

# 웹기반 멀티미디어 저작 시스템의 설계 및 구현

## (Design and Implementation of the Web-based Multimedia Authoring System)

박 영 미\*    박 종 운\*\*    송 한 영\*\*\*    신 현 신\*\*\*\*    박 두 영\*\*\*\*\*    장 종 환\*\*\*\*\*  
(Young-Mi Park) (Jong-Woon Park) (Han-Young Song) (Hyun-San Shin) (Doo-Yeong Park) (Jong-Whan Jang)

### 요 약

아이콘 기반의 저작 환경에서 멀티미디어 문서를 생성 및 편집할 수 있으며, 저작 과정에서의 오류 검증 및 피드백 기능, 사전실행 기능을 통하여 저작내용을 확인하면서 저작을 진행할 수 있고, 자료의 내부 구조를 트리구조로 채택하여 시간관계와 공간관계를 갖게 되는 미디어 객체들간의 상속성 표현기능을 사용하여 프리젠테이션에 참여하는 미디어 객체들간의 동기와 사용자 상호작용에 의한 비동기를 효율적으로 처리할 수 있는 웹기반 멀티미디어 저작 시스템을 구현하였다.

### ABSTRACT

The web-based multimedia development tool was implemented to create and edit multimedia contents under the graphic user interface environment with icons. The development process enables the developers to verify the contents being developed by checking for errors, exchanging feedbacks and utilizing the pre-execution feature. By applying the tree structure to the internal data format, the tool uses inherent display features among media objects with time and spatial relations to efficiently process the synchronism of media objects that participate in the presentation and asynchronism according to user interaction.

## 1. 서론

### 1.1 개요

웹 사용이 증가함에 따라 인터넷을 이용하는 다수의 사용자가 서버에 접속하여 사용할 수 있는 멀티미디어 저작 시스템의 필요성이 증가하게 되었다. 그러나 기존의 저작 및 프리젠테이션 시스템을 사용하여 저작하기에는 그 저작 환경이 어렵고 복잡하며,

아울러 저작에 따른 자동화된 멀티미디어 데이터 처리 환경이 미비한 실정이다. 기존의 멀티미디어 저작 시스템이 갖는 문제점을 사용자 저작 환경과 멀티미디어 데이터 처리 환경 측면에서 정리하면 다음과 같다[1,40].

최근의 인터넷의 급속한 확산과 사용자 환경이 웹 환경으로 바뀌어 가고 있다. 기존에 개발된 저작 시스템들은 주로 스크립트를 기반으로 하는 프로그래밍 언어를 사용하기 때문에 웹과 윈도우에 익숙한

\* 정회원 : 배재대학교 정보통신공학과 시간 강사

논문접수 : 2001. 12. 24.

\*\* 정회원 : 배재대학교 정보통신공학과

심사완료 : 2002. 2. 23.

\*\*\* 정회원 : 배재대학교 정보통신공학과

\*\*\*\* 정회원 : 충청대학교 컴퓨터학부

\*\*\*\*\* 정회원 : 배재대학교 정보통신공학과

\*\*\*\*\* 정회원 : 배재대학교 컴퓨터 전자 정보통신공학부 교수

※ 이 논문은 2001년도 두뇌한국 21 사업에 의하여 지원되었음.

일반 사용자들에게는 배우기도 어렵고 사용 방법이 까다롭다. 또한 저작시 발생하는 일관성 오류를 즉시 사용자에게 알려주는 동적 저작 기능이 미흡하다. 동적 저작은 편집 단계에서부터 실행 단계까지 일관성 있는 프로그램이 되도록 하는 저작 도구의 중요한 기능이다. 전체적인 실행 시간 등을 고려하여 시나리오 저작 중간에 실행 내용을 알려주는 사전실행 기능이 미흡하다. 사전실행 기능은 각 미디어 리소스 내용뿐만 아니라 현재까지 저작된 문서의 사전실행을 지원할 수 있어야 한다. 즉, 멀티미디어 저작 환경과 함께 데이터 처리 환경 측면에서는 다양한 미디어 리소스 제공 기술 및 저장기술 연구가 미흡하다. 사용자가 저작 화면상에서 직접 미디어 객체를 선택하고 간단한 조작으로 멀티미디어 문서에 포함시킬 수 있는 연구도 중요한 멀티미디어 저작 시스템의 일부분이다.

본 논문에서는 기존 저작 시스템의 문제점을 해결하기 위해 아이콘 기반의 사용자 인터페이스를 제공하여 줌으로써 다양한 미디어 리소스를 사용하여 간단한 조작으로 컴퓨터 프로그램 전문가가 아닌 일반 사용자도 쉽게 멀티미디어 문서를 생성하고 편집할 수 있는 프리젠테이션 저작 도구와 미디어간 시간과 공간 관계를 통하여 프리젠테이션에 참여하는 미디어간의 동기 및 사용자 상호작용에 의한 비동기를 효율적으로 지원하는 프리젠테이션 엔진을 다음과 같이 구현하고자 한다.

첫째, 아이콘을 기반으로 WYSIWYG(What You See Is What You Get)를 제공하는 멀티미디어 문서 자료 편집 환경을 제공한다. 사용자는 복잡한 스크립트 언어를 사용하지 않고도 간단한 조작으로 멀티미디어 문서를 저작할 수 있다.

둘째, 클라이언트-서버 환경에서 사용할 수 있도록 한다. 사용자는 서버에 접속하여 저작 시스템을 다운 받아 멀티미디어 문서를 저작할 수 있고, 필요한 리소스 정보를 서버로부터 제공받을 수 있다.

셋째, 웹상에서 사용할 수 있도록 한다. 인터넷 사용자가 웹 브라우저상에서 멀티미디어 문서를 저작할 수 있고, 저작한 문서로 프리젠테이션을 실행할 수 있다.

넷째, 미디어 리소스 및 저작된 프리젠테이션 자료의 사전실행 기능을 제공한다. 저작시 수시로 피드백 과정을 반복하며, 작업 내용을 즉시 확인할 수

있다.

다섯째, 프리젠테이션 진행 중 사용자가 미디어 객체의 실행 순서를 원하는 형태로 바꿀 수 있는 동기 및 비동기 지원 매커니즘을 지원한다.

여섯째, 미디어 객체들간의 관련성을 시간과 공간 관계로 정의하고, 이를 내부 구조로 정의한다. 미디어 객체들간에는 시간적, 공간적 관계를 가질 수 있으며 이를 내부에 트리 구조로 정의하여 저장하고 관리한다.

일곱째, 프리젠테이션에 참여하는 미디어 객체들은 객체 상태 테이블을 통하여 관리한다. 각 미디어 객체들은 일정한 주기로 생성되고, 실행되며, 실행 후 휴면상태에 들어가는 과정을 테이블 정보로 관리하며 이러한 객체상태정보는 스케줄러와 연동해서 사용한다.

본 논문의 구성은 다음과 같다. 2장에서는 멀티미디어 저작 시스템과 관련된 국제표준화기구의 표준화 활동, 국내의 저작 시스템의 개발 현황 및 최근 연구 동향을 기술하였다. 3장에서는 기존 저작 시스템의 문제점을 해결하기 위해 제안한 시스템의 구조와 기능, 사용자 인터페이스 화면 설계, 저작 컴포넌트 및 프리젠테이션 실행 엔진을 설계하였다. 4장에서는 저작 시스템을 구현하고, 구현된 저작 시스템의 실행 과정을 기술하였다. 마지막으로 5장에서는 본 논문의 결론 및 향후 연구내용을 기술하였다.

## 2. 관련 연구 분석

본 장에서는 멀티미디어 저작 시스템과 관련된 국제표준화기구의 표준화 활동, 국내의 저작 시스템 및 최근 연구 동향 등을 조사하여 분석하였다.

### 2.1 멀티미디어 문서 표준화 활동

MHEG1,5(ISO/IEC JTC1/SC29/WG12)[18], Hytime (ISO/IEC 10744 : 1992)[18,19,21,22], SGML(ISO 8879)[25,26,27], ODA(ISO 8613)[20], HyperODA (ISO/IEC JTC1/SC18/WG3)[23]등의 국제표준화 기구에서 멀티미디어 문서를 정의하고 전자문서를 지원하는 저작 시스템의 구조를 제안하고 있다.

