

# 멀티미디어 데이터베이스의 프레젠테이션을 위한 멀티미디어 검색 질의어

나 연 목<sup>†</sup>

요 약

본 논문에서는 검색 빈도가 높고 검색 형태가 다변적인 멀티미디어 응용 분야에서 간단한 질의어만을 사용해 멀티미디어 데이터가 포함된 질의 검색 결과를 멀티미디어 답게 시공간상에 배치해 프레젠테이션해주는 멀티미디어 검색 질의어(SQL/MR:SQL/Multimedia Retrieval)를 제안하였다. 특히, SQL/MR의 주요 구문과 질의예, 그리고 SQL/MR이 사용하는 주요 태그의 처리 방법을 제시하였다. SQL/MR은 SQL의 SELECT절만을 확장한 것으로 사용자가 매번 질의시마다 저작 도구를 사용해야 하는 번거로움을 제거해 준다. SQL/MR은 모노미디어 데이터 타입(또는 플라스) 형태로 멀티미디어 데이터를 분리 저장하는 대부분의 현존 관계 DBMS, 객체지향 DBMS, 객체관계 DBMS에서 모두 활용될 수 있다.

## A Multimedia Retrieval Query Language for Multimedia Database Presentation

Yunmook Nah<sup>†</sup>

ABSTRACT

This paper proposes a multimedia retrieval query language SQL/MR(SQL/Multimedia Retrieval) to present query results including multimedia data temporal-spatially for retrieval-oriented multimedia applications. Especially, we describe the syntax of SQL/MR, query examples, and algorithms to process the temporal-spatial layout tags. SQL/MR extends the SELECT clause of SQL to eliminate time-consuming authoring efforts for every query retrieval. SQL/MR can be utilized in relational DBMSs, object-oriented DBMSs, and object-relational DBMSs, which store multimedia data separately in monomedia type fields.

### 1. 서 론

컴퓨터 하드웨어의 급속한 발전, 광학 저장 장치의 출현, 컴퓨터 통신 기술의 성숙으로 인하여 수치와 문자 이외의 대형 멀티미디어 데이터를 입력, 저장,

처리하는 일이 가능해지고 있다. 컴퓨터의 처리 대상으로 보면 텍스트, 이미지, 오디오, 비디오 등의 멀티미디어 데이터는 크기도 크고 처리도 복잡하지만 인간의 입장에서 보면 단순한 수치와 문자 데이터 보다 훨씬 대하기가 자연스럽고 이해가 용이하다.

기존의 데이터베이스 시스템에서 관리해온 데이터는 숫자와 문자 데이터였다. 예를 들어, 사무 응용의 경우, 직원 테이블은 직원번호(숫자), 이름(문자), 주소(문자), 봉급(숫자) 등으로 이루어져 있다. 멀티미디어

※본 연구는 1995년도 단국대학교 대학 연구비의 지원으로 연구되었음.

† 정 회 원: 단국대학교 전자·컴퓨터공학부  
논문접수: 1996년 10월 14일, 심사완료: 1996년 4월 25일

어 기술의 발전으로 직원 테이블에 직원의 사진(이미지), 자기소개 음성(오디오), 이력서(텍스트) 등을 추가하는 일이 가능해지고 있다. 본 논문에서는 기존의 숫자나 문자 데이터는 정형 데이터(formatted data)라 하고, 정형 데이터 이외의 새롭게 추가된 데이터를 총칭하여 멀티미디어 데이터(multimedia data)라고 하고, 이러한 멀티미디어 데이터를 저장한 데이터베이스를 멀티미디어 데이터베이스라고 부르기로 한다. 텍스트, 이미지, 오디오, 비디오 중 하나를 치칭할 때는 모노미디어 데이터(monomedia data)라는 용어를 사용하기로 한다.

멀티미디어 데이터를 지원해주려는 노력은 관계, 객체 지향, 객체 관계 DBMS에서 각기 시도되고 있다. Sybase, Informix-OnLine, Rdb, Interbase, Starburst 등 대부분의 상용 관계 DBMS는 모노미디어 데이터를 위한 BLOB(Binary Large Object) 타입이나 TEXT, IMAGE, AUDIO 데이터 타입과 이들을 조작하기 위한 연산(캡처, 압축, 프레젠테이션)을 DBMS 개발자가 추가적으로 정의 지원하는 방법을 사용하고 있고, 모노미디어 데이터를 포함한 질의어 사용이 가능하다. Orion, VODAK, Jasmine 등의 객체 지향 DBMS는 모노미디어 데이터와 사용자 정의 클래스를 모두 지원하므로 복잡한 멀티미디어 정보의 저장 관리도 가능하다. 그러나 멀티미디어에 대한 질의 기능은 취약한 편이다. UniSQL이나 Illustra 등의 객체 관계 DBMS(object-relational DBMS)는 관계 DBMS와 객체 지향 DBMS의 기능을 모두 지원하므로 복잡한 멀티미디어 데이터의 지원도 가능하고 이러한 정보에 대한 질의 수행도 가능하다. 객체 지향 DBMS와 마찬가지로 모노미디어 클래스와 사용자 정의 클래스를 모두 지원한다.

