

□ 特 輯 □

멀티미디어 DBMS의 구조 및 모델링

한국항공대학 전자계산학과 황 수 찬*

충남대학교 컴퓨터공학과 이 규 철*

서울대학교 컴퓨터공학과 나연목* · 이석호*

● 목

차 ●

I . 서 론	3.1 관련 연구
II . 멀티미디어 데이터의 특성	3.2 멀티미디어 데이터 모델의 요건
2.1 멀티미디어 데이터의 분류	IV . 멀티미디어 DBMS의 구조
2.2 시공간적인 특성	4.1 관련 연구
2.3 멀티미디어 데이터의 구성 요소	4.2 멀티미디어 DBMS의 프로토타입 구조
III . 멀티미디어 데이터 모델	V . 결 론

I. 서 론

컴퓨터 처리 정보의 형태가 문자, 수치 중심의 단순 정형(formatted) 데이터에서 텍스트, 그래픽, 영상, 음성 등 비정형(unformatted) 데이터를 포함하는 정보로 확장됨에 따라 이러한 멀티미디어 정보의 처리가 컴퓨터의 각 분야에서 관심 대상이 되고 있다.

데이터베이스 분야에서도 멀티미디어 데이터의 표현 및 처리에 관련된 기술에 대해 많은 관심이 모아지고 있다. 1983년경부터 대두된 초기 연구의 관심은 화일 시스템을 기반으로 한 멀티미디어 데이터의 표현에 관한 연구[29]와 멀티미디어 통신과 관련된 멀티미디어 문서 구조 표현 및 검색에 관한 연구[4]로 나누어 볼 수 있다.

이 시기의 주요 연구 결과로는 국제표준사무문서구조(ODA: ISO 8613), 사무문서 교환 형식(ODIF), ODA를 기반으로 한 이태리 IEI의 문서 검색 시스템 MULTOS [20] 등을 들 수 있다.

1980년대 후반에 들어서면서 주로 화일을 이용하여 처리하던 멀티미디어 데이터를 데이터베이스 시스템 내

에서 정형화된 데이터와 함께 처리하려는 시도를 하였다. 이러한 연구의 시초로 미국 MCC의 멀티미디어 응용에 대한 요구사항 분석 및 기초적인 표현 기법[30,31,32]에 관한 연구를 들 수 있다. 그 이후로 일본 정보과학대의 멀티미디어 데이터베이스의 조직 관리 기법[18,19]이나 미국 시라큐스 대학의 분산 환경 하의 동기화 표현 기법 [15,16] 등이 발표되었고 실험적인 멀티미디어 데이터 베이스 시스템[34]이 제안된 바도 있다. 이 연구들은 멀티미디어 데이터와 관련된 요구 사항을 모델링[18,31, 32]과 데이터베이스 시스템[5,30,31,32] 측면에서 제시 하고 있으며, 기타 연구의 이슈들을 살펴보면 멀티미디어 데이터베이스 관리 시스템의 구조, 멀티미디어 데이터의 처리 및 저장 기법, 또는 DBMS 본래의 기능에 관련된 문제 등을 들 수 있다.

그러나 이들 많은 연구에도 불구하고 아직은 데이터 베이스 시스템 내에서 멀티미디어 데이터들을 제대로 처리하지 못하고 있으며, 멀티미디어 응용 고유의 특성을 반영하는 모델 및 DBMS 시스템을 제시하는 데에는 미치지 못하고 있는 실정이다. 따라서 본 고에서는 멀티미디어 데이터의 특성을 살펴보고, 이러한 데이터를 데이터베이스에 제대로 표현하기 위한 데이터 모델의

* 종신회원

요건과 DBMS의 구조 등을 기존 연구 결과들을 바탕으로 고려해 보기로 한다.

본 고의 구성은 먼저 다음 2장에서 멀티미디어 데이터의 특성을 살펴보고, 3장에서는 이러한 데이터를 표현하기 위해 데이터 모델이 지원해야 하는 기능을 기존 연구 모델을 기반으로 설명한다. 다음 4장에서는 이러한 모델을 지원하는 멀티미디어 DBMS의 구조를 검토해 보며, 마지막으로 5장에서 결론 및 앞으로 해결해야 할 연구 과제들을 살펴보기로 한다.

II. 멀티미디어 데이터의 특성

2.1 멀티미디어 데이터의 분류

문서(document)와 같이 현실 세계에서 사용되는 정보는 일반적으로 기존 컴퓨터의 처리 대상이던 문자나 숫자 뿐만 아니라, 도표나 그림, 도면, 지도, 설명을 위한 긴 텍스트, 사진, 음성 데이터 등과 같이 다양한 타입의 데이터들로 구성된다. 보통 이러한 각 데이터 타입을 미디어(media)라고 하고, 여러 미디어가 혼합된 데이터를 멀티미디어(multimedia) 데이터라 한다. 이러한 혼합 데이터를 다루는 멀티미디어 응용은 멀티미디어에 고유한 새로운 데이터 타입과 연산을 필요로 한다. 이 멀티미디어 응용이 대상으로 하는 주요 미디어 타입은 다음과 같다.

- ◆ 텍스트(text)
- ◆ 그래픽(graphics): 벡터 그래픽(vector graphics)
- ◆ 이미지(images): 정적 이미지(still images), 사진(picture), 비트맵 이미지(bitmaped images)
- ◆ 비디오(video): 모션 비디오(motion video), 애니메이션(animation), 시청각 화면(audiovisual sequence)
- ◆ 오디오(audio): 소리(sound), 음악(music), 음성(voice)

이들 각각을 모노미디어(monomedia)라 하고, 이들이 혼합된 경우를 멀티미디어(multimedia)라 한다. 즉, 멀티미디어 데이터는 전통적인 정형 데이터(formatted data) 외에 텍스트, 소리, 영상 등의 다양한 비정형 데이터(unformatted data)를 시공간적으로 통합한 것으로 정의할 수 있다[14.31,32].