## 2.2 국내 저작 시스템 분석

국내 업체들의 저작 시스템은 주로 교육 목적으로 만들어졌다. 새빛은 초기버전(윈도우3.1)에 비해 많은 발전을 해 왔으나 처리 속도가 늦어 리듬이 끊어지고, 미디어의 상세 행위를 기술하기가 어렵고, 다른 언어와의 호환이 안되는 문제점이 있다. 한울의 경우도 사용 범위가 제한적이고, 초기 편집된 시나리오를 수정하는 작업이 어려운 저작 환경을 제공한다. 아울러 저작 중에 그 내용을 사전 실행할 수 있는 기능이 부족하였다. 초기에 외국의 제품을 우리 것으로 만들다 보니 기능 위주로 될 수 밖에 없어 나중에 시스템이 산만해지는 경우가 많았다. 또한 모든 기능을 다 제공하려는 욕심 때문에 일반 사용자가 원하는 소프트웨어를 만들기가 어렵다. 현재 베타 버전으로 현장 시험 중인 PASS2000의 경우 교육용 코스웨어 제작 목적을 충실히 이행하였으나 원하는 소프트웨어를 만들어 나가는 과정이 복잡하고, 기능은 많으나 일반 사용자에게는 초기 작업 시간이 길 것으로 예측된다[39,45].

## 2.3 국외 저작 시스템 분석

주로 대학에서 연구 개발되고 있는 저작 시스템은 시간과 동기화, 시간과 공간의 동기화, 멀티미디어 문서의 새로운 파싱 알고리즘, 효율적인 내부 저장 구조 등에 초점을 맞추어 개발하였기 때문에 특정 분야의 응용에 그 사용이 국한된다.[39]

멀티미디어 저작 시스템은 저작 모델을 정의하고 이 모델을 바탕으로 프리젠테이션 되는 시나리오를 저작할 수 있는 시스템이다. Harmony[24] 시스템은 일본 오사카 대학에서 연구된 멀티미디어 프리젠테이션을 위한 하이퍼텍스트 패러다임을 확장한 시스템이다. Harmony 시스템은 시간 종속적인 미디어 객체 정의와 객체간에 이벤트에 의해서 메시지가 전달되는 방식, 그리고 부객체 개념 등을 제공하고 있었지만, 미디어간에 동기화가 불분명하고, 실시간 처리가 미흡한 단점을 가지고 있다.

CMIF[13,14] 시스템은 네덜란드 암스테르담 대학의 저작 시스템으로서, 프리젠테이션에 참여하는 미디어들을 처리 순서에 따라 계층화하여 순차적으로 혹은 병렬로 수행시킬 수 있는 구조를 갖는 시스템

이다. CMIF는 Harmony 시스템을 발전시킨 모델로써, 다양한 동기화 함수를 제공하고는 있지만, coarse-grain 동기화와 사용자 상호작용에 의한 비동기 처리 지원이 어렵고, 그밖에 공동 작업 등 새로운 저작 환경 지원을 위한 확장성이 미흡했다.

FLIPS[6] 시스템은 미국 미네소타 대학의 멀티미디어 프리젠테이션 시스템으로써, 융통성 있는 프리젠테이션 제공을 목표로 하고 있다. FLIPS 시스템은 융통성 있는 프리젠테이션을 제공하였지만, 동기화에 있어서 coarse-grain 동기화만을 대상으로 하기 때문에 fine-grain 동기화에는 문제점이 있었다. 사용자 인터페이스에서 저작하는 시나리오 구성이 계층적인 형태를 제공하지 못하여 이를 파싱하고 내부 구조화하기가 어렵다.

Firefly[11] 시스템은 미국 Xerox의 저작 시스템이다. 현재 연구가 진행 중에 있으며 연구의 목적이 본 연구와 유사하지만, 연구 추진 전략이나 연구 방법에 있어서는 그 접근 방식이 다른 시스템이다. Firefly 시스템은 최적의 스케줄링을 위한 알고리즘이 정의되었으나 만들어진 알고리즘 구성이 모듈 형태가 아니기 때문에 이를 디버깅하기가 쉽지 않았다. 아울러 사용자 인터페이스 부분에 있어서 그 시나리오 구성을 트리 구조와 같은 계층화된 형태로 표현하는 것에 관하여 연구 중에 있다.

상용 제품들은 주로 미국 제품들로서 애플사, 매크로미디어사, 테크미디어사, 오라클사에서 개발되고 있으며 Authorware, IconAuthor, Multimedia Viewer, Toolbook, Audio Visual, Hypermedia, Action!, Director, Autound, Hypercard, MediaView, PowerPoint 등이 있다[1,39,42,43,44].

상용화된 저작 시스템들은 주로 사용자 인터페이스를 강화하였다. 따라서 시스템별로 그 특징과 전문성이 상이했다. 초보자들이 멀티미디어 저작에 대한 개념 이해와 간단한 저작 작업에 들어갈 때 비교적 쉬운 제품으로는 Authorware를 들 수 있다. Authorware는 매크로미디어사 제품으로 저작 메타포어 방식이 흐름도 중심이기 때문에 기존의 프로그래밍 방식과 유사했다. 사람의 직관에 의해서 진행순위에 미디어를 갖다 놓는 방식이기 때문에 초기 작업 시간을 최소화 할 수 있었다. 시나리오의 구성을 계층화하여 각 모듈의 구성이 명확한 반면, 시나리오의 동기점을 명확히 표현하기가 어려운 단점이 있

었다. 상용화된 제품들은 대부분 미디어의 공간 특성, 객체의 공간 이동 등의 애니메이션 효과 제공에는 탁월했지만 미디어의 시간 특성인 미디어간 동기점 표현, 미디어의 시간 속성 등에는 여러 면에서 기능이 미흡했다[42,43,44].

## 2.4 최신 연구동향

멀티미디어 저작 시스템의 최신 연구 동향은 다음과 같다.

첫째, WYSIWYG(What You See Is What You Get) 멀티미디어 문서 편집 환경 제공으로 다형성 특징을 갖고 있는 각 미디어를 화면상에서 사용자가 직접 통합 편집 하는 기능을 제공한다.

둘째, 통합 미디어의 빠른 스토리보딩(story boarding) 작업을 지원하는 것으로 빠른 시나리오의 편집과, 편집 중에 그 수행 결과를 알수 있는 사전실행 기능을 제공한다.

셋째, 문서의 논리적 구조 정의의 필요성이 저작 시스템 구조에서 중요한 부분으로 세그멘테이션 기법이나, 각 미디어의 시간적, 공간적 컴포지션 등을 통해 얻어진 결과를 정형화된 시스템의 논리적 내부 구조 형태로 저장한다.

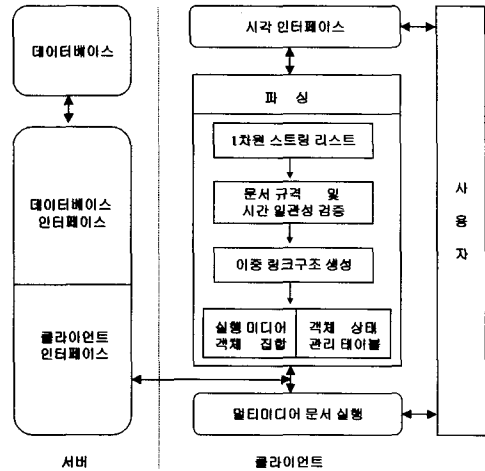
그밖에 3차원 그래픽 표현 기법의 제공과, 분산 환경하에서의 다양한 도구 및 자원의 활용 가능성 등이 연구되고 있다. 이러한 연구 동향을 분석해 보면, 크게 사용자 인터페이스 부분과 멀티미디어 문서 파싱과 관련한 저작 시스템의 논리적 내부 저장 구조 부분으로 나눌 수 있다[10,15,16,17].

