

# 멀티미디어 데이터베이스에서 SMIL을 이용한 질의어 확장 및 프리젠테이션 시스템의 구현

이 중 화<sup>†</sup>·이 중 환<sup>†</sup>·유 영 호<sup>†</sup>·김 경 석<sup>††</sup>

## 요 약

멀티미디어 데이터베이스 시스템에서 멀티미디어 데이터를 질의하고 결과를 어떻게 프리젠테이션 할 것인가 하는 것은 매우 중요하다. 또한 질의 결과를 다양한 응용에서 사용하기 위해서는 보다 일반화된 형태로 질의결과를 제공해야 한다.

본 논문에서는 W3C의 권고안인 SMIL에서 제공하는 멀티미디어 프리젠테이션의 기능적 요소들을 질의어에서 지원하도록 함으로써 보다 일반화된 형태로 질의어를 확장하고, 또한 다양한 응용에서 사용하도록 하기 위해서 질의 결과를 SMIL 형태로 제공하도록 한다.

키워드 : 프리젠테이션, 멀티미디어, 데이터베이스

## Query Language Extension and Implementation of MM Presentation System using SMIL in MMDB

Jung-Hwa Lee<sup>†</sup> · Jong-Hwan Lee<sup>†</sup> · Young-Ho Yu<sup>†</sup> · Kyong-Sok Kim<sup>††</sup>

## ABSTRACT

It is very important how users query MM data and how the query results are presented in MMDB systems. Also, the query results are represented as the generalized form so as to be used in various applications. In this paper, we propose a generalized method for querying and presenting MM data by extending query language based on presentation elements of SMIL(SMIL is W3C recommendation for synchronized MM presentation). Our system provides the query results for users as SMIL format in order to be used for various applications.

Key word : presentation, multimedia, database

### 1. 서 론

컴퓨터의 성능 및 기술 향상으로 멀티미디어 데이터의 중요성이 날로 부각되고 있는 현 시점에서, 데이터 처리의 핵심 기술이라고 할 수 있는 데이터베이스 분야에서도 멀티미디어 데이터를 어떻게 처리할 것인가에 대한 연구가 활발히 진행되고 있다.

멀티미디어 데이터베이스 시스템에 관한 연구는 멀티미디어 데이터를 어떻게 표현할 것인가, 어떻게 저장할 것인가, 어떻게 질의할 것인가, 질의한 결과를 어떻게 나타낼 것인가 등의 큰 범주로 나누어서 연구가 진행되어 왔다. 특히 사용자가 데이터베이스에서 데이터를 질의하기 위해서는 SQL과 같은 질의어를 사용하게 되는데, 멀티미디어 데이터베이스를 위해서는 기존의 데이터베이스의 기능적 요

소가 확장되어야 하듯이 사용자가 직접 접하게 되는 질의어의 확장은 무엇보다도 중요하다.

멀티미디어 질의어를 확장하는데 있어서는 멀티미디어 데이터를 모델링하고 이를 질의해 낼 수 있는 연구[1,2] 뿐만 아니라 멀티미디어 데이터가 특징적 요소로 가지고 있는 시공간적 요소를 어떻게 표현할 것인가, 또한 사용자가 질의해 낸 결과를 어떻게 시·공간적으로 정의된 형태로 제공할 것인가에 대한 연구도 중요하다. 이를 멀티미디어 프리젠테이션이라고 한다.

기존의 멀티미디어 데이터베이스 질의어에서 프리젠테이션을 지원하는 방법으로는 기존의 구문에 프리젠테이션을 위한 새로운 구문을 추가하는 형태가 일반적인데, 이는 프리젠테이션을 위해 새로운 질의어를 디자인하는 방법에 비해 사용자가 새로운 질의어를 배울 필요가 없으며, 기존의 질의어의 장점을 그대로 살릴수 있다는 등의 여러 가지 장점을 가지고 있기 때문에 대부분 이 방법을 사용한다[3-6].

\* 이 논문은 1998년도 부산대학교 학술연구조성비를 지원받아 수행된 연구임.

† 준 회 원 : 부산대학교 대학원 전자계산학과

†† 정 회 원 : 부산대학교 전자전기정보컴퓨터공학부 교수

논문접수 : 2000년 11월 17일, 심사완료 : 2001년 1월 30일

그러나 새로운 구문을 추가하는 방법에 있어서도 기존의 여러 연구들을 살펴보면 사용자 요구사항의 분석에 따른 시공간 관계의 정의에서부터 서로 다른 견해를 가지고 있으며 제공되는 기능들도 차이점이 많기 때문에 기존의 SQL과 같이 보편적으로 사용되기에는 문제점이 있다[5, 7-9]. 따라서 본 논문에서는 질의결과를 사용자에게 보여주는 멀티미디어 프리젠테이션에 있어서 보다 일반적인 기능적 요소를 정의하고 이를 질의어에서 제공하고자 한다. 이를 위해서 본 논문에서는 WWW에서의 동기화된 멀티미디어 문서를 기술하는 언어인 SMIL에서 제공하는 멀티미디어 프리젠테이션의 기능적 요소들을 분석하여 이를 멀티미디어 데이터베이스 질의어에서 지원하도록 함으로써 보다 일반화된 형태로 프리젠테이션을 정의하고, 또한 질의 결과를 SMIL의 형태로 제공함으로써 멀티미디어 데이터베이스에서 질의된 결과들을 SMIL을 지원하고 있는 다양한 응용에서 사용할 수 있는 방법을 제공하고자 한다.

본 논문의 구성은 다음과 같다. 먼저 2장에서는 멀티미디어 질의어 확장에 관련한 연구들과 SMIL에 대해 살펴보고 3장에서는 질의어를 확장하기 위해 SMIL을 통해 정의된 시공간 관계 및 표현 방법을 분석하고 4장에서 질의어를 확장하기 위한 문법을 정의한다. 5장에서는 확장된 질의어를 수행하고 이를 SMIL형태로 제공해주기 위한 프리젠테이션 시스템 설계 및 구현에 대해 살펴보고 6장에서는 결론 및 향후 연구과제에 대해 살펴본다.

## 2. 관련 연구

멀티미디어 데이터베이스에서 프리젠테이션을 하기 위한 기존의 연구들은 PICQUERY+, OVID, SQL+D 등이 있다.

PICQUERY+는 의료 이미지 데이터베이스 질의어로서, 질의와 프리젠테이션 부분을 분리하여 템플릿 형식으로 질의를 준다[9]. 질의의 결과는 프리젠테이션 템플릿의 입력이 되며, 프리젠테이션 템플릿 내의 정보에 따라 결과 미디어의 프리젠테이션을 수행한다. 특히, 교차 관계(intersects), 포함 관계(contains), 일직선 관계, 뒷면 관계(behind) 등의 공간을 표현하기 위한 연산자와 포함 관계(during), 사이 관계(between), 동시 관계(equivalent), 순차 관계(adjacent), 이전 관계(precedes) 등의 시간 연산자가 있다. 또한 두 개의 미디어를 하나의 미디어로 합치는 연합 연산자(fuses\_into), 하나의 미디어를 두 개의 미디어로 분리하는 분리 연산자(split\_into) 등 풍부한 시공간 연산자들을 가지고 있다. 그러나, 기존에 널리 사용되고 있는 SQL 등과 같은 질의어와 구조가 다르기 때문에 일반 사용자들이 질의어를 구성할 때 어려움이 따를 수 있으며, 특정 응용을 위한 인터페이스를 설계하고 있으므로, 이를 광범위한 멀티미디어 질의어로 이용하는 데는 적합하지 못하다.

