

삼차원 시·공간을 이용하는 프레젠테이션 공동저작 시스템

(Collaborative Authoring System using 3D Spatio-Temporal Space)

이도형[†] 성미영^{**}
(Do Hyung Lee) (Mee Young Sung)

요약 본 논문에서는 시간 편집 환경과 공간 편집 환경을 하나의 삼차원 저작 환경으로 통합한 멀티미디어 프레젠테이션 공동 저작 시스템을 소개한다. 본 저작 시스템은 하나의 삼차원 좌표계에서 멀티미디어 프레젠테이션의 공간 속성과 시간 속성을 동시에 표현한다. 한 개의 축은 일반적인 시간 정보(T-zone)를 표현하고 나머지 두개의 축은 공간 정보(XY-zone)를 나타내며 시간적인 미디어 객체들은 삼차원 육면체로 표현하고 오디오 객체는 원통 모양으로 표현하고 있으므로, 사용자들이 자유자재로 삼차원공간을 이동하며 시간 편집과 공간 편집을 할 수 있어 효율적이다.

또한 본 논문에서는 제안하는 공동 저작 시스템에서 생성되는 공유 객체들에 대한 효율적인 동시성 제어 기법을 설계하고 구현하였다. 구현된 동시성 제어 기법은 다중버전 기법(multiple versions)을 가시화하여 공동 저작에서의 사용자 인지(user awareness)를 높였으며 공유 객체에 대한 접근허가권(access permissions)을 이용하여 네트워크 통신에서의 지연이나 장애로 발생할 수 있는 충돌(collision)을 최소화하였다. 그리고 이 동시성 제어 기법은 잠금 단위를 세밀하게 하여 사용자의 편집 자유도를 최대한 보장하였으며, 공유 객체의 일관성을 유지하면서 낙관적 동시성 제어(optimistic concurrency control)를 지원하므로 사용자들에게 멀티미디어 프레젠테이션을 동시 편집할 때 빠른 응답 시간을 제공할 수 있다.

키워드 : 공동저작, 삼차원, 시·공간 관계, 동시성 제어, SMIL

Abstract In this paper, we propose a collaborative multimedia authoring system. Our authoring system represents a multimedia presentation in a 3D coordinate system. One axis represents the traditional timeline information (T-zone), and the other two axes represent spatial coordinates (XY-zone). Our system represents a visual media objects as a 3D parallelepipeds and audio media objects as cylinders. This interface allows for simultaneous authoring and manipulation of both the temporal and the spatial aspects of a presentation. Using our system, users can design multimedia presentations collaboratively in the unified spatio-temporal space while freely traversing the spatial domain and the temporal domain without changing the context of authoring.

In addition, we suggest an efficient mechanism of concurrency control for shared objects generated by our collaborative writing system. The mechanism is mainly based on the user awareness, the multiple versions, and the access permission of shared objects. Our concurrency control mechanism is designed to keep data consistency by minimizing the collision due to the delay or the failure of network communication and to allow maximum responsiveness for users using optimistic concurrency control. Also, the mechanism maximize the responsiveness by refining the locking granularity and applying different concurrency control mechanisms to each.

Key words : Collaborative Authoring, Three dimension, Spatio-Temporal relation, concurrency control, SMIL(Synchronized Multimedia Integration Language)

· 본 연구는 한국과학재단 지정 인천대학교 멀티미디어연구센터의 지원에 의한 것입니다.

† 비회원 : 인천대학교 컴퓨터공학과
oldkill@incheon.ac.kr

** 종신회원 : 인천대학교 컴퓨터공학과 교수
mysung@incheon.ac.kr

논문접수 : 2003년 5월 22일

심사완료 : 2003년 9월 5일

1. 서론

멀티미디어 저작 시스템은 시간관계와 공간관계를 동시에 편집할 수 있는 환경을 제공해야 한다[1-4]. 또한, 공동 멀티미디어 저작시스템은 사용자 상호작용을 지원할 필요가 있다[1-3]. 비디오, 사운드, 애니메이션과 같

은 미디어는 시간적 관계 표현이 필요하고, 비디오, 이미지, 텍스트와 같은 미디어는 공간적 관계 표현이 요구된다[2,5]. 많은 개발자들이 이차원 데스크 탑 메타포어를 대체할 수 있는 삼차원 상호작용 메타포어를 찾으려고 노력하면서 인간과 컴퓨터 상호작용을 재정의 하고 있다. 또한, 머지않아 컴퓨터를 사용하는 기본환경이 현재의 이차원 데스크 탑 환경에서 삼차원 가상공간으로 옮겨갈 것이다[6]. 결국 네트워크화 된 삼차원 가상공간에서 사용자들은 토론 및 의사소통 등의 공동작업을 위하여 모이게 될 것이며, 이 때 가능한 공동작업 중 하나인 프레젠테이션 공동저작에 대한 요구가 증가함에 따라 삼차원 가상공간에서 공동저작을 잘 지원해 주는 시스템이 절실히 필요하게 될 것이다[7]. 그래서 본 연구에서는 이와 같은 요구가 발생할 것에 대비하여 삼차원 시·공간을 이용하는 멀티미디어 공동저작 시스템을 개발하게 되었다.

프레젠테이션 저작의 핵심은 객체 사이의 공간관계 조합과 시간관계의 조합에 있다. 기존의 저작도구는 보통 공간정보와 시간정보가 두 개의 서로 다른 이차원 그래픽 인터페이스에서 독립적으로 편집되는 환경을 제공한다. 그러나 이러한 서로 다른 두 개의 이차원 그래픽 인터페이스(공간 인터페이스 및 시간 인터페이스)를 지원하는 저작 환경은 두 개의 인터페이스를 빈번하게 이동하며 저작을 해야 하므로 사용자를 불편하게 한다. 또한 분리된 두 개의 인터페이스 각각은 비디오와 애니메이션 등 시간 속성과 공간 속성을 동시에 가지고 있는 미디어의 속성을 완벽하게 표현할 수 없다.

본 연구에서는 미디어 데이터가 출력되는 이차원 공간에 시간 차원을 추가한 삼차원 시·공간 영역 안에 공간 저작 인터페이스와 시간 저작 인터페이스를 동시에 표현하고자 한다. 이렇게 함으로써 본 연구에서 제안하는 멀티미디어 저작 시스템에는 삼차원 인터페이스의 장점을 충분히 활용하여 시간과 공간 정보를 하나의 연속적인(seamless) 삼차원 공간에서 동시에 표현하고 편집할 수 있다. 본 시스템은 통합 스트리밍 미디어를 위한 XML 기반의 마크업 언어인 SMIL(Synchronized Multimedia Integration Language)[8,9]을 사용한다. 또한, 인터넷의 발달로 인하여 사용자들은 원거리의 사용자와의 토론식 공동저작에 대한 욕구가 증가되는 추세이다. 그러므로 효과적인 멀티미디어 공동저작 시스템 및 효율적인 동시성 제어방법에 대한 구체적인 연구가 더욱 필요한 실정이다.