## 3. 저작 도구 설계

### 3.1 개요

본 논문에서 제안하는 저작 시스템은 멀티미디어 문서 저작이 직관적이고, 사용하기가 쉬운 아이콘 기반의 시각 인터페이스를 지원하고, 문서 저작중 일관성 오류를 검증하여 피드백 해 주는 기능을 제공한다. 또한 저작 중간에 그 내용을 사전 실행하는 기능을 지원한다. 실행 미디어 객체간의 관련성을

시간과 공간관계로 정의하고 이를 트리 구조의 내부 저장 구조로 정의하여 사용하므로 객체 노드간 상속성을 표현할 수 있어 객체 미디어간 동기화 및 사용자 상호작용에 의한 비동기를 효율적으로 지원한다. 멀티미디어 문서 저작을 위한 시스템의 처리 구성도는 [그림 1]과 같다.



[그림 1] 저작 시스템 처리 구성도  
[Fig. 1] development tool process structure

## 3.2 저작 시스템 모듈 구성

저작 시스템은 크게 저작 도구 인터페이스와 프리젠테이션 엔진으로 구분되며 각각의 명칭은 [그림 2]와 같고 각 모듈별 기능은 다음과 같다.

### 3.2.1 저작 도구 인터페이스

#### 3.2.1.1 저작도구 관리자

##### 1) 사용자 인증 및 환경 설정 모듈

클라이언트의 권한을 설정하고 각각의 클라이언트 환경을 저장, 관리한다. 모듈 기능으로는 사용자 등록 기능, 사용자 인증 기능, 저작 환경 설정 기능이 있으며 환경 설정은 크게 GUI 설정과 컨트롤 설정을 할 수가 있다.

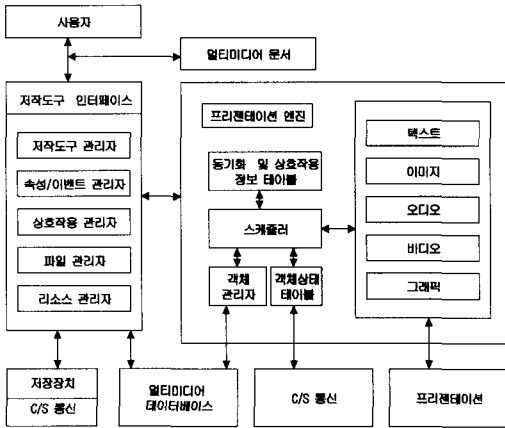
##### 2) 컨트롤 제어 모듈

사용자의 환경 설정에 따라 컨트롤의 디스플레이를 결정하는 모듈로서 기능으로는 블록기능, 환경

설정 로딩 기능, 동적 컨트롤 로딩 기능 등이 있다.

3) **저작판 수행 모듈**

사용자가 선택한 객체를 디스플레이하는 모듈로서 기능으로는 객체 레이아웃 기능, 객체 이동 기능, 객체 리사이징 기능, 페이지 네비게이션 기능 등이 있다.



[그림 2] 저작 시스템 모듈 구성도  
[Fig. 2] development tool module structure

4) **편집 모듈**

저작시에 객체들에 대한 편집 기능을 수행하는 모듈로서 기능으로는 복사 기능, 붙여넣기 기능, 전체 선택 기능, 지우기 기능, 실행취소(undo) 기능 등이 있다.

5) **프로젝트 관리 모듈**

프로젝트 파일을 삭제, 이동, 복사하며 아울러 프로젝트 파일에 대한 정보를 변경한다. 기능으로는 삭제 기능, 복사 기능, 이동 기능, 정보 변경 기능 등이 있다.

6) **저작도구 관리 모듈**

저작도구를 사용자가 사용할 수 있도록 다운로드 하고 캐싱기능을 가능케 한다. 또한 저작도구의 기능 추가 및 향상을 위하여 업그레이드가 필요한 경우 사용자의 브라우저가 버전을 비교하여 최신의 것 이면 다운로드하여 관리한다.

7) **백업복구튜닝 모듈**

사용자 정보, 리소스 정보 등 데이터 베이스와 실행 파일을 백업하고 필요시 복구한다.

8) **모니터링 및 제어 모듈**

클라이언트에 대한 모니터링 및 트래픽 등 통계

데이터를 관리한다. 아울러 임계영역(수행능력)에 이르면 서비스를 생성하지 않도록 지시한다.

3.2.1.2 **속성 및 이벤트 관리자**

1) **속성이벤트 제어 모듈**

사용자가 선택한 객체에 대한 속성 및 이벤트를 정의한다.

2) **이벤트 플로우 생성 모듈**

사용자의 액션을 플로우 형태로 기술한다. 사용자의 액션 사이는 연속적 구간으로 설정된다.

3) **링크 생성 모듈**

선택된 객체에 대한 하이퍼 링크를 설정한다.

3.2.1.3 **파일관리자**

1) **프로젝트 로딩 및 생성 기능 모듈**

저작결과를 저장할 파일을 생성하고 저작결과를 저장하며 저장된 저작물을 편집하기 위해 로딩한다.

3.2.1.4 **리소스 관리자**

1) **리소스 관리 모듈**

그룹별/사용자별 리소스를 관리한다. 리소스마다 접근권한을 설정하여 관리한다.

2) **리소스 뷰어 모듈**

서버에 있는 자원을 디스플레이 하며 리소스를 클라이언트로 전송한다.

3.2.2 **프리젠테이션 엔진**

저작 결과를 실행하는 저작 시스템 엔진으로 사용자에게 저작 결과물을 브라우저를 통해 디스플레이 하는 기능을 수행한다. 세부 기능으로는 엔진 로딩 기능, 결과물 파싱 기능, 멀티스레드/상태 테이블을 통한 디스플레이 기능 등이 있다. 아울러 미리보기(preview) 모듈로서 저작된 결과물을 저작시에 동적으로 수행시키는 모듈로 기능으로는 결과물 파싱 기능, 멀티스레드/상태 테이블 등을 이용한 디스플레이 기능 등이 있다.

3.2.2.1 **동기화 및 상호작용 정보테이블**

프리젠테이션 미디어 동기화 처리를 위하여 사용되는 테이블로서 파일관리자로부터 받은 저작 모듈에서의 저작 내용과 아울러 저작중 발생가능한 이벤

트 정보에 대한 입출력 기능을 실행한다.

### 3.2.2.2 스케줄러

파일관리자에게서 받은 파일을 분석하여 실행에 참여하는 미디어 객체 정보, 객체상태 정보, 각 미디어별 스케줄러를 관리한다. 전체 프리젠테이션을 담당하는 마스터 스케줄러로서 각 미디어 객체의 시작 시간/종료시간을 정의하고, 이벤트버튼에 따른 스케줄러 스프레드를 실행시키며, 객체상태 테이블(SMT: State Management Table)을 관리한다.

### 3.2.2.3 객체 관리자

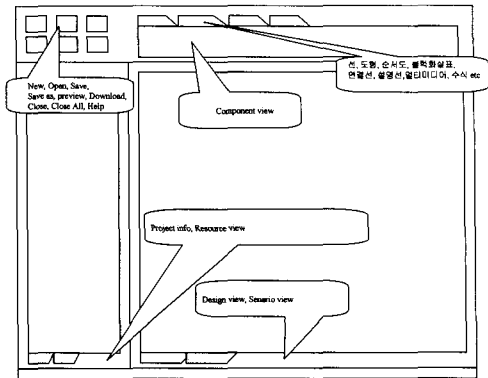
파일관리자를 통해 엔진에 전송되는 각 미디어 객체 정보를 관리한다.

### 3.2.3 문서 파일 구조

저작 시스템 파일관리자는 프로젝트 파일과 스크립트 파일을 관리한다. 프로젝트 파일은 자바의 실행결과 내용을 저장하는 것이고, 스크립트 파일은 슬라이드 및 객체 실행정보인 파일의 생성 정보, 슬라이드 번호, 시간, 공간, 리소스이고, 동기화 정보는 각 슬라이드내 객체간 수행 순서 정의이며, 이벤트 정보는 사용자 상호작용에 대한 정보로서 각 객체들에 대하여 미리 정의된 사용자의 상호작용시 수행되는 액션 정보를 갖고 있다.

## 3.3 사용자 인터페이스 화면 설계

클라이언트 사용자에게 멀티미디어 문서를 저작, 편집하는 기능을 제공하고, 저작 결과물을 사전 실행해 볼 수 있는 저작 환경을 제공하는 사용자 인터페이스 화면을 설계하였다.



