

# 웹 기반 멀티미디어 공동 작업 환경에서의 오류 공유 시스템

고응남

천안대학교 정보통신학부

e-mail : [ssken@cheonan.ac.kr](mailto:ssken@cheonan.ac.kr)

## An Error Sharing System on a Web based Multimedia Collaboration Environment

Eung-Nam Ko

Division of Information & Communication, Cheonan University

### 요 약

본 논문에서는 웹 기반의 멀티미디어 공동 작업 환경에서의 오류 공유 시스템에 대하여 제안한다. 오류 공유 시스템은 윈도우 방식을 사용하고 응용 및 오류 공유 구조로는 분산 복제형 구조(CARV)이다. 분산 복제형 구조는 중앙 집중형 구조와 복제형 구조 두개의 장점을 취합한 형태이지만 복제형 구조처럼 완전한 대칭적인 구조를 가지고 있지는 않다. 감지된 오류는 웹 상의 오류 공유 시스템을 이용하여 신속히 전달하여서 오류를 공동으로 공유할 수 있게 하고 신속히 복구할 수 있도록 한다.

### 1. 서론

컴퓨터 성능의 급속한 발전과 고속 네트워크 기술의 발달로 인한 오늘날의 컴퓨팅 환경이 개방화됨에 따라 컴퓨터 지원 협동 작업(CSCW: Computer Supported Cooperative Work) 응용은 화상 회의 시스템, 상호 협력적 원격 교육, 원격 진료, 그룹 게임 분야 등으로 실생활의 거의 모든 분야로 확대되고 있다.[1,2,3]. 최근 들어 이러한 컴퓨터 지원 협동 작업 환경이 증가하고 있는데 반하여 이러한 시스템에서의 응용 공유, 접근 제어, 동시성 제어, 상호 인지 등 응용 오류에 대한 연구는 미흡한 실정이다[4,5]. 본 논문에서는 웹 기반 응용 공유 기능을 갖고 있는 멀티미디어 공동 작업 환경에서 오류 공유에 대하여 기술한다. 응용 공유는 멀티미디어 공동 작업 시스템에서 사용자들에게 공동 작업 환경을 제공하는 핵심 요소로써 단일 사용자용으로 설계되어서 네트워크를 인지하지 못하는 응용 프로그램을 네트워크에서 여러 명의 사용자

자가 공동 작업 환경으로 사용할 수 있도록 지원한다 [6]. 이러한 현재의 방향에도 불구하고 상호작용(interactive) 하는 멀티미디어 환경의 구성 요소에서는 그 시스템에서 계산될 수 있는 결함 허용 응용에서조차도 충분한 신뢰성(reliability)을 항상 보장하는 것은 아니다[7]. 따라서, 본 연구에서는 웹 기반 멀티미디어 공동작업 환경에서 응용 소프트웨어와 미디어 등의 프로세스(process) 요소들에 대한 오류를 미리 감지하여 알려주고 오류 공유 시스템을 이용하여 신속하고 정확하게 전달하여서 복구하는데 빠르게 대처할 수 있는 시스템을 제안한다.

본 논문의 구성은 2 에서 관련 연구를 기술하고, 3에서는 제안하는 웹 기반 멀티미디어 공동 작업 환경에서의 오류 공유 시스템에 대해서 기술하고, 4에서는 시스템 평가, 5에서는 결론을 기술한다.

### 2. 관련 연구

기존의 멀티미디어 공동 작업 환경의 대표적인 중

류는 다음과 같다[8,9,10,11]. 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. 이 시스템은 상호 작용 과정의 모든 동작을 중앙 세션 관리기를 통하여 하기 때문에 서버의 부담이 많아진다는 단점이 있다[8]. 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[9]. MMConf는 미국의 캠브리지에서 개발된 분산형 응용 공유 구조를 선택하였으며, X-윈도우즈를 기반으로 설계되어 있다[10]. CECED는 SRI international에서 개발된 중앙 집중형 구조와 복제형 구조의 혼합 구조를 지원하며, 화면 공유 개념을 확장하였다[11].