본 논문에서는 시간 편집 환경과 공간 편집 환경을 하나의 삼차원 저작 환경으로 통합한 멀티미디어 프레젠테이션 공동 저작 시스템을 소개한다. 본 논문의 초점은 멀티미디어 프레젠테이션의 공동저작을 위한 삼차원

시·공간 인터페이스이다. 본 시스템의 저작 인터페이스는 다른 시스템에서는 찾아볼 수 없는 고유한 방법이다. 본 시스템은 삼차원 시·공간 편집기(3D Spatio-Temporal Editor)와 태그 편집기(Tag Editor), 속성 편집기(Attribute Editor), 텍스트 편집기(Text Editor), 미디어 객체 관리자(Media Object Manager)를 포함하고 있다. 미디어 객체 관리자는 삼차원 공간에 미디어 객체를 시각화시키고, 프레젠테이션의 일관성을 유지시키며 모든 수정 정보를 한 편집기에서 다른 편집기로 전달한다. 이러한 미디어 객체 관리자의 기능을 위해서는 프레젠테이션의 구조를 관리할 내부 자료구조가 필요하게 되는데, 기존 연구에서 발표된 통합 멀티미디어 데이터의 내부 자료구조에는 OCPN (Object Composition Petri Nets)[10], DTPN (Dynamic Timed Petri Nets)[11], XOCPN (eXtended OCPN)[12], TCN (Temporal Constraint Network)[1] 등이 있다. DTPN 과 XOCPN은 OCPN의 변형된 형태로 페트리 넷(Petri Net)에 기반하고 있고, 그 인터페이스는 눈금화된 타임라인(scaled time-line)을 바탕으로 하고 있다. TCN은 비순환 그래프(Directed Acyclic Graph) 구조이며, 그 인터페이스는 신축적 시간 모델(Elastic Time Model)을 표현하는 시간 상자(time-box)에 기반한다. 본 연구에서는 TRN(Temporal Relation Network)[13,14]이라는 멀티미디어 프레젠테이션의 내부 표현을 사용한다. 또한 본 논문에서는 멀티미디어 공동저작을 위한 동시성 제어 방법으로 낙관적인 잠금을 이용하는 다중버전 기법을 제시한다. 이 다중버전 기법은 허가권 얻기를 통하여 충돌을 최소화하는 한편, 동시 접근에 대한 사용자 인식을 이끌어 내는 방법이다. 이 방법은 전통적인 동시성 제어 기법인 실행 취소(reversible execution 또는 undo) 기법[15]이 적용되는 일련의 과정을 사용자가 인식할 수 있도록 그래픽하게 가시화한 것이라고 할 수 있다.

본 논문의 2절에서는 삼차원에 시·공간 관계를 표현하는 메커니즘을 설명한다. 3절에서는 본 시스템의 내부 표현으로 제안하는 TRN의 자동생성에 대해 기술한다. 이 연구에서 제안하는 동시성 제어 모델은 4절에서 논의한다. 5절에서는 이 시스템의 동시성 제어 기법을 다른 기법들과 비교 분석해 보고, 6절에서는 본 시스템의 구현에 대해 알아본다. 마지막 절에서는 결론과 앞으로의 과제를 제시한다.

2. 삼차원 공간에서 시간과 공간 표현

본 저작 시스템은 하나의 삼차원 좌표계에서 멀티미디어 프레젠테이션을 표현한다. 한 개의 축은 일반적인 시간 정보(T-zone)를 표현하고 나머지 두 개의 축은 그

림 1에서 보는 것처럼 공간 정보(XY-zone)를 나타내고 있다. 본 시스템에서는 시간과 공간 정보를 동시에 표현해야 하는 미디어 객체들은 삼차원 육면체로 표현하고 오디오 객체와 같은 시간정보만을 표현하게 되는 미디어 객체들은 원통 모양으로 표현하고 있다. 시간 축에 따른 길이는 미디어의 재생시간(duration)에 해당하며, XY 축으로 본 육면체의 크기는 그 시각 미디어의 공간 영역을 보여주는 것이다.

그림 1, 그림 2, 그림 3은 사용자가 의도한 표 1과 같은 멀티미디어 프레젠테이션을 본 저작 시스템에서 표현한 예이다. 이 프레젠테이션은 오디오 1개, 비디오 2개, 이미지 2개, 텍스트 2개의 총 7개의 미디어 객체로 구성되어 있다. 시간축(T) 흐름에 따라 공간(X-Y)에 미디어들이 재생되는 내용을 편집한 것으로 저작자들은 이러한 미디어 객체를 원하는 위치에 생성시키고 마우스 드래그를 이용하여 객체의 재생시간의 길이나 공간적 영역을 변경할 수 있다.

그림 2는 그림 1에서 제시된 예를 시간 관점에서 바라본 것이고, 그림 3은 공간 관점에서 바라본 것이다. 저작자들은 관점 변경 버튼을 이용하여 각 관점으로 신속하게 이동할 수 있다. 저작자들은 자신의 컴퓨터에 있는 파일이나 네트워크 파일 객체를 삼차원 시·공간으로 드래그와 드롭 함으로써 원하는 위치에 객체를 생성

표 1 예제의 미디어들의 시간 속성

파일이름	시작시간(sec)	재생시간(sec)
smil.mpg	2	23
smil2.mpg	0	50
smil.jpg	21	28
smil2.jpg	50	35
smil.mp3	50	60
smil.txt	65	35
smil2.txt	50	20

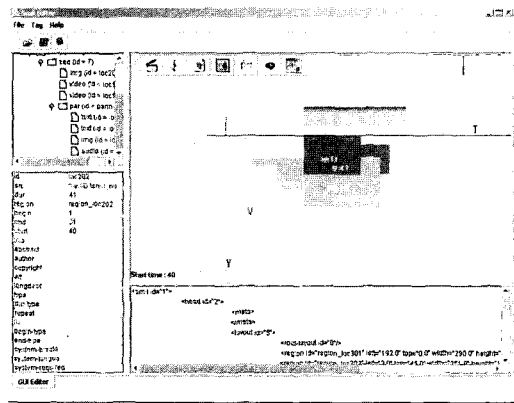


그림 2 그림 1 예의 시간적인 관점

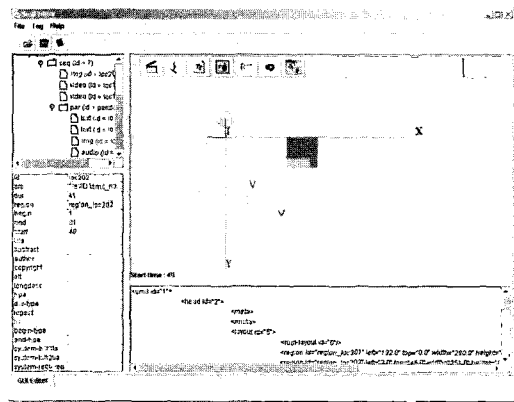


그림 3 그림 1 예의 공간적인 관점

할 수 있다. 또한, 삼차원 시·공간 영역의 모든 물체는 시간적인 관점뿐만 아니라 공간적인 관점에서 확대, 축소, 재배치될 수 있다. 또한 삼차원의 장점인 시점 자유변환을 통하여 시간과 공간의 관계를 한눈에 알아볼 수 있어 직관적인 인터페이스를 지원하며 저작자가 시간적 관계를 직접 설정할 필요가 없어(TRN 자동생성에 의하여) 편리한 인터페이스를 제공한다. 본 시스템은 이와 같이 삼차원 시·공간 인터페이스를 통하여 멀티미디어 프레젠테이션을 저작하는 사용자에게 보다 쉽고 직관적인 인터페이스를 제공한다.