〔그림 3〕 메인 프레임 화면  
[Fig. 3] main frame image

사용자 인터페이스 화면은 메인 프레임 화면, 리소스 뷰 화면, 속성 뷰 화면, 디자인 뷰 화면, 시나리오 뷰 화면으로 구성된다. 메인 프레임 화면은 저작환경 사용자 인터페이스의 기본틀을 구성하고, 메뉴바, 컴포넌트 바, 상태바를 가진다. 또한 저작파일에 관한 일반 사항과 파일명, 생성날짜, 변경날짜, 파일 크기, 사용자 정보를 표시해 준다. 리소스 뷰 화면은 동영상, 소리, 이미지등 리소스 파일 정보를 표시해 주며, 속성 뷰 화면은 미디어 객체의 속성을 설정한다. 디자인 뷰 화면에서는 저작활동을 수행하며, 컴포넌트 배치, 시간설정, 슈트관리기능을 수행하고, 시나리오 뷰 화면에서는 컴포넌트 상호간에 공간관계와 시간관계를 설정한다. [그림 3]은 메인 프레임 화면의 설계내용을 보여준다.

## 3.4 실행 엔진 설계

### 3.4.1 저작문서 정의

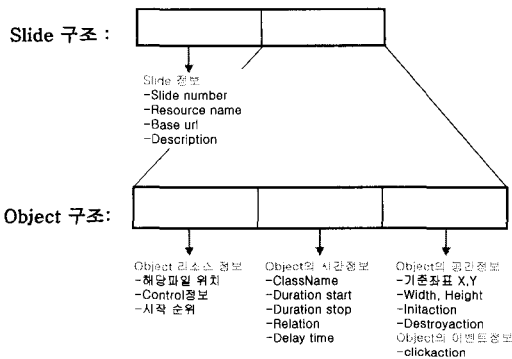
저작문서는 저작활동으로 이루어진 문서정보 및 각 슬라이드와 객체들의 공간 및 시간, 이벤트 정보를 XML(eXtensible Markup Language)을 이용하여 작성한다.

DTD(Document Type Definition)란 문서 원형 정의로, 이는 문서의 구조와 구성 요소들의 속성, 객체, 표기법을 정의한 것이다. DTD는 XML문서의 기초로서 개발자가 구조, 내용, XML의 형식을 설정하는 요소들을 만들 수 있게 한다. 저작문서에서도 DTD

를 작성하여 문서의 유효성 검사를 하도록 한다.

본 시스템에서 정의하여 사용하는 파일 구조는 [그림 4]와 같다. 각 슬라이드(slide)는 객체(object)구조를 여러 개 갖을 수 있으며, 각 구조엔 슬라이드와 객체에 관한 시간, 공간, 이벤트, 리소스 정보를 저장한다.

여러 객체에 대한 효율적인 처리를 위해 슬라이드와 객체의 구조는 링크드 리스트로 연결이 되어 있으며 순서대로 슬라이드와 객체의 정보에 접근할 수 있다.



[그림 4] 슬라이드와 객체의 구조  
[Fig. 4] slide and structure of object

### 3.4.2 파싱

파싱은 저작활동으로 만들어진 시나리오를 스캐닝을 통해 정보를 추출하는 단계와 시간 일관성 검증 및 더블 링크드 리스트 구조로 변환하는 단계로 나뉜다. 정보추출단계에서는 문서 및 슬라이드와 객체에 대한 공간 및 시간, 이벤트, 리소스 정보를 XML 파서를 이용하여 확인 추출해 낸다. 시간 일관성 검증단계에서는 프리젠테이션에 참여하는 미디어간의 시간 일관성 검증은, 객체간에 발생될 수 있는 시간 관계가 기존의 시간 관계에 영향을 주는지를 판별하는 것이다. 시간 일관성을 검증하는 방법은 미디어간에 가능한 파생 시간 관계를 비교하면 된다. 여기서 파생 시간 관계란 객체간의 삼각 관계에 의해서 발생될 수 있는 시간 관계를 말하는 것으로, 새로 저작하는 미디어 객체가 아직 다른 미디어와 관계가 없거나, 또는 발생 가능한 삼각 관계에 포함될 때 시간 일관성은 유지되는 것이다. 각 컴포넌트간의

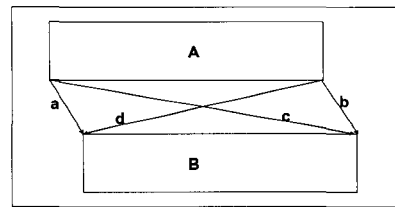
시간 일관성 검증은 객체간의 시간적 우선순위를 파악하여 먼저 시작되는 객체부터 정렬을 하게 된다. 또한 객체들의 수행시간을 계산한다.

### 3.4.3 실행엔진 설계

저작 결과를 실행하는 시스템 엔진으로 사용자에 게 저작 결과물을 브라우저를 통해 디스플레이하는 기능을 수행한다. 엔진 로딩 기능과 결과물 파싱기능, 그리고 이어서 설명할 스케줄링(시간 인과성 관계, 동기화)에 의한 디스플레이 기능 등이 있으며, 미리보기 (preview)모듈로서 저작된 결과물을 저작시에 동적으로 수행시키는 모듈로 이를 구분하면 결과물 파싱, 스케줄링등을 이용한 디스플레이 기능 등이 있다.

#### 3.4.3.1 인과성 시간 관계 정의

인과성 관계는 이유, 목적, 양보, 의뢰로써, 프리젠테이션 상에 실행되는 미디어들은 서로간에 어떤 이유나 목적에 의해 서로 관련성을 맺게 된다.



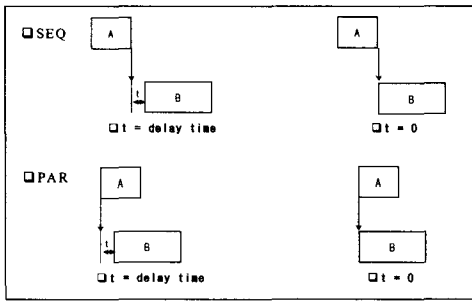
[그림 5] 객체간의 인과성 관계  
[Fig. 5] Causality between object and object

인과성을 표현하는 시간 관계는 하나의 객체(A)에 존재하는 두 개의 동기점인 시작점(A), 끝점(A)을 사용해서 [그림 5]와 같이 크게 4가지로 구성할 수 있다. 시작점은 한 객체의 시작을 나타내고, 끝점은 한 객체의 멈춤을 나타낸다.

[그림 5]는 두 개의 객체 A, B 간에 존재하는 인과성 관계를 표현한 것이다. A, B 각 동기점에 의해서 한 객체는 cause 객체가 되고, 다른 객체는 effect 객체가 된다. 이런 관계는 화살표 방향에 따라 그 역할이 결정된다. 즉 cause 이벤트를 발생하는 A 객체는 능동 객체의 역할을 수행하고, effect 이벤트가 발생하는 다른 한 객체는 수동 객체의 역할을 수행

하는 것이다. 4가지 관계를 설명하면, a는 A가 시작하면서 B를 시작시키는 관계, b는 A가 끝나면서 B를 끝내는 관계, c는 A가 시작하면서 B를 끝내는 관계, 마지막으로 d는 A가 끝나면서 B를 시작시키는 관계를 나타낸다.

[그림 5]를 바탕으로 본 시스템의 수행 엔진에서 정의한 인과성 기반 시간 관계는 [그림 6]과 같다. 기본적인 2가지 인과성 관계와 지연시간을 포함하여 4가지를 정의했다



[그림 6] 인과성 시간 관계  
[Fig. 6] causality time relation

3.4.3.2 동기화

앞서 정의한 인과성 시간 관계 명세를 바탕으로 각각 객체들의 실행 시점과 미디어간에 시간 관계를 적용한 동기화 메커니즘을 지원한다. 멀티미디어 동기화 방식으로 참조점(reference point) 방식을 사용한다. 참조점 정의는 인과성 관계를 갖는 객체간의 참조 위치로써 정적 객체인 경우는 미디어 객체의 시작점과 끝점이 되고, 동적 객체는 동영상의 프레임 번호 순서가 된다. 각 참조점에서는 이벤트를 사용하여 인과성 관계를 유지한다.