이러한 멀티미디어 데이터들은 관점에 따라 다시 여러 형태로 그룹지어 구분하기도 한다. 일반적으로 텍스트와 그래픽을 수동 미디어(passive media)로, 모션 비디오와

음악을 능동 미디어(active media) 또는 시간 미디어(temporal media)로 구분한다[26]. [32]에서는 비트맵 이미지를 공간 멀티미디어 객체(spatial multimedia object), 텍스트, 오디오를 선형 멀티미디어 객체(linear multimedia object)라 하고 애니메이션된 이미지는 이를 혼합한 형태로 보고 있다. 여기서, 공간 멀티미디어 객체는 특정 사각 영역(specific rectangular portion)에 표현이 되고, 선형 멀티미디어 객체는 시각 위치에서 거리나 시간에 의한 오프셋(offset)으로 표현된다.

또한 텍스트, 이미지 등 공간적 성향을 갖는 미디어를 정적 미디어(static media), 음성 주석, 비디오 이미지, 애니메이션 등 시간 제약 조건도 갖는 미디어를 동적 미디어(dynamic media), 정적 미디어와 동적 미디어가 혼합 구성된 미디어를 혼합 미디어(mixed media)라고 구분하기도 한다[15]. 멀티미디어 통신 분야에서는 텍스트와 그래픽을 이산 미디어(discrete media)로, 비디오와 디지털 오디오를 연속 미디어(continuous media)로 구분하고 있다[11].

2.2 시공간적인 특성

모노미디어 데이터들이 통합되어 구성되는 멀티미디어 데이터는 각 구성 미디어의 특성들에 따라 서로 복잡한 공간적(spatial)이고 시간적(temporal)인 관계를 갖는다. 시공간적인 특성을 고려하기 위해서는 앞에서 설명했던 다양한 분류 중에서 멀티미디어 응용에서 통합 표현되는 미디어 데이터를 [26]에서 처럼 텍스트, 그래픽, 이미지 등의 수동적 미디어와 음성 주석(오디오 미디어), 일반 애니메이션(시청각 미디어) 등의 능동적 미디어로 구분하는 것이 편리하다

공간(space)은 이차원 레이어아웃의 평면으로 분수 있으며 해당 장치의 화면이나 윈도우 상에 표현된다. 공간은 시간축 상에서 시간 p (spatial time interval)을 갖는다. 수동 미디어는 능동 미디어와 순차적으로 관련될 때 사용자/시스템이 정의한 고정 단위 시간동안 보여지며, 능동 미디어와 동시에 보여질 경우에는 능동 미디어의 지속 시간동안만 보여지게 된다. 수동 미디어가 능동 미디어와 아무런 관련이 없을 경우에는 '마우스 누름'과 같은 외부 사건(external event)에 의해 프리젠테이션을 끝내게 된다.

공간은 하나 또는 여러 시각 미디어를 동시에 혹은 순차적으로 보여주기 위해 사용된다. 하나의 공간이 여러 시각 미디어를 순차적으로 보일 경우, 그 공간은 여러 개의 공간 샷(space shot)으로 나누어지며 주어진 공간의

크기보다 큰 시각 미디어는 그 공간 내에서 스크롤되는 것으로 가정한다.

시간(time)은 일차원축으로 표현되며 오디오 프리젠테이션 장치에 오디오 미디어를 표현하거나 시간적으로 순차화된 혼합 미디어를 표현하기 위해 사용된다. 일차원 시간축 상의 시간 간격은 τ (temporal time interval)로 표현되는데 τ 는 능동 미디어의 지속 시간이나 ρ 의 부분 단위 시간을 나타낸다. 능동 미디어는 내부 지속 시간을 내재하고 있다.

2.3 멀티미디어 데이터의 구성 요소

분류된 미디어 객체 클래스에는 다양한 멀티미디어 데이터들이 포함될 수 있다. 즉, 각 미디어 객체 클래스에 포함될 수 있는 데이터들은 그 데이터의 기본 단위와 적용 범위 및 목적에 따라 그 분류를 달리할 수 있다. 본 절에서는 질의 처리시에 필요한 미디어의 내용을 효과적으로 참조하기 위해 실제 어떤 데이터들이 지원 가능한가를 살펴보고 데이터의 내부 구성 요소도 함께 고려해 본다. 데이터의 종류는 크게 5가지 형태인 원시 데이터(raw data), 서술 데이터(description data), 등록 데이터(registration data), 구조적 데이터(structural data), 공학/기술 데이터(engineering/technical data)로 분류되고 그 각각의 특성은 아래와 같다.

2.3.1 원시 데이터

원시 데이터는 텍스트, 그래픽, 이미지, 음성 및 사운드, 비디오 등 가장 기본적인 데이터 타입 형태로 저장될 수 있는 것을 말한다. 다음은 각 데이터의 구성 단위가 될 수 있는 것들을 열거한 것이다.

◆ 텍스트 데이터

- string의 연속으로 구성되는 데이터

◆ 그래픽 데이터

- 여러 구성 요소가 있다.

- point, node(특정 위치 및 하나의 접합점이 되는 요소),

- chain(공통 속성을 가지고 있는 link들의 연속)

- link(속성을 갖지 않은 선형 요소)

- segment(두점을 잇는 한 벡터 요소)

- polygon(폐쇄된 chain을 가진 2차원 요소)

- spot(한 노드나 포인트에 연관된 실세계 객체)

- composite spot(하나 이상의 spot가 연관된 실세계 객체)

- line(한 chain과 연관된 실세계 객체)

- composite line(하나 이상의 chain과 연관된 실세계 객체)

- area, composite area(하나 이상의 area와 연관된 2차원 객체)

◆ 이미지 데이터

- 그래픽 이미지의 구성 요소와 중복되는 부분이 많다.

pixel, area, composite area, spot, composite spot, line, composite line, frame, clipping 등.

◆ 음성 및 사운드 데이터

- segment(임의의 시간 및 다른 정보와 연관된 시간 단위)

- sampling 등.

◆ 비디오 데이터

- frame, scene, 동기화된 단위 등.

◆ 애니메이션 데이터

- frame 등.

2.3.2 서술 데이터

어떤 객체에 대한 정보를 좀 더 자세히 설명해 놓거나 자세한 주석문 등을 달아 놓고자 할때 자연 언어 형태로 기술하여 저장할 수 있다. 이러한 데이터 형태는 비정형 정보 자체를 질의 대상으로 삼기 어려운 연산에서 이용될 수 있는 데이터이다. 예를 들어 현재의 기술 수준으로는 이미지 데이터에 대해 특정 추출(feature extraction)이나 패턴 인식(pattern recognition) 등을 통한 내용 검색(content search)이 불가능하므로 이를 보조하기 위한 키워드나 서술 데이터를 첨가하여 저장하고, 이를 이용하여 내용 검색을 하게 할 수 있다.