그림 4는 그림 1에 나타나 있는 것처럼 삼차원 인터페이스를 이용하여 편집된 멀티미디어 프레젠테이션의 내용이 SMIL 코드로 변환된 것이다. SMIL 태그는 <head>에 공간정보를 표현하며 <body>부분에 미디어 객체의 정보와 미디어들 간의 시간관계가 표현된다. 그림 1, 그림 2, 그림 3을 보면 시간의 흐름에 따라 비디오 미디어객체 2개, 이미지 미디어객체 1개가 일정한 간격의 시작시간을 가지며 병렬관계(parallel)를 가지고 있

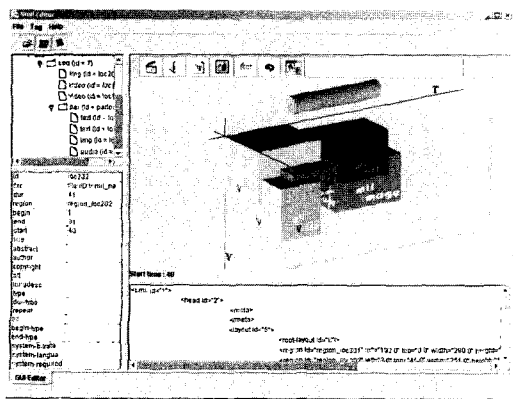


그림 1 멀티미디어 프레젠테이션의 한 예

```

<smil id="1">
  <head id="2">
    <meta>
    </meta>
    <layout id="5">
      <root-layout id="6"/>
      <region id="region_loc101" left="0.0" top="0.0" width="254.0" height="275.0"/>
      <region id="region_loc107" left="192.0" top="150.0" width="299.0" height="245.0"/>
      <region id="region_loc201" left="182.0" top="111.0" width="218.0" height="245.0"/>
      <region id="region_loc301" left="138.0" top="84.0" width="257.0" height="239.0"/>
      <region id="region_loc304" left="0.0" top="81.0" width="308.0" height="143.0"/>
      <region id="region_loc213" left="24.0" top="210.0" width="383.0" height="173.0"/>
    </layout>
  </head>
  <body id="3">
    <seq id="7">
      <par id="parloc301">
        <video id="loc101" src="file://D:\smil2.mpg" dur="50" region="region_loc101" begin="0"/>
        <video id="loc107" src="file://D:\smil.mpg" dur="23" region="region_loc107" begin="2"/>
        
      </par>
      <par id="parloc302">
        <audio id="loc401" src="file://D:\smil.mp3" dur="60" region="" begin="0"/>
        <text id="loc301" src="file://D:\smil.txt" dur="35" region="region_loc301" begin="15">
        </text>
        <text id="loc304" src="file://D:\smil2.txt" dur="20" region="region_loc304" begin="0">
        </text>
        
      </par>
    </seq>
  </body>
</smil>

```

그림 4 편집기를 통해 생성된 SMIL 코드

는 것을 확인할 수 있고 다시 오디오 미디어 객체 1개, 이미지 미디어 객체 1개, 텍스트 미디어 객체 2개가 병렬관계를 가지고 있는 것을 확인할 수 있다. 병렬관계를 가지는 두 개의 미디어 집합은 하나의 연속적인(sequential) 관계를 가지는데, 그림 4의 SMIL 코드와 그래픽 표현이 정확하게 일치함을 확인할 수 있다.

멀티미디어 프레젠테이션이 저작자에 의해 단계적으로 디자인됨에 따라 그에 해당하는 SMIL 코드가 실시간으로 생성되고 생성된 SMIL 코드는 태그 편집기상에 출력된다[16].

3. 삼차원 그래픽 표현으로부터 TRN 자동 생성

멀티미디어 프레젠테이션 저작 과정은 콘텐츠 편집을 위한 일련의 사용자 상호 작용으로 구성된다. 즉, 사용자의 입력에 대해 시스템은 즉각적인 반응과 적당한 피드백을 제공하여야 한다. 이를 위하여 프레젠테이션의 상태를 별도의 내부 자료구조로 지속적으로 유지하고 관리할 필요가 있다[1,11,12,17]. 본 연구에서 사용하는 멀티미디어 프레젠테이션의 내부 표현은 본 연구의 이전 연구에서 제안한 TRN이다. TRN 구조는 멀티미디어 프레젠테이션의 개념적 시간구조와 정확하게 일치한다[14].

본 절에서는 프레젠테이션 저작자에 의해 제작된 삼차원 그래픽 표현으로부터 TRN을 자동생성할 수 있는 알고리즘에 대해서 간략하게 설명한다. 이 알고리즘은 세 개의 모듈, build_TRN, insertSeqNode(), insertPar()로 구성되어 있다.

Build_TRN은 실제 문서구조의 모든 노드를 순회하는 기능을 담당한다. 이 알고리즘에서 각각의 구성모듈은 문서구조의 직접 또는 순차 순회를 하는데 필요한 모든 메소드, 예를 들어 getNextSibling, getChildNode, getChildNodes, getParentNode 등을 포함한다.

insertSeqNode()은 매개변수로 전달받은 node의 상속성을 이용해 미디어 객체를 생성하고 필요에 따라 지연객체를 추가적으로 생성하여 TRN에 삽입한다. insertSeqNode()에서는 지연객체의 재생시간에 따라 순차적 시간관계인 'meets'와 'before' 중 하나의 시간관계를 설정한다.

insertPar() 모듈은 객체들의 병렬관계('equals', 'starts', 'finishes', 'during', 'overlaps' 등[14])를 다루기 위한 것이다. 모든 병렬 관계는 하나의 객체로 축약될 수 있다. 병렬관계로 연결되어 있는 객체들의 그룹을 병렬 블록(parallel block)이라 부른다. insertPar()는 다음과 같은

작업을 수행한다.

- ① par 노드의 자식 노드 수를 알아낸다.
- ② par노드 전체시간을 산출한다.
- ③ par블록과 par 노드 안의 미디어 노드들의 관계를 설정한다.
- ④ 만약 자식 노드의 수가 둘 이하일 경우에는, 블록의 전체시간과 각 미디어의 시간 정보를 가지고 두 미디어의 시간관계를 산출하여 각 미디어 객체를 TRN에 삽입하고 시간관계를 설정한 다음 종료한다.

InsertPar()는 각 노드들에 대해서 재귀적으로 수행하여 TRN에 객체를 삽입한다. 또한, 이 알고리즘은 M이 속성수이고 N이 TRN에 있는 노드 수일 때 O(MN)의 시간이 걸린다. O(N)시간 안에 문서 트리를 순회하고, 각각의 노드들에 대해서 O(M)시간이 걸리는 insertObject()를 호출한다. 실제로, 트리를 순회하는데 필요한 가장 좋은 알고리즘은 O(N)의 시간이 걸리며 데이터 구조 자체는 O(N)만큼의 기억공간에 저장되어야 한다.

4. 동시성 제어

모든 공동작업 시스템에서는 사용자의 동시작업 사이에 충돌을 해결하기 위해 동시성 제어가 필요하다. 동시성 제어 기법에 대하여 데이터베이스 분야에서 오랜 세월 동안 많은 연구가 있어 왔다[21]. 그러나 이들 기법을 공동작업에 그대로 적용하기에는 한계가 있다[29]. 왜냐하면 공유 데이터에 대한 추가, 삭제, 변경, 읽기, 쓰기 등 다양한 연산이 동시 다발적으로 발생하고 이를 실시간으로 사용자에게 보여줘야 하는 공동작업 시스템에서의 공유 데이터에 대한 연산이 읽기와 쓰기 작업만 있는 데이터베이스 시스템에서의 연산보다 훨씬 복잡하기 때문이다. 데이터베이스 분야의 동시성 제어 기술을 바탕으로 하여 연구되어 온 공동 작업 분야의 전통적인 동시성 제어 기법에는 단순 잠금(simple locking) 및 잠금 객체 가시화[22], 낙관적 잠금[23], 트랜잭션 순서화(transaction serialization)[24], 낙관적 트랜잭션 순서화[25], 순서 규약(turn-taking protocols)[26], 의존관계 검사(dependency detection)[21], 역 실행(reversible execution; undoing)[18,27], 연산 변환(operation transformation)[28], 다중버전(multiple versions)[30]과 같은 많은 메커니즘이 있다. 그러나 특정 공동 작업 응용 시스템에서 동시성 제어를 위한 가장 좋은 알고리즘을 찾는 것은 쉬운 일이 아니며 응용의 의미(semantics), 시나리오, 지연 시간, 사용자의 상호작용 모델의 반응 방식(feedback mechanism) 등을 신중하게 검토해서 결정해야 한다[22].