OVID(Object-oriented Video Information Database)에서는 객체 지향 데이터베이스를 기반으로 비디오 데이터를 위한 질의어 확장 및 사용자 인터페이스를 구현하였다[5]. 이 연구에서 정의된 VideoSQL에서는 SQL을 기반으로 사용자가 요구한 데이터가 화면에 보여질 때의 위치와 시간 관계를 표현 할 수 있다. 그러나, VideoSQL은 특정 미디어에 의존적이며, VideoChart라는 전용 사용자 인터페이스에서 실행하기 때문에 인터페이스가 일반적이지 못하다는 단점을 가지고 있다.

SQL+D는 기존의 SQL 질의어에 프리젠테이션 절을 확장한 형식의 질의어로서, 질의 결과를 위한 공간적인 그래픽의 인터페이스를 디자인할 수 있도록 확장하였다[10]. 확장된 프리젠테이션 절은 정형화된 GUI 요소들인 박스화된 텍스트, 체크 박스, 리스트, 버튼 등을 제공하며, 사용자에게 결과 화면을 디자인할 수 있도록 한다.

위의 연구들을 살펴보면 각 응용에 맞는 멀티미디어 데이터를 처리하기 위해 질의어를 확장하고 이를 화면에 나타내기 위한 사용자 인터페이스 시스템을 개발하였다. 위의 연구들에서는 멀티미디어 데이터베이스에서 프리젠테이션하기 위한 다양한 시공간 연산자 및 키워드를 정의하고 있으나 그 연산자나 키워드들이 비디오 데이터나 이미지 데이터 등과 같은 특정 미디어에 제한적이기 때문에 여러 가지 다양한 응용에 사용할 수 없다는 단점을 가지고 있다. 또는 질의의 결과를 저장하고 제공하는 형식이 각 연구에서 다르게 제공되기 때문에 이를 상호 운용할 수 없다는 단점이 있다.

프리젠테이션을 위한 시공간 연산자를 정의함에 있어서 프리젠테이션 객체들을 모델링하고, 모델링 결과를 통해 연산자로 정의하여 일반응용에서 사용하고자 하는 시도는 [11]과 [12]에서 찾아볼 수 있다. 하지만 이들 연구에서 정의된 연산자들 역시 프리젠테이션을 위한 표준화된 시공간 관계를 정의하고 있지 못하고 있으며, 위의 연구들은 질의어를 설계하는 데에 중점을 두고 있기 때문에, 질의어의 결과들을 다른 응용에서 사용하도록 표준화하는 방안에 대해서는 언급하지 않고 있다.

따라서 본 연구에서는 멀티미디어 질의 결과를 프리젠테이션하기 위한 연산자의 정의뿐만 아니라 질의 결과를 보여주는 방법에 있어서 표준화된 형태로 결과를 저장하고, 저장된 결과를 보편화된 프리젠테이션 프로그램을 사용할 수 있도록 한다.

이를 위해 본 연구에서는 아래에서 살펴볼 SMIL에서 제공하는 멀티미디어 표현 기법들을 질의어에서 수용하고 질의 결과를 SMIL형태로 구성할 수 있도록 함으로써 데이터베이스에서 질의한 결과를 SMIL을 지원하는 일반적인 멀티미디어 사용자 인터페이스에서 수용할 수 있도록 하였다.

### 3. SMIL(Synchronized Multimedia Integration Language)

SMIL은 다양한 브라우저와 플레이어로 읽고 처리할 수 있는 멀티미디어 정보를 표현하는 언어로 웹을 통해 멀티미디어 데이터를 배포하기 위한 목적으로 W3C 표준으로 개발되었다. SMIL은 독립적인 미디어들을 동기화된 멀티미디어 프리젠테이션으로 나타내며, 프리젠테이션의 시간 동기화와 화면상의 프리젠테이션 배치를 기술하며, 다양한 미디어와 포맷을 지원한다.

SMIL을 개발하기 위해 W3C, Bell Labs, RealNetworks, Apple, Netscape, DEC 등의 영향력 있는 기관들이 참여하였으므로, 이미 웹과 함께 널리 배포되어 있는 RealNetworks사의 RealPlayer, Oratrix Development VB 사의 Grins 등 많은 멀티미디어 플레이어들이 SMIL을 지원하고 있다[13-14].

#### 3.1 SMIL 분석

SMIL은 XML(eXtended Markup Language)로 작성된 문서이며, 크게 머리 부분인 HEAD와 몸체 부분인 BODY로 나뉘며, HEAD 요소는 프리젠테이션 될 미디어의 공간적인 배치 정보를 지정하는 LAYOUT 요소와 SMIL 문서에 대한 메타 정보를 나타내는 META 요소로 나뉜다. BODY는 미디어들간의 동기화를 나타내기 위한 SEQ, PAR의 동기화 요소와, 미디어와 미디어의 시공간 정보를 기술하는 미디어 요소로 구성된다[13-14].

SMIL 문서에서 사용될 각 요소들을 보다 자세히 살펴보면 다음과 같다.

#### ● ROOT-LAYOUT 요소

프리젠테이션 될 전체 화면의 크기와 배경 색을 지정하며, 이 요소가 가지는 속성은 아래와 같다.

- 속성 : Background-COLOR, HEIGHT, ID, TITLE, WIDTH

#### ● REGION 요소

전체 화면에서 미디어가 프리젠테이션 될 영역을 정의하며, 다음의 속성들을 가진다.

- 속성 : BACKGROUND-COLOR, FIT=(FILL | HIDDEN | MEET | SCROLL | SLICE), HEIGHT, ID, LEFT, TITLE, TOP, WIDTH, Z-INDEX

Z-INDEX 속성은 프리젠테이션 도중 영역이 겹쳐질 때, 영역의 우선 순위를 나타낸다. FIT속성은 영역내의 미디어 크기의 조정방법에 대한 속성들으로써 영역의 크기에 맞게 미디어어의 크기를 늘이거나 줄이거나, 또는 자르도록 지정한다.

#### ● PAR, SEQ 요소(동기화 요소)

미디어들간의 시간 동기화를 맞추기 위한 요소로서, 이 두 요소는 블록 형태이며, 블록 안에는 미디어 요소 또는 다른 시간 동기화 요소들이 올 수 있다. 본 논문에서는 블록 안의 요소들을 편의상 자식 요소라고 부른다. 시간 동기화 요소는 자식 요소들의 시작 시간을 동기화하고 자식 요소들의 마치는 시간을 제한할 수 있으며, 자식 요소들의 반복 여부를 결정할 수 있다.

동기화 요소는 다음의 속성들을 가진다.

- 속성 : ABSTRACT, AUTHOR, BEGIN, COPY-RIGHT, DUR, END, REPEAT.

속성에서 BEGIN과 DUR, END는 자식 요소들의 전체적인 시작시간과 마치는 시간을 지정하며, REPEAT는 자식 요소의 프리젠테이션 반복 횟수를 나타낸다.

#### ● 미디어 요소(ANIMATION, AUDIO, IMG, TEXT, VIDEO)

미디어 요소에서는 프리젠테이션 될 미디어들과 미디어의 시공간 정보를 정의한다.

