 기술한다. 오류 발생시에 복제형 구조에 기반한 응용 공유의 동작 원리를 이용하여 동기화를 잘 수행할 수 있는 환경을 제공한다.

본 논문의 구성은 2장에서는 관련연구, 3장에서는

화이트보드 및 오류제어, 4장에서는 시스템 평가와 결론을 기술한다.

## 2. 관련연구

본 절에서는 기존의 멀티미디어 공동 작업 환경의 종류 및 구조에 대해서 기술한다.

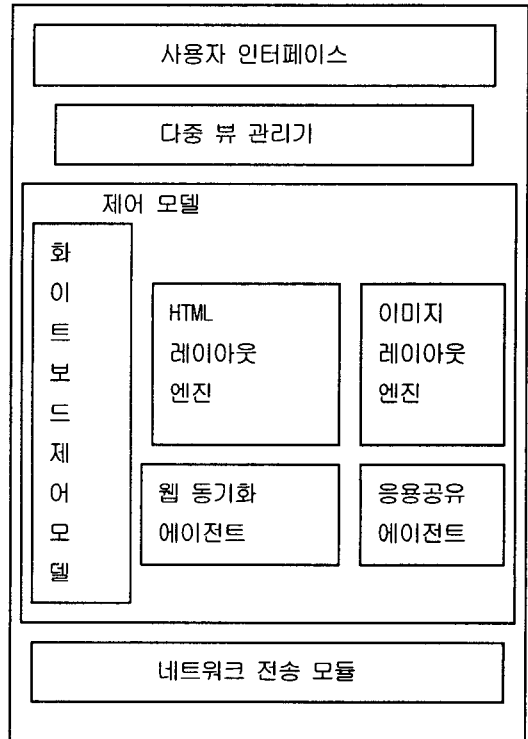
Shastra는 Purdue 대학교에서 개발된 UNIX를 기반으로 멀티미디어 협력 작업 설계 환경을 제공하는 시스템이다[4]. MERMAID는 일본의 Kansai C&C Lab과 NEC사에서 개발된 분산형 응용 공유 구조를 선택하면서, 공유 이벤트의 분배를 이벤트 발송 부분에서 처리함으로써 다양한 응용의 지원을 고려하는 시스템이다[5]. MMConf는 미국의 캠브리지에서 개발된 분산형 응용 공유 구조를 선택하였으며, X-윈도우즈를 기반으로 설계되어 있다[6]. CECEd는 SRI international에서 개발된 중앙 집중형 구조와 복제형 구조의 혼합 구조를 지원하며, 화면 공유 개념을 확장하였다[7]. 기존의 멀티미디어 공동작업 환경의 구조는 응용 구조에 따라 집중형(Centralized), 분산형(Distributed), 복제형(Replicated)으로 구분할 수 있다. 집중형 구조는 모든 구성 요소가 하나의 워크스테이션에서만 실행되는 구조이다. 분산형 구조는 구성 요소가 분산되어 있으며, 이들은 여러 워크스테이션에 걸쳐서 존재하는 구조이다. 복제형 구조는 분산형 구조의 변형된 형태로서, 대응하는 구성 요소가 아주 동일하거나 복제된다. 기존 멀티미디어 공동 작업 공간의 복제형 구조에서는 화이트보드를 위하여 응용 공유의 동작 원리를 이용하지만 오류 처리 및 제어에 대한 연구는 미흡한 실정이다.

## 3. 화이트보드 및 오류 제어

본 절에서는 기존 화이트보드 모델이 가지고 있던 문제점들을 해결하기 위해 다양한 멀티미디어 공동 작업 공간에 대한 일괄된 인터페이스를 제공하는 화이트보드 모델 및 여기서 발생하는 오류 및 제어에 대하여 기술한다.

### 3.1 화이트보드 모델

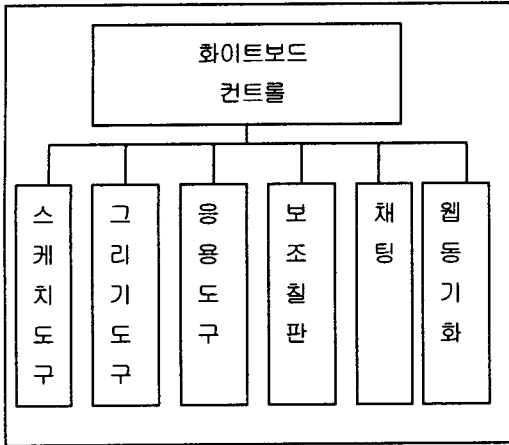
화이트보드 모델의 구성은 [그림 1]과 같다.