그 동안 여러 공동작업 시스템 개발을 수행해 온 경

험으로 볼 때, 멀티미디어 프레젠테이션 공동저작에 대한 동시성 제어에 대한 완벽한 해답은 없으며 아래의 조건을 충족시킬 수 있다면 만족스러운 결과를 이끌어 낼 수 있다는 결론을 내리게 되었다[19].

- 공동작업에 대한 사용자의 인식
- 미세한 공유 단위 제공
- 시스템 반응성(responsiveness) 극대화
- 충돌(collision)을 최소화

본 연구에서는 위의 조건들을 만족시킬 멀티미디어 프레젠테이션 공동저작 시스템의 효과적인 동시성 제어를 위하여 다음과 같은 방법들을 복합적으로 사용한다.

• 공동 접근 객체를 투명하게 표현해줌으로써 계속되는 모든 동시 접근을 인식하게 한다. 이 방법은 WYS-IWYS(What You See Is What I See)[22] 개념을 모든 동시 접근 연산들에도 적용하여 실시간으로 충돌 상황과 내용을 알려주는 접근이다.

• 같은 객체에 대한 작업이라도 시간편집 작업과 공간편집 작업을 서로 다른 별 개의 작업으로 간주하여 동시성 제어 기법을 적용함으로써 잠금의 대상이 되는 단위를 최소한으로 줄인다.

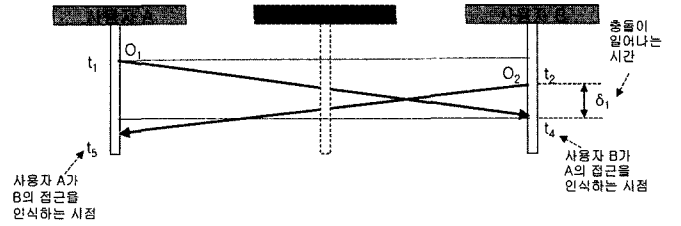
• 동시 접근이 일어난 상황에서 일어난 서로 다른 사용자의 저작 내용을 서로 다른 버전으로 유지하여 가시화하고 최종적으로 하나의 버전으로 통합하는 다중버전(multiple versions) 기법을 지원하여 시스템 반응성(responsiveness)을 최대화한다. 이 때, 하나의 버전으로 통합하는 일은 사용자들이 스스로 조정하는 가운데 양보하여 하나의 버전만 남도록 한다. 서로 양보하지 않는 경우에는, 공유 객체의 편집을 최후에 해제(release)하는 사용자의 버전으로 통합된다.

• 사용자로 하여금 접근허가권(access permission)을 얻은 후 편집하도록 제약함으로써 동시 조작에 따른 충돌(collision)을 최소화한다.

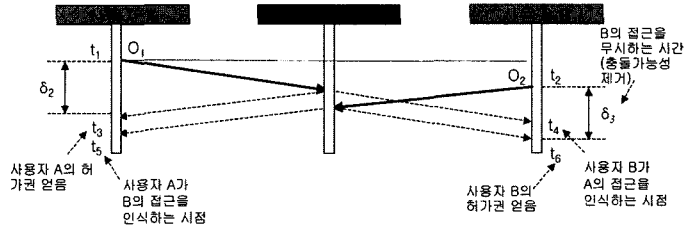
그림 5는 본 연구에서 설계한 동시성 제어 기법 중 네트워크 지연시간에 의해 예상되는 충돌을 막기 위해 접근허가권을 이용하는 동시성 제어 기법의 구체적인 내용을 보여주고 있다.

O1 과 O2는 공유객체에 대한 두 사용자의 공동 작업을 보여주고 있다. 그림 5에서 시간의 위치와 시간간격은 다음과 같다.

- T1: O1 시작
- T2: O2 시작
- T3: A의 접근권한 도착
- T4: B가 동시접근 인식(t2에서 t4까지 B는 블록된 상태임)
- T5: A가 동시접근 인식
- T6: B가 A의 접근인식



(a) 단순한 잠금기법을 이용한 동시성 제어 방법



(b) 허가권 얻기를 이용한 동시성 제어 방법

그림 5 접근허가권을 이용하는 동시성 제어

```

mediaEditStart(Object mediaObject) // 편집 시작 (편집시작 메시지 전달시 호출)
begin
    // 미디어객체에 대한 허가권 얻기 및 복사 미디어 객체 cp_mediaObject 생성
    cp_mediaObject ::= getPermission(mediaObject);
end

mediaEditing(Object cp_mediaObject) // 편집 중 (편집되는 과정의 메시지 전달시 호출)
begin
    // 복사 미디어객체에 대해 사용자 편집에 따른 편집적용
    // 복사 미디어객체의 편집상황 표현 및 타 사용자에게 편집상황 전달
    .....
end

mediaEditEnd(Object cp_mediaObject) // 편집 종료 (편집종료 메시지 전달시 호출)
begin
    // 원본 미디어객체의 타사용자의 편집상황 확인
    if (getMediaStatus(cp_mediaObject))
        // 사용자의 편집 적용: 복사된 투명 미디어 객체를 원본 객체로 대체
        mediaEditApply(cp_mediaObject);
    else
        // 사용자의 편집 취소: 복사된 미디어 객체 삭제
        mediaEditUndo(cp_mediaObject);
    endif
end

getMediaStatus(Object cp_mediaObject) // 원본미디어 객체의 공유 접근 여부 확인
begin
    // 원본미디어 객체 접근 수가 1명이 아닐 경우 취소처리 반환 (false)
    if (cp_mediaObject.orginal_mediaObject.cp_count != 1)
        return false;
    // 원본미디어 객체 접근 수가 1명(클라이언트 자신의 접근)일 경우 편집 적용처리 반환 (true)
    else if (cp_mediaObject.orginal_mediaObject.cp_count == 1)
        return true;
end
    
```

그림 6 동시성 제어관련 이벤트 처리 알고리즘

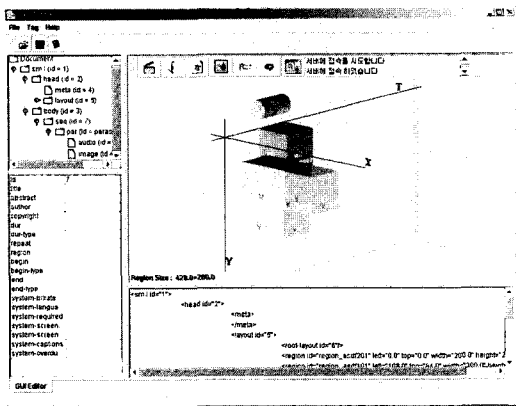


그림 7 공동저작의 한 예

- 81: B가 접근을 요구하고 이미 A에 의해 접근되고 있음을 깨닫는 시간(충돌이 일어날 수 있음)
- 82: A가 허가권을 받는 시간(블록된 상태임)
- 83: B가 허가권을 받는 시간(블록된 상태임)

본 연구에서 제안하는 기법은 동시성 제어기법의 분류 중 낙관적 잠금(locking) 기법에 속한다. 만약 하나의 미디어 객체에 다수의 사용자들이 동시접근을 할 경우 각 클라이언트 프로세스들은 그에 대응하는 미디어 객체의 복사본(clone)을 생성하여 작성을 시작하게 된다. 낙관적인 잠금에서 작업이 완료될 때까지 권한을 얻지 못하는 사용자는 'undo'(실행취소) 되어버린다. 본 시스템에서는 다중버전 기법을 이용하여 동시작업을 표시하여 줌으로써 사용자들의 동시 접근에 대한 인식을 높여주며 그에 따른 타협 아래 최종 편집이 수행될 수 있다. 그림 6은 본 시스템의 동시성 제어관련 이벤트 처리 알고리즘을 요약한 것이다.