사용될 수 있는 미디어로는 애니메이션, 오디오, 이미지, 텍스트, 비디오 등이 있으며 각 미디어가 가질 수 있는 속성은 다음과 같다.

- 속성 : BEGIN, DUR, END, REGION, SRC, CLIP-BEGIN, CLIP-END

SRC는 미디어의 파일 위치를 나타내며, BEGIN, DUR, END는 시간을 정보를 나타낸다. 그리고, REGION은 REGION 요소에서 지정한 공간 영역 중 미디어가 프리젠테이션 될 영역 식별자를 지정한다.

## 4. 멀티미디어 프리젠테이션을 위한 질의어 확장

사용자 질의 결과가 SMIL 문서로 자연스럽게 표현될 수 있도록 하기 위해서는 SMIL에서 제공하는 시공간 정보들을 질의어에서 모두 표현할 수 있어야 한다. 따라서 이 장에서는 앞에서 분석한 SMIL 문법을 바탕으로 질의어를 확장하기 위한 프리젠테이션 문법을 정의하고, 이를 통해 기존의 데이터베이스 질의어인 SQL을 확장한다.

### 4.1 프리젠테이션 문법 정의

본 연구에서는 질의결과의 프리젠테이션 정보를 나타내기 위해 새로운 질인 present질의 추가한다.

추가될 프리젠테이션 질의 문법은 아래 (그림 4.1) 과 같이, 크게 공간을 정의하는 LAYOUT 절과, 미디어들 간의 동기화 및 미디어의 시공간 정보를 정의하는 SHOW절로 나누어 정의한다.

```
<present> ::= [LAYOUT <layout_spec>] SHOW <show_spec>
```

(그림 4.1) 프리젠테이션 문법

위의 LAYOUT 절은 (그림 4.2)과 같이 정의된다.

```
<layout_spec>
:: = [<root_layout_spec>,<region_spec>{,<region_spec>}

<root_layout_spec>
:: = "Main" REGION(<left>,<top>[,<width>,<height>])

<region_spec>
:: = <spatial_id> REGION(<left>,<top>)[WITH <priority>] |
<spatial_id> REGION(<left>,<top>,<width>,<height>)[WITH <priority>] AS <fit>

<fit> ::= "fill" | "hidden" | "meet" | "scroll" | "slice"
```

(그림 4.2) LAYOUT 절 문법

확장된 SQL의 LAYOUT 절은 SMIL의 ROOT-LAYOUT 요소와 REGION요소를 표현하도록 정의하며 키워드 "REGION"을 사용한다.

먼저 공간배치 정보를 나타내는 속성인 TOP, LEFT, HEIGHT, WIDTH를 추가하고, 앞에서 살펴본 영역내의 미디어 조정을 위해 fit절을 정의한다. 또한 미디어들의 영역이 겹칠 때 화면에 나타날 순서는 WITH절의 priority를 통해 정할 수 있다.

LAYOUT 절 안에서 정의한 영역과 영역 메타 정보들은 다음에 설명할 SHOW 절에서 미디어의 시공간 정보를 나타내는 데 참조하기 위해 공간 식별자를 추가로 둔다. 일반 미디어가 나타날 영역과 배경화면은 서로 구분하기 위해 배경화면의 식별자는 항상 "MAIN"이 되도록 하며, 다른 공간을 지정할 때는 이를 식별자로 사용할 수 없도록 한다.

(그림 4.3)은 동기화 정보를 나타내는 SHOW절의 문법이다.

SHOW 절은 미디어의 시공간 정보와 미디어들간의 동기를 나타내기 위한 것으로, SMIL에서 BODY 요소가 포함

```
<show_spec>
:: = (<s-t_sync_spec> AND) <sync_spec>

<s-t_sync_spec>
:: = <selected_attribute_name> <time_constraints> ON <region_id>

<sync_spec>
:: = (<sync_operator> (<present_ele>) <time_constraints>) <time_constraints>

<present_ele>
:: = <attribute_name> |
<attribute_name> {,<present_ele>} |
<sync_operator>(<present_ele>,<present_ele>))
<time_constraints>

<time_constraints> ::= [from <begin>][for <dur> | until <end>]

<sync_operator> ::= SEQ | PAR
```

(그림 4.3) SHOW 절의 문법

하는 동기화 요소와 미디어 요소를 표현한다.

동기화 요소를 프리젠테이션 문법으로 정의하기 위해 SMIL 문법과 마찬가지로, "SEQ"와 "PAR" 키워드를 사용하며, 이들이 하는 역할은 SMIL 과 같다. 동기화 요소의 블록은 괄호로 표현하며, 괄호 내에는 동기화하기 위한 SELECT 절의 애트리뷰트 또는 동기화 키워드가 올 수 있고, 이들 간의 구분은 ","로 한다.

동기화 요소의 속성들 중에서 시간 정보 속성은 "FROM", "FOR", UNTIL"을 사용하여 표현한다.

프리알리기 위해서 SELECT 절의 애트리뷰트 이름을 제일 앞에 기술한다. 그리고, 속성들 중, 시간을 나타내기 위한 BEGIN, DUR, END, REGION, CLIP-BEGIN, CLIP-END 속성들을 프리젠테이션 문법으로 표현한다. BEGIN, DUR, END 는 동기화 요소와 마찬가지로 BEGIN, FOR, UNTIL 키워드로서 정의하며, CLIP-BEGIN과 CLIP-END는 FROM 과 UNTIL 뒤에 시간 값과 선택적으로 기술될 수 있도록 한다. REGION 속성을 나타내기 위해 ON 키워드를 정의하며, ON 키워드를 사용해서 미디어의 공간 정보인 LAYOUT 절의 해당되는 공간 식별자를 지정할 수 있도록 한다.

#### 4.2 확장된 질의어의 예

4.1절에서 정의한 프리젠테이션 문법은 (그림 4.4) 와 같이 기존의 질의어에 프리젠테이션 절로 확장되어 사용된다.

```
<multimedia_query> ::= <query> [<present>]
```

(그림 4.4) 프리젠테이션 문법을 기존 질의어에 확장

사용자의 요구사항이 아래와 같을 때, 본 논문에서 정의한 프리젠테이션스에 질의하기 위해 작성한 질의문은 (그림 4.5)와 같다.