[그림 1] 화이트보드 모델의 구성

제안하는 화이트보드 모델은 기존 공동 작업 공간이 기능적으로 분화되어 각기 다른 도구로써 사용되었던 점을 개선하여 공동 작업 공간에 대한 단일 사용자 인터페이스를 제공한다. 이를 위해 사용자 인터페이스를 통한 각 기능 모듈별 인터페이스에 접근가능하며 현재 동작 중인 화이트보드 구성 요소에 대한 정보를 표시할 수도 있다.

사용자 인터페이스의 각 요소는 [그림 2]와 같다. 사용자는 해당 도구에 대한 조작이 허용되는 한 언제라도 필요로 하는 도구를 선택할 수 있으며 사용자 인터페이스는 이를 받아들여 선택된 도구에 해당하는 뷰를 생성하거나 조작할 수 있도록 활성화시킨다.



[그림 2] 사용자 인터페이스

다양한 형태의 공유된 공유 객체들은 그 속성에 따라 다르게 윈도우 창에 출력된다. 이러한 공유 객체의 출력은 다중화된 뷰의 개념을 적용하여 수행되는데 이렇게 서로 다른 속성을 가지는 공유 객체의 출력을 위해 본 모델에서는 세분화된 레이아웃을 지원하는 다중 뷰 관리를 두고 있다. 다중 뷰 관리는 내부적으로 지원되는 여러 가지 레이아웃 엔진을 통해 표현하고자 하는 공유 객체의 정보를 분석하고 이를 그래픽 화면으로 출력하는 작업을 수행한다. 이는 다중 뷰 관리기가 각 뷰와 그에 속한 공유 객체들의 접근 권한을 비교하여 해당 객체에 대한 사용 권한이 있는 경우에 한해 처리하며, 원격지 사용자에게 의한 입력 이벤트가 전달될 경우에도 이를 분석하여 처리한다.

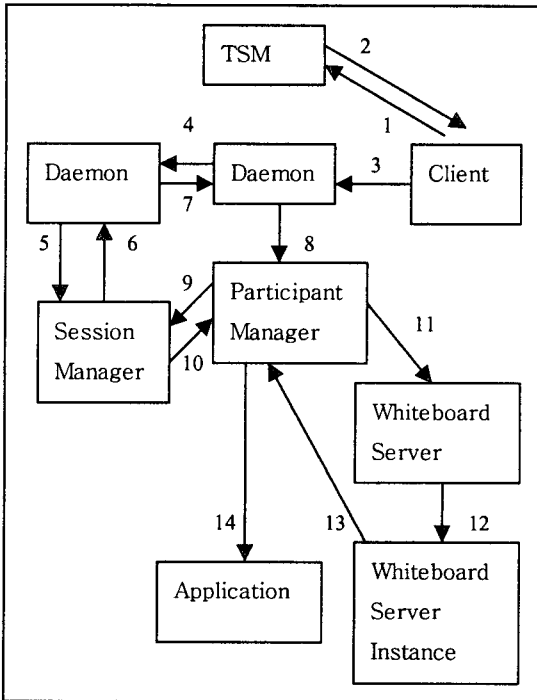
제어 모듈은 화이트보드 제어 모듈과 응용 공유, 웹 동기화 에이전트로 구성된다. 화이트보드 제어 모듈은 사용자가 화이트보드에 입력하는 화이트보드 제어 명령을 수행하고 화이트보드가 포함된 세션 관리 모듈과의 인터페이스를 이룬다. 세션 관리 모듈은 전체적인 세션 진행과 관련된 제어를 수행하게 되며 이를 바탕으로 화이트보드에 대한 제어를 요구하게 된다. 화이트보드 제어 모듈은 이러한 요구를 받아 다중 뷰 관리기와 각 에이전트 등에 필요한 메시지를 전달한다. 또한 화이트보드 내부의 수행 결과를 세션 관리 모듈 혹은 그 상위의 응용 프로그램에 공

지하는 기능을 포함한다. 화이트보드 제어 모듈은 이러한 기능 외에 전체 세션과 관련된 부가 정보를 저장하여 필요할 경우 이를 사용자에게 제공해 준다. 응용 공유 에이전트는 사용자가 응용공유 도구를 선택할 경우 동작하며 선택된 응용 프로그램에 대한 뷰를 생성하여 원격지 사용자들에게 전달해 주는 기능을 수행한다.

전송 관리 계층은 플랫폼의 각 계층의 구성 요소들이 사용하는 각종 전송 프로토콜(TCP, UDP, IP, Multicast, SMTP 등)을 제공하는 역할을 한다. 즉 모든 구성 요소 들은 이 계층을 통해서 원격지에 보내게 된다.