기존의 멀티미디어 공동작업 환경의 구조는 응용 구조에 따라 집중형(Centralized), 분산형(Distributed), 복제형(Replicated)으로 구분할 수 있다. 집중형 구조는 모드 구성 요소가 하나의 워크스테이션에서만 실행되는 구조이다. 이는 모든 요소가 하나의 워크스테이션에서만 실행되기 때문에 가장 간단한 방법이다. 분산형 구조는 구성 요소가 분산되어 있으며, 이들은 여러 워크스테이션에 걸쳐서 존재하는 구조이다. 복제형 구조는 분산형 구조의 변형된 형태로서, 대응하는 구성 요소가 아주 동일하거나 복제된다. 이는 각자의 워크스테이션에 실행에 필요한 모든 요소가 존재하는 구조이다.

### 3. 웹 기반 멀티미디어 공동 작업 환경에서의 오류 공유 시스템

본 논문에서는 웹 기반 멀티미디어 공동 작업 환경에서의 오류 공유 방식에 대하여 제안한다.

#### 3.1 웹 기반 멀티미디어 공동 작업 환경

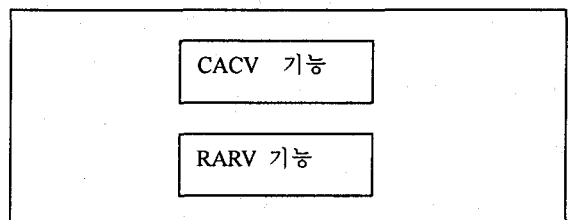
본 연구의 제안 모델에서는 IP 계층에서 호스트에 제공하는 멀티캐스트를 이용하였다. UDP/IP 브로드캐스팅도 다수의 호스트에 동시 전송이 가능하지만 호스트를 지정할 수 없어 그룹 전송을 하지 못하고 그룹의 가입과 탈퇴가 자유롭지 못하기 때문에 IP 멀티캐스트를 사용한다. 본 논문에서 제안하는 시스템은 여러 기능의 에이전트가 존재하며 원활한 오류 감지 및 복구 기법을 수행하는 멀티 에이전트 시스템이다. 시스템 계층으로는 윈도우 98/NT 등이 사용된다. CAI는 Common Application Interface의 약어이다. 오류 동기화가 가능하게 하기 위해서는 혹 서버, 응용 및 오류 공유 서버, 웹 URL 동기화 서버 등이 있어야 한다. 혹 서버에서는 윈도우와 응용 사이의 사건 큐(event queue)에 사건 처리기와 사건 재지향기, 사건 여파기를 설치한다. 이들은 사건 큐를 지나가는 사건들을 윈

도우로부터 응용 프로그램으로 전달되는 과정에서 가로채서 가공한 다음에 다시 윈도우나 응용 프로그램으로 전달한다. 이 과정에서 오류 공유 시스템이 필요한 처리를 행한다.

#### 3.2 웹에서의 응용 공유 구조

제안하는 오류 공유 시스템은 윈도우 공유 방식을 사용한다. 응용 및 오류 공유 구조로는 분산 복제형 구조(CARV)이다. 분산 복제형 구조는 (그림 1)처럼 중앙 집중형 구조(CACV)와 복제형 구조(RARV) 두개의 장점을 취합한 형태로 모든 시스템이 프로세스를 수행할 수 있는 능력을 가지고 있지만 복제형 구조와 같이 완전한 대칭적인 구조를 가지고 있지는 않다.