2.3.3 등록 데이터

비정형 데이터의 특성 및 필요한 정보를 따로 추출하여 이러한 정보들만 저장한 데이터이다 이 데이터에 포함되는 정보는 원시 데이터만으로 표현하기 어려운 것들을 지닌다. 예를 들면.

◆ 이미지 데이터

- 해상도, 픽셀 수, 색상, 크기 및 이미지 포맷(format)

◆ 텍스트 데이터

- 제목, 폰트 크기와 형식, 페이지 정보

◆ 기타

- 데이터 생성 날짜, 디바이스의 종류 등이 있다.

2.3.4 구조적 데이터

전체 스키마 구성이 어떻게 이루어져 있는가를 명시하는 데이터이다. 한 클래스 객체의 상위 클래스 및 하위 클래스를 명시하거나 각 클래스 내의 에트리뷰트와 메소드들의 종류를 명시하여 저장할 수 있는 데이터이다.

2.3.5 공학/기술 데이터

이러한 데이터는 보통 시스템 관리자가 장비 및 설치 환경에 대한 기술적인 정보를 명시할 수 있는 데이터이다.

III. 멀티미디어 데이터 모델

본 장에서는 2장에서 설명한 멀티미디어 데이터를 데이터베이스에 표현하기 위한 데이터 모델에 대해서 고찰해 보기로 한다. 앞에서 설명한 바와 같이 멀티미디어 데이터는 기존의 정형 데이터와는 매우 다른 특성을 가지므로 이를 표현하기 위한 데이터 모델도 기존의 데이터 모델과는 그 기능이 매우 다르게 된다. 이 장에서는 데이터 모델에 관련된 기존 연구 결과들을 살펴보고 멀티미디어 데이터를 처리하기 위한 데이터 모델의 요점에 대해 설명하기로 한다.

3.1 관련 연구

멀티미디어 데이터의 구조 및 연산을 표현하는 데이터 모델링에 대한 접근 방법은 크게 두 가지로 구분할 수 있다. 첫번째는 기존의 관계형 데이터베이스 시스템을 확장하여 멀티미디어 데이터를 처리하려는 시도이고, 둘째는 객체 지향 데이터 모델을 멀티미디어용으로 확장하는 접근 방법이다.

먼저 관계 데이터 모델을 이용한 경우는 기존의 관계 모델에 멀티미디어 응용에 필요한 새로운 데이터 타입을 추가하는 방법으로써 이 방법은 초기에 텍스트나 이미지 등의 대용량 데이터의 저장 및 관리에 많이 사용되었다. 텍스트가 저장된 기억 장소에 대한 포인터나 화일명 등을 릴레이션의 속성으로 유지하여 가변 길이의 텍스트를 관리하거나[29,27], 여러 사용자가 일반 데이터처럼 공유할 수 있도록 텍스트의 각 행을 튜플로 간주하는 순서성 릴레이션으로 유지하여 텍스트를 관리하는 방법[27] 등이 있다. 이외에 GENESIS[1], STAIRS 등이 관계 데이터 모델을 기반으로 텍스트를 관리하고 있다.

그래픽 데이터의 경우는 그림 내에 존재하는 요소 객체들을 종류별로 나누어 릴레이션으로 관리하는 방법

[3]과 이미지를 별도의 장소에 저장하고 각 이미지에 대한 식별자를 릴레이션으로 관리하는 방법[25] 등이 제시되었다. 이러한 시스템들은 텍스트, 이미지, 그래픽 등의 대용량 비정형 데이터를 위해 이진 대형 객체(BLOB: Binary Large Object)라는 타입을 지원하고 이러한 타입을 가지는 대용량 객체들을 별도의 공간인 BLOB 공간에 저장하여 관리하고 있다. 사실상 이런 방법은 최근 Sybase, Informix-Online과 같은 일부 상용 관계 DBMS에서도 지원하고 있다.

이와 같이 관계 모델을 확장한 방법은 기존 관계 모델의 안정된 이론적 배경과 시스템 구현 기법 등을 이용할 수 있어서 초기에 많이 이용되었다. 그러나 이들은 비정형 데이터의 단순 저장과 검색의 일부 기능만을 제공할 뿐 실제 멀티미디어 데이터 자체의 시공간적인 특징 표현, 다양한 미디어 데이터들의 통합 모델링 기능, 사용자의 요구에 맞는 다양한 연산 표현 및 조각 기능 등이 거의 제공되지 못하고 있다.

한편 1980년대 후반에 들면서 객체 지향 데이터 모델을 멀티미디어 응용을 위한 데이터 모델로 이용하려는 시도가 나타났다. 객체 지향 데이터 모델은 집단화(agggregation)와 일반화(generalization) 등의 데이터 추상화, 데이터 인캡슐레이션, 특성 계승, 연산의 표현, 다양한 관계성 표현의 융통성, 모델의 확장성 등 실제 멀티미디어 데이터를 모델링하는데 필요한 기능들을 많이 제공하고 있다. 이 객체 지향 개념을 이용한 멀티미디어 데이터 모델링 시도에는 MCC의 ORION 시스템[30,31, 32]을 비롯하여, GMD-IPSI의 VODAK 복합 객체 지향 모델[10]에 기반한 개방 데이터 모델[13,14], ODA에 기반한 문서 검색 시스템 MULTOS[20,8,9,24,28], 시카고 대학의 멀티미디어 프리젠테이션용 위한 모델[15, 16,2], 토론토 대학의 문서 정보 시스템 MINOS[4,5,6,7] 등이 포함된다. 이 모델들의 공통적인 특징은 멀티미디어 데이터를 객체와 클래스로 표현하여 객체지향 개념이 개용하고 있는 데이터의 추상화를 지원하고 있으며, 다양한 미디어 데이터들로 구성되는 멀티미디어 문서의 구조들을 통합 정의할 수 있도록 하는 것이다. 그러나 이러한 객체 지향 개념을 이용한 모델들도 멀티미디어 데이터의 복잡하고 다양한 모델링 요구 사항들을 만족시키는 데에는 부족한 점이 많다. 여기서는 대표적으로 ORION과 MINOS의 데이터 모델의 특징을 간단히 살펴보기로 한다.