검색 질의의 수행 결과는 세 가지 유형의 DBMS에서 모두 기존의 방법에 비해 개선되지 않고 있다. 검색 결과는 단순히 테이블 형태로 보여지며 멀티미디어 데이터 필드는 데이터 타입이나 관련 파일명만을 보여준다. 멀티미디어 데이터는 관련 필드마다 매번 클릭해 살펴보아야 한다. 정형 데이터와 멀티미디어 데이터가 포함된 검색 결과를 기존의 테이블 방식으로 보여주는 것은 다수의 검색 결과를 일괄적으로 살펴보는 데는 편리하지만, 각 튜플(또는 인스턴스 객체)의 세부 내용을 관련지어 볼 수 없다는 불편이 있

다. 이러한 정형 데이터와 멀티미디어 데이터를 포함한 화면은 저작 도구(authoring tool)나 멀티미디어 보고서 작성기를 이용한 추가의 작업을 통해 제작하여야 한다.

멀티미디어 데이터의 프레젠테이션은 여러 가지 방법으로 수행할 수 있다. 첫째, 응용 프로그램에서 직접 수행하거나 ToolBook이나 IconAuthor 등의 별도의 저작 도구를 이용한다. 그러나, 매번 검색할 때마다 저작 도구를 이용해 화면을 제작하는 일은 일반 사용자에게는 번거로운 일이다. 특히, 검색 빈도가 높고 검색 결과가 수시로 다양하게 변할 경우 매번 별도의 저작 작업을 수행하는 일은 어렵다고 볼 수 있다. 둘째, DBMS에서 제공하는 사용자 인터페이스 설계 도구나 멀티미디어 보고서 작성기를 이용하여 수행한다. 이러한 방법은 최종적으로 고정된 멀티미디어 프레젠테이션 화면의 구성에는 바람직하지만 수시로 질의 대상을 바꿔가며 다양한 구조의 멀티미디어 정보를 검색하는 경우에는 과도한 노력이 소모된다. 셋째, 스키마 수준에서 대략적으로 기술된 프레젠테이션 정보를 이용하여 프레젠테이션을 수행한다[2, 13]. 그러나, 테이블(또는 클래스)당 고정된 프레젠테이션 스키마를 정의해 유지하면 다양한 형태로 검색 결과를 변경해 볼 수가 없다. 또한 여러 테이블에서 취합된 정보에 대해서는 프레젠테이션 구조를 미리 정의할 수 없다.

본 논문에서는 검색 빈도가 높고 검색 형태가 다양한 응용 분야에서 간단한 질의어만을 사용해 멀티미디어 데이터가 포함된 질의 검색 결과를 멀티미디어 답게 보여주기 위한 멀티미디어 검색 질의어(SQL/MR:SQL/Multimedia Retrieval)를 제안한다. 본 논문에서 제안한 SQL/MR은 복잡한 구조를 갖는 멀티미디어 구조를 모델링하는 ORM(Object Relationship Model)[7, 8, 9]에 기반한 멀티미디어 데이터 질의어 MQL(Multimedia Query Language)[10, 11]을 변형 확장한 것이다. ORM과 MQL은 시간적으로 구조화된 멀티미디어 데이터를 표현하고 검색하는 기능을 제공하고 있으나, 그 기능을 기존의 DBMS에 추가하기가 어려워 전용 멀티미디어 DBMS[11, 12]로 개발되었다. 본 논문의 목적은 ORM과 MQL의 다양한 멀티미디어 구조 표현 기능을 기존의 DBMS에서 직접 활용할 수 있도록 하는 구현이 용이한 방안을 제

〈표 1〉 저작 도구 및 DBMS의 검색 형태 비교  
 <Table 1> Comparison of Retrieval Styles of Authoring Tools and DBMSs

분류	제품/시스템 이름	데이터 저장 장소	데이터 저장 형태	저작/검색 결과
저작도구	ToolBook, IconAuthor	화일	모노미디어	시공간에 배치된 멀티미디어
모노미디어 DBMS	Oracle, Sybase, UniSQL, Illustra	테이블 또는 클래스	모노미디어	테이블 또는 모노미디어
ORM 및 MQL 기반 멀티미디어 DBMS		PART-OF로 연결된 클래스	시공간 구조가진 멀티미디어	시공간에 배치된 멀티미디어
SQL/MR 기반 모노미디어 DBMS		테이블 또는 클래스	모노미디어	시공간에 배치된 멀티미디어

시하는데 있다. 표 1은 멀티미디어 저작용 저작도구, 모노미디어 타입을 지원하는 DBMS, 구조화된 멀티미디어를 저장하는 DBMS, SQL/MR을 지원하는 DBMS의 특징을 비교하고 있다.

SQL/MM이나 MQL은 멀티미디어를 구성하는 모노미디어와 멀티미디어의 구조를 기술하는 데이터 언어로 멀티미디어 데이터의 저장 형태 표현과 이러한 저장 구조를 이용한 검색 방법에 치중하고 있다 [4, 5, 11]. ODMG 93의 OQL은 객체 지향 데이터베이스를 위한 검색 언어로 객체 지향 DBMS에 관계 DBMS SQL과 같은 질의 표현 기능을 추가하는 것을 목적으로 하고 있다 [3]. 이러한 데이터 언어들의 최종 검색 결과는 객체 또는 복합 객체가 되며 이러한 객체를 시공간상에 어떻게 프레젠테이션해야 할 지를 기술하는 기능은 결여되어 있다. MQL 관련 연구에서는 저장된 멀티미디어에 내재된 시공간 구조 정보를 실제로 시공간 상에 프레젠테이션해주는 스케줄링 기능을 제시하고 있지만 모노미디어 데이터가 분리 저장된 데이터베이스에서 검색 결과를 시공간적으로 구성하는 기능은 제공하지 않고 있다 [11, 12].