CACV 구조의 기능은 View 분배 기능을 중점적으로 지원한다. 또한, 생성자 측의 응용으로 입력을 전달하기 위해서 자신의 인스턴스 영역 내의 사건을 판별하여 생성자 측의 응용 공유 인스턴스의 입력으로 하는 기능을 가지며, 자신과 관계 없는 참여자의 사건을 인식해서 무시하는 기능도 있다. 또한 참여자가 발생시킨 사건을 공유 되는 응용의 입력으로 사용하기 위해서 가공하는 기능도 가진다. RARV 구조의 기능은 이벤트 분배 기능과 공유 되는 응용들간의 일치화 문제를 해결한다. 특히, 다른 참여자의 사건을 동시에 자신의 응용 프로그램으로 전달해서 모든 참여자 사이의 응용에 대한 일관된 처리를 행하며, 참여자들 간에 공유 되는 응용들의 일관성을 감시하여 항상 일관성을 유지하도록, 공유 윈도우 이동, 크기 조정, 사건 사상 함수 변환 등을 행한다. 특히, 사건 사상 함수는 모든 참여자들 간에 공유되는 응용이 반드시 같은 위치에 있지 않아도 공유가 원활히 이루어질 수 있도록 지원한다. 즉, 각각의 복제된 응용들은 서로 상대적인 위치를 유지하더라도 복제된 윈도우 내의 위치만 일정하게 유지가 되면 공유에 지장이 없다. 이렇게 함으로써 각자의 사용자는 융통성 있는 응용의 배치를 가능케 하며, 각자의 독립적인 작업을 방해하지 않으면서 공동 작업을 수행해나갈 수 있다.



(그림 1) 분산 복제형 구조의 구성

CACV 구조에서의 응용 및 오류 공유 과정은 먼저, 참여자가 공유되는 응용 및 오류 사건을 발생시키는 데서 시작한다. 사용자가 발생시킨 사건은 윈도우 메시지의 형태로 참여자 측 사건 분배기로 재지향 되고,

이는 다시 생성자 측 사건 분배기로 네트워크를 통해서 전송된다. 생성자 측 사건 분배기는 수신한 사건을 다시 공유되는 응용 프로그램으로 재지향 한다. 여기까지가 참여자 측에서 생성된 사건이 생성자 측 응용까지 전달되는 경로이다. 생성자 측의 응용은 전달 받은 사건을 수행해서 수행 결과인 view 를 화면 상에 생성하며, 이를 생성자 측 뷰 분배기가 감지해서 전송할 수 있는 형태로 압축한 후, 이를 여러 개의 전송 가능한 패킷으로 만들어서 참여자 측으로 보낸다. 참여자 측에서는 뷰 분배기가 생성자 측에서 보낸 패킷 들을 받아서 다시 하나의 뷰 정보로 재구성해서 압축을 해제한 다음, 참여자 측 화면으로 출력한다. 이렇게 해서 참여자는 응용 프로그램을 가지지 않고서도 생성자의 응용 프로그램을 통해서 공동 작업을 수행할 수 있다.

RARV 구조에서의 응용 및 오류 공유 과정은 한 사용자가 발생시킨 응용 및 오류 사건은 윈도우 메시지의 형태로 사건 분배기로 재지향 되고, 이는 다시 다른 사용자들의 사건 분배기로 네트워크를 통해서 전송된다. 다른 사용자들의 사건 분배기는 수신한 사건을 다시 공유되는 응용 프로그램으로 재지향 한다. 여기까지가 한 사용자 측에서 생성된 사건이 다른 사용자들이 소유한 공유 응용까지 전달되는 경로이다. 다른 사용자들의 응용은 각자 전달 받은 사건을 수행해서 수행 결과인 view 를 화면 상에 생성하며, 이를 각자의 화면으로 출력한다. 이렇게 해서 사용자들은 각자의 워크스테이션에 응용 프로그램을 가지고 각자 생성한 사건을 사건 분배기를 통해서 분배하는 방법으로 각자의 응용 프로그램을 가지고 공동 작업을 수행한다.

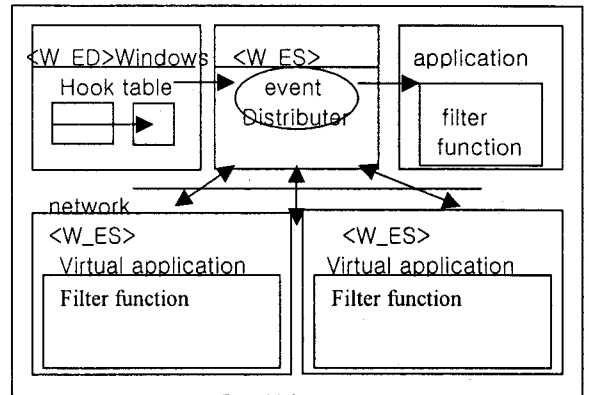
### 3.3 오류 공유 시스템

본 논문에서 제안하는 ESS\_WMCE 는 여러 기능의 에이전트가 존재하며 원활한 오류 감지 및 복구 기법을 수행하는 멀티 에이전트 시스템이다. ESS\_WMCE 를 구성하는 구성 모듈로는 W\_ED(Web based Error Detection)와 W\_ES(Web based Error Sharing), W\_ER(Web based Error Recovery)이다.