먼저 ORION의 경우는 복잡한 멀티미디어 문서의 구조를 집단화 계층(agggregation hierarchy)을 이용하여 효율적으로 표현할 수 있도록 하고 있다. 이 집단화 계

층은 한 문서가 다양한 미디어의 요소들로 구성되는 것을 표현하고 있으며, 기존 객체 지향 모델과는 달리 그 구성요소들이 다른 집단화 계층에도 동시에 속할 수 있도록 하여 대용량 데이터의 공유를 모델링에서부터 표현할 수 있도록 한다. 또한 연산의 정의인 베스드도 별도의 객체로 정의하여 공유의 대상이 될 수 있도록 하였으며 관계성 객체(relationship object)를 제공하여 일반화, 집단화 외에 다양한 객체 간의 관계들을 표현할 수 있도록 하고 있다. 그러나 이 모델에서는 문서의 논리적 구조는 표현할 수 있지만 실제 프리젠테이션을 위한 레이아웃 구조를 표현하는 기능이 부족하고, 멀티미디어 데이터의 시공간적 의미를 표현하지 못함으로써 문서의 구조나 형태, 시공간적 특성 등을 이용한 정보 검색이 불가능한 문제점이 있다.

토론토 대학의 MINOS는 화일에 기반한 멀티미디어 문서 정보 검색 시스템으로서 문서를 표현하기 위한 모델로 객체 지향 모델을 이용하고 있다. MINOS의 경우도 객체의 클래스를 이용하여 문서를 정의하고 있는데 이들 간의 일반화, 집단화 계층 표현 등을 이용하여 정보의 공유를 강조하고 있으며 문서간의 다양한 관계성을 표현할 수 있도록 하고 있다. 이 모델은 문서 구조에 대한 정의를 클래스 수준에서 뿐만 아니라 인스턴스 수준에서도 필요한 표현을 할 수 있는 융통성을 제공한다. MINOS의 멀티미디어 문서 모델은 멀티미디어 문서의 논리적 구성요소들을 표현하는 논리적 모델과 실제 출력 장치에 문서를 프리젠테이션하는데 필요한 구성요소들을 명세하는 물리적 모델, 이 두 모델 사이의 사상(mapping) 등으로 구성된다. 따라서 멀티미디어 문서의 인스턴스는 논리적 문서 인스턴스, 물리적 문서 인스턴스, 사상 인스턴스 등으로 구성된다. 이와 같이 논리적 요소와 물리적 프리젠테이션 요소를 분리함으로써 하나의 논리적 객체를 서로 다른 물리적 객체에 사상할 수 있어 다양한 프리젠테이션이 가능한 장점이 있다. 그러나 MINOS는 문서의 고유한 시공간 정보를 분리 표현함으로써 문서 자체의 고유한 제약조건을 표현할 수 없으며 문서의 검색시 정해진 프리젠테이션 형식에서만 출력 가능하므로 등적으로 다양한 형식의 출력을 구할 수 없다. 또한 문서에 대한 단순 삽입, 삭제 등 몇 가지의 고정된 연산만을 지원하므로 사용자가 필요로 하는 다양한 연산을 표현할 수 없는 문제점이 있다.

3.2 멀티미디어 데이터 모델의 요건

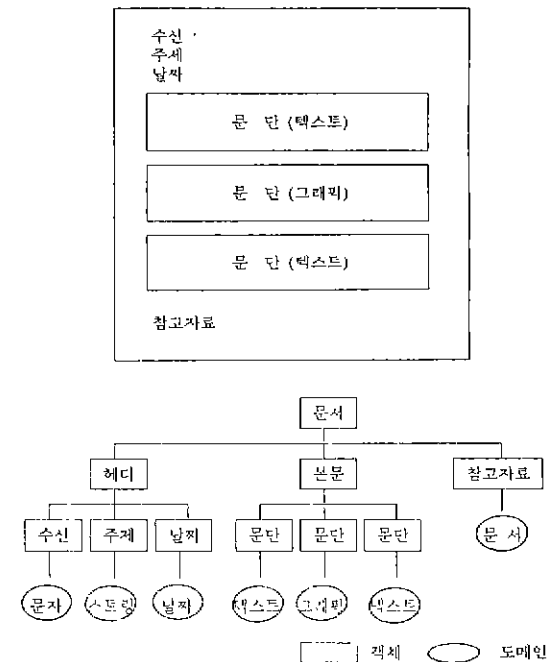
2장에서 설명한 멀티미디어 데이터의 특성과 기존

멀티미디어 모델들이 지원하지는 데이터 모델링 요소들을 중심으로 멀티미디어 데이터 모델의 필수적인 요건들을 고찰하기로 한다.

3.2.1 집단화 계층(aggregation hierarchy)

멀티미디어 데이터는 각 미디어의 데이터들이 복잡한 구조를 가지며 서로 연관되어 구성되므로 이들의 구성 관계는 기존 객체 지향 데이터 모델에서 많이 연구되었던 복합 객체(complex object)의 성격을 가진다[12]. (그림 1)은 멀티미디어 문서와 그 문서를 표현한 집단화 계층을 예로 보인 것이다. 멀티미디어 모델에서 집단화 계층은 다양한 미디어 객체들 간의 IS-PART-OF 관계를 이용한 멀티미디어 정보의 통합을 가능케 하는 필수적인 요소이다. 또한 동일한 멀티미디어 정보가 여러 문서에서 동시에 이용되는 경우가 많으므로 동일한 객체가 여러 집단화 계층에 참가할 수 있는 정보의 공유가 가능해야 한다. 특히 멀티미디어 데이터는 대용량이므로 데이터의 공유가 필수적이다.

또한 집단화 계층에 참가하는 각 구성요소들 사이에 시공간적 순서를 정의할 수 있어야 한다. 예로 (그림 1)에서 헤더, 본문, 참고자료 등은 반드시 이 순서대로 문서에 나타나야 하므로 문서에 나타나는 절이나 문단들 간의 순서도 정의될 수 있어야 한다.