사용자의 요구) 영화 테이블에서 주인공이 "브루스 윌리스"인 사진과 줄거리를 보이며, 줄거리와 사진을 동시에 보이며, 튜플간의 간격은 2초로하고, 하나의 튜플 시간이 10초를 넘지 않게 하며, 두 미디어는 영역(0,0,100,100)에서 보이며, 실제 결과의 미디어 크기가 영역과 맞지 않으면 미디어를 같은 비율로 늘이거나 줄여라.

```
SELECT photo p, synopsis s
FROM movie
WHERE cast = 'Bruce Willise'
LAYOUT "r1" REGION (0,0,100,100) AS "meet"
SHOW p ON r1 AND
s FROM 2 ON r1 AND
PAR(p ,s) FROM 2 DUR 10
```

(그림 4.5) 제안한 프리젠테이션 문법을 사용한 보기

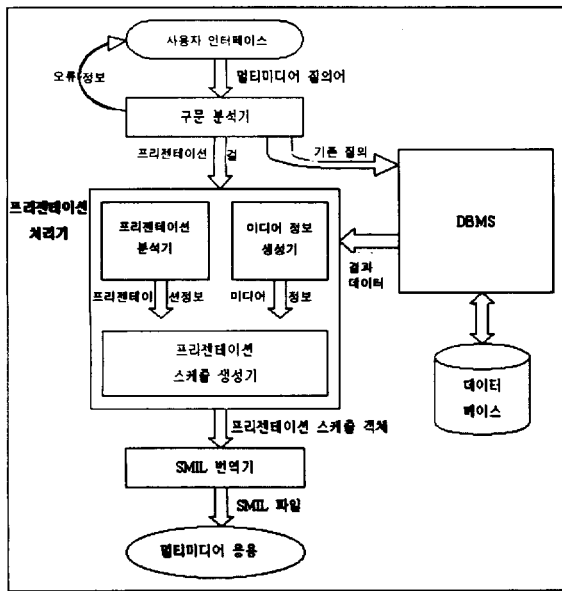
### 5. SMIL을 이용한 프리젠테이션 시스템

사용자가 4장에서 정의한 프리젠테이션 문법을 사용하여 멀티미디어 질의를 주었을 때, 질의를 수행하고 그 결과를

SMIL 문서로 제공하기 위해서, 본 연구에서는 기존의 멀티미디어 데이터베이스 시스템을 확장한 프리젠테이션 시스템을 설계한다.

5.1 시스템 구성

멀티미디어 데이터베이스 시스템에서 프리젠테이션을 지원하기 위한 시스템 구성은 (그림 5.1)과 같이 크게 기존의 멀티미디어 데이터베이스 시스템과 프리젠테이션을 지원하기 위한 모듈인 프리젠테이션 처리기, SMIL 번역기로 구성된다.



(그림 5.1) 프리젠테이션을 지원하는 멀티미디어 데이터베이스 시스템 구성

사용자가 인터페이스를 통해 멀티미디어 질의를 주면, 구문 분석기에서는 프리젠테이션 질의 오류를 검사하고 오류가 없으면, 기존 질의와 프리젠테이션 질을 분리하여 각각을 기존의 데이터베이스 시스템과 프리젠테이션 처리기로 넘겨준다. 그리고, 데이터베이스 시스템은 기존 질의의 오류 여부를 검사한 후 질의를 수행하고 질의 결과를 프리젠테이션 처리기로 넘겨준다.

프리젠테이션 처리기는 프리젠테이션 질과 질의 결과를 프리젠테이션 분석기와 미디어 정보 생성기를 통해서 프리젠테이션 정보와 미디어 정보를 생성하여 프리젠테이션 스케줄 생성기로 넘겨주며, 프리젠테이션 스케줄 생성기는 두 정보들을 사용하여 프리젠테이션 스케줄을 생성하여 SMIL 번역기로 넘겨준다. SMIL 번역기는 메타 객체와 프리젠테이션 스케줄을 통해 SMIL 파일을 생성하며, 생성된 SMIL 파일은 다양한 멀티미디어 응용에서 사용될 수 있다.

5.2 프리젠테이션 처리기

프리젠테이션 처리기는 구문 분석기로부터 받은 프리젠테이션

질과 데이터베이스 시스템으로부터 받은 결과 데이터를 해석하여 프리젠테이션 정보와 미디어 정보를 생성하고, 이 두 정보를 사용해서 프리젠테이션 스케줄 객체를 생성한다.

5.2.1 미디어 정보 생성기

미디어 정보 생성기는 기존의 멀티미디어 데이터베이스 시스템으로부터 질의 결과를 받고, 결과에서 SELECT 절의 각 애트리뷰트에 해당하는 미디어의 종류와 파일 이름, 바이너리 데이터를 차례로 꺼낸 뒤, 미디어 종류와 파일 이름으로 (그림 5.2)의 구조를 가지는 결과 미디어 객체를 생성한다.

클래스 이름 : <b>ResultMedia</b> (결과 미디어 클래스)		
설명 : 결과 미디어의 정보		
변 수		
filename	String	미디어의 파일 이름
mediaType	int	미디어 종류 IMG : 1, ANIMATION : 2, AUDIO : 3, TEXT : 4, VIDEO : 5

(그림 5.2) 결과 미디어 객체 구조

생성된 결과 미디어 객체는 (그림 5.3)의 결과 미디어 집합인 resultMediaSet에 순서대로 삽입되며, 미디어의 바이너리 데이터는 SMIL 문서에서 참조할 수 있도록 해당하는 이름의 파일로 생성된다.

미디어 정보 생성기가 최종적으로 반환하는 (그림 5.3)의 미디어 정보 객체이다.

클래스 이름 : <b>MediaInfo</b> (미디어 정보 클래스)			
설명 : 데이터베이스에서 검색된 결과 미디어 정보			
변 수			
selectedAttrList	String []	Select 절의 애트리뷰트들의 순서화 된 리스트	
attrCount	int	Select 절의 애트리뷰트 개수	
resultMediaSet	ResultMedia[]	결과 미디어들의 정보	
mediaCount	int	결과 미디어들의 개수	
메 소 드			
MediaInfo(String[], ResultMediaInfo[])	선택된 속성들의 순서화 된 리스트, 결과 미디어 정보	없음	생성자로서, 미디어 정보 클래스의 오브젝트를 생성한다.
String[] getAttrOrderedList()	없음	선택된 속성들의 리스트	애트리뷰트의 순서화된 리스트를 얻는다.
int getAttrCount()	없음	선택된 속성들의 개수	애트리뷰트의 개수를 얻는다.
ResultMediaInfo[] getResultMediaSet()	없음	결과 미디어 정보	결과 미디어들의 정보를 얻는다.
int getMediaCount()	없음	결과 미디어 개수	결과 미디어의 개수를 얻는다.

(그림 5.3) 미디어 정보 객체 구조

5.2.2 프리젠테이션 분석기

프리젠테이션 분석기는 프리젠테이션 절인 LAYOUT 절과 SHOW 절을 분석하여 (그림 5.4)의 프리젠테이션 정보 객체를 생성한 후 프리젠테이션 스케줄 생성기로 넘겨준다.

프리젠테이션 정보 객체는, 미디어들이 공간상에 배치될 정보들을 가지는 spatialInfoSet 객체, 미디어들의 시공간 정보들을 가지는 STInfoSet 객체, 미디어들을 동기화하기 위한 syncInfoSet 객체, 그리고 동기화 연산자의 시간 정보 등을 나타내는 syncOptInfoSet 객체로 구성된다. spatialInfoSet 객체와 STInfoSet 객체를 구성하는 공간 정보 객체와 시공간 정보 객체는 프리젠테이션 절을 차례로 읽어가며 해당되는 정보를 추출하여 생성한다.