본 논문이 기여하는 바는 다음과 같다. 첫째, 간단한 시공간 구성 기능을 가진 질의어의 사용을 통해 사용자가 매번 질의시마다 저작 도구를 사용해야 하는 번거로움을 제거해 준다. 둘째, 시공간상의 배치 형태를 질의의 변형만으로 간단히 변경할 수 있다. 셋째, 본 논문의 질의어는 모노미디어 데이터 타입 (또는 클래스) 형태로 멀티미디어 데이터를 분리 저장하는 대부분의 현존 관계 DBMS, 객체지향 DBMS,

객체관계 DBMS에서 모두 활용될 수 있으며, 후처리 기 형태로 쉽게 추가되어 검색 기능의 수준을 향상시킬 수 있다.

본 논문의 구성은 다음과 같다. 2절에서는 SQL/MR의 시발점으로 사용된 멀티미디어 데이터 모델 ORM과 멀티미디어 질의어 MQL을 간략히 소개한다. 3절에서는 멀티미디어 데이터를 미디어별로 분리 저장하는 기존 DBMS의 검색 방법과 SQL/MR 이용시에 개선된 검색 방법을 비교하고, SQL/MR의 주요 구문, 시공간 배치를 위한 태그, 기능을 설명한다. 4절에서는 SQL/MR의 구현을 위한 세부 알고리즘을 소개한다. 마지막으로 5절에서는 결론을 맺는다.

## 2. ORM과 MQL 개관

ORM은 객체지향 모델을 기반으로 멀티미디어 데이터를 클래스와 객체로 표현하여 멀티미디어 데이터간의 다양한 시공간 관련성을 표현할 수 있도록 한 멀티미디어 데이터 모델이다. 멀티미디어 객체는 클래스 객체와 인스턴스 객체로 구분된다. 멀티미디어 객체들은 공간 구성(spatial composition), 공간 순차(spatial sequence), 시간 순차(temporal sequence), 병행 모임(parallel collection) 등의 시공간 관계로 연결되며, 클래스 객체들은 다중구조 표현을 위한 등치 관계로도 연결되고, 인스턴스 객체들은 이외에 하이퍼링크와 버전 관계로도 연결된다. 여기서 공간은 화면이나 윈도우 또는 페이지를 나타내고, 병행 모임은 시간 동기화의 표현에 사용된다. 멀티미디어 객체들

의 복잡한 상호 관계는 ORD(Object-Relationship Diagram)로 표현된다.

ORM에서는 멀티미디어 복합 객체가 시공간 특성을 반영할 수 있도록 그 도메인 유형을 확장하여 도메인 객체가 지닐 수 있는 가능한 값/객체의 범위, 객체의 구조, 적용 가능한 연산 등을 정의할 수 있도록 한다. 도메인 유형으로는 기본 클래스인 INT, REAL, CHAR, STRING 등과, 모노미디어 클래스인 TEXT, GRAPHIC, IMAGE, AUDIO 등이 있으며, 복합 도메인 구성을 위해 클래스 정의시 태그를 이용한다. 기본적인 태그로는 []가 투플을, {}가 집합을, <>가 순서를 나타낸다. 공간상에 배치된 성분 객체들의 관련성을 나타내는 공간 구성의 표현에는 sc[, sc{} 태그가, 공간상의 순서를 나타내는 공간 순차의 표현에는 s<>, s{}가, 시간 순서를 표현하는 시간 순차의 표현에는 t<>, t{}가, 시간상의 병행 발생을 표시하는 태그로는 p[]가 사용된다. 그림 1은 ORM을 기반으로 한 MQL의 클래스 정의어를 이용해 사용자 정의 멀티미디어 데이터를 정의한 예이다. 멀티미디어 책 MBook은 텍스트, 그래픽, 이미지의 공간 구성으로 이루어진 페이지로 구성되고 각 페이지의 내용은 오디오로 설명된다. '!' 표시는 선택적 도메인을 나타낸다. 즉, 한 페이지내에 텍스트, 이미지, 그래픽 객체가 여러 개 섞여서 나올 수 있음을 표시한다.

```
CREATE CLASS MBook
(content: p[pages:s{Page},
      sound:Audio])

CREATE CLASS Page
(content: sc{Text!Image!Graphic})
```

(그림 1) ORM에서 사용자 정의 클래스 정의 예  
(Fig. 1) Example of User-defined Class Definition in ORM

사용자 클래스가 정의되면 SQL과 유사한 MQL 멀티미디어 데이터 질의어를 이용해 멀티미디어 데이터를 삽입, 삭제, 변경, 검색할 수 있다. 예를 들어, 그림 1의 사용자 정의 멀티미디어 클래스 MBook에서 '그림이 있는 페이지를 검색하라'는 질의는 다음과 같이 표현할 수 있다.

```
SELECT g
FROM MBook. pages g
WHERE g CONTAINS GRAPHIC
```

위 질의에서 'CONTAINS'는 공간 구성 연산을 제공하는 SpatialComposition 클래스의 수퍼 클래스인 Set 클래스내에 구현된 연산이다. 즉, 이 질의어의 WHERE절은 이질적인 모노미디어 객체들로 구성된 페이지 객체 내부에 매개변수로 주어진 GRAPHIC 타입의 객체가 존재하면 참이 된다.