우선적으로, 모든 사용자는 그림 5에서 살펴 본 것처럼 네트워크 지연시간에 의해 예상되는 충돌을 막을 수 있도록 편집할 대상 객체에 대하여 접근허가를 얻는다. 그림 7에서 보는 바와 같이 접근허가권에 따라 수정되는 모든 객체들에 대해 복사본(반투명하게 표현)을 생성한다. 이 때, 이후 작업에서 발생할 수 있는 실행취소 등의 동시성 제어를 위해 원본 객체를 남겨둠으로써 다른 사용자의 접근에 대한 불만을 제거할 수 있다. 다른 사용자의 두 번째 접근은 접근허가권에 따른 또 다른 복사본을 생성하게 된다. 이렇게 각 사용자의 접근에 따라 생성된 복사본들은 사용자의 인식을 높여주며 서로의 수정에 대해 타협을 할 수 있게 해주기 때문에 충돌을 최소화하며 동시성 제어를 할 수 있게 해준다.

5. 시스템 구현

상호작용의 멀티미디어 저작을 위한 삼차원 인터페이

스는 멀티미디어 정보를 분석하고 삼차원 인터페이스에 적절한 정보를 전달하기 위하여 삼차원 엔진과 실시간 제어 모듈이 필요하다. 효율적인 저작환경을 제공하기 위하여 다양한 편집도구 역시 필요하다. 본 시스템의 전체 구성은 그림 8과 같으며 크게 편집 시스템(Editing System), 미디어 객체 관리자(Media Object Manager; MOM), 그리고 협동 관리자(Collaboration Manager) 부분으로 나뉜다. 시스템의 내부에는 멀티미디어 프레젠테이션을 효과적으로 표현하기 위하여 TRN이 존재하며 시스템의 각 모듈은 이 TRN 구조를 기반으로 하여 상호 통신한다. TRN 모듈에는 삼차원 인터페이스에서 저작된 내용을 TRN 구조에 삽입하기 위하여 삼차원 공간 파서(3D Space Parser)가 존재한다. 삼차원 인터페이스에 의해 입력된 미디어 정보는 시간과 공간의 관계를 포함하지 않기 때문에, 이 삼차원 파서는 편집된 미디어들의 시간정보와 공간의 정보를 비교하여 논리적인 시간관계와 공간관계를 구성한다. 또한, 스마일 파일을 생성하고 스마일 코드를 화면에 보여주기 위하여 XML 기반 문서를 쉽게 처리할 수 있는 DOM(Document Object Model) [20] 구조를 사용하였다.

구현은 자바 기술을 이용하였다. 기본적으로 J2SDK1.4을 사용하였고, 삼차원 인터페이스 구현은 J3DSDK 1.3을 사용하여 구현하였다. 공동저작을 위한 통신은 자바 소켓(Java Socket)을 사용하여 구현하였다.

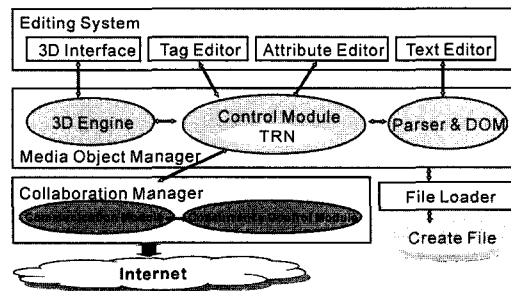


그림 8 전체 시스템 구성도

5.1 편집 시스템(Editing System)

편집 시스템에는 미디어들의 시·공간 관계를 표현할 수 있는 주 편집 환경인 삼차원 편집기(3D Spatio-Temporal Editor) 외에도 태그 편집기(Tag Editor), 속성 편집기(Attribute Editor), 그리고 텍스트 편집기(Text Editor)가 있다. 이들 편집기는 미디어 객체 관리자를 통하여 정보를 교환하고 서로 협동하여 사용하기 편리하고 효율적인 저작 환경을 제공한다. 편집 시스템의 사용자 인터페이스는 그림 1, 그림 2, 그림 3, 그리고 그림 7에서 보여주고 있다. 각 편집기의 역할을 아래에

정리해 본다.

- 삼차원 시·공간 편집기: 드래그 & 드롭 기능을 지원하여 자신의 컴퓨터에 있는 파일이나 네트워크 파일 객체를 삼차원 시·공간으로 드래그&드롭 함으로써 삼차원 공간의 원하는 위치에 객체를 생성할 수 있게 한다. 또한, 마우스 드래그를 이용하여 삼차원 시·공간 영역의 모든 미디어 객체의 시간 또는 공간 차원에서의 확대, 축소, 그리고 재배치가 가능하다. 멀티미디어 프레젠테이션이 단계적으로 저작됨에 따라 그에 해당하는 SMIL 코드가 실시간으로 생성되고 생성된 SMIL 코드는 태그 편집기상에 출력된다.
- 태그 편집기: 각 편집기를 통해 저작되어 자동 생성된 SMIL 코드를 트리 형식으로 관리할 수 있으며, SMIL 태그 요소를 추가, 삭제, 또는 선택할 수 있는 기능을 제공한다.
- 속성 편집기: 각 편집기에서 표현하지 못하는 미디어들의 세부적인 속성들에 대한 편집을 가능하게 해주며 미디어의 세밀한 옵션 및 값을 편집할 수 있어 해당 미디어 타입에 따른 정밀한 편집이 가능하게 한다.
- 텍스트 편집기: 각 편집기를 통해 저작되어 자동 생성된 SMIL 코드를 실시간으로 보여주는 편집기로서 다른 편집 시스템에서 수정되고 교환되는 정보가 미디어 객체 관리자를 통해 직접 텍스트 편집기로 갱신되므로 사용자로 하여금 자신의 입력에 대한 결과를 즉시 확인할 수 있게 하는 기능을 제공한다.

5.2 미디어 객체 관리자(Media Object Manager)

미디어 객체 관리자는 본 시스템에서 중요한 역할을 하는 부분이다. 미디어 객체 관리자는 삼차원 미디어 객체를 화면에 나타내게 해주고 프레젠테이션 정보를 일관성 있게 유지시켜준다. 또한, 각 인터페이스들과 연결되어 실시간으로 정보를 교환하게 된다. 미디어 객체 관리자는 다음과 같이 세 가지의 모듈로 구성된다.

- 삼차원 엔진(3D Engine): 이 부분은 본 시스템의 핵심이라 할 수 있다. 이 엔진은 미디어 객체를 사용자의 입력에 따라 시간적, 공간적 편집이 가능하게 한다. 사용자의 입력에 따른 편집은 삼차원 인터페이스에 보여주는 삼차원 미디어 객체의 모양을 실시간으로 변경시켜 사용자에게 시간적, 공간적으로 변경되는 내용을 즉시 확인할 수 있게 주며, 미디어의 변경된 속성 값을 TRN에 전달하여 편집을 유지시켜 준다.
- 제어 모듈(Control Module): 이 모듈은 각 인터페이스 사이의 정보를 교환하며 실시간으로 정보를 갱신해 줌으로써 일관성을 유지시켜준다. 또한, 삼차원공간 파서를 이용하여 프레젠테이션의 내부구조인 미디어 객체들의 시간관계를 표현하는 TRN을 생성한다.
- 파서와 DOM 생성기(Parser & DOM generator): 파

서는 SMIL 코드를 번역하고 분석한다. 또한, TRN에 정의된 내용에 따라 자동적으로 DOM을 생성한다. 생성된 DOM에 정의된 대로 SMIL 코드를 생성하여 파일로 저장하고 텍스트 편집기를 통하여 화면에 보여준다.

