

# 이벤트 기반 다중 쓰레드에 의한 멀티미디어 프리젠테이션에 관한 연구

이규남<sup>o</sup>, 나인호  
군산대학교 전자정보공학부  
{knlee<sup>o</sup>, ihra}@kunsan.ac.kr

## A Study on Multimedia Presentation by Event based Multi-thread

Kyunam Lee<sup>o</sup>, Inho Ra  
School of Electronic & Information Engineering, Kunsan National University

### 요 약

본 논문에서는 멀티미디어 프리젠테이션 과정에서 발생할 수 있는 여러 동기화 요소를 고려한 프리젠테이션 제어 방안에 대하여 기술하였다. 효율적인 프리젠테이션을 위하여 모노 미디어의 개별적인 효과 표현과 시나리오를 기반으로 한 스케줄러, 그리고 여러 하드웨어 및 소프트웨어 관리 쓰레드 등으로부터 동기화와 직·간접적으로 관련된 이벤트를 발생시키고, 이러한 이벤트를 종합적으로 관리하도록 하였다. 발생된 이벤트는 불규칙하고 동시다발적이기 때문에 발생 환경에 따라 적절하게 폐기하거나 처리의 선후관계를 다시 정의하는 등 프리젠테이션 환경에 따른 최적의 서비스 품질을 유지할 수 있도록 고려하였다.

### 1. 서 론

컴퓨터와 더불어 네트워크 기술의 발전으로 과거 텍스트 위주의 정적 미디어(static media)를 다루던 컴퓨팅 환경이 오디오와 비디오 데이터의 처리량이 급격히 증가하면서 동적 미디어(dynamic media) 환경으로 빠르게 전환하고 있다. 특히, 텍스트, 이미지, 오디오, 비디오 등 여러 미디어의 고유 특징을 활용하여 다양한 형태의 새로운 멀티미디어 콘텐츠를 생성하고, 이를 로컬 혹은 네트워크를 이용한 서비스 기술의 개발에 관심이 집중되고 있다.

멀티미디어 서비스는 다양한 모노 미디어(mono media)의 시공간적 조합에 대한 프리젠테이션 과정이며, 이러한 멀티미디어의 프리젠테이션 과정에서는 반드시 구성된 미디어들의 시간적 동기화에 대한 문제를 고려해야 한다. 특히 이용자가 구성한 멀티미디어 프리젠테이션을 수행할 때 시나리오에 기반하여 정확한 시간적 동기화(synchronization)가 이루어지지 않으면 멀티미디어의 요구된 서비스 품질을 기대하기 어렵다[1, 2, 3].

이러한 멀티미디어 프리젠테이션에 대한 동기화 기술은 다양한 관점에서 많은 연구가 진행되었으며 대부분 데이터의 버퍼링이나 멀티미디어 저장장치 관리 및 전송 스케줄링 등과 같은 부분에 집중되어 왔다. 본 논문에서는 그러한 연구 결과를 토대로 멀티미디어 프리젠테이션을 통합적으로 운용 관리할 수 있는 멀티미디어 프리젠테이션의 제어 및 관리에 대한 내용을 주제로 하였다.

본 논문에서는 먼저 멀티미디어 프리젠테이션의 정의에 대하여 간단하게 기술하고, 동기화된 프리젠테이션 수행을 위하여 이와 관계된 여러 미디어의 취급 방법, 효과(effect), 그리고 버퍼와 같은 각종 자원들을 개별적으로 관리하는 쓰레드(thread)를 중심으로 이벤트(event)

기반 프리젠테이션 제어 기법의 설계에 대하여 자세하게 기술하였다.

### 2. 관련 연구

멀티미디어 프리젠테이션에서는 사용자가 작성하고자 하는 멀티미디어 콘텐츠에 포함된 다양한 모노 미디어의 시간적인 조합(temporal composition)과 공간적 조합(spacial composition)이 요구된다[3]. 시간적인 조합은 모노 미디어들 간의 시간 관계를 정의하는 것으로써 7가지의 시간적인 관계를 정의하는 것이며, 공간적 조합은 모노 미디어들의 출력 위치를 지정하는 것이다[3].