ORM과 MQL은 구조를 가진 멀티미디어의 구조 표현과 멀티미디어 구조를 이용한 정보 검색에 치중한다. 그러므로 MQL은 FROM절과 WHERE절이 관계 DBMS나 객체관계 DBMS의 SQL과 많이 다르다. MQL의 SELECT절은 검색 결과가 시공간 구조를 가진 복합 객체라는 것을 제외하고는 기존 DBMS의 SELECT문과 큰 차이가 없다.

본 논문에서는 스키마 상에서 멀티미디어 구조가 명시되지 않고 모노미디어 데이터(또는 객체)가 여러 속성에 분리 저장된 대부분의 기존 DBMS를 위하여, 그 검색 결과에 대해 시공간상의 레이아웃을 구성해 줄 수 있도록 MQL의 SELECT문을 변형 확장한다. 결과적으로 제안된 SQL/MR의 FROM절과 WHERE절은 관계 DBMS의 SQL과 동일하다. 그러므로, SQL/MR은 질의 처리기의 SELECT절만을 변경해 구현하거나, SELECT절이 SQL 질의 처리의 최종 단계라는 사실을 이용하여 SQL 질의 처리기를 변경하지 않고 SQL 검색 결과에 대한 후처리기 형태로 구현할 수 있다.

### 3. SQL/MR: 멀티미디어 검색 질의어

#### 3.1 기존 검색 질의와 SQL/MR의 비교 예

관계 DBMS, 객체지향 DBMS, 객체관계 DBMS에서 멀티미디어 데이터를 포함한 검색 질의의 수행 결과는 단순히 테이블 형태로 보여진다. 예를 들어, 그림 2(a)의 직원(EMP) 테이블(또는 클래스)에서 그림 2(b)의 SQL 질의를 사용하여 100번 사원의 정보를 검색한 결과는 대부분 그림 2(c)와 같다. 텍스트, 이미지, 오디오의 세부적인 내용은 해당 위치에서 마우스를 클릭하여 별도로 프레젠테이션하여야 한다. 그림

3은 그림 2(a)의 테이블에 대한 SQL/MR 예와 검색 결과를 보이고 있다.

```
CREATE TABLE Emp
(eno      SMALLINT,
 name    CHAR(15),
 dept    CHAR(20),
 addr    CHAR(30),
 face    IMAGE,
 intro   AUDIO,
 profile TEXT);
```

(a)직원(EMP) 테이블

```
SELECT name, dept, addr, face, intro, profile
FROM Emp
WHERE eno = 100;
```

(b) 검색 질의

name	dept	addr	face	intro	profile
Y. Nah	Dept. of R & D	1234, North Ave.	Image	Audio	Text

(c) 특정 튜플(또는 인스턴스 객체)의 검색 결과

(그림 2) 기존 방식의 멀티미디어 정보 검색

(Fig. 2) Traditional Multimedia Information Retrieval

```
SELECT <SC[name, dept, addr, face, intro], profile>
FROM Emp
WHERE eno = 100;
```

(a) SQL/MR 질의

(b) 프레젠테이션 결과

(그림 3) SQL/MR의 공간 순차를 이용한 프레젠테이션 예 (Fig. 3) Presentation Example using Spatial Sequence of SQL/MR

### 3.2 SQL/MR

SQL/MR의 목표 DBMS는 BLOB/IMAGE/AUDIO 클러스터와 SET-OF 키워드를 통해 집합값(set value)을 지원하는 객체지향 DBMS 및 객체관계 DBMS이다. 멀티미디어 검색 질의어 SQL/MR은 다음과 같은 가정을 전제로 한다.

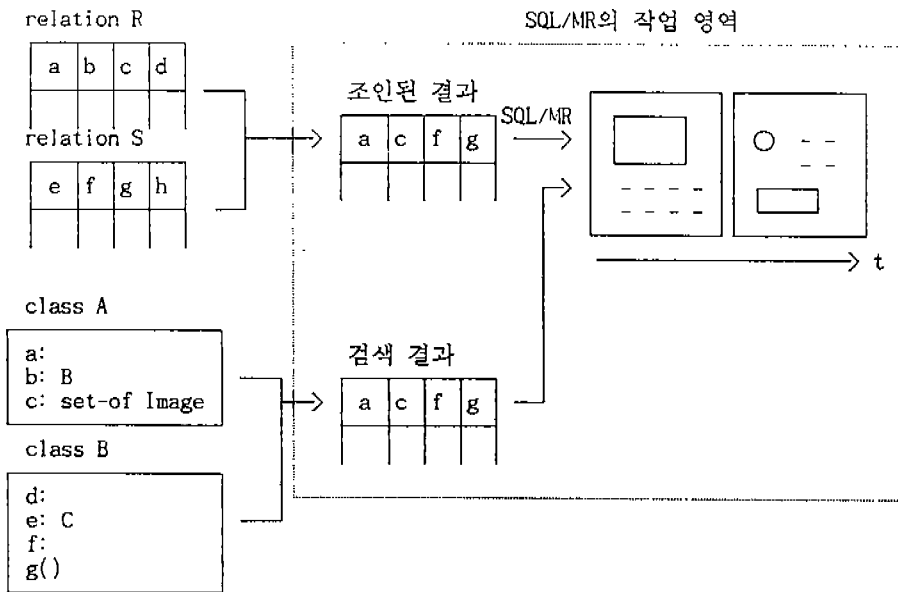
- SET-OF 키워드는 모노미디어 타입에 대해서만 사용할 수 있는 것으로 제한한다.
- 검색 결과의 프레젠테이션은 일정한 크기의 페이지(화면)상에서 수행된다.
- 한 속성에 속한 이미지 객체나 비디오 객체의 공간상의 크기(즉, 해상도)는 일정하다.