### 3.2 오류 제어

특정 세션이 진행 중에 발생할 수 있는 오류 처리 및 제어는 좀 더 복잡한 관리가 필요하다. [그림 3]은 이미 진행 중인 세션에 복구된 화이트보드가 참여하는 과정을 나타내고 있다. 복구된 화이트보드가 특정 세션에 참여하기 위해서는 어느 곳에서 어떤 세션이 진행 중인지에 대한 정보를 알아야 한다. 이를 지원하기 위한 서비스 객체로는 총괄 세션 관리자(TSM: Total Session Manager)가 있는데 총괄 세션 관리자는 모든 연결된 네트워크의 세션에 대한 정보와 복구된 화이트보드 등 세션에 참여하기 위한 접근 점에 대한 정보를 가지고 있다. 복구된 화이트보드는 클라이언트를 통하여 총괄 세션 관리자로부터 이러한 정보를 얻게 되고(1,2과정), 데몬의 복구된 화이트보드 인터페이스를 통하여 복구된 화이트보드의 참여 서비스를 요청(3과정)하게 된다. 이러한 서비스 요청을 받은 데몬은 원격지의 데몬에게 복구된 화이트보드의 참여 요청을 해당되는 세션 관리자에게 전달해 주기를 요청하고 세션 관리자의 응답을 요청의 역순으로 전달 받는다(4,5,6,7,8과정). 이 때 복구된 화이트보드의 참여가 허락되면 데몬은 초청 받았을 경우 세션에 참여하는 과정과 같은 순서에 의해 복구된 화이트보드를 위한 객체가 생성되고 복구된 화이트보드의 참여자는 세션에 참여하게 된다(8,9,10,11,12,13,14과정).



[그림 3] 오류 처리 및 제어

#### 4. 시스템 평가와 결론

제안된 시스템은 Visual C++로 설계 및 구축 가능하다. 기존 시스템과 기능 비교를 하면 <표 1>과 같다.

<표 1>

기능	Shas-tra	MERMAID	MMConf	CECED	제안 논문
화이트보드	있음	있음	있음	있음	있음
화이트보드에 서의 오류 제어	없음	없음	없음	없음	있음

본 논문에서는 상호 작용을 지원하는 멀티미디어 공동 작업 공간을 이용한 화이트보드 및 오류 제어에 대해서 제안하였다. 화이트보드 모델은 기존 공동 작업 공간이 기능적으로 분화되어 각기 다른 도구로써 사용되었던 점을 개선하여 공동 작업 공간에 대

한 단일 사용자 인터페이스를 제공 하였다. 즉, 기존 멀티미디어 공동 작업 공간의 복제형 구조에서는 화이트보드를 위하여 응용 공유의 동작 원리를 이용하지만 오류 처리 및 제어에 대한 연구는 미흡한 실정 이었지만 본 연구에서는 세션의 화이트보드 인스턴스가 비정상적으로 종료되는 경우에 화이트보드를 중단시키지 않고 미디어처럼 취급하여 재 활성화시키는 장점을 가지고 있다.

향후 연구 계획으로는 동기 비동기 모드를 겸용하는 웹 공유 시스템에서의 오류제어에 대한 연구 등이다.

#### [참고문헌]

- [1] 윤보열, 송승헌, 김응곤, “협동 설계 시스템을 위한 오브젝트 Picking과 Concurrency”, 한국 정보과학회 2001 봄 학술발표 논문집(B) 제 28권 1호, 한국 정보과학회, 2001년 4월 26일, pp.631 - 633.
- [2] F. Faure, C.Faisstnauer, G.Hesina, “Collaborative animation over the net”, IEEE 1999, pp.107-116.
- [3] 김문석, 성미영, “동기적 웹브라우저 공유를 지원하는 협동 작업 시스템”, 한국정보처리학회논문지 제8-B권 제 3호, 2001.6, pp.283-288.
- [4] A. Anupam and C.L.Bajai, “Collaborative Multimedia Scientific Design in Shastra”, Proceeding of the ACM Multimedia’93, Aug.1993, pp.447-456.
- [5] T. Ohmori and K. Watabe, Distributed Cooperative Control for Application Sharing Based on Multiparty and Multimedia Desktop Conferencing Systems:MERMAID, 4<sup>th</sup> IEEE ComSoc International Workshop on Multimedia Communications, April 1-4, 1992.
- [6] Torrence Crowley and Raymond Tomlinson, MMConf: An Infrastructure for Building Shared Multimedia Applications, CSCW ’90 Proceedings, October 1990.
- [7] Earl Craighill and Keith Skinner, CECED: A System For Informal Multimedia Collaboration, Proceedings ACM Multimedia ’93, August 1-6 1993.