3.4.3.3 스케줄러

서버로부터 받은 파일을 분석하여 실행에 참여하는 컴포넌트 객체 정보, 객체 상태 정보, 스케줄을 관리한다. 전체 프리젠테이션을 담당하는 master 스케줄러로서 각 컴포넌트 객체의 시작시간/종료시간/지연시간을 정의한다.

3.4.3.4 객체상태 테이블

객체 상태는 초기(wait), 대기(ready), 실행(running)

g), 완료(complete)로 구분하며, 이 사이클에 의해서 각 미디어 객체가 실행된다. 이벤트는 내부 이벤트(start, stop, display)와 외부 이벤트(skip:(forward, backward), pause, restart)로 구분된다

저작을 통해 만들어진 프리젠테이션은 두 가지로 나눌 수 있다. 사용자의 스케줄링처리가 들어간 동적인 프리젠테이션과 스케줄링 되지 않은 정적인 프리젠테이션으로 크게 나눌 수 있다. 정적인 프리젠테이션은 동적인 변화가 없기 때문에 시간에 따른 상태 테이블이 필요치 않다. 그러나 동적 프리젠테이션을 실행 시 각각의 객체가 시간에 따라 어떠한 상태를 갖고 있는지에 대한 테이블 관리가 필수이다. 따라서 시간에 따른 상태 테이블 관리를 해주고, 동영상과 같은 미디어 객체들의 경우는 index정보를 사용하여 동영상의 프레임에 접근할 수 있게 해줌으로써 프리젠테이션 수행을 일관성 있게 해줄 수 있다. [그림 7]은 5개의 객체가 시간에 따라 어떠한 상태를 갖는지를 보여주는 객체 상태 테이블의 예이다.

| 시간<br>객체 | Index(현재 위치) |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
|----------|--------------|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|-----|
|          | 1            | 2   | 3   | 4   | 5   | 6   | 7   | 8   | 9   | 10  | 11  | 12  | 13  | 14  | 15  | 16  | 17  | 18  | 19  |
| 1 동영상    | run          | run | run | run | run | run | run | run | run | run | run | run | run | run | run | run | run | run | run |
| 2 이미지1   |              |     |     | run | run | run | run | run | run | run | run | run | run | run | run | run | run | run | run |
| 3 텍스트    |              |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| 4 웨이브    |              |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |
| 5 이미지2   |              |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |     |

run      wait      complete

[그림 7] 객체상태 테이블  
[Fig. 7] object state table

3.4.3.5 객체 관리자

파일관리자를 통해 엔진에 전송되는 각 미디어 객체 정보를 관리한다. 스레드를 이용한 버퍼를 마련하여 다음에 실행될 객체의 리소스를 미리 서버로부터 다운 받아 실행준비를 하고, 이미 실행이 끝난 객체에 대해서도 어느 정도의 기간동안 관리를 함으로써 프리젠테이션의 수행을 유연하게 해준다.



3.4.3.6 상호작용

외부이벤트(skip(forward/backward), pause, stop등) 발생시 그에 적절한 반응을 수행한다. 프리젠테이션 관리자는 사용자 상호 작용에 의해 프리젠테이션의 순서가 동적으로 변환하는 경우에, 각 미디어 객체의 상태를 관리하고 이를 효과적으로 처리해준다. 본 시스템에서 지원하고 있는 사용자 상호 작용에 대한 이벤트들은 위에서 정의한 바와 같으며 특히 skip 이벤트에 대하여 알고리즘을 기술하면 다음과 같다.

이벤트 모니터는 실행시간에 사용자로부터 오는 상호작용을 감시하다가 한 이벤트가 발생되는 경우, 프리젠테이션 관리자에게 신호를 보낸다. 프리젠테이션 엔진은 사용자 상호 작용에 대한 신호를 받고 이에 대한 처리를 시작한다.

1) 전진방향 스킵(forward skip) 이벤트

사용자가 현재 프리젠테이션 중인 미디어의 수행을 전진방향 스킵하라는 명령을 입력하면, 현재 수행중인 모든 미디어 객체들에 대한 미디어 플레이어에게 stop 신호를 보내 프리젠테이션을 정지시키고 객체 상태를 run에서 ready로 변경한다. 스킵시간을 계산한 후, 스킵시간이 지난 다음에 프리젠테이션 되어야 할 미디어 객체들을 체크하여 각 미디어들의 수행시간을 남은 시간으로 갱신한다. 또한 그 미디어들을 수행시키기 위해 그 미디어 플레이어들에게 start 신호를 보내고, 객체들의 상태 정보를 ready에서 run으로 혹은 idle에서 run으로 변경한다.

2) 후진방향 스킵(backward skip) 이벤트

현재 수행중인 모든 미디어 객체들에 대한 미디어 플레이어에게 stop 신호를 보내 프리젠테이션을 정지시키고 객체 상태를 run에서 ready로 변경한다. 스킵시간을 계산한 후, 스킵시간 이전에 프리젠테이션 되어야 할 미디어 객체들을 체크하여 각 미디어들의 수행시간을 갱신한다. 또한 그 미디어들을 수행시키기 위해 그 미디어 플레이어들에게 start 신호를 보내고, 객체들의 상태 정보를 ready에서 run으로 혹은 idle에서 run으로 변경한다.

3) 프리젠테이션 속도 조절(scaling of presentation speed) 이벤트

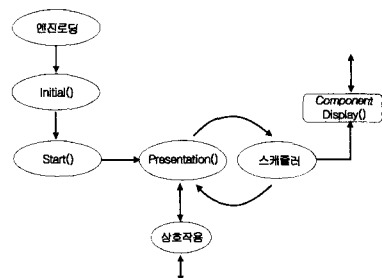
사용자가 프리젠테이션 수행 중인 미디어의 프리젠테이션 속도를 늦추거나 빠르게 하는 프리젠테이

션 속도 조절 명령을 입력하면, 지정된 시간 범위에서 수행되어야 할 모든 미디어들을 체크하여 그 미디어의 수행 속도를 변화시킨다. 또한, 지정된 시간 뒤에 수행될 예정인 미디어의 속도는 변경되지 않지만 동기화를 위해 프리젠테이션 속도의 조절비에 따라 프리젠테이션 시간을 연장 또는 단축시켜야 하는 등의 프리젠테이션 변화가 반영된다.

3.4.4 실행엔진 관리자 수행 알고리즘

[그림 8]은 실행엔진의 메인 알고리즘을 도식화한 것이다.

- 1) 엔진 로딩 : JAVA 가상머신을 로딩하여 실행 엔진을 구동한다.
- 2) Initial() : 초기화를 담당하는 부분이다. 저작활동으로 서버에 만들어진 저작문서를 불러와 읽고, 문서의 내용 중 슬라이드와 객체에 대한 정보를 파싱하여 미리 정의된 파일구조에 저장하는 역할을 한다.
- 3) Start() : 타이머를 작동하면서 1초단위로 Presentation()를 호출한다.
- 4) Presentation() : 파싱을 통해 정보를 저장하고 있는 파일구조에 접근하여 저장된 순서대로 동기화 및 스케줄링을 수행한다.
- 5) ComponentDisplay() : 각 컴포넌트들의 play와 stop을 수행한다.
- 6) 상호작용 : 외부 이벤트 발생시 timer의 작동을 멈추고 이벤트에 대한 적절한 작용을 한다. 종류로는 skip(forward & backward), pause, restart 등이 있다.



[그림 8] 실행엔진 메인 알고리즘 [Fig. 8] execution engine main algorithm

## 4. 저작 도구 구현

### 4.1 개요

제3장에서 제안한 저작 시스템의 설계 내용을 토대로 아이콘 기반의 사용자 인터페이스와 사용자가 사용자 인터페이스를 통하여 저작한 멀티미디어 문서를 웹상에서 실행할 수 있는 프리젠테이션 실행엔진을 구현하였다.