검색 결과 화면의 크기를 페이지로 제한하는 것은 대부분의 저작 도구들이 한 화면 크기 이내에서 원도우를 구성하는 것을 고려해 보면 타당한 가정으로 볼 수 있다. 데이터베이스의 웹 인터페이스로 고려되고 있는 HTML과 같은 방법은 페이지 개념은 존재하지만 한 페이지가 한 화면내에 보여지지 않고 스크롤이 된다. SQL/MR에서는 한 튜플의 내용을 상하로 스크롤해 보여주는 것보다는 한 튜플의 내용을 여러 개의 페이지로 배치해 보여주는 방법을 사용한다. 한 화면내에 배치된 이미지나 애니메이션 블록 등의 크기도 페이지 방식의 저작 도구와 마찬가지로 일정한 크기로 가정한다.

SQL/MR은 관계 DBMS의 조인 결과 테이블이나 객체지향/객체관계 DBMS의 검색 결과 클래스를 입력으로 하여 멀티미디어적인 프레젠테이션을 위한 시공간적 배치를 구성해준다 (그림 4). 최종 검색 결과를 대상으로 하므로 기반 DBMS는 분산 DBMS라도 무방하다. 분산 시스템의 경우는 SQL/MR의 SELECT절이 최종 프레젠테이션 사이트에서 실행되는 것으로 생각하면 된다.

SQL/MR은 정형 데이터, 모노미디어, 모노미디어의 집합을 입력으로 하여 시공간 프레젠테이션을 제공해주는 함수라고 생각할 수 있다.

SQL/MR에서는 검색 목표 테이블(또는 클래스)의 구조가 단순하므로 ORM과 MQL의 태그 중 멀티미디어 데이터의 프레젠테이션에 필요한 일부 태그만



(그림 4) SQL/MR의 작업 영역  
(Fig. 4) Working Boundary of SQL/MR

을 사용한다. SQL/MR의 SELECT절을 BNF 문법으로 표현하면 다음과 같다. 굵은 문자는 키워드를 나타낸다.

```

<select-stmt> := SELECT <attr-list>
                | <constructed-attr-list>
<constructed-attr-list> := <start-tag> <attr-list> <end-tag>
<attr-list> := <attr> {, <attr>}*
                | <set-type-attr> FLAT
<attr> := <simple-attr>
                | <set-type-attr> FLAT ((index))
                | <spatially-arranged-attr> [DUR <time>]
                | <temporally-ordered-attr>
<simple-attr> := <attr-name>
<spatially-arranged-attr> := <attr-name> [AT <coordinate-list>]
<temporally-ordered-attr> := <attr-name> [DUR <time>]
                | DUROBJ <time>
<start-tag> := <{SC||T<|P[
<end-tag> := >|]
<coordinate> := <x-coordinate> @ <y-coordinate>
<time> := <integer> <time-unit>
<time-unit> := min | sec | ms
    
```

태그로는 순서를 나타내는 <, 동일 공간(화면)상의 배치를 나타내는 SC[], 시간적인 전후 순서를 나타내는 T<, 시간의 병행성을 나타내는 P[]만을 사용한다. < > 태그에 의해 순서를 갖도록 지정된 속성(또는 객체)들은 페이지(화면) 단위로 프레젠테이션된다. 다음 페이지 또는 이전 페이지로의 이동은 화면 오른쪽 하단의 화살표를 클릭해 수행한다. SC[] 태그에 의해 화면상에 공간 구성되는 객체들은 AT절을 이용해 화면의 왼쪽 상단을 원점으로 해서 그에 상대적인 좌표를 표시해 위치를 지정한다. 텍스트와 이미지와 같은 2차원적인 데이터는 오른쪽 하단을 나타내는 하나의 좌표를 더 명시해 크기를 제한할 수 있다. AT절이 생략되면 디폴트로 이전 속성의 위치에 대해 다음 라인에 배치한다. T< > 태그에 의해 시간축상에 순서대로 나열되는 객체들은 텍스트나 이미지와 같은 시간-독립 객체의 경우 DUR절에 지속 시간을 지정해주고, 오디오나 비디오와 같은 시간-종속 객체들은 내재된 지속시간 만큼 플레이백한다. P[] 태그에 의해 동일한 시간대에 병행 발생하는 객체들은 시간-종속 객체의 내재된 지속시간이나 시간-독립 객체에 대해 명시된 지속시간 만큼 플레이백된다. 집합 객체의 경우는 FLAT 키워드에 의해 속성 리스트

트로 바꾼후 원하는 태그를 적용하면 된다. FLAT을 적용하지 않으면 첫번째 객체만 프레젠테이션된다. 집합 객체의 각 원소의 위치를 ()안에 표시한 후 개개의 위치를 지정할 수도 있다.