클래스 이름 : PresentInfo (프리젠테이션 정보 클래스)			
설명 : 프리젠테이션을 위한 정보			
변 수			
속 성 명			
spatialInfoSet	SpatialInfo[]	미디어가 프리젠테이션 될 공간 객체들 정보	
STInfoSet	STInfo[]	미디어의 시공간 정보 객체들 정보	
syncInfoSet	syncInfo[]	미디어들간의 시간 동기화 정보	
syncOptInfoSet	syncOptInfo[]	동기화 연산자의 정보	
메 소 드			
메 소 드 명			
void PresentInfoSet(Spatial[], STsync[], String)	공간 객체들, 시공간 동기화 객체들, 동기화정보	없음	생성자로서 프리젠테이션 정보 오브젝트 생성
SpatialInfo[] getSpatialInfoSet()	없음	공간 객체들	미디어들의 공간 정보 얻음
STInfo[] getSTInfoSet()	없음	시공간 동기화 객체들	미디어들의 시공간 동기화 정보를 얻음
String getSyncInfoSet()	없음	동기화객체들	미디어들의 시간 동기화 정보를 얻음

(그림 5.4) 프리젠테이션 정보 객체 구조

동기화 정보들을 가지는 syncInfoSet 객체와 동기화 연산자들에 대한 시간 정보와 메타 정보를 가지는 syncOptInfoSet 객체를 구성하는 동기화 정보 객체의 구조는 (그림 5.5)에 나와 있으며, 동기화 연산자 정보 객체에 대한 구조는 (그림 5.6)에 나와 있다.

클래스 이름 : syncInfo (동기화 정보 클래스)		
설명 : 동기화 정보		
변 수		
속 성 명		
elementType	int	요소 형, 1: SEQ, 2: PAR, 0: Media
elementId	String	동기화 연산자의 식별자 또는 select절의 속성 이름
status	int	동기화 연산자들의 블록의 시작과 끝을 지정 동기화 연산자의 블록의 시작은 0, 끝은 1

(그림 5.5) 동기화 정보 객체 구조

동기화 정보 클래스에서 동기화 연산자의 블록은 속성, status로 나타내며, 이 속성으로서 동기화 연산자가 여러 미디어 또는 다른 동기화 연산자 블록을 포함할 수 있도록 한다. syncInfoSet 객체와 syncOptInfoSet 객체를 구하기 위한 알고리즘은 (그림 5.7)에 나와 있다.

클래스 이름 : SyncOptInfo (동기화 연산자 정보 클래스)		
설명 : 동기화 연산자를 위한 시간 정보 및 기타 정보		
변 수		
속 성 명		
id	String	동기화 연산자 식별자 (root는 id: "root")
type	int	동기화 연산자 타입 (1: SEQ, 2: PAR, 3: 선택된 속성 이름)
timeInfo	TimeInfo	연산자 블록 내의 자식들의 시간 제약
loop	int	자식들의 프리젠테이션 반복 횟수

(그림 5.6) 동기화 연산자 정보 객체 구조

초기화 :

- SHOW 절 내에서 동기화를 기술한 문장을 추출하여 syncClause에 할당
- syncInfoSet, syncOptInfoSet 초기화  
syncInfoSet : ("(", ")", ",", " ") 개수만큼 SyncInfo 배열 할당  
syncOptInfoSet : ("SEQ", "PAR") 개수 + 1만큼 SyncOptInfo 배열 할당
- Index 초기화  
Sindex = 0 : syncInfoSet의 내부 인덱스를 0으로 함  
Oindex = 0 : syncOptInfoSet의 내부 인덱스를 0으로 함
- 스택 초기화  
Stack의 인덱스를 0으로 함

수행 : syncClause를 단어 단위로 스캔하면서 null을 만날 때까지 다음을 수행.

- PAR 또는 SEQ를 만나면
  - 아이디 생성기에서 ID 생성하여, Stack에 삽입.
  - SyncInfo 객체 생성  
elementId = 생성된 ID, elementType = 1 또는 2(PAR 또는 SEQ)  
status = 0(BEGIN)
  - syncInfoSet에 객체 삽입, Sindex 증가
- 키워드 이외의 단어 즉, SELECT 절의 에트리뷰트 이름을 만나면
  - SyncInfo 객체 생성  
elementId = 에트리뷰트 이름, elementType = 0(Media) status = 1(END)
  - syncInfoSet에 객체 삽입, Sindex 증가
- )"를 만나면
  - SyncInfo 객체 생성  
elementId = 스택에서 꺼낸 ID 입력, elementType = 스택에서 꺼낸 type 입력,  
status = 2(END)
  - syncInfoSet에 삽입, Sindex 증가
  - )" 다음 단어가 ", " 또는 ")"가 아니라면 즉, 동기화 연산자에 대한 시간 정보가 뒤에 오는 경우
  - SyncOptInfo 객체 생성  
ID, TYPE = 현재의 Sindex가 가리키는 elementId, elementType 입력
  - TimeInfo 클래스의 객체 생성  
BEGIN = FROM 이하 문자 입력, DUR = FOR 이하 문자 입력
  - syncOptInfoSet에 객체 삽입, Oindex 증가

(그림 5.7) syncInfoSet 객체 및 syncOptInfoSet 객체 생성 알고리즘

5.2.3 프리젠테이션 스케줄 생성기

프리젠테이션 스케줄 생성기는 프리젠테이션 분석기에서 반환한 프리젠테이션 정보 객체와 결과 미디어 생성기에서 반환한 미디어 정보 객체를 입력으로 받아들여 프리젠테이션 스케줄 객체를 생성하고, SMIL 변환기로 반환한다. 프리젠테이션 스케줄 객체는 SMIL 변환기 내에 위치하고, 내부 카운터를 사용하여 SMIL 변환기가 SMIL의 요소들을 작성하는 데 필요한 메타 정보 객체들을 차례대로 하나씩 반환한다.

SMIL 변환기가 SMIL 문서의 HEAD를 작성할 때와 BODY의 동기화 요소 부분, 그리고 미디어 요소 부분을 작성할 때, 필요한 정보가 각각 다르기 때문에, 프리젠테이션 객체는 입력으로 들어온 프리젠테이션 스케줄 객체와 미디어 정보 객체 내의 필요한 객체들을 참조하여 메타 정보 객체를 생성한다.

SMIL 변환기가 HEAD를 작성할 때 프리젠테이션 스케줄 객체는 프리젠테이션 정보 객체 내의 spatialInfoSet 객체로부터 해당 카운터인 (그림 5.13)의 spatialCounter가 가리키는 공간 정보 객체를 추출하여 (그림 5.8)의 공간 메타 정보 객체를 생성하여 반환하며, SMIL 변환기의 다음 요청을 위해 spatialCounter를 증가시킨다. 또한, SMIL 변환기가 BODY를 작성할 때, 프리젠테이션 스케줄 객체는 프리젠테이션 정보 객체의 syncInfoSet 객체를 참조하며, 동기화 타입 속성인 syncType가 SEQ 또는 PAR의 동기화 연산자이고 status가 1, 즉 동기화 연산자의 블록의 시작이면 syncOptInfoSet 객체에서 해당 동기화 카운터인 syncCounter의 연산자의 식별자와 같은 동기화 연산자 정보 객체를 추출하여 (그림 5.9)의 연산자 메타 정보 객체를 생성하여 반환한다.