(그림 1) 멀티미디어 문서와 집단화 계층 표현

3.2.2 일반화 계층(generalization hierarchy)

집단화 계층에 속한 각 클래스들은 집단화 계층과는 독립적으로 IS-A 관계를 표현하여 특성의 계승이 이루어져야 한다. 이 특성 계승은 매우 복잡한 구조를 가지는 멀티미디어 데이터를 모델링하기 위해서 반드시 필요한 개념이다. 즉 기존의 클래스에 새로운 특성과 연산을 첨가하여 서브클래스를 정의함으로써 복잡한 클래스의 구조를 제정의하지 않고 간단히 정의할 수 있다.

3.2.3 관계성 표현

멀티미디어 객체들 간에는 일반화와 집단화 관계 외에도 다양한 관계성이 존재하는데 멀티미디어 응용을 위해서는 이를 지원할 수 있어야 한다. 예를 들어 멀티미디어 문서에 그림이 포함된 경우 이 그림에 대한 설명이나 주석(annotation) 등을 표현해야 하는 경우가 빈번히 생긴다. 이 경우 그림 객체와 텍스트나 음성으로 표현된 주석 객체와의 관련성을 데이터베이스에 표현할 수 있어야 한다. 멀티미디어 정보들 간의 관계성으로는 멀티미디어 객체들 간의 참조(reference) 관계, 구성(composition) 관계, 시공간적 관계, 존재 종속 관계, 동치(equivalence) 관계 등이 있을 수 있으며, 이외에도 사용자가 필요한 관계성들을 정의할 수도 있어야 한다. 이러한 정보는 일반적으로 클래스 수준에서 정의되어지지만 시공간적 관계 등은 객체 수준에서도 정의될 수 있어야 한다.

3.2.4 프리젠테이션 정보 표현

멀티미디어 정보의 모델링을 위해서는 멀티미디어 데이터의 물리적 프리젠테이션 정보의 모델링도 필수적인 요건이다. 그러나 기존 멀티미디어 시스템들의 가장 미흡한 기능이 바로 이 프리젠테이션 정보의 모델링 부분이다. 대부분의 시스템들은 문서의 논리적 구조를 표현하는데 치중을 하며 이 프리젠테이션 정보의 모델링을 제대로 지원하지 못하고 있다. 멀티미디어 정보는 데이터의 특성상 시공간적 프리젠테이션에 관한 정보를 논리적 구조와 분리하여 표현하기가 매우 어렵다. 프리젠테이션 정보는 단순히 데이터를 사용자에게 보여주기 위한 입출력 형식일 뿐만 아니라 멀티미디어 데이터 자체의 고유 제약조건을 의미하는 경우도 있으므로 데이터베이스의 스키마에 포함되어야 할 정보인 것이다. 예를 들어 음성 정보를 스피커로 출력을 할때 음성 출력시간은 사용자가 임의로 지정하기 보다는 실제로 저장된 데이터의 크기에 의존하기 때문에 영상 정보 등과 동기화(synchronization)를 하고자 할때 큰 제약조건으로

작용하게 된다.

3.2.5 연산 및 데이터 언어

멀티미디어 데이터를 조작하기 위해서는 다양한 연산이 제공되어야 한다. 기존의 정형 데이터를 조작하기 위한 연산만으로는 멀티미디어 데이터를 조작하는 것이 불가능하다. 멀티미디어 데이터는 각 미디어 데이터별로 적용할 수 있는 연산의 형태가 서로 상이하므로 각 미디어별로 미디어 데이터를 조작할 수 있는 다양한 연산이 제공되어야 할 뿐 아니라 이들을 통합한 멀티미디어 정보에 대해서도 내용식 검색과 멀티미디어 데이터들 간의 일반화, 집단화, 참조관계, 시공간 관계 등 다양한 관계성을 통한 효율적 연산들이 지원되어야 한다. 또한 각 미디어 데이터에 대한 사용자 정의 연산 등을 쉽게 첨가할 수 있어야 하며, 질의 형태는 SQL과 같은 고급의 비절차적 데이터 언어가 지원되어야 한다.

3.2.6 버전 관리

버전의 생성과 제어 기능이 제공되어야 한다. 대용량의 멀티미디어 데이터를 관리하는 멀티미디어 데이터베이스 시스템이나, CASE 시스템에서 처럼 버전을 관리해야 하는 경우가 많이 발생한다. 따라서 여러 버전들의 이력 통제라든지, 묵시적(default) 버전의 선택, 수정된 버전의 참조 관계 관리 등의 기능이 제공되어야 한다.

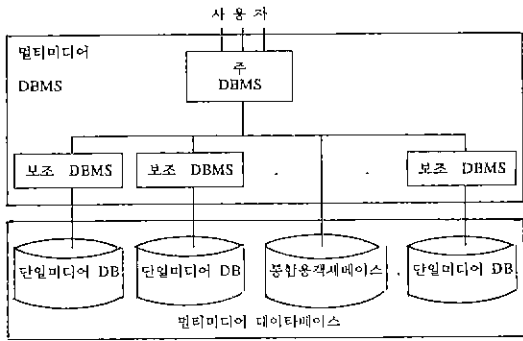
이외에도 다양한 멀티미디어 장치와 도구의 개념적 지원과 동적인 스키마의 수정, 뷰, 하이퍼 링크의 지원 등 다양한 기능들을 필요로 하고 있다.

IV. 멀티미디어 DBMS의 구조

멀티미디어 DBMS는 취급하는 데이터가 기존의 정형화된 데이터가 아니고, 그 모델 또한 매우 다르므로 DBMS의 구조(architecture)도 기존의 관계 DBMS와는 상이한 점이 많게 된다. 그러던 멀티미디어 데이터를 처리하기 위한 멀티미디어 DBMS의 구조를 기존 시스템의 구조를 기반으로 살펴 보기로 한다.