SQL/MR의 태그가 단순해 보이지만 알렌이 제안한 13가지 시간 릴레이션(중복을 제거하면 7가지)[1]을 T<> 태그와 P[] 태그와 지연객체(DUROBJ)를 이용해 다음과 같이 모두 표현할 수 있다. a와 b를 결과 테이블의 임의의 속성이라고 하자.

- before T(a, DUROBJ, b)
- meets T(a, b)
- overlaps P[T(a, DUROBJ<sub>i</sub>), T(DUROBJ<sub>j</sub>, b)]
- during P[T(DUROBJ<sub>i</sub>, a, DUROBJ<sub>j</sub>), b]
- starts P[T(a, DUROBJ), b]
- finishes P[T(DUROBJ, a), b]
- equals P[a, b]

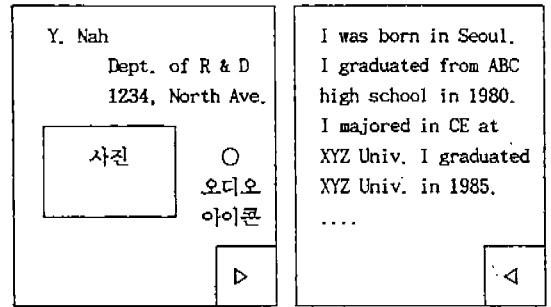
SQL/MR에서는 T<>, P[], 지연객체를 이용해 임의의 두 속성(객체)간의 시간 관계는 물론이고 임의의 객수의 속성(객체)간의 시간 관계를 모두 표현할 수 있다. 또한 SC[] 태그를 이용해 공간 배치를 수행할 수 있으며, <> 태그를 이용해 시간 특성이 명시되지 않은 순서도 표현할 수 있다.

그림 5는 SC[] 태그를 이용한 공간 구성에서 성분 객체들의 상대적 위치를 명시한 예를 보이고 있다. dept와 addr 속성의 위치가 이동하였고, intro 속성에 대한 오디오 아이콘의 위치도 변경되었다.

그림 6은 페이지별 순서를 나타내는 <> 태그 대신 시간적인 순서를 나타내는 T<> 태그를 이용한 프레젠테이션 예를 보이고 있다. 오디오 데이터를 저장한 'intro' 속성은 공간 구성 태그 밖으로 빠져나와 SC[] 태그가 기술하는 화면과 시간적으로 동기화되어 플레이백된다. 이 병행 모임의 플레이백 시간은 시간-중속 객체인 오디오 객체의 지속시간에 의해 결정된다. 오디오의 지속시간보다 더 오래 플레이백을 원할 경우에는 SC[] 태그 다음에 DUR 값을 명시하면 된다. 그림 6(c)는 그림 6(a)의 프레젠테이션 순서를 시간선을 따라 표시하고 있다. name, dept, addr, face의 공간 구성 화면과 intro 오디오의 플레이백이 끝나면 profile 텍스트가 지정된 시간(10초)동안 화면에 보여

```
SELECT <SC[name AT 10@10,
      dept AT 20@20,
      addr AT 20@30,
      face AT 10@40,
      intro AT 50@50],
      profile>
FROM Emp
WHERE eno = 100;
```

(a) SQL/MR 질의

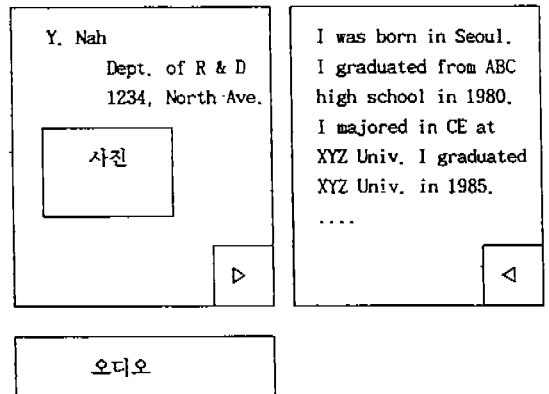


(b) 프레젠테이션 결과

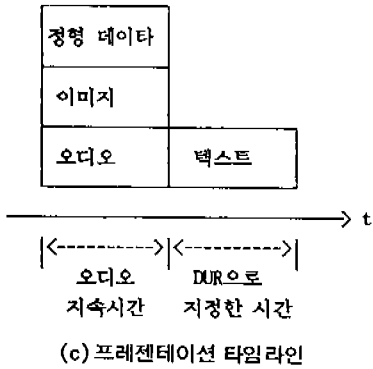
(그림 5) SQL/MR의 공간 배치를 지정한 프레젠테이션 예 (Fig. 5) Presentation Example with Spatial Layout Specification of SQL/MR

```
SELECT T<P[SC[name AT 10@10,
      dept AT 10@20,
      addr AT 10@30,
      face AT 10@40],
      intro],
      profile DUR 10 sec>
FROM Emp
WHERE eno = 100;
```

(a) SQL/MR 질의



(b) 프레젠테이션 결과



(그림 6) SQL/MR의 병행 모임을 이용한 프레젠테이션 예 (Fig. 6) Presentation Example using Parallel Collection of SQL/MR

지게 된다.