5.3 협동 관리자(Collaboration Manager)

협동 관리자는 원거리에 있는 사용자와의 토론 형식의 편집을 지원하기 위한 공동저작을 가능하게 하는 모듈이다. 공동저작에 필요한 이벤트를 처리하고 공동저작에서 예상되는 발생하는 충돌(공유 데이터에 동시에 접근할 경우 발생할 수 있는 공유 데이터의 불일치)을 제거하는 기능을 한다. 이 모듈은 통신모듈과 동시성 제어 모듈로 나뉘며 자바 소켓(Java socket)을 사용하여 구현하였다.

- 통신 모듈(Communication Module): 이 모듈은 사용자의 이벤트에 대한 모든 통신을 담당한다. 서버와 TCP(Transmission Control Protocol)로 연결되어 사용자의 편집에 따라 그에 해당하는 패킷을 생성하여 서버에게 보내며 서버로부터 전송된 패킷을 분석하여 그에 해당하는 시스템을 호출하게 된다.
- 동시성 제어 모듈(Concurrency Control Module): 이 모듈은 동시성 제어를 위하여 통신 모듈과 연결되어 있으며, 서버에게 미디어 객체에 대한 허가를 요청하고 접근이 허가된 객체들 중 동시접근 중인 미디어들에 대하여 충돌이 없도록 제어하고 관리한다.

6. 시스템 성능 분석

본 절에서는 본 시스템의 성능을 두 가지 관점에서 분석해 보고자 한다. 첫 번째 분석은 본 시스템이 제안하는 삼차원 인터페이스의 유용성(usability) 분석이다. 두 번째 분석은 본 시스템이 제안하는 동시성 제어 기법의 성능에 대한 분석이다.

6.1 삼차원 인터페이스의 유용성 분석

본 시스템의 삼차원 인터페이스는 기존 상용 제품인 Tagfree Smil Editor의 이차원 인터페이스와 실제로 실험을 통하여 비교해 봄으로써 성능을 분석해 보았다. Tagfree의 저작 인터페이스는 그림 9에서 보는 바와 같이 이차원 인터페이스로 표현된 시간관계 편집기와 공간편집기의 두 개 인터페이스를 이용해야 한다. 이와는 다르게, 본 시스템의 저작 인터페이스는 그림 7에서 보는 바와 같이 하나의 삼차원 인터페이스로 표현되어 있음을 확인할 수 있다.

실험은 컴퓨터를 잘 다룰 수 있는 대학생 10명에게 SMIL 태그에 대한 기본적인 이해와 시간과 공간, 미디어들에 대한 간단한 설명을 해주고, 프로그램에 대한 대략적인 사용법을 설명한 후 10명에게 두 가지 모두 사용

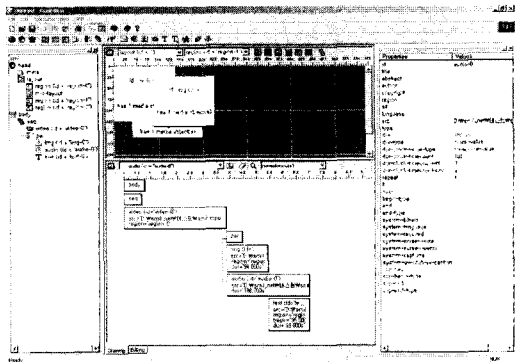


그림 9 Tagfree의 SmilEditor

해보도록 하였다. 여러 가지 미디어(audio, video, image, text)들을 어떻게 표현할 것인가를 제시해 주고 5명의 학생은 삼차원 인터페이스를 먼저 사용한 후 이차원 인터페이스를 사용토록 하였고, 나머지 5명의 학생은 이차원 인터페이스를 먼저 사용한 후 삼차원 인터페이스를 사용하여 프레젠테이션을 편집하도록 하였다.

또 다른 실험에서는 실험 대상자 10명에게 객체 수가 각각 1개, 3개, 5개, 7개, 9개, 11개인 프레젠테이션의 시나리오를 각각 주고 제작해 보도록 실험해 보았다. 이 경우에도 두 그룹으로 나누어 삼차원 인터페이스와 이차원 인터페이스의 사용 순서를 교차시켜 실험하였다.

표 2에 나타나 있듯이 모든 학생이 삼차원 인터페이스를 사용할 때 더 빠르게 프레젠테이션 제작하는 것을 확인할 수 있었다. 그림 10에서 보여준 것처럼 삼차원 인터페이스를 사용하는 것이 이차원 인터페이스를 사용하는 것보다 더 빠르게 저작할 수 있다는 실험 결과가

표 2 각 인터페이스에 대한 편집시간

	3차원 인터페이스	2차원 인터페이스
삼차원 인터페이스 우선편집 평균	6분 06초	8분 32초
이차원 인터페이스 우선편집 평균	5분 35초	8분 48초

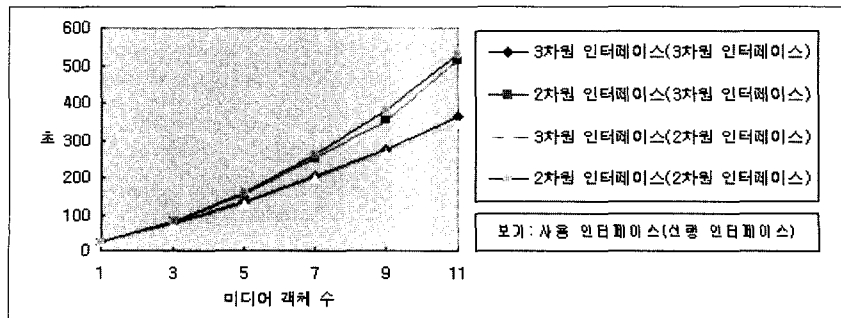


그림 10 미디어 객체 증가(복잡도 증가)에 따른 제작시간 평균 표

나왔다. 또한, 저작 객체 수가 증가할수록 저작 시간의 차이가 더 많아짐도 확인되었다.

이차원 인터페이스를 사용하는 경우 저작 시간이 더 많이 소요된 이유는 시간관계(sequential 또는 parallel)를 일일이 삽입해 가며 편집을 해야 하는 것과 미디어 객체수가 증가할 때마다 복잡도가 증가하여 수정이 더 어려워지기 때문인 것을 들 수 있다. 이에 비해, 삼차원 인터페이스는 미디어들의 시간관계를 파악하고 시간관계(sequential 또는 parallel)를 자동으로 생성하고 삽입함으로써 사용자의 부담이 줄어들었음을 확인할 수 있었다. 뿐만 아니라, 이차원 인터페이스 편집기는 공간 편집을 분리해서 제작한 후 사용자가 속성 편집기를 통하여 수동으로 시간 편집 인터페이스의 해당 객체와 연결해 주는 작업이 필요하였으며 시간과 공간의 관계가 한눈에 확인되지 않음으로 인한 불편함이 확인되었다.

한편, 삼차원 인터페이스는 공간 편집을 삼차원 미디어 객체를 통하여 직접 편집하므로 시간과 공간의 관계가 한눈에 드러나며 따로 미디어에 공간정보를 연결할 필요가 없어 사용자에게 보다 직관적인 인터페이스임이 확인되었다. 삼차원 인터페이스에 있어서 실험 대상 학생들이 제시한 문제점으로는 미디어들의 시간과 공간 관계가 삼차원 인터페이스 상에 겹쳐져서 표현되었을 경우 편집이 불편한 점이 있음이 제시되었다. 이것을 제외하곤 좋은 평을 받았고 실제로 편집에 있어서 삼차원 인터페이스를 이용한 경우에 직관적이고 빠른 편집이 가능하였다.

6.2 동시성 제어 기법 분석

공동저작을 위한 동시성 제어 기법 분석은 본 시스템과 유사한 다른 시스템이 현재 존재하지 않기 때문에 여러 동시성 제어 기법 중 두 가지 대표적인 기법인 단순 잠금 기법 및 타임스탬프 기법(트랜잭션 순서화 개념을 구현하는 대표적인 방법)과 비교해 봄으로써 성능을 평가해 보고자 한다.