### 4.2 사용자 인터페이스 구현

#### 4.2.1 사용기술 및 응용 소프트웨어

##### 4.2.1.1 사용자 인터페이스 구현 환경

- 서버
  - H/W : 멀티미디어 PC 서버
  - OS : 레드햇 리눅스 6.0(커널 2.2.5)
  - 웹서버 : 아파치 웹 서버, PHP4.0
  - 프로그램 개발 환경 : JDK1.3
- 클라이언트
  - IE4.0 ~ IE5.0, NE 4.0 ~ 5.0
- 4.2.1.2 사용 기술
  - 저작 애플릿
    - 자바빈즈, 자바 2D/3D, RMI, JMF(Java Media Framework), Swing componenet
  - 실행 애플릿
    - 애플릿 - 자바빈즈, 자바 2D/3D, RMI, JMF, Swing componenet
  - 기타 비즈니스 로직 부분
    - PHP script, Java script, DOM(Dynamic Object Model)

#### 4.2.3 사용자 인터페이스 구현

웹상에서 아이콘을 기반으로 멀티미디어 리소스를 선택하고, 편집하여 멀티미디어 문서를 저작할 수

있는 사용자 인터페이스를 구현하였다. 사용자 인터페이스는 크게 멀티미디어 문서의 생성과 편집을 제공하는 주메뉴와 시간과 공간 그리고 실행의 결과를 프리젠테이션하는 상호작용 서브 메뉴로 구성된다. 사용자는 각 슬라이드의 문서를 저작하면서 사전실행을 해 볼 수 있다. 주메뉴에는 객체간의 시간 관계 정의, 미디어 특성 값과 시간 관계 특성값을 입력하고, 아울러 실행되는 미디어들에 대한 공간 관계를 정의하고 공간 특성 값을 입력할 수 있도록 구현하였다.

### 4.3 실행엔진 구현

#### 4.3.1 실행 엔진

실행엔진은 응용프로그램을 구동하면서 저작환경에서 만들어진 저작문서를 서버로부터 읽어 들여 각 슬라이드와 객체에 대한 정보를 파싱하여 링크드 리스트 구조에 정보를 저장하고, 컴포넌트(이미지, 사운드, 동영상, 화살표 등) 사이의 동기화 처리 및 실행, 스케줄링, 상호작용 기능을 수행한다. 애플릿으로 구현하였으며 자바빈즈, 자바 2D/3D, JMF, JDK1.3를 사용하였다.

#### 4.3.2 실행과정

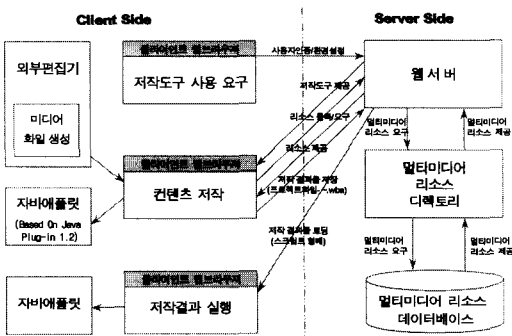
저작활동을 통해 만들어진 멀티미디어 문서파일은 서버의 파일관리 폴더에 저장되어 사용자로 하여금 리스트로 생성된 파일을 확인할 수 있고, 리스트에 열거된 파일 항목 중 자신이 프리젠테이션 하고자 하는 파일을 더블 클릭하여 실행시킬 수 있다.

실행엔진을 일반 HTML문서상에 끼어넣어 다른 하이퍼 텍스트 및 미디어와 함께 사용함으로써 프리젠테이션의 효과를 높일 수 있고, 단독으로 프리젠테이션을 하기 위해선 수행화면을 더블클릭 하여 프리젠테이션만을 수행할 수도 있다.

### 4.4 저작 시스템 실행

사용자가 클라이언트에서 저작 시스템의 사용을 요청하면 서버에서 사용자 인증을 거친 후 사용자별 환경을 설정한다. 사용자 환경을 설정한 후 서버에

서 저작 시스템을 사용자에게 제공하여 저작 활동을 수행할 수 있도록 해준다. 저작중 필요한 리소스를 서버로 제공 받게 되고, 필요한 경우 사용자는 외부 편집기를 이용하여 필요한 리소스를 저작할 수도 있다. 저작 결과물은 서버에 저장되며, 인터넷 접속이 가능한 임의의 장소에서 저작 결과물을 클라이언트로 내려 받아 실행할 수 있다. [그림 9]은 본 논문에서 구현한 저작 시스템의 구조 및 수행절차를 보여 주고 있다.



[그림 9] 저작 시스템 운용구조 및 수행절차

[Fig. 9] development tool structure and implement procedure

#### 4.4.1 사용자 인증

본 저작 시스템을 사용하려면 먼저 사용자 인증을 받아야 한다. 사용자가 등록한 아이디와 패스워드를 입력한 후 확인 버튼을 클릭한다. 등록된 사용자인 경우 저작 시스템에서 제공하는 리소스를 이용하여 멀티미디어 문서를 저작할 수 있고, 저작한 내용을 저장하고, 필요시 인터넷 접속이 가능한 임의의 장소에서 저장된 내용을 서버로부터 다운로드 받아 실행해 볼 수 있다.

#### 4.4.2 저작환경 다운로드

사용자의 PC에 자바 플러그인이 설치되지 않은 경우에는 플러그인 설치 과정이 완료된 후에 해당 애플릿이 화면에 나타난다.

#### 4.4.3 프로젝트 생성 및 로딩

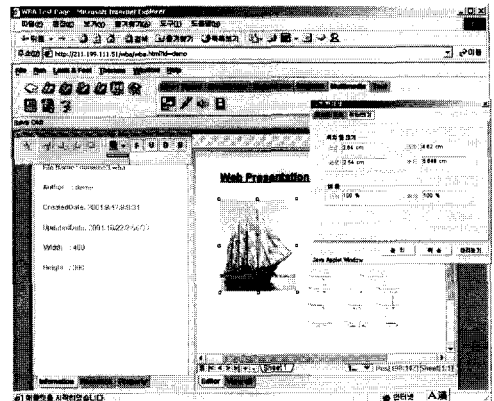
메뉴바에서 File->New가 선택되어지면 새 파일 작성 다이얼로그가 화면에 보이게 되고 기본으로 noname.wba라는 파일명이 나타난다. 사용자가 원하는 이름으로 바꾼 후 OK 버튼을 클릭한다.

#### 4.4.4 문서 저작

##### 4.4.4.1 컴포넌트 배치 및 속성 설정

화면에 저작판이 나타나면, 원하는 컴포넌트를 화면 상단 우측에서 선택한 저작판에 원하는 크기만큼 드래그 앤 드롭하면 해당 컴포넌트가 화면에 나타나게 된다.

특정한 컴포넌트의 기능을 변경하고자 하면, 해당 컴포넌트를 선택한 후 오른 쪽 마우스를 클릭하면 위의 화면과 같이 속성 편집창이 나타나게 된다. 원하는 속성으로 변경 후 확인 버튼을 클릭하면 컴포넌트의 속성이 바뀌게 된다.

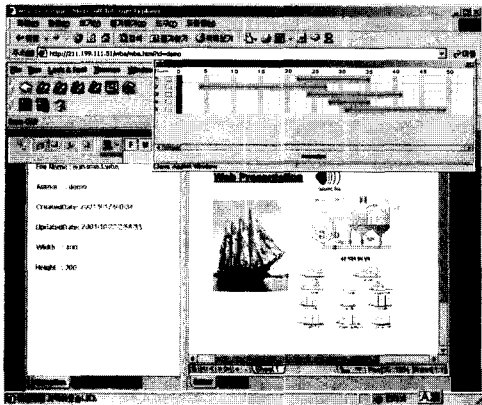


[그림 10] 미디어 객체 배치 및 속성 설정

[Fig. 10] media object arrangement and attribution institution

##### 4.4.4.2 관계 설정

컴포넌트의 배치를 마치고 난 후, 컴포넌트들의 시간 관계를 설정하고자 한다면 Ctrl+S키를 누르면 위의 화면과 같이 시나리오 창이 나타나게 된다. 컴포넌트의 마우스로 드래그하여 기간을 설정 후 "X" 버튼을 누르면 시간 설정 관계가 저장이 된다.