4.1 관련 연구

멀티미디어 DBMS의 구조는 실제 그 시스템이 각 미디어 데이터들을 어느 정도로 통합하는가에 따라 다르게 된다. 단순히 기존 DBMS에 대량 데이터로 텍스트나 영상, 그래픽 등 어느 하나만 지원하는 것에서부터, 텍스트, 영상, 그래픽, 음성, 모션 비디오까지 통합하여 지원하는 것이 있다. 여기서는 어느 한 두 가지의 미디어



(그림 2) 주-보조 구조의 멀티미디어 DBMS

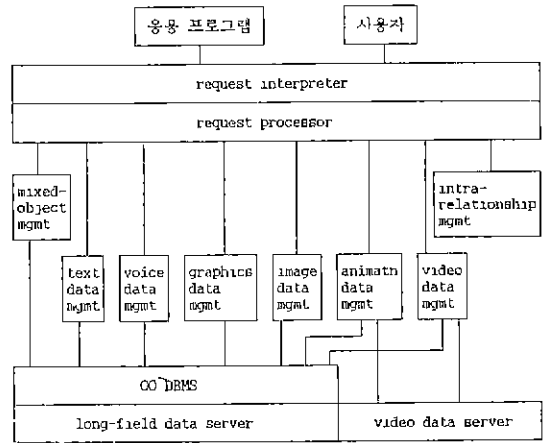
어떤 지원하는 시스템이 아니라 모든 미디어를 지원할 수 있는 시스템의 구조를 고려하고 있다.

초기 멀티미디어 DBMS의 구조는 (그림 2)와 같이 단일 미디어를 지원하는 DBMS들을 사용하여 각 미디어 데이터를 처리하고 상위의 멀티미디어 DBMS가 이를 통합하는 형식의 주-보조(primary-secondary) 구조로 대표할 수 있다[18]. 이러한 구조에서 객체 지향 DBMS를 이용한 좀 더 진보된 구조가 (그림 3)과 같이 [17]에서 제시한 구조이다.

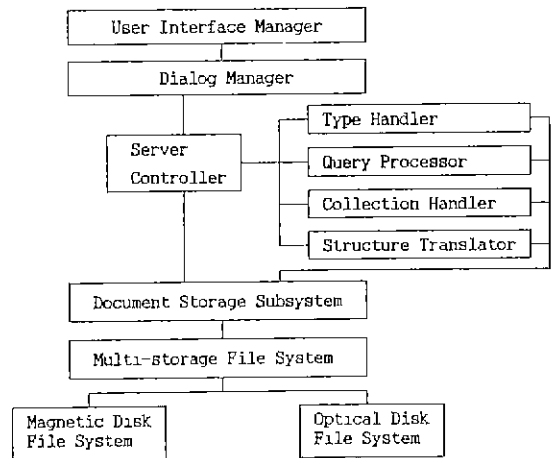
(그림 3)에서 OODBMS(Object-Oriented DBMS)는 멀티미디어 데이터의 구조를 관리하며, 요구 해석기(request interpreter)는 멀티미디어 데이터를 캡처(capture), 검색, 프리젠티하는 기능을 하며, 요구 처리기(request processor)는 통신 제어나 트랜잭션 관리 등 여러 구성 요소들의 실행을 절합(bind)하는 역할을 한다. 내부 관계성 관리(intra-relationship management) 모듈은 동기화, 미디어 변환 등과 같은 객체 간의 관계성을 관리하며, 혼합 객체 관리(mixed object management) 모듈은 내부 관계성 관리 모듈에 의한 혼합 객체를 저장하게 된다. 긴 필드 데이터 서버(long-field data server)는 텍스트, 영상, 음성 등과 같이 크기에 개한이 없는 대량 데이터를 관리하며, 비디오 데이터 서버(video data server)는 애니메이션이나 비디오 필름과 같은 데이터를 관리하는 모듈이다.

다음은 MULTOS 시스템의 구조에 대해 알아보자. MULTOS 시스템은 서버-클라이언트 구조로 워크스테이션들로 구성된 분산 환경에서 사용되며 그 구조는 (그림 4)와 같다[28].

MULTOS의 구조에서 사용자 인터페이스 관리기는 클라이언트 쪽에 위치하며 대화 관리기는 서버와 클라이언트 모두에 존재하고, 나머지 부분은 서버에 위치한



(그림 3) OODBMS를 이용한 멀티미디어 DBMS의 구조



(그림 4) MULTOS 시스템의 구조

다. 각 모듈의 역할을 보면 먼저 사용자 인터페이스 관리기는 질의 구성, 브라우징, 문서 작성, 프리젠티이션 등 사용자가 이 시스템을 이용하는데 관련된 기능을 수행한다. 대화 관리기는 서버-클라이언트 간의 통신을 담당하며, 서버 제어기는 서버에 대한 명령의 실행이나 병행수행 제어, 일관성 제어, 문서 관리 및 시스템 관리 기능 등을 수행한다 이 기능을 위해 서버 제어기는 문서의 타입 관리를 위한 타입 관리기, 문서의 그룹핑을 관리하는 집단(collection) 관리기, 질의처리기, 구조 변환기(structure translator) 등을 이용한다 이 구조 변환기는 문서에 대한 연산을 저장 서브시스템에 대한 연산으로 사상시키는 기능을 수행한다. 다음 문서 저장 서브시스템은 문서의 내용과 구조의 저장을 관리하며,

내용식 문서 검색을 위한 접근방법을 제공한다. 그리고 실제 멀티미디어 데이터들은 자기 디스크와 광 디스크의 파일 시스템에 저장되며, 서로 다른 두 매체에 저장된 데이터를 통합하는 역할을 다중 저장 파일 시스템(multi-storage file system)이 담당하고 있다.

마지막으로 ORION은 OODBMS의 구조에 멀티미디어 정보 관리기(MIM; Multimedia Information Manager)를 두어 이것이 멀티미디어 클래스들을 포함한 클래스 계층의 정의와 메시지 전달 프로토펙트를 처리하고 있다 [31,32]. MIM은 ORION의 객체 서브시스템에 포함되어 있다.

4.2 멀티미디어 DBMS의 프로토타입 구조

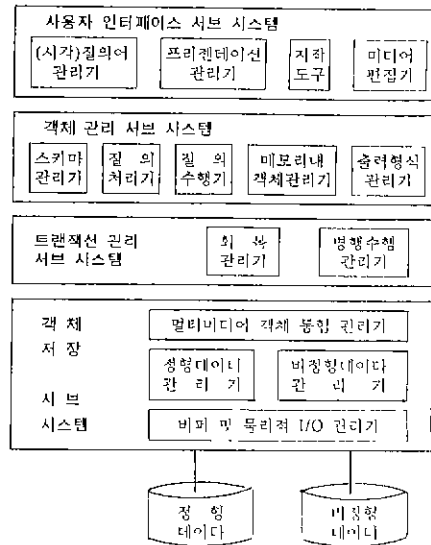
본 절에서는 앞절의 관련 연구 및 3장에서 검토했던 데이터 모델의 요건을 만족시키기 위한 멀티미디어 DBMS의 프로토타입 구조를 제시하기로 한다. 바람직한 멀티미디어 DBMS는 사용자 인터페이스, 데이터 모델의 지원을 위한 객체 관리, 병행수행과 회복 기능을 제공하는 트랜잭션 관리, 객체 저장 기능 등을 제공하여야 한다. 이를 위한 멀티미디어 DBMS의 프로토타입 구조 (그림 5)를 모듈 별로 설명한다.