단순 잠금 기법은 기본적으로 두 개의 연산 lock과

unlock으로 미디어 객체를 제어한다. 잠금 기법은 상호 배제(mutual exclusion)를 제공하므로 lock이 걸린 미디어 객체에는 이 lock을 건 클라이언트만 독점적으로 접근할 수 있고 lock을 건 클라이언트에 의해서만 unlock이 될 수 있다. 이점을 개선한 낙관적 잠금(optimistic locking) 기법은 잠금 권한을 얻기를 기다리는 동안 미리 그 객체에 대해 조작할 수 있도록 지원하는 방법이다. 잠금을 요청한 후에 잠정적인 승인을 얻어 임시적인 연산을 수행하다가 요청한 잠금 권한을 얻으면 실제로 그 연산을 실시한다. 만약, 요청한 잠금 권한을 얻지 못하면 미리 수행한 내용은 'undo' 처리된다.

타임스탬프 기법은 시스템이 클라이언트들에게 부여한 타임스탬프에 따라 여러 클라이언트들이 어떤 미디어 객체에 접근하고자 할 때 시스템이 부여한 타임스탬프에 따라 접근하는 순서를 정하는 방법으로 직렬성(serialization)도 보장되고 교착상태(deadlock)도 발생하지 않는다.

그림 11, 그림 12, 그림 13은 공유객체에 대한 동시성 제어 기법들의 편집순서와 각 동시성 제어기법의 성능을 비교한 것이다.

그림 11, 그림 12, 그림 13에서 확인할 수 있듯이, 일반적으로 잠금 기법은 충돌이 전혀 일어날 수 없는 확실한 방법이라 할 수 있으나 잠금이 적용된 객체에 대해 사용자들이 전혀 접근할 수 없게 만드는 것은 동시성 제어 면에서 보았을 때 굉장히 비효율적이며 실시간성을 지원하기 어려우므로 공동작업 시스템에 사용하는 것은 바람직하지 않다.

또한, 타임스탬프 기법의 경우에는 각 사용자의 접근이 매우 자유로울 수가 있으나 그 객체들에 대한 접근이 제대로 통제되지 않은 상태이므로 충돌의 우려가 굉장히 많다. 또한 특정 작업의 지연으로 사용자의 조작이 실시간으로 반영되지 않을 수 있으므로 타임스탬프 기법 역시 실시간 공동 저작 시스템에 적용되기에는 부적절하다.

사용자 인식에 대하여 각 세 가지의 동시성 제어방법을 분석한 결과, 잠금 기법은 한 사용자의 편집에 대해 다른 사용자의 편집을 허락하지 않으므로 사용자의 인식이 전혀 없으며, 타임스탬프 기법의 경우에는 동시 접근 시 서로 다른 사용자의 작업을 나타내 줄 수는 있지만 다른 사용자의 의도를 알 수 없어 사용자의 인식에 의한 대화형 공동 작업을 할 수 없다. 본 시스템에서 제안하는 동시성 제어기법은 모든 사용자의 동시 접근에 대하여 각각의 투명한 미디어를 모든 사용자에게 보여줌으로써 상대방의 의도와 작업상황을 확인할 수 있어 대화형 공동작업, 즉 토론식의 공동작업을 할 수 있어 사용자 인식이 높은 것으로 평가된다.

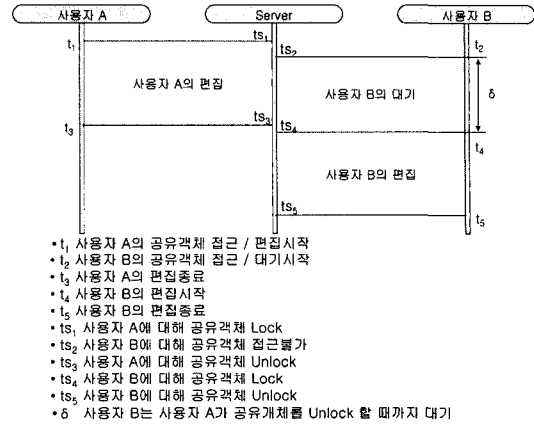


그림 11 단순 잠금 기법의 동시 접근 처리 절차

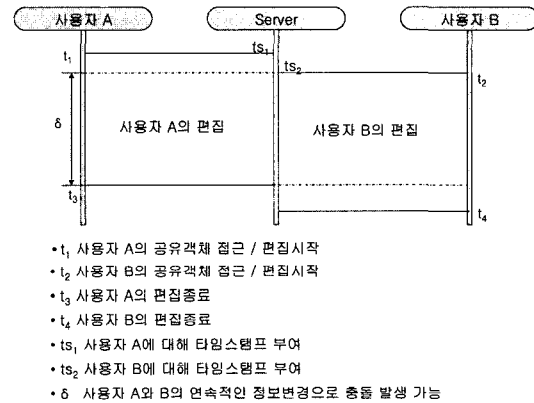


그림 12 타임스탬프 기법의 동시 접근 처리 절차

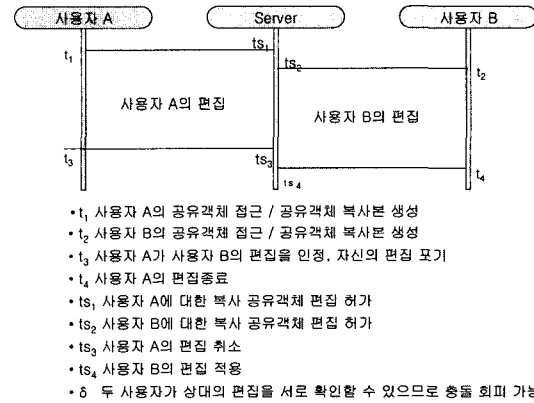


그림 13 본 시스템의 다중버전 기법의 동시 접근 처리 절차

본 시스템의 다중버전 기법은 기본적으로는 잠금이 내재되어 있으면서도 자유로운 사용자 접근이 가능하다. 또한, 사용자의 인식을 높여주므로 다른 두 기법보다 나

은 동시성 제어 방법이라 할 수 있다. 네트워크 전송을 비교해 보았을 때, 미디어의 움직임을 전송함에 있어서 미디어 객체의 편집 변화량 등의 기본적인 미디어 정보를 보내기 때문에 다른 기법들과 별반 차이는 없다고 할 수 있다. 그러나 편집되는 모든 미디어 객체들에 대해 복사 미디어 객체를 생성하기 때문에 동시 접근자가 매우 많아질 경우 각 클라이언트 시스템의 네트워크 전송 부하를 무시할 수는 없을 것이다.

7. 결론

본 논문에서는 분산되어 있는 사용자들이 삼차원 시·공간 안에서 동시에 멀티미디어 프레젠테이션을 디자인하고 제작할 수 있는 공동저작 시스템을 제안하였다. 본 저작 시스템의 공유 삼차원 인터페이스는 이차원의 공간편집 환경과 일차원의 시간편집 공간을 단 하나의 편집 윈도우로 통합한 것으로 다른 시스템에서는 찾아볼 수 없는 고유한 방법이다. 본 시스템에서 저작된 멀티미디어 프레젠테이션으로부터 내부표현인 TRN이 자동으로 생성되며 프레젠테이션 내용이 수정됨에 따라 자동으로 변환된다. TRN 구조는 멀티미디어 프레젠테이션의 개념적 시간구조와 정확히 일치하며 이 TRN 구조로부터 SMIL 코드가 자동으로 생성된다. 본 연구에서는 사용자인식, 다중버전, 그리고 공유객체에 대한 접근허가권을 기반으로 하는 효율적인 동시성 제어 매커니즘을 설계하고 구현하였다.