이렇게 작성된 멀티미디어 콘텐츠는 프리젠테이션 과정에서 대용량 데이터의 실시간적 처리라는 특성으로 인하여 처리 지연현상을 필연적으로 수반하게 된다[4]. 또한, 네트워크를 이용한 멀티미디어의 프리젠테이션 과정에서는 네트워크의 예측 불가능한 불규칙성으로 인하여 전송 및 출력 동기화 문제가 발생하므로 사용자가 원하는 서비스 품질을 유지하기 어렵다.

멀티미디어 프리젠테이션의 동기화는 크게 미디어내 동기화(Intra-media synchronization)와 미디어간 동기화(Inter-media synchronization)로 나눌 수 있으며[3], 그 처리 방법은 데이터의 저장에 대한 멀티미디어 저장장치나 동적·정적 버퍼링 기법[5], 네트워크의 지연을 흡수할 수 있는 기법[4], 멀티미디어 프리젠테이션 시나리오 작성 과정에서 미디어간의 관계를 기술하는 기법[3, 6] 등 다양한 방안들이 연구되어 활용되고 있는 추세이다.

본 논문은 멀티미디어의 시공간적 조합을 이루고 있는 각 미디어들과, 그러한 미디어가 표현하는 여러 가지 효과, 프리젠테이션 시나리오, 그리고 프리젠테이션과 관련된 컴퓨터의 여러 주요한 자원 등에 개별적인 작업 쓰레드를 생성하도록 하였다. 특히, 프리젠테이션 과정에서

생성된 여러 쓰레드들을 유기적으로 통합 관리하기 위하여 쓰레드간 주요한 정보를 상호 교환할 수 있는 이벤트를 생성하고, 생성된 여러 이벤트를 종합적으로 관리하여 프리젠테이션 각 상황에 맞는 동기화에 대한 처방을 할 수 있도록 하는 이벤트 기반의 다중 쓰레드를 이용한 프리젠테이션 제어 기법을 설계하였다.

프리젠테이션 과정에서 이미 정의된 프리젠테이션 시나리오에 의한 스케줄 이외에 네트워크 환경을 비롯한 모든 하드웨어 및 소프트웨어적 환경은 가변적이어서 예측 불가능하기 때문에 미리 특정 환경을 대상으로 동기화 방안을 기술하는 것은 그 한계가 있다. 따라서 실시간으로 변하게 되는 프리젠테이션 동기화와 관련된 여러 변수를 종합적으로 고려하였다. 또한 동시에 발생한 여러 이벤트를 시간적인 전후관계와 그 중요도를 감안하여 먼저 처리하거나 지연하도록 하였으며, 경우에 따라 여러 이벤트를 하나로 통합할 수 있도록 하여 사용자의 입장에서 프리젠테이션을 원활하게 수행할 수 있도록 하는데 중점을 두었다.

### 3. 이벤트(Event)

멀티미디어의 프리젠테이션과 관련된 이벤트는 여러 종류가 있을 수 있다. 본 논문에서는 발생할 수 있는 이벤트에 대해 여러 가지를 고려하였으며, 주요한 몇 가지 이벤트에 대하여 기술하면 다음과 같다.