#### 4. 시공간 배치 기법

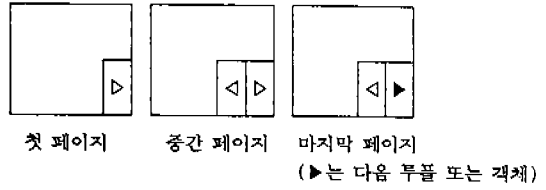
SQL/MR의 시공간 태그의 처리 알고리즘은 다음과 같다. <> 태그를 처리하는 sequence 알고리즘은 소속 요소들을 페이지상에 순서대로 프레젠테이션하고 이전 페이지로 이동, 또는 이후 페이지로 이동하라는 화살표를 클릭하면 해당 페이지로 이동한다.

```

algorithm sequence // <> tag
for all the elements contained in <> tag do
repeat
present it on a page
until previous or next page event occurs
end
end
    
```

알고리즘 sequence에서 생성하는 각 페이지의 기본적인 레이아웃은 그림 7과 같다. 한 투플에 대한 마지막 페이지에 도달하면 다음 투플로 넘어가기 위한 화살표가 보여진다. T<>, P[] 태그는 소속 요소들을 내재된 지속시간 또는 명시된 지속시간 만큼 보여주지만, 사용자의 인터럽트가 가능하도록 그림 7과 같은 전후 페이지 이동 화살표를 <> 태그와 마찬가지로 갖게 된다.

✓ T<> 태그를 처리하는 temporal\_sequence 알고



(그림 7) <> 태그에서의 기본적인 페이지 레이아웃 (Fig. 7) Basic Page layout of <> Tag

리즘은 소속 요소들을 페이지상에 프레젠테이션하고 시간-종속 미디어에 내재된 지속시간, 시간-독립 미디어나 SC[]에 대해 명시된 지속시간, SC[]에 포함된 요소 미디어의 지속시간, 또는 이러한 지속시간 중 최대 지속시간에 도달하거나 전후 페이지로 이동하라는 화살표가 눌러지면 다음 페이지로 이동한다.

```

algorithm temporal_sequence // T<> tag
for all the elements contained in T<> tag do
repeat
present it on a page
until internal duration reaches // time-dep media
or explicit DUR specified by DUR clause
reaches // time-independent
or max duration among elements reaches // SC[]
or previous or next page event occurs
end
end
    
```

P[] 태그를 처리하는 parallel\_collection 알고리즘은 소속 요소들을 페이지나 스피커 등의 출력 매체를 통해 동시에 플레이백하고 소속 요소에 관련된 지속시간, 또는 여러 요소의 지속시간 중 최대 지속시간에 도달하거나 전후 페이지로 이동하라는 화살표가 눌러지면 다음 페이지로 이동한다.

```

algorithm parallel_collection // P[] tag
repeat (in parallel)
if presentation medium conflicts among elements
then accept user feedback
else present all the elements on the target
presentation medium
until max duration among elements reaches
    
```



or interrupting event occurs  
end

SC[] 태그를 처리하는 spatial\_composition\_present 알고리즘은 소속 요소들을 다음 라인 위치, 또는 AT 절에서 명시한 위치에 배치하고 화면에 프레젠테이션한다. 집합 객체의 구성시 한 페이지 경계가 초과하면 자동적으로 다음 페이지로 넘어간다. 그러나 AT절을 이용하여 위치를 명시할 경우에는 SQL/MR 사용자가 두 개 이상의 SC[] 태그를 이용하여 표현하여야 한다. 오디오 데이터는 SC[] 태그 내에서는 아이콘화되어 마우스 클릭시 프레젠테이션되고, 페이지를 차지하는 다른 요소들과 P[] 태그로 관계될 때는 화면을 차지하지 않고 스피커를 통해 바로 플레이백된다.

```
algorithm spatial_composition_present // SC[] tag
for all the elements contained in SC[] tag do
  if location not specified
    then [if size not exceeds the screen page
          then locate it or audio icon at the
            next line position
          else proceed to the next page or
            accept user feedback]
    else locate it or audio icon at the position
      specified by AT clause
  end
repeat
  if not iconized
    then present
    else present upon clicking
  until condition expires
end
```

애니메이션과 비디오 데이터는 S<>, T<>, P[] 태그내에서는 한 페이지를 차지하며, SC[] 태그 내에서는 한 페이지내의 일부 블록을 차지한다. 애니메이션이나 비디오 데이터 페이지나 블록에는 MS-Windows의 미디어 플레이어와 같은 제어 버튼(play, stop, pause, FF, REW)이 부착되는 것으로 가정한다.

## 5. 결 론

본 논문에서는 멀티미디어 데이터를 미디어별로 분리 저장한 데이터베이스에 대해 멀티미디어 데이터가 포함된 질의 검색 결과를 멀티미디어 담개 시공간상에 배치해 프레젠테이션해주는 멀티미디어 검색 질의어(SQL/MR:SQL/Multimedia Retrieval)를 제안하였다. SQL/MR은 SQL의 SELECT절을 확장한 것으로 사용자가 매번 질의시마다 저작 도구를 사용하여 하는 번거로움을 제거해 준다. SQL/MR은 검색 결과인 정형 데이터, 모노미디어, 모노미디어의 집합을 입력으로 하여 시공간 프레젠테이션을 수행한다. SQL/MR의 시공간 배치를 위한 태그로 순서를 나타내는 <>, 동일 공간상의 배치를 나타내는 SC[], 시간적인 전후 순서를 나타내는 T<>, 시간의 병행성을 나타내는 P[]를 사용하였다. 또한 이러한 태그를 이용해 시공간상에 멀티미디어 데이터를 배치해주는 sequence, temporal\_sequence, parallel\_collection, spatial\_composition\_present 알고리즘을 제시하였다.