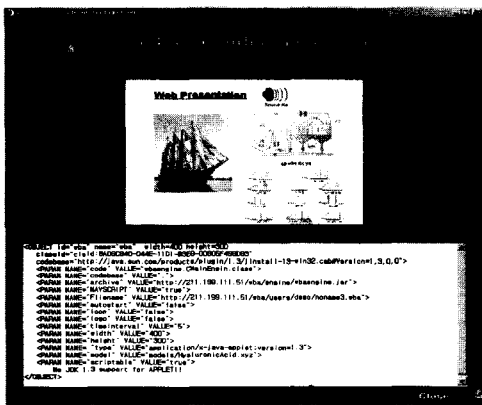


[그림 11] 관계설정  
[Fig. 11] relation institution

4.4.5 저작결과물 저장 및 실행

저작활동을 완료한 후, 작업 내용을 저장하려면 File->Save, Save As, Save All을 선택한다. 저장이 정상적으로 완료되면 상태바에 “Save OK” 메시지가 나타난다.

저작결과를 사전실행해 보고자 할 때는 메뉴바의 Run->Preview 메뉴를 선택하면 다른 브라우저창에 저작 결과가 실행되어 진다.



[그림 12] 저작결과물 사전실행  
[Fig. 12] pre-execution of development results

5. 결론

교육, 정보통신, 오락, 키오스크(kiosk)등 사회 모든 분야에서 멀티미디어 응용 분야의 확대와 이에 따른 멀티미디어 저작 시스템의 필요성이 매우 커졌다. 인터넷 인구의 폭발적인 증가로 웹상에서 사용 가능한 시스템의 필요성과 일반 사용자도 쉽게 사용할 수 있고 시스템과 자연스럽게 상호 대화하면서 편집할 수 있는 멀티미디어 저작 시스템이 요구되었다.

본 논문에서는 아이콘 기반의 사용자 인터 페이스를 제공하여 컴퓨터 프로그램 전문가가 아닌 일반 사용자도 간단한 조작으로 멀티미디어 문서를 생성하고 편집할 수 있으며, 프리젠테이션에 참여하는 미디어 객체들 사이의 관련성을 시간 관계와 공간 관계로 정의하고 이를 객체상태 테이블을 통하여 관리하므로 미디어 객체들 간의 동기와 사용자 상호작용에 의한 비동기를 효율적으로 지원할 수 있는 웹 기반 멀티미디어 저작 시스템을 구현하였다.

본 논문에서 구현한 저작 시스템의 특징을 요약하면 다음과 같다.

첫째, 아이콘을 기반으로 WYSIWYG(What You See Is What You Get)를 제공하는 멀티미디어 문서 편집 환경을 제공한다. 사용자는 복잡한 스크립트 언어를 사용하지 않고도 간단한 조작으로 멀티미디어 문서를 저작할 수 있다.

둘째, 클라이언트-서버 환경에서 사용할 수 있다. 사용자는 서버에 접속하여 저작 시스템을 다운 받아 멀티미디어 문서를 저작할 수 있고, 필요한 리소스 정보를 서버로부터 제공받을 수 있다.

셋째, 웹상에서 사용할 수 있도록 한다. 인터넷 사용자가 웹 브라우저상에서 멀티미디어 문서를 저작할 수 있고, 저작한 문서로 프리젠테이션을 실행할 수 있다.

넷째, 미디어 리소스 및 저작된 멀티미디어 문서의 사전실행 기능을 제공한다. 멀티미디어 문서 저작시 수시로 피드백 과정을 반복하며, 작업 내용을 즉시 확인할 수 있다.

다섯째, 프리젠테이션 진행 중 사용자가 미디어 객체의 실행 순서를 원하는 형태로 바꿀 수 있는 동기 및 비동기 지원 메커니즘을 지원한다.

여섯째, 미디어 객체들간의 관련성을 시간과 공간 관계로 정의하고, 이를 내부 구조로 정의한다. 미디어

어 객체들간에는 시간적, 공간적 관계를 가질 수 있으며 이를 내부에 트리 구조로 정의하여 저장하고 관리한다.

일곱째, 프리젠테이션에 참여하는 미디어 객체들은 객체 상태테이블을 통하여 관리한다. 각 미디어 객체들은 일정한 주기로 생성되고, 실행되며, 실행 후 휴면상태에 들어가는 과정을 테이블 정보로 관리하며, 이러한 객체상태정보는 스케줄러와 연동해서 사용한다.

본 논문의 연구결과는 웹 교육용 자료의 제작 및 원격 교육용 코스웨어 제작에 사용될 수 있고, 다양한 프리젠테이션 분야에 사용될 수 있다. 또한 홈페이지의 배너 제작 및 멀티미디어 동영상 메일, 연하장 제작에 사용될 수 있다.

본 논문의 향후 연구 내용은 다음과 같다.

첫째, 웹상에서 실행되는 대부분의 시스템이 갖는 문제점으로 실행 속도를 들 수 있다. 이는 실행 환경 측면에서 여러 가지를 해결해야 하는 데 특히 서버와 클라이언트간에 전송되는 다양한 데이터들을 어떻게 미리 효과적으로 처리하느냐 하는 것이 기술적으로 중요한 요소이다. 이를 위하여 통신 프로토콜의 연구와 시스템의 스케줄링 전략 프로그램을 연구해야 한다.

둘째, 저작환경은 수 많은 사용자의 요구들을 만족해야 한다. 미디어 리소스 측면에서는 본 시스템에서는 프리젠테이션 시 필요한 다양한 그래픽 컴포넌트를 개발하였지만 앞으로 추가 개발이 필요하고 아울러 다른 미디어 컴포넌트들도 연구 개발하여 다양한 리소스들을 제공해야 한다. 사용자 상호작용 측면에서는 프리젠테이션에 참여하는 객체들에 다양한 이벤트를 통해 실행 순서를 바꾸어 사용자의 요구를 만족할 수 있는 비동기 메커니즘이 제공되어야 한다.

셋째, 클라이언트가 윈도우에서 사용 가능하지만 앞으로 리눅스 환경에서도 사용할 수 있는 버전의 개발이 요구된다.

### ※ 참고문헌

- [1] Multimedia Authoring System FAQ, <http://www.tiac.net/users/jasiglar/MMASFAQ.html>.
- [2] Richard L. Phillips, "Media View", CACM, Vol. 34, No. 7, pp.75-83, July 1991.
- [3] Hui-Jung Chang, Tai-Yuan Hou, Arding Hsu, and Shi-Kuo Chang, "Tele-Action Objects for an Active Multimedia System", IEEE Multimedia Computing Systems, pp.106-113, 1995.
- [4] Ephraim P. Glinert and Steven L. Tanimoto, "Pict: An Interactive Graphical Programming Environment", IEEE Computer, Nov., 1984.
- [5] J. F. Koegel and J. M. Heines, "Improving Visual Programming Languages for Multimedia Authoring", EDMEDIA'93, pp.286-293, 1993.
- [6] J. Schnepf, J. Konstan, and D. Du, "Doing FLIPS: Flexible Interactive Presentation Synchronization", IEEE Journal on Selected Areas in Comm., pp.114-125, Jan., 1996.
- [7] G. Blakowski, J. Hubel, U. Langrehr, and M. Muhlhauser, "Tool Support for the Synchronization and Presentation of Distributed Multimedia", Computer Comm., Vol.15, No. 10, pp. 611-618, Dec., 1992.
- [8] B. Prabhakaran and S. V. Raghavan, "Synchronization Models for Presentation with User Participation", in Proc., ACM Multimedia '93, pp.157-165, 1993.
- [9] R. Steinmetz, K. Nahrstedt, Multimedia: Computing, Communications, and Applications, Prentice Hall, Chap. 13-18, 1995.
- [10] John F. Koegel Buford, Multimedia Systems, ACM Press, Addison-Wesley, New York, Chap.7,11,12, 1994.