4.2.1 편리한 사용자 인터페이스

멀티미디어 DBMS의 인터페이스는 기존 DBMS가 지원하는 텍스트 형태의 질의어 인터페이스 만으로는 효율적 사용이 어렵다. 데이터 자체의 종류가 다양해짐에 따라 새로운 검색 방법의 제공과 검색 결과의 출력 방법이 다양하게 지원되어야 한다. 예를 들어 SQL과 같은 텍스트 형태의 질의어 외에도 시각 언어(visual language)나 메뉴 방식으로 질의를 할 수 있어야 하며, 검색 결과도 멀티미디어 데이터의 특성에 맞도록 프리젠테이션 되어야 한다. 또한 검색 결과의 브라우징 및 검색 결과분 바탕으로 한 질의문 재구성(query reformulation) 등의 기능도 제공되어야 한다. 그리고 다양한 미디어 데이터의 입출력을 위한 저작 도구(authoring tool)와 미디어 편집기들이 제공되어야 한다.

4.2.2 객체의 관리

일반적으로 객체 관리 시스템은 사용자 인터페이스 시스템과 저장 시스템 사이에서 질의의 실행과 객체의 관리를 수행한다. 그 주요 기능은 스키마의 관리, 질의어의 번역 및 최적화, 질의의 수행, 출력형식 생성, 메모리내 객체 관리 기능 등이다. 각 기능을 자세히 설명하기로



(그림 5) 멀티미디어 DBMS의 프로토타입 구조

한다

가) 스키마 관리

데이터베이스의 스키마를 관리하는 모듈로서 클래스 계층, 클래스 구조, 메소드 등의 관리와 데이터 정의어(DDL)를 처리하는 기능을 수행하며, 스키마에 대한 정보 요청시 저장된 스키마 정보를 검색하거나, 브라우징, 수정할 수 있도록 지원한다. 또한 클래스 정의에 포함된 프리젠테이션에 관한 시공간 정보를 분리하여 관리한다

나) 질의 처리 및 수행

질의 처리는 사용자 인터페이스로부터 텍스트 형태의 질의문을 받아 이를 분석하여 최적의 접근 계획을 수립, 수행하는 역할을 한다. 사용자 인터페이스에서 입력되는 질의는 사용자가 직접 텍스트 형태로 입력한 것일 수도 있고, 메뉴나 시각 언어 형태로 표현한 것을 중간 형태의 언어로 변환한 것일 수도 있다. 입력된 질의는 구문 분석, 의미 분석, 질의 최적화 과정을 거쳐 접근 계획을 수립하게 된다 여기서 질의 최적화 과정은 적절한 비용 평가 과정을 필요로 하는데 대용량의 데이터를 다루므로 데이터의 입출력 양이나 중간 결과의 크기 등 멀티미디어 데이터의 효율적 처리에 가장 큰 비중을 두어야 할 것이다. 다음 생성된 접근 계획은 질의의 수행 기능에 의해 실행되는데 저장 시스템과의 효율적 인터페이스가 요구되어 진다. 수행 결과는 출력 형식 생성 기능에 전달된다.

다) 메모리 내 객체 관리

메모리 내 객체 관리(in-memory object management)는 객체 관리 기능을 수행하는 여러 모듈들이 저장 시

시스템과의 인터페이스를 의식하지 않고 항상 객체 버퍼 안의 메모리 내 객체만을 대상으로 모든 작업을 수행할 수 있도록 하는 중간 역할을 담당한다. 이 기능에 의해 다른 모듈들은 객체의 저장 형식이나 접근 방법 등을 고려할 필요없이 모든 객체가 항상 메모리 내 객체 형태로 존재하는 것으로 간주할 수 있게 된다. 이를 위해서는 객체 식별자를 통한 객체의 접근 요청을 받아 객체 버퍼 내의 메모리 내 객체 포인터를 반환하는 일을 수행하는 접근 관리의 기능과 메모리 상의 한정된 버퍼를 여러 객체들이 나누어 쓸 수 있도록 버퍼를 관리하는 기능이 필요하다.

라) 출력 형식 생성

출력 형식 생성 기능은 기존 DBMS에서는 요구되지 않았거나 보고서 생성기와 같이 별도의 응용 프로그램으로 지원되는 부분이다. 그러나 멀티미디어 DBMS에서는 데이터베이스 스키마 자체에 프리젠테이션 정보가 포함되어 있으므로 이를 이용하여 검색된 개념적 결과에 시공간적인 정보를 첨가한 출력 형태로 만들어야 한다. 이를 위해서는 메모리 내 객체 관리 기능과 스키마 관리의 기능을 이용하여 필요한 정보를 전달받고, 경우에 따라서는 사용자의 질의 자체에 프리젠테이션에 대한 요구 사항이 포함되어 있을 수도 있으므로 이러한 정보들을 취합하여 적절한 출력 형식을 만들어야 한다. 또 질의의 결과가 조인이나 프로젝션 등의 연산으로 인해 원래 클래스의 구조를 유지하지 않은 경우에도 대처할 수 있는 기능이 제공되어야 한다.

4.2.3 데이터베이스 제어

멀티미디어 응용을 위한 데이터베이스에도 일반 데이터베이스와 같이 병행수행 제어(concurrency control)와 회복(recovery) 등과 같은 데이터베이스 제어 기능이 제공되어야 한다.

병행수행 제어는 동일한 데이터를 다수의 사용자가 동시에 접근하는 것을 제어하는 기능이다. 예를 들어 한 문서에 대한 변경 작업 중에는 이 변경 작업이 확정(committed) 될 때까지는 다른 문서에 의해 참조되거나 이용될 수 없다. 그런데 멀티미디어 문서는 매우 방대한 크기이므로 변경 작업이나 확정 작업에 많은 시간이 걸리게 되어 데이터의 이용에 제약을 받게 된다. 따라서 데이터의 이용에 가능하면 제약을 적게 줄 수 있는 병행수행 제어 방법이 제공되어야 한다.