SQL/MR은 태그를 이용해 알렌의 13가지 시간 릴레이션을 T<> 태그와 P[] 태그와 지연객체(DUROBJ)를 이용해 표현할 수 있고, 하나의 공간(화면)상에 텍스트, 그래픽, 이미지, 오디오, 비디오 데이터를 배치할 수 있다. 오디오와 같은 시간-종속 미디어는 공간상에 배치하지 않고 공간과 병행적으로 플레이백되도록 표현할 수도 있다.

SQL/MR은 일반 저작 도구가 지원하느 복잡하고 정교한 형태의 저작 기능을 제공하기 보다는 수시로 검색 결과의 프레젠테이션 형태를 간단한 표현만으로 쉽게 바꿀수 있도록 하는데 치중하였다. SQL/MR은 모노미디어 데이터 타입 또는 클래스 형태로 멀티미디어 데이터를 분리 저장하는 대부분의 현존 관계 DBMS, 객체지향 DBMS, 객체관계 DBMS에서 모두 활용될 수 있다. SQL/MR의 <> 태그와 SC[] 태그만을 이용한 검색 결과는 <> 태그를 스크롤되는 페이지로 간주하면 HTML 형태로 바로 표현할 수 있어 모노미디어 지원 DBMS의 웹 인터페이스를 개발하는데도 활용할 수 있다.

참 고 문 헌

[1] Allen, J. F., "Maintaining Knowledge about Temporal Intervals," *Communications of ACM*, 26(11), pp. 832-843, 1983.

[2] Andre, E., "Intellimedia: Making Multimedia Usable by Exploiting AI Methods," *ACM Computing Surveys*, 27(4), pp. 560-563, 1995.

[3] Cattell, R.G.G, *The Object Database Standard: ODMG-93*, Morgan Kaufmann Publishers, 1994.

[4] Gallagher, L. J., "Object-SQL: Language Extensions for Object Data Management," in *Proc. CIKM '92*, 1992.

[5] Lee, K., Lee, Y. K., Yoo, S-J., and Berra, P. B., "Object-Oriented Modeling, Querying, and Indexing for Multi-structured Hypermedia Document Databases," in *Proc. IEEE Int'l Workshop on Multimedia Database Management Systems*, 1996.

[6] Little, T. D. C. and Ghafoor, A., "Interval-Based Conceptual Models for Time-Dependent Multimedia Data," *IEEE Trans. on Knowledge and Data Eng.*, Vol. 5, No. 4, pp. 551-563, August 1993.

[7] Nah, Y. and Lee, S., "Object-Relationship Model for Conceptual Modeling of Multimedia Data," *Future Databases '92, Advanced Database Research and Development Series-Vol. 3*, World Scientific, pp. 125-132, 1992.

[8] Nah, Y. and Lee, S., "Two-level Modeling Schemes for Temporal-Spatial Multimedia Data Representation," in *Proc. DEXA*, pp. 102-107, 1992.

[9] Nah, Y., Lee, S., and Hwang, S., "Multimedia Composite Object having Temporal-Spatial Structures," in *Proc. Int'l Computer Science Conference(ICSC) Data and Knowledge Engineering: Theory and Applications*, pp. 314-320, 1992.

[10] Nah, Y., Lee, S., Park, Y., and Hwang, S., "A Representation of Operations on Temporal-Spatial Multimedia Data," in *Proc. Int'l Sympo. on Next Generation Database Systems and Their Applications*, pp. 241-248, 1993.

[11] Nah, Y., Lee, S., Park, Y., and Hwang, S., "A Query Language for Multimedia Databases," in *Proc. InfoScience '93*, pp. 329-336, 1993.

[12] Park, J., Lee, S., Cha, J., and Nah, Y., "A Presentation Scheduling Scheme for Multimedia Databases," in *Proc. COMPSAC*, pp. 284-289, 1996.

[13] Weitzman, L. and Wittenburg, K., "Automatic Presentation of Multimedia Documents Using Relational Grammers," in *Proc. ACM Multimedia Conf.*, pp. 443-451, 1994.

[14] Woelk, D., Luther, W., and Kim, W., "Multimedia Applications and Database Requirements," in *Proc. IEEE-CS Office Automation Sympo.*, pp. 180-189, 1987.



나 연 묵

1986년 서울대학교 전자계산기 공학과(공학사)  
 1988년 서울대학교 전자계산기 공학과(공학석사)  
 1993년 서울대학교 컴퓨터공학 과(공학박사)  
 1991년 미국 IBM T.J.Watson 연구소 객원연구원

1993년~현재 단국대학교 공과대학 전자·컴퓨터공학부 조교수  
 관심분야: 데이터베이스, 멀티미디어 시스템, 객체지향 시스템