- ① 사용자 이벤트(User Event) : 사용자 이벤트는 프리젠테이션의 진행에 직접 및 간접적으로 참여하고 있는 사용자가 발생시키는 이벤트로써 예를 들면, 프리젠테이션의 시작, 중지, 일시 정지 등의 프리젠테이션 조작을 표현하는 이벤트이다. 대화형 멀티미디어 프리젠테이션 환경이 요구됨에 따라 프리젠테이션 과정에 있어 사용자 참여가 보장되는 시스템의 중요성이 증가되고 있다.
- ② 프리젠테이션 스케줄러 이벤트(Presentation Scheduler Event) : 이것은 사용자의 프리젠테이션 시나리오 작성에 따른 모노 미디어들과 효과들의 시공간적 조합에 대하여 프리젠테이션 수행 직전에 전체적인 이벤트 스케줄을 작성한다. 이러한 프리젠테이션 스케줄은 프리젠테이션 과정에서 미디어에 대한 효과 발생을 위한 이벤트로 통지하게 되며, 이 이벤트에 의하여 멀티미디어 컨텐츠의 목적을 사용자에게 표현할 수 있다. 프리젠테이션 스케줄 즉, 이벤트 스케줄은 다른 이벤트들과 달리 프리젠테이션 진행 전에 구성되어야 하고, 그 시간적인 순서에 의하여 정렬되게 된다.
- ③ 버퍼 관리자 이벤트(Buffer Manager Event) : 네트워크를 통한 데이터의 임시 저장과 출력할 데이터의 관리를 위한 버퍼를 관리하는 이벤트이다.
- ④ 네트워크 관리자 이벤트(Network Manager Event) : 네트워크 관리자 이벤트는 원격지에 존재하는 미디어 데이터의 프리젠테이션 과정에서 필수적인 이벤트로써 네트워크의 상태, 데이터의 송수신 상황 등을 통지할 수 있는 이벤트이다.

이러한 다양한 이벤트들은 여러 소스(source)로부터 발생될 수 있으며, 발생한 이벤트의 내용과 발생 시점 등 중요한 정보와 함께 생성 및 전달되고 마지막으로 처

리되어야 한다. 따라서 이러한 이벤트의 데이터를 다음과 같은 형식으로 설계하였다.

- ① 이벤트 내용 : 발생한 이벤트의 종류를 나타낸다.
- ② 이벤트 처리대상 미디어 혹은 쓰레드 : 발생한 이벤트는 "Presentation Manager"에 의하여 분석되고 해당 처리 쓰레드나 모듈로 전달되어야하므로 처리 대상에 대한 정보를 표현한다.
- ③ 이벤트 발생 시간 : 이벤트의 발생 시간을 표현하며, 동일한 이벤트의 중복 발생이나 우선순위 처리 등을 위하여 사용된다.

### 4. 프리젠테이션 관리자(Presentation Manager)

"Presentation Manager"는 프리젠테이션에 대한 제어를 처리할 수 있는 여러 가지 협력 쓰레드로 설계하였으며, 그 구성과 운용은 그림 1과 같다.

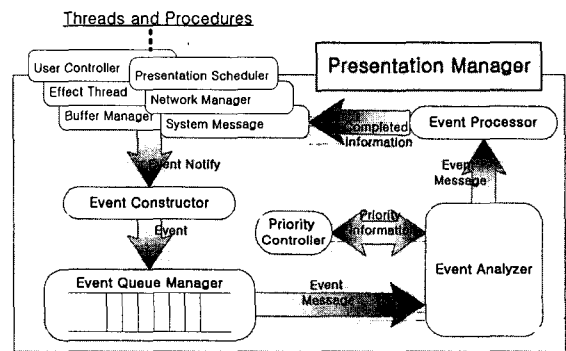


그림 1. Presentation Manager의 운용

#### 4.1 이벤트 통지(Event Notify)

"Event Notify"는 여러 가지 작업 쓰레드와 관리자들의 처리 도중에 발생하는 사건을 이벤트로 발생시켜서 그 사실을 "Presentation Manager"에게 통보하기 위한 메시지이다. "Event Constructor" 쓰레드를 비롯한 여러 가지 관리자 쓰레드가 백그라운드로 운용되는 과정에서 프리젠테이션 효과 지정이나 사용자의 프리젠테이션 제어 등의 사건이 발생할 경우 그 내용을 "Presentation Manager"에게 통보하여 적절한 이벤트를 만들 수 있도록 하였다.