본 시스템의 장점의 하나는 시간 편집을 하다 공간 편집을 하려고 할 때 편집 환경을 바꿀 필요가 없다는 점이다. 다른 장점은, 본 시스템에 구현된 동시성 제어 기법이 사용자들이 삼차원 공유 객체를 동시에 편집할 때 네트워크 지연으로 생길 수 있는 충돌을 최소화해 주고 사용자에게 최대한의 동시 편집 자유도를 줄 수 있는 효율적인 방법이라는 점이다. 본 시스템은 SMIL 기반의 멀티미디어 프레젠테이션 저작 시스템뿐만 아니라 삼차원 가상공간에서의 다양한 협동작업 환경에서 활용될 수 있다.

앞으로, 멀티미디어 프레젠테이션 저작 환경에서 공유 객체의 잠금을 현재 시스템에서보다 더 줄이면서 시스템의 부하를 최소화 할 수 있는 공동작업 시스템에 가장 효과적인 새로운 동시성 제어 매커니즘에 대해 깊이 연구하고자 한다. 또한, TRN을 이용하여 미세한 동기화를 보장하는 최적의 자동 스케줄링 매커니즘에 대해서도 연구할 계획이다.

참고 문헌

[1] J. Song, G. Ramalingam, R. Miller, B. K. Yi, "Interactive authoring of multimedia documents in

a constraint-based authoring system," *Multimedia Systems*, Vol.7, pp.424-437, Springer-Verlag, 1999.

[2] M. Y. Kim, J. Song, "Multimedia Documents with Elastic Time," *Proceedings of ACM Multimedia '95*, November 5-9, 1995, San Francisco, California, USA, pp.143-154, 1995.

[3] M. Vazirgiannis, I. Kostalas, T. Sellis, "Specifying and Authoring Multimedia Scenarios," *IEEE Multimedia*, Vol.6, No.3, pp.24-37, July-September 1999.

[4] L. Weitzman, "Automatic Presentation of Multimedia Documents Using Relational Grammars," *Proceedings of ACM Multimedia 94*, October 15-20, 1994, San Francisco, California, USA, pp.443-451, 1994.

[5] A. Karmouch, J. Emery, "A Playback Schedule Model for Multimedia Documents," *IEEE Multimedia*, Vol.3, No.1, pp.50-61, Spring 1996.

[6] <http://www.technologyreview.com/articles/tristram1201.asp>

[7] R. M. Baecker, D. Nastos, I. R. Posner, K. L. Mawby, "The User-centred Iterative Design Of Collaborative Writing Software," *Proceedings of ACM INTERCHI '93*, April 24-29, 1993, pp.339-541.

[8] W3C, Synchronized Multimedia Integration Language (SMIL) 1.0 Specification: W3C Recommendation 15-June-1998, <http://www.w3.org/TR/REC-smil/>

[9] W3C, Synchronized Multimedia Integration Language (SMIL 2.0): W3C Recommendation 07 August 2001, <http://www.w3.org/TR/smil20/>

[10] T. D. C. Little, A. Ghafoor, "Spatio-Temporal Composition of Distributed Multimedia Objects for Value-Added Networks," *IEEE Computer*, Vol.24, No.10, pp.42-50, October 1991.

[11] M. Iino, Y. F. Day, and A. Ghafoor, "An Object-Oriented Model for Spatio-Temporal Synchronization of Multimedia Information," *Proceedings of IEEE International Conference on Multimedia Computing and Systems*, May 14-19, 1994, Boston, Massachusetts, USA, pp.110-119, 1994.

[12] N. U. Qazi, M. Woo, and A. Ghafoor, "A Synchronization and Communication Model for Distributed Multimedia Objects," *Proceedings of ACM Multimedia 93*, August 1-6, 1993, Anaheim, California, USA, pp.147-155, 1993.

[13] J. F. Allen, "Maintaining Knowledge about Temporal Intervals," *Communications of the ACM*, Vol.26, No.11, pp.832-843, November 1983.

[14] 노승진, 장진희, 성미영 "개념적 시간관계 기반의 멀티미디어 프레젠테이션 저작 시스템", *정보과학회논문지: 컴퓨팅의 실제*, 제9권 제3호, pp.266-277, 2003. 6.

[15] A. Prakash, M. J. Knister, "A Framework for Undoing Actions in Collaborative Systems," *ACM Transactions on Computer-Human Interaction*, Vol.1, No.4, pp.295-330, December 1994.

- [16] S. J. Rho, D. H. Lee, M. Y. Sung, "Designing Multimedia Presentations in 3D Spatio-Temporal Space," *Proceedings of International Society on VSMM(Virtual Systems and Multimedia)*, September 25-27, 2002, Gyeongju, Korea, pp.466-474, 2002.
- [17] B. Prabhakaran and S. V. Raghavan, "Synchronization Models For Multimedia Presentation With User Participation," *Proceedings of ACM Multimedia 93*, August 1-6, 1993, Anaheim, California, USA, pp.157-166, 1993.
- [18] C. A. Ellis, S. J. Gibbs, G. L. Rein, Groupware: Some Issues and Experiences. *Communications of the ACM*, Vol.34, No.1, pp.38-58, January 1991.
- [19] M. Y. Sung, D. H. Lee, S. J. Rho, S. Y. Rhee "Authoring Together in a 3D Spatio-Temporal Space," *Proceedings of ACM Multimedia 2002 Workshop ITP(International Conference on Immersive Telepresence)*, December 6, 2002, Juanles-Pins, France, 4 pages, 2002.
- [20] <http://www.w3.org/DOM>
- [21] J. Gray, A. Reuter, *Transaction Processing: Concepts and Techniques Morgan Kaufmann*, 1070 pages, 1993.
- [22] M. Stefik, D. G. Bobrow, G. Foster, S. Lanning, D. Tatar, "WYSIWIS Revised: Early Experience with Multiuser Interfaces," *ACM Transactions of office Information Systems*, Vol.5, No.2, pp.147-167, April, 1987.
- [23] S. Greenberg, D. Marwood, "Real Time Groupware as a Distributed System: Concurrency Control and its Effect on the Interface," *Proceedings of ACM CSCW '94*, October 22-26, 1994, Chapel Hill, North Carolina, USA, pp.207-217, 1994.
- [24] L. Lamport, "Time, Clocks, and the Ordering of Events in a Distributed System," *Communications of the ACM*, Vol.21, No.7, pp.558-565, July 1978.
- [25] D. R. Jefferson, "Virtual Times," *ACM Transactions on Programming Language and Systems*, Vol.7, No.3, pp.404-425, July 1985.
- [26] K. A. Lants, "An Experiment in Integrated Multimedia Conferencing," *Proceedings of ACM CSCW '86*, pp.267-275, 1986.
- [27] S. Sarin, I. Grief, "Computer-Based Real-Time Conferences," *IEEE Computer*, Vol.18, No.10, pp.33-45, October 1985.
- [28] C. A. Ellis, S. J. Gibbs, "Concurrency Control in Groupware Systems," *Proceedings of ACM SIGMOD International Conference on Management of Data*, Seattle, Washington, USA, pp.399-407, 1989.
- [29] J. Munson, P. Dewan, "A Concurrency Control Framework for Collaborative Systems," *Proceedings of ACM CSCW '96*, November 16-20, 1996, Boston, Massachusetts, USA, pp.278-287, 1996.
- [30] C. Sun, D. Chen, "Consistency Maintenance in

Real-Time Collaborative Graphics Editing Systems," *ACM Transactions on Computer-Human Interaction*, Vo.9, No.1, pp.1-41, March 2002.



이도형

인천대학교 컴퓨터공학과 재학중. 관심분야는 멀티미디어 시스템, 실시간 시스템

성미영

정보과학회논문지 : 컴퓨팅의 실제 제 9 권 제 3 호 참조