- [11] M. Cecelia Buchanan, Polle T. Zellweger, "Scheduling Multimedia Documents Using Temporal Constraints", Workshop on Network & Operating Systems Supported for Digital Video and Audio. pp.237-249, Nov., 1992.
- [12] Linda Hardman, Dick A.A. Bulterman, and Guido van Rossum, "Structured Multimedia Authoring", CWI Report CS-R9304, 1993.
- [13] Guido van Rossum et al. "CMIFed: A Presentation Environment for Portable Hypermedia Documents", CWI Report CS-R9305, 1993.
- [14] Linda Hardman, Dick A.A. Bulterman, and Guido van Rossum, "The Amsterdam Hypermedia Model: Extending Hypertext to Support Real Multimedia", CWI Report CS-R9306, 1993.
- [15] Robert M. Akscyn, Donald L. Mccracken, and Elise A. Yoder, "KMS: A Distributed Hypermedia System for Managing Knowledge in Organizations", CACM, Vol.31, No.7, pp 820-835, July, 1988.
- [16] Michael Bleber and Tomas Isakowitz, "Applications", CACM, Vol.38, No.8, pp.27-29, Aug., 1995.
- [17] Shi-Kuo Chang, "Extending Visual Languages for Multimedia", IEEE Multimedia, pp. 18-26, 1996.
- [18] Brian D. Markey, "Emerging Hypermedia Standards Hypermedia Marketplace Prepares for HyTime and MHEG", USENIX-Summer'91, pp.59-74, 1991.
- [19] C. F. Goldfarb, "HyTime; A Standard for Structured Hypermedia Interchange", IEEE Computer, Vol.34, No.8, pp.81-84, 1991.
- [20] Petra Hoepner, "Synchronizing the Presentation of Multimedia Objects - ODA Extensions", SIGOIS bulletin, Vol.19, No.1, pp.19-32, 1991.
- [21] John F. Koegel, Lloyd W. Rutledge, John L. Rutledge, and Can Keskin, "HyOctane: A Hytime Engine for an MMIS", ACM Multimedia'93, pp.19-32, 1993.
- [22] Steven R. Newcomb, Neill A. Kipp, and Victoria T. Newcomb, "The HyTime", CACM Vol.34, No.11, pp.67-83, 1991.
- [23] ISO/IEC JTC 1/SC18/WG3/N, "HyperODA - Extensions for Temporal Relationships", Jan., 1993.
- [24] K. Fujikawa et al. "Multimedia Presentation System 'Harmony' with Temporal and Active Media", USENIX, Summer, pp.75-93, 1991.
- [25] Wolfgang Herzner and Erwin Hocevar, "CDAM - Compound Document Access and Management", "SIGOIS bulletin, Vol.19, No. 1, pp.1-18, 1991.
- [26] S. Christodoulakis, M. Theodoridou, F. Ho, M. Papa, and A. Pathria, "Multimedia Document Presentation, Information Extraction, and Document Formation in MINOS: A Model and a System", ACM Trans. On Office Information Systems, Vol.4, No.4, pp.345-383, 1986.
- [27] Chung-Ming Huang and Chung-Ming Lo, "An EFSM-Based Multimedia Synchronization Model and the Authoring System", IEEE Journal on Selected Areas in Comm., Vol.14, No.1, pp.138-152, Jan., 1996.
- [28] B. Prabhakaran, Multimedia Database Management Systems, Chap. 3-5, Kluwer Academic Publishers, 1997.
- [29] Hyun-san Shin, Yoo-jae Won, Woo-jong Yoo, Kwan-Jong Yoo, "Presentation Jockey: Multimedia Authoring System for Automatic Presentation", IASTED Applied Informatics'97, pp. 259-262, Feb., 1997.

- [30] Kathy Walrath, Mary Campione, "The JFC Swing Tutorial: A Guide to Constructing GUIs", Addison-Wesley Pub Co, Jul., 1999
- [31] Mary Campione, Kathy Walrath, Alison Huml, Tutuorial Team, "The Java (TM) Tutorial Continued: The Rest of the JDK(TM): The Java Series", Addison-Wesley Pub Co, Dec., 1998
- [32] Mary Campione, Kathy Walrath, "The Java Tutorial Second Edition: Object-Oriented Programming for the Internet (Java Series)", Addison-Wesley Pub Co, Mar., 1998
- [33] Andrzej Duda, Cherif Keramane, "Structured Temporal Composition of Multimedia Data", International Workshop on Multimedia Database, pp. 136-142, Aug. 1995.
- [34] Eitzman, K.Wittenburg, "Automatic Presentation of Multimedia Documents using Relational Grammars", Multimedia'94, pp. 443-451, 1994.
- [35] Y. Kim, J. Song,, "Multimedia Documents with Elastic Time", ACM Multimedia'95 pp. 143-154, 1995.
- [36] Thomas D. C. Little, A. Ghafoor, "Spatio-Temporal Composition of Distributed Multimedia Objects for Value-Added Networks", IEEE Computer, pp. 42-50, 1991.
- [37] ISO/IEC JTC 1/SC18/WG3/N, "HyperODA-Extensions for Temporal Relationships" Jan., 1993.
- [38] Hyun-San Shin, Woo-Jong Yoo, and Kwan-Jong Yoo, "Presentation Jockey: Multimedia Authoring System using Structured Document Parsing", HCI'97 Design of Computing System", Aug., 1997, pp. 659-662.
- [39] 신현산, "동적 저작 모델에 기반한 통합 멀티미디어 저작 시스템", 충남대학교 대학원, 박사학위 논문, 1998. 8.
- [40] 신현산, 유관중, "멀티미디어 문서저작 통합 시스템 Presentation Jockey 설계 및 구현", 정보과학회 추계 학술발표논문, 제24권 2호, pp.343-346, 1997.
- [41] 송한영, 신현산, 유대상, 김신형, 전찬원, 장종환, "웹 환경의 교육용 프리젠테이션 저작 도구에 관한 연구", 제16회 산학연 멀티미디어 산업기술 학술대회, pp.157-161, 2000. 11.
- [42] 황병선, 김성겸, 이정현, "확장성 있는 대화형 멀티미디어 저작도구를 위한 개념적 모델의 설계 및 구현", 한국정보과학회 가을 학술발표논문, 제20권 제2호, pp.1225-1228, 1993.
- [43] 이범렬, 송경준, 이근영, 양광호, 최락만, "멀티미디어 저작 시스템 개발", 한국정보과학회 가을 학술발표논문, 제20권, 제2호, pp.1217-1220, 1993.
- [44] 차재혁, 이석호, "멀티미디어 DBMS를 위한 사용자 인터페이스", 정보과학회지, 제14권, 제9호, pp.22-30, 1996. 9.
- [45] 멀티미디어교육지원센터, "교육용 멀티미디어 저작 도구 PASS", 2000.

박 영 미



1988년 한양대학교  
전자공학과 졸업 (학사)  
1990년 한양대학교  
전자공학과 졸업 (석사)  
1990 ~ 1992 오텔코  
(현재 한국화약) 중앙연구소  
연구원  
1992 ~ 1997 한국전자통신  
연구원 선임연구원  
1997 ~ 현재 배재대학교  
정보통신공학 배재대학교  
정보통신공학과 시간강사  
2001 ~ 현재 박사과정  
관심분야 : 영상처리,  
멀티미디어

장 중 환



1979년 - 한양대학교  
전자통신공학과 졸업 (학사)  
1986년 - North Carolina  
주립대학, 전기 및  
컴퓨터공학과 (공학석사)  
1990년 - North Carolina  
주립대학, 전기 및  
컴퓨터공학과 (공학박사)  
  
1995년 - 현재 (사)  
한국정보처리학회  
멀티미디어시스템 연구회  
운영위원  
1995년 - 현재  
(사) 한국통신학회  
대전충남지부 이사  
1998년 - 현재 정보통신부  
지정 정보통신창업지원센터  
소장  
1999년 - 현재 대전광역시  
최고정보화 책임관 (CIO)  
1999년 - 현재 대전광역시  
과학기술위원회 전문위원  
1999년 - 현재  
한국소프트웨어진흥원  
대전S/W지원센터 운영위원  
2000년 - 현재  
충청상거래지원센터  
기획운영위원  
2000년 - 현재 배재대학교  
공과대학장  
1990년 - 현재 배재대학교  
컴퓨터전자정보통신공학부  
교수  
1992년 - 1996년 배재대학교  
전자계산소장  
관심분야 : 멀티미디어검색,  
비디오편집