W\_ED는 오류를 감지하는 핵심 에이전트로 고장 감지 정보 흐름은 윈도우의 훅킹(hooking) 방법을 이용하여 그 상태를 분석하여 오류의 발생 여부를 감지한다. 이 과정에서 오류를 감지한 내용, 즉, 포인팅 하는 함수를 가로채서 전달하는 방식이다. W\_ED는 저널 레코드 또는 셸 훅 등의 훅킹 메시지를 사용하여 오류를 감지한다.

오류가 감지되면 W\_ES에서 오류 공유가 발생한다. 그 사건은 윈도우 메시지 형태로 사건 분배기로 재지향 되고, 이는 다시 다른 사용자들의 사건 분배기로 네트워크를 통해서 전송된다. 다른 사용자들의 사건 분배기는 수신한 사건을 다시 공유되는 응용 프로그램으로 재지향 한다. 즉, 한 사용자에서 발생된 오류가 다른 사용자들의 공유 응용까지 오류를 전달하는 경로이다. 다른 사용자들의 응용은 각자

전달 받은 오류 사건을 수행해서 수행 결과인 뷰(view)를 화면상에 생성하며, 이를 각자의 화면으로 출력한다. 이렇게 해서 사용자들은 각자의 워크스테이션에 응용 프로그램을 가지고 각자 발생한 오류 사건을 사건 분배기를 통해서 분배하는 방법으로 각자의 응용 프로그램을 가지고 공동 작업을 수행하거나 오류를 인식한다. 오류 공유 과정은 (그림 2)와 같다.



(그림 2) 오류 공유 과정

W\_ER 에서는 오류 복구가 이루어진다. 이 때 단순 재 실행인 경우에는 미디어 자원과 미디어 서버 인스턴스를 생성시켜서 복구시킨다. 세션이 진행 중인 경우에는 검사점 설정까지 설정된 지점까지 복귀하여 복구한다.

### 4. 시스템평가

제안된 시스템은 Visual C++로 설계 및 구축하였다. 기존 멀티미디어 공동 작업 환경은 웹 기반의 기능이 없고 오류 공유 등의 기능도 없다. 웹 기반 멀티미디어 공동 작업 환경에서의 오류 전달에서 기존 방법보다 제안된 방법의 나은 점을 시뮬레이션을 통하여 [표 1]과 같이 비교하였다.

[표 1] 웹 기반 멀티미디어 공동 작업 환경에서의 오류 공유 적용 비교

	Shastra	MERM AID	MMc-onf	CECED	제안된 논문
오류 공유	지원 안됨	지원 안됨	지원 안됨	지원 안됨	지원
응용 공유	지원 안됨	지원 안됨	지원 안됨	단일 응용	단일 응용, 응용 복제
통신 프로토콜	TCP/IP	TCP/IP	TCP/IP (multi cast)	TCP/IP (multi cast)	UDP/IP (multi cast)

### 5. 결론

본 논문에서 기술한 방식은 웹 기반 멀티미디어

공동 작업 환경에서 세션을 잘 유지하기 위하여 오류 공유를 위한 시스템을 제안하였다. 응용 공유는 단일 사용자용으로 설계된 응용을 네트워크를 통해서 동시에 여러 사용자가 사용할 수 있도록 함으로써 공동 작업 환경을 제공한다. 그러므로, 기존에 사용되던 응용 프로그램을 수정하지 않고 공동 작업 환경에서도 재사용할 수 있다는 장점이 있다.