클래스 이름 : <b>SpatialMetaInfo</b> (공간 메타 정보 클래스)		
설명 : SMIL 문서에서 REGION 요소와 LAYOUT 요소를 작성하기 위한 공간 배치 정보		
변 수		
spatialInfo	Spatial	공간 객체

(그림 5.8) 공간 메타 정보 객체 구조

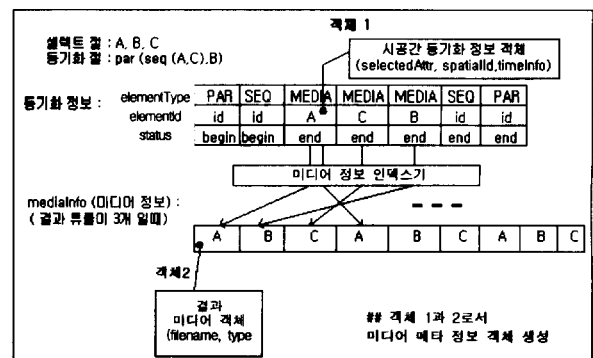
클래스 이름 : <b>OptMetaInfo</b> (연산자 메타 정보 클래스)		
설명 : 동기화 연산자 요소, 즉 SEQ, PAR 요소를 작성하기 위한 시간정보 및 자식 요소들의 반복 정보		
변 수		
optType	int	동기화 연산자 타입, 1 : SEQ, 2 : PAR, ( ) : Media
timeInfo	TempInfo	동기화 연산자의 시간 정보로서, 동기화 정보 객체의 status가 1이면 (end) null 값이 들어감
loop	int	연산자 블록 내의 자식들 반복 횟수 정보

(그림 5.9) 연산자 메타 정보 객체 구조

그리고, SMIL 변환기의 다음 요청을 위해 syncCounter를 증가시킨다. 만약 상태가 2, 즉 동기화 연산자의 블록의 끝이면 연산자 메타 객체의 시간 정보 값을 널로 하여 반환한다. 또한 syncInfoSet 객체의 현재 syncCounter가 가리키는 요소형인 elementType이 미디어라면, 현재 미디어 카운터를 미디어 인덱스를 통해 변환시켜 미디어 정보에서 해당하는 결과 미디어 객체를 얻고, STInfoSet 객체에서 현재 미디어 식별자와 같은 미디어 정보 객체를 추출한 후 (그림 5.10)의 미디어 메타 정보 객체를 생성한 후 반환하고, 미디어 카운터와 동기화 카운터를 증가시킨다. 미디어 메타 정보 객체를 생성하는 과정은 (그림 5.11)과 같다. 만약, 동기화 카운터가 가리키는 동기화 정보가 널 값을 가진다면, 즉 syncInfoSet 객체를 모두 읽었다면, mediaInfoSet 객체에서 결과 미디어 객체가 더 있는지 점검하는 함수를 통해 여부를 확인하고, 결과 미디어 객체가 남아 있다면, 동기화 카운터를 초기화시킴으로써 다시 처음부터 동기화 정보를 읽도록 한다. 이로써, 결과 튜플이 여러 개 있을 때, 모든 튜플의 미디어들을 동기화 시킬 수 있으며, 튜플들 간의 동기화를 시킬 수 있도록 한다.

클래스 이름 : <b>MediaMetaInfo</b> (미디어 메타 정보 클래스)		
설명 : 미디어 요소를 완성하기 위한 메타 정보		
변 수		
mediaType	int	미디어 타입, 1 : IMG, 2 : ANIMATION, 3 : AUDIO
timeInfo	TimeInfo	미디어의 시간 정보
filename	String	미디어의 파일 이름
region	String	미디어가 프리젠테이션 될 영역 식별자

(그림 5.10) 미디어 메타 정보 객체 구조



(그림 5.11) 미디어 메타 정보 객체 생성

(그림 5.11)에서 미디어 정보 인덱스기는 프리젠테이션 정보 객체 내의 syncInfoSet 객체의 미디어 순서와 mediaInfoSet 객체의 미디어 순서의 동기화를 맞추기 위한 것이다. 미디어 정보 인덱스기는 syncInfoSet 객체의 해당 동기화 카운터가 가리키는 요소형이 미디어일 때, 현재 미

디어 카운터를 mediaInfoSet 객체 내의 결과 미디어 정보 인 결과 미디어 객체를 가져오기 위한 mediaInfoSet 객체의 인덱스로 변환한다. mediaInfoSet 객체의 인덱스를 구해 내는 과정은 다음과 같다.

mediaInfoSet 객체로부터 SELECT 절의 애틀리뷰트의 순서를 가지고 있는 selectedAttrList와 syncInfoSet 객체 내의 속성 순서를 비교하여 동기화 정보 내의 속성이 SELECT 절의 속성 리스트의 몇 번째에 속하는지를 파악 하여 인덱스 리스트를 구성하고, (그림 5.12)의 식을 통해서 미디어 정보 내의 결과 미디어 객체의 순서를 얻어낸다.

현재 미디어 카운터 : counter (그림 4.13)의 예  
 속성 개수 : attrCount attrCount = 3  
 인덱스 리스트 : I I : [0] [2] [1]  
 결과 미디어 위치 = counter / attrCount \* attrCount + I[counter MOD attrCount]  
 예) counter : 5 ⇒ 정보 미디어의 순서는 (5/3) \* 3 + I[2] = 1 \* 3 + 1 = 4  
 따라서 현재 미디어 카운터가 5이면, 미디어 정보에서 4번째 결과 미디어 객체 참조

(그림 5.12) 미디어 인덱스

클래스 이름 : PresentSchedule (프리젠테이션 스케줄 클래스)			
설명 : SMIL 변환기에서 태그를 만들기 위한 메타 정보 객체를 반환			
변수			
속성명	타입	설명	비고
spatialInfo	Spatial[]	공간 정보	
stSyncInfo	STsyncInfo[]	시공간 동기화 정보	
syncInfo	SyncInfo[]	동기화 정보	
syncOptInfo	SyncOptInfo[]	동기화 연산자 정보	
mediaInfo	ResultMedia[]	미디어 정보	
mediaCounter	int	미디어 정보 객체 카운터	
spatialCounter	int	공간 정보 객체 카운터	
syncCounter	int	동기화 정보 객체 카운터	
mediaCount	int	결과 미디어 개수	
메소드			
메소드명	타입	설명	비고
void PresentSchedule(SpatialInfo[], STsyncInfo[], SyncInfo[], SyncOptInfo[], MediaInfo[], MediaCount)		프리젠테이션 정보들	없음 각 카운터들 초기화, 프리젠테이션 스케줄러 객체 생성
Spatial getNextSpatialInfo()		없음	공간 객체 spatialInfo에서 현재 Spatial 객체 반환, spatialCounter++
OptMetaInfo getNextOptInfo()		없음	동기화 태그를 위한 메타 정보 syncOptInfo에서 생성된 동기화 연산자 메타 정보 객체를 반환, syncCounter++
MediaMetaInfo getNextMediaInfo()		없음	미디어 태그를 위한 메타 정보 mediaInfo와 st-syncInfo로부터 미디어 메타 정보 객체를 생성, 반환, syncCounter++
boolean checkMedia()		없음	현재 syncInfo가 미디어인지 여부 현재 syncInfo가 미디어인지 점검
boolean checkNextMedia()		없음	미디어 정보가 더 남아있는지 여부 현재 미디어 카운터의 결과 미디어 객체가 더 있는지 여부를 반환

(그림 5.13) 프리젠테이션 스케줄 객체 구조

프리젠테이션 정보 객체와 mediaInfoSet 객체로부터 메타 정보들을 생성하고 하나씩 반환하는 스케줄을 처리하는 프리젠테이션 스케줄 객체는 (그림 5.13)과 같다.

5.3 SMIL 변환기

SMIL 변환기는 프리젠테이션 처리엔지니어링 스케줄 객체와 SMIL의 태그 정보와 태그내 속성정보를 가지고 있는 (그림 5.14)의 SMIL 메타 객체를 통해서 결과에 대한 SMIL 문서를 작성한다.