#### 4.2 이벤트 생성자(Event Constructor)

"Event Constructor"는 통지된 이벤트의 정보에 다음과 같은 정보를 추가로 부여함으로써 이벤트에 대한 충분한 정보를 구성하도록 하였다.

- 이벤트 내용.
- 이벤트 처리 대상 미디어 혹은 쓰레드
- 이벤트 발생 시간

#### 4.3 이벤트 큐 관리자(Event Queue Manager)

"Event Queue Manager"는 발생한 이벤트를 큐에 저장하고 관리하는 기능을 수행한다. 프리젠테이션 스케줄러에 의한 이벤트를 제외하고 이벤트의 발생은 예측할 수 없으며, 그 처리 시점도 가정할 수 없기 때문에 발생하는 이벤트에 대한 주요 정보를 관리하기 위하여 본 논문에서는 이중 큐 시스템을 사용하여 발생하는 이벤트를 저장 및 처리하였다. 다시 말해, 두 개의 큐  $Q_1$ ,  $Q_2$ 를

"Event Queue Manager"가 관리하면서 "Event Constructor"에 의해 생성된 이벤트를 Q<sub>1</sub>에 저장한다면 "Event Analyzer"에 의한 큐 접근은 Q<sub>2</sub>로 한다. 반대로 "Event Constructor"에 의해 발생한 이벤트를 Q<sub>2</sub>에 저장한다면 "Event Analyzer"에 의한 이벤트 읽기는 Q<sub>1</sub>을 대상으로 한다. 그림 2는 이와 같은 큐의 운용 방식을 나타낸 것이다.

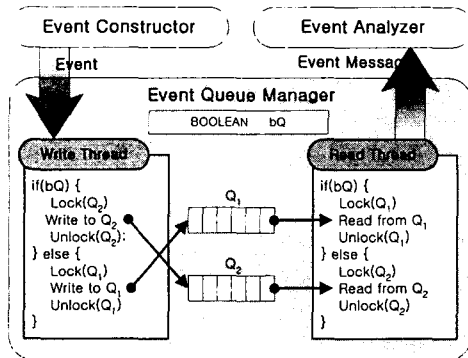


그림 2. Event Queue Manager의 운용

그림 2의 이진 데이터 bQ는 읽기 또는 쓰기에 대한 큐의 선택을 위한 변수로서, 최소의 이벤트 처리 임계 시간 내에서 논리적 시간의 흐름에 맞춰 1과 0의 값으로 반복하여 설정된다. 예를 들어, 프리젠테이션 진행과 관련하여 0.01ms에 한번씩 bQ의 설정치를 변경할 경우 1/100초의 주기로 큐에 저장된 이벤트를 처리할 수 있게 하였다.

#### 4.4 이벤트 분석기(Event Analyzer)와 우선 순위 제어기(Priority Controller)

"Priority Controller"는 발생 가능한 여러 이벤트에 대하여 미리 우선순위(Urgency, Normal, notUrgency)를 부여하였다. "Urgency"는 긴급한 처리를 요하는 이벤트로써, 사용자의 프리젠테이션 개입으로 인한 이벤트나, 네트워크의 과부하 등의 이벤트이다. "Normal" 이벤트 그룹은 스케줄러 등의 이벤트들이 속하며, 동기화에 있어 중요하지만 긴급하지 않은 이벤트이다. "notUrgency" 그룹의 이벤트는 긴급하지 않은 이벤트이며, 예를 들어 소모된 데이터의 버퍼 해제 등과 같은 이벤트이다. 각 이벤트 그룹에 속한 이벤트들은 다시 그 중요도에 따라 3등급으로 분류하였으며, 프리젠테이션 과정에 동적으로 발생하는 각 이벤트에 대하여 우선순위를 비교 분석하여 선후 처리 관계 및 통합이나 폐기 등을 처리할 수 있도록 하였다.