제한한 오류 공유 시스템은 UDP 브로드 캐스팅 및 IP 멀티캐스팅을 지원하며 적은 부하로 LAN 및 WAN 환경에서 응용 프로그램에서의 오류를 다수의 사용자가 공유하도록 지원하였다. 특히 기존의 TCP를 이용한 방법과는 달리 참여자의 수에 관계없이 일정한 전송률과 동일한 응답 시간을 보장함으로써 동시에 모든 참여자들이 같은 결과의 오류를 공유하게 되며 사용자의 수의 증가에 따른 영향을 적게 받는다. 본 논문에서 제안하는 시스템은 여러 기능의 에이전트가 존재하며 원활한 오류 감지 및 복구 기법을 수행하는 멀티 에이전트 시스템이다.

오류 공유 시스템은 오류 감지, 오류 공유, 오류 복구로 구성되었다. 오류 감지 에이전트는 고장 감지 정보 흐름은 윈도우의 훅킹(hooking) 방법을 이용하여 그 상태를 분석하여 오류의 발생 여부를 감지한다. 이 과정에서 오류를 감지한 내용, 즉, 포인팅 하는 함수를 가로채서 전달하는 방식이다.

오류가 감지되면 오류 공유가 발생한다. 그 사건은 윈도우 메시지 형태로 사건 분배기로 재 지향 되고, 이는 다시 다른 사용자들의 사건 분배기로 네트워크를 통해서 전송된다. 다른 사용자들의 사건 분배기는 수신한 사건을 다시 공유되는 응용 프로그램으로 재지향 한다. 즉, 한 사용자에서 발생된 오류가 다른 사용자들의 공유 응용까지 오류를 전달하는 경로이다.

오류 복구 에이전트에서는 단순 재 실행인 경우에는 미디어 자원과 미디어 서버 인스턴스를 생성시켜서 복구시키고 세션이 진행 중인 경우에는 검사점 설정까지 설정된 지점까지 복귀하여 복구한다.

향 후 연구과제로는 웹 기반 다중 환경을 지원하는 오류 공유 기능에 대한 연구 등이다.

※ 본 연구는 2003 정보통신연구진흥원 기초기술 연구 지원 사업의 지원으로 수행되었음

참고문헌

[1] A. Versey and Ajay Paul Sravana, "CASE as Collaborative Support technologies", Communication of ACM, pp.83-94, Jan. 1995.  
 [2] James D.Palmer and N. Ann Fields, "Computer Supported Cooperative Work", IEEE Computer, pp.1517, May 1994.  
 [3] 탁진현, 이세훈, 왕창중, "다중 참여자를 지원하는 3 차원 가상 환경에서 공동 작업을 위한 어플리케이션 공유 시스템 설계", 한국 정보처리회 논문지 제 7 권 제 2 호, pp.355-364, 2000

2 월.  
 [4] Victor P. Nelson and Bill D. Carroll, "Fault-Tolerant Computing", IEEE Computer Society Order Number 677, Library of Congress Number 86-46205, IEEE Catalog Number EH0254-3, ISBN 0-8186-0677-0.  
 [5] Eung-Nam Ko, Chul Hwang, Dae-Joon Hwang, "Implementation of an Error Detection-Recovery Software for Interactive Multimedia Environment by using Hook Technique: EDRSHT", In proceedings of IEEE/IEE ICT'99, Cheju, Korea, June 15-18, 1999, pp.340-344.  
 [6]Klaus H. Wolf and Peter Schulthess, "Multimedia Application Sharing in a Heterogeneous Environment" ACM Multimedia'95, November 5-9,1995.  
 [7] Hiroaki Higaki, Kenji Shima, Takayuki Tachikawa, Makoto Takizawa, Checkpoint and Rollback in Asynchronous Distributed Systems, IEEE INFOCOM97, Proceedings Volume 3.  
 [8]A. Anupam and C.L.Bajai, "Collaborative Multimedia Scientific Design in Shastra", Proceeding of the ACM Multimedia'93, Aug. 1993, pp.447-456.  
 [9]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.  
 [10]Torrence Crowley and Raymond Tomlinson, MMConf: An Infrastructure for Building Shared Multimedia Applications, CSCW '90 Proceedings, October 1990.  
 [11]Earl Craighill and Keith Skinner, CECED: A System For Informal Multimedia Collaboration, Proceedings ACM Multimedia '93, August 1-6 1993.