SMIL 변환기는 SMIL 메타 객체로부터 SMIL의 전체 태그들의 순서값을 가지고 있는 속성인 basicTagInfoSet으로부터 해당 태그의 위치를 가리키는 속성인 tagPointer를 사용해서 태그를 하나씩 읽으면서 SMIL 파일을 작성한다. SMIL 변환기가 basicTagInfoSet으로부터 "LAYOUT" 태그를 읽으면, 프리젠테이션 스케줄 객체에 공간 메타 정보 객체를 요청한다. 프리젠테이션 스케줄 객체는 spatialInfoSet 객체를 spatialCounter에 따라 하나씩 공간 메타 정보 객체로 생성하여 반환하며, SMIL 변환기는 받은 객체로부터 공간 식별자가 "Main"인지 아닌지를 점검하고, SMIL 파일에 ROOT-LAYOUT 요소를 작성할 것인지 REGION 요소를 작성할 것인지 결정하며, 이에 따라 SMIL 메타 객체로부터 rootLayoutTagInfo 객체 또는 regionTagInfo 객체를 가져와서 SMIL 파일을 작성한다.

클래스 이름 : SmilMeta (SMIL 메타 클래스)		
설명 : SMIL 파일을 작성하기 위한 태그 정보와 태그 내 속성 정보를 가짐		
변수		
속성명	타입	설명
basicTagInfoSet	String[]	SMIL의 전체 틀을 구성하는 기본 태그들
tagPointer	int	basicTagInfoSet을 읽어나가는 지시자
layoutTagInfo	LayoutTagInfo	LAYOUT의 태그 정보를 가짐
seqTagInfo	SEQTagInfo	SEQ 태그를 작성하기 위한 정보
parTagInfo	PARTagInfo	PAR 태그를 작성하기 위한 정보
mediaTagInfo	MediaTagInfo	미디어 태그 정보들
메소드		
메소드명	타입	설명
void Meta()		없음
getBasicTag(int)	tagPointer	기본 태그
		인덱스에 해당하는 기본 태그를 반환한다.

(그림 5.14) SMIL 메타 객체 구조

"LAYOUT" 요소를 모두 작성한 후, 다시 SMIL 메타 객체의 basicTagInfoSet 속성으로부터 "/LAYOUT" 이하의 태그들을 결과 SMIL 파일에 쓴다. 이때, SMIL 변환기가 "BODY" 태그를 읽으면, 프리젠테이션 스케줄 객체에 현재 syncInfoSet 객체 내의 요소 형 속성인 elementType이 미디어인지 동기화 연산자인지를 점검하여 미디어 정보 객체 또는 동기화 연산자 정보 객체를 요청하며, 프리젠테이션 스케줄 객체는 해당되는 정보를 내부 카운터들에 의해



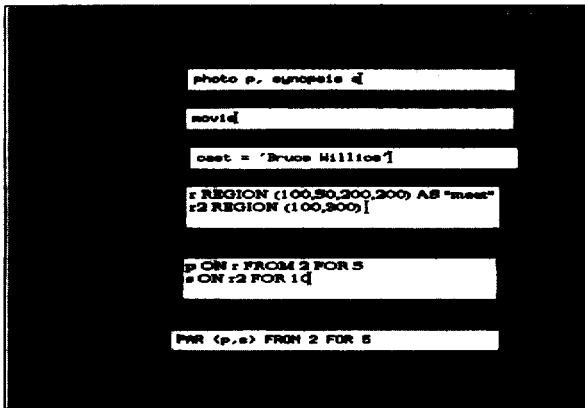
메타 객체로 생성하여 반환한다.

프리젠테이션 스케줄 객체로부터 반환된 메타 객체의 요소형이 연산자형 또는 미디어형인가에 따라 "SEQ", "PAR", "IMG", "ANIMATION" 등의 요소를 작성하며, 만약 OptsyncInfo 객체내의 시간 정보인 timeInfo 속성의 값이 널(null)이면, 해당 동기화 연산자를 담기 위한 태그 정보를 가져와서, SMIL 파일에 작성한다. 프리젠테이션 스케줄 객체가 모든 syncInfoSet 객체를 읽기를 마쳤으면 널 값을 반환하며, SMIL 변환기는 "/BODY" 이하의 태그를 작성하며, "/SMIL" 태그를 만나면 SMIL 파일을 닫는다. 생성된 SMIL 파일은 사용자 또는 멀티미디어 응용 프로그램으로 전달되어 SMIL 플레이어를 통해 프리젠테이션 되거나 여러 멀티미디어 응용들에서 사용될 수 있다.

5.4 사용자 질의 및 질의 결과 프리젠테이션

본 논문에서는 멀티미디어 데이터베이스로 오브젝트 디자인사의 오브젝트 스토어(ObjectStore)를 사용하였으며, 프리젠테이션 처리기 및 사용자 인터페이스는 JAVA1.2를 사용하여 구현하였다.

사용자가 본 논문에서 정의한 프리젠테이션 문법을 사용하여 질의를 (그림 5.15)와 같이 질의 입력창을 통해 작성



(그림 5.15) 사용자 질의 입력 창

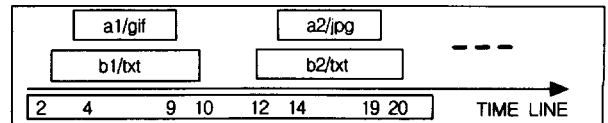
```

<SMIL>
<HEAD>
<LAYOUT>
<REGION "r" top="50" left="100" width="200" height="200"
fit="meat"/>
<REGION "r2" top="300" left="100"/>
</LAYOUT>
<BODY> <!-- 전체 결과가 3 개일 때 -->
<SEQ>
<PAR begin="2" dur="10">
<IMG src="a1.gif" region = "r1" begin="2" dur="5"/>
<TEXT src="b1.txt" region="r2" dur="10"/>
</PAR>
<PAR begin="2" dur="10">
<IMG src="a2.jpg" region = "r1" begin="2" dur="5"/>
<TEXT src="b2.txt" region="r2" dur="10" />
</PAR>
<PAR begin="2" dur="10">
<IMG src="a3.jpg" region = "r1" begin="2" dur="5"/>
<TEXT src="b3.txt" region="r2" dur="10"/>
</PAR>
</SEQ>
</BODY>
</SMIL>
    
```

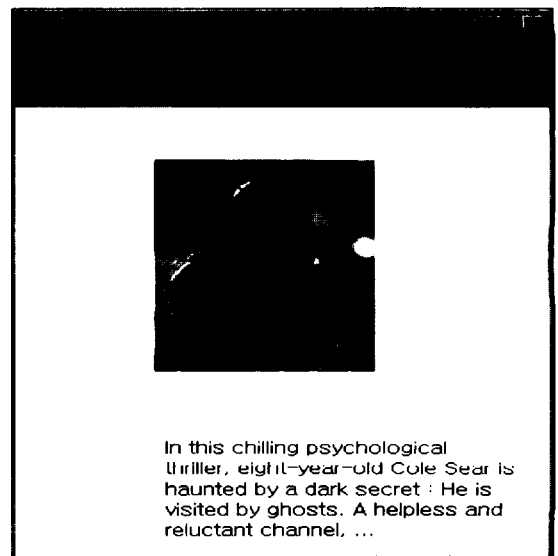
(그림 5.16) 번역된 SMIL 파일

하였을 때, 프리젠테이션 처리기와 SMIL 번역기를 통해 결과로 제공되는 SMIL 파일은 (그림 5.16) 과 같다.