"Event Analyzer"는 이벤트의 우선순위를 참조하여 현재 발생한 이벤트의 관계를 재정렬할 수 있도록 하였다. 이벤트 큐에 저장된 이벤트는 주기적으로 이벤트 분석기에 의하여 분석되며, 이 과정에서 메시지의 선후관계가 다시 정렬된다. 또한 중복된 이벤트에 대하여는 하나의 이벤트로 통합 또는 폐기하게 되는데 예를 들어, 네트워크의 전송 지연에 대한 이벤트가 여러번 발생하였다면, 그 내용을 참조하여 하나의 전송지연 이벤트로 묶어 처리될 수 있다. 특히 이벤트 분석기는 "notUrgency" 등급의 이벤트 그룹에 준하는 이벤트들에 대하여 그 처리를 뒤로 미룰 수 있으며, 이 경우 "Event Queue Manager"

로 다시 해당 이벤트를 전송함으로써 처리가 급하지 않은 이벤트를 뒤로 미루고 좀더 주요한 이벤트를 먼저 처리할 수 있도록 고려하였다.

이러한 프리젠테이션 관리자의 운용은 프리젠테이션 동기화 요소의 불규칙하고 동적인 발생을 흡수할 수 있도록 하는데 목표를 두었으며, 다양한 동기화 정보를 종합적으로 처리할 수 있고, 특히, 실시간 대화식 처리를 지향함으로써 멀티미디어 프리젠테이션 효율이 향상될 수 있도록 하였다.

#### 5. 결 론

본 논문에서는 멀티미디어 프리젠테이션 과정에서 발생할 수 있는 여러 동기화 요소를 고려한 효율적인 프리젠테이션을 위한 프리젠테이션 관리부의 설계에 대하여 기술하였다. 이를 위하여 각각 모노 미디어의 개별적인 프리젠테이션과 스케줄러, 그리고 여러 하드웨어 및 소프트웨어 관리자 등을 각각 담당하여 처리할 수 있는 쓰레드를 두고, 각 쓰레드들을 이벤트로 운용함으로써 유기적이며, 동적으로 프리젠테이션을 제어할 수 있도록 하였다.

프리젠테이션의 동기화와 관련된 여러 가지 변수들은 그 발생 상황에 대하여 정확히 예측하기 불가능하며, 또한 다수가 한번에 발생할 수 있기 때문에 종합적인 고찰이 필요하게 된다. 따라서 다양한 환경에서 여러 가지 실험이 필수적이고, 좀더 효율적인 프리젠테이션 제어를 위하여 각 이벤트의 우선순위 조절 및 새로운 이벤트의 추가 등이 필요하다. 향후 본 논문의 설계 결과를 기반으로 프리젠테이션의 서비스 품질을 좀더 향상시킬 수 있도록 많은 실험을 진행하고 완성된 시스템을 구현하고자 한다.

#### [참고문헌]

- [1] Y.Kato, K.Hakozaki, "Application QoS Management for Distributed Computing Systems", 2001 IEEE International Conference on Communications, Vol 4of10, pp.1201-1205, 2001
- [2] B. Prabhakaran, S. V. Raghovan, "Synchronization Models for Multimedia Presentation with User Participation", Multimedia Systems Springer-Verlag, pp. 53-62, 1994.
- [3] T.D.C. Little, A.Ghafoor, "Distributed Multimedia Objects Composition and Communication", IEEE Network Magazine, pp.72-84, Nov.1990
- [4] Y.Ishibashi, S.Tasaka, Y.Tachibana, "Adaptive Causality and Media Synchronization Control for Networked Multimedia Applications", 2001 IEEE International Conference on Communications, Vol 3of10, pp. 952-958, 2001
- [5] Kun-Lung Wu, Philip S. Yu, "Increasing Multimedia System Throughput with Consumption-Based Buffer Management", Multimedia System 6:421-428, 1998.
- [6] S. Ricarel, W. Tonathen, M. David, "Actuality-of-Service Specification for Multimedia Presentation System", Springer-Verlag, pp. 251-263, 1995