위의 SMIL 파일에 따른 결과 미디어들 간의 시간 동기화는 (그림 5.17)과 같으며, 4초에서 9초에 해당하는 질의 결과를 Oratrix Development VB사의 Grins 플레이어를 통해 프리젠테이션하는 화면은 (그림 5.18)과 같다.



(그림 5.17) 미디어들 간의 시간 동기화



(그림 5.18) Grins 플레이어를 통한 프리젠테이션

6. 결 론

본 논문에서는 멀티미디어 데이터베이스에서 보다 일반화된 멀티미디어 프리젠테이션을 위하여 SMIL을 기반으로 질의어를 확장하고 질의 결과를 다시 SMIL 문서로 재구성할 수 있는 방법을 제공함으로써 여러 멀티미디어 응용에서 사용할 수 있도록 하였다.

SQL을 확장하기 위한 멀티미디어 프리젠테이션 문법은 SMIL 문법을 분석하여 시공간을 표현하는 요소 및 속성을 찾아내어 프리젠테이션 문법의 키워드로 정의하고 이를 통해 SQL을 확장하였다. SMIL에서 지원하는 연산자를 이용하여 튜플 내 미디어들간의 동기화 뿐 아니라, 튜플들 간의 동기화 또한 가능하도록 프리젠테이션 문법을 정의했다. 또한 데이터베이스에서 질의를 수행하여 얻은 질의 결과 데이터를 분석하여 결과 미디어 수, 결과 미디어 형, 결과 미디어의 크기 및 시간 정보를 추출하여 질의 결과를 동적으로 SMIL 에 적용할 수 있도록 프리젠테이션 처리기시스템을 설계 및 구현하였다.

본 논문에서 정의한 프리젠테이션 문법과 프리젠테이션 시스템은 기존의 시스템과는 독립적으로 설계 및 구현함으로써 여러 가지 기존의 데이터베이스 시스템을 쉽게 확장할 수 있도록 하였으며, 멀티미디어 질의 결과의 W3C의 멀티미디어 프리젠테이션 표준인 SMIL 문서로 제공되므로, 광범위하게 배포되어 있는 기존의 플레이어들을 통해서 결과를 프리젠테이션하거나 다양한 멀티미디어 응용에서 결과를 사용할 수 있다. 또한 SMIL의 멀티미디어 프리젠테이션 표준을 따라 시공간 관련 정보들을 표현했으므로, 멀티미디어 질의어 측면에서도 보다 일반화 할 수 있었다는 측면에서 의의를 가진다.

향후 연구과제로는 확장이 가능한 질의어 구조와 구문 분석기에 대한 연구, 그리고 SMIL 플레이어에서 실시간으로 데이터베이스에 접근하는 방안 및 이를 스케줄링하는 알고리즘에 대한 연구가 필요하다.

**참 고 문 헌**

[1] G. Blakowski, et. al., "A Media Synchronization Survey : Reference Model, Specification, and Case Studies," IEEE Journal on Selected Areas in Communications, Vol.14, No.1, Jan. 1996.

[2] T. D. C. Little, A. Ghafoor, "Multimedia Object Models for Synchronization and Databases," Proc. of 6th Data Engineering, 1990.

[3] Setrag, Khoshafian, A. Brad Baker, "Multi Media and Imaging Databases," Morgan Kaufmann Publishers, Inc. 1996.

[4] John Z, Li "Modeling and Querying Multimedia Data," University of Alberta, University of Alberta, Department of Computing Science, 1998.

[5] E. Oomoto, K. Tanaka, "OVID : Design and implementation of a video-object database system," ACM Multimedia Systems, 5(4) : 629-643. 1993.

[6] N. Roussopoulos, C. Faloutsos, T. Sellis. "An efficient pictorial database system for PSQL," IEEE Transactions on Software Engineering, 14(5) : 639-650. 1998.

[7] R. Ahad, A. Basu. "ESQL : A query language for the relational model supporting image domains," In Proceedings of the 7th International Conference on Data Engineering, 550-559. 1991.

[8] John, Z. Li, M.T. Ozsu. "MOQL : A multimedia object query language," Laboratory for Database Systems Research, Department of CS, University of Alberta. 1997.

[9] A. F. Cardenas, I. T. Jeong, R. K. Taira. "The knowledge based object oriented PICQUERY + language," IEEE Transactions on knowledge and Data Engineering, 5(4) : 644-657, August 1993.

[10] Graciela Gonzalez, Chitta Baral, Amarendra Nandigam, "SQL+D : Extended Display Capabilities for Multimedia Database Queries," University of Texas at El Paso Dept. CS, ACM Multimedia 98-Electronic Proceedings, 1998.

[11] M. Vazirgiannis, "Multimedia Database object and application modeling," Multimedia Database System, Kluwer Academic Pub. 1996.

[12] M. Iino, Y. F. Day, and A. Ghafoor, "An Object-Oriented

Model for Spatio-Temporal Synchronization of Multimedia Information." In Proc. IEEE Intl. Conference on Multimedia computing and Systems, pp.110-119, Boston, Ma, May 1994.

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

[14] 이승준. "XML", KMK 정보산업연구원. 1999.



**이 증 화**

e-mail : jhlee@asadal.cs.pusan.ac.kr  
 1992년 부산대학교 전자계산학과(이학사)  
 1995년 부산대학교 전자계산학과(이학석사)  
 1995년~현재 부산대학교 전자계산학과  
 박사과정  
 관심분야 : 데이터베이스, 멀티미디어, XML,  
 한글/한말 정보처리



**이 증 환**

e-mail : jhwlee@asadal.cs.pusan.ac.kr  
 1996년 부산대학교 전자계산학과(이학사)  
 1998년 부산대학교 전자계산학과(이학석사)  
 1998년~현재 부산대학교 전자계산학과  
 박사과정  
 관심분야 : 데이터베이스, 인터넷 응용 프로  
 토클



**유 영 호**

e-mail : yhyou@asadal.cs.pusan.ac.kr  
 1994년 부산대학교 전자계산학과(이학사)  
 1997년 부산대학교 전자계산학과(이학석사)  
 1997년~현재 부산대학교 전자계산학과  
 박사과정  
 관심분야 : 데이터베이스, 멀티미디어, XML



**김 경 석**

e-mail : gimgs@asadal.cs.pusan.ac.kr  
 1977년 서울대학교 무역학과 졸업(경제학사)  
 1979년 서울대학교 전자계산학과(이학석사)  
 1988년 일리노이 주립대(어바나-샴페인)  
 전자계산학 박사  
 1988년~1992년 미국 노스다코타 주립대학  
 교 전자계산학과 조교수

1992년~현재 부산대학교 전자전기정보 컴퓨터 공학부 부교수  
 1993년~현재 문자 코드 전문위원회 의장  
 1994년~현재 국어 정보학회 이사  
 1997년~현재 문헌 정보 및 한글 로마자 전문위원회 위원  
 1992년~현재 국제 표준화 기구(ISO) 회의에서 국가 대표로 활동  
 관심분야 : 데이터베이스, 멀티미디어, 한글/한말 정보처리