한편 어떤 종류의 응용이건 데이터베이스 회복 기능은 매우 중요한 요구조건이다. 이것은 보조기억 장치의 회복과 트랜잭션의 회복을 모두 포함한다. 보조기억 장치의

회복을 위해 기존 DBMS들은 주기적으로 백업 장치에 덤프를 받는 방법을 이용하고 있지만 이 또한 데이터베이스의 크기가 클 경우는 문제점이 많이 발생하여 여러 다양한 덤프 기법들을 연구하고 있다. 멀티미디어 데이터베이스의 경우는 데이터베이스의 크기가 더욱 크므로 시스템의 사용에 영향을 주지않고 효율적으로 백업을 받을 수 있는 방법들이 제공되어야 한다.

트랜잭션의 회복은 단일 트랜잭션으로 여러 멀티미디어 데이터들을 갱신할 경우 이들 간의 상호 일관성을 유지하도록 하는 중요한 작업이다. 예로 시스템에 이상이 생겼을 경우 모든 데이터에 대한 변경이 동일하게 확정되거나 취소(abort)되어야 한다는 것이다. 그러나 일반적으로 CAD 환경과 유사하게 멀티미디어 응용을 위한 트랜잭션은 그 처리 시간이 매우 길 수가 있으며, 그 안에서 서브 트랜잭션들이 다수 포함될 수도 있으므로 기존 방법과는 다른 처리 방법이 필요하게 된다.

4.2.4 멀티미디어 데이터의 효율적 저장

멀티미디어 DBMS의 저장 시스템은 정수, 실수, 문자열 등의 기본적 타입과 튜플, 집합 타입의 정형 데이터, 그리고 텍스트, 그래픽, 이미지, 오디오 등의 비정형 데이터들이 상호 연관된 복합 객체 구조로 표현된 멀티미디어 데이터를 효율적으로 저장, 관리하며 신속한 탐색이 이루어질 수 있는 기능을 제공해야 한다.

먼저 멀티미디어 복합 객체의 각 부속 객체 사이에 존재하는 소유(own) 혹은 참조(reference) 관계를 저장 시스템에서 효율적으로 지원해야 한다. 소유 관계는 멀티미디어 데이터들을 결집(clustering)시켜 저장하여 최소한의 디스크 접근으로 관련 객체들을 접근할 수 있도록 하는 통합 관리 기능이 필요하다. 참조 관계의 경우는 일반적으로 저장 시스템 내에서 객체 식별자를 통해 구현되는데 각 미디어 객체는 각각의 서버 시스템에 의해 독립적으로 관리되므로 참조하는 객체의 신속한 접근과 여러 객체에 의한 공유가 가능해야 한다.

복합 객체 형태의 멀티미디어 데이터는 미디어 데이터의 크기가 대용량일 뿐 아니라 중첩된 튜플과 집합 데이터를 많이 포함하므로 물리적 디스크 불력을 초과하는 대용량의 레코드가 될 가능성이 크다. 이러한 특성으로 인해 튜플의 크기를 디스크 페이지의 크기로 제한하는 기존의 데이터베이스 저장 시스템으로는 그 처리가 용이하지 않다[33]. 따라서 멀티미디어 DBMS를 위한 저장 시스템에서는 여러 디스크 불력에 걸쳐있는 대용량 멀티미디어 데이터의 관리기법, 즉, 임의크기의 가변 길이 멀티미디어 데이터를 저장/관리하는 기법이

필요하다. 특히, 비정형 데이터는 대용량 데이터이므로 사용자가 원하는 부분을 신속하게 접근할 수 있도록 하는 객체 내 인덱싱(intra-object indexing) 기법이 제공되어야 한다.

한편 멀티미디어 데이터들은 저장시 정형과 비정형 데이터로 분리되어 저장되므로 상위 시스템과의 인터페이스를 담당하며, 멀티미디어 객체의 분리/통합을 담당하는 저장 객체의 통합 기능이 필요하다. 즉 복합 객체 구조를 가진 멀티미디어 객체는 저장 매체에 저장되기 위해 참조 관계가 분리된 1차원 형태로 변환되어야 하며, 상위 시스템으로 멀티미디어 객체를 전달하기 위해서는 1차원의 디스크 객체를 다시 멀티미디어 객체로 통합시키는 과정이 필요하다. 이를 위해 멀티미디어 객체의 메모리 내 형식과 디스크 내부 형식과의 변환, OID와 레코드 식별자의 변환, 객체에 대한 인덱스의 제공, 정형 데이터와 비정형 데이터 관리 모듈의 제어 기능 등이 요구된다.

V. 결 론

이상에서 멀티미디어 데이터의 특성을 살펴보고, 이러한 데이터의 특성을 효율적으로 반영하는 멀티미디어 데이터 모델에 관해 기존 연구 결과를 비교하여, 이를 기반으로 추출한 데이터 모델의 요건을 제시하였다. 또한 멀티미디어 데이터베이스 시스템의 구조에 관하여 기존 시스템의 구조를 분석하고, 바람직한 멀티미디어 DBMS의 프로토타입을 제안하고, 이의 각 구성 요소에 대하여 살펴 보았다.

멀티미디어 데이터 모델에 관하여 현재까지의 연구 결과를 종합해 볼 때, 대부분의 연구가 객체 지향 데이터 모델 및 문서 구조 모델링에 기반을 두고 있어 멀티미디어 데이터의 특성을 모델링하는데 많은 장점을 지니고 있는 반면, 시공간적인 특성 및 프리젠테이션 정보의 모델링, 각 미디어와 멀티미디어 데이터 입출력 기기의 정보 및 연산의 표현, 하이퍼미디어의 비순차적 동적 관계성의 표현 등에 있어서는 좀 더 많은 연구가 이루어져야 할 필요성이 있다. 또한, 객체 지향 모델 자체에 대하여 공통된 이론적 토대가 마련되지 않고 있는 점도 멀티미디어 데이터 모델의 정립을 어렵게 하는 하나의 이유가 되며, 3장 후반에 언급한 바와 같이 멀티미디어 데이터를 조작하는 연산을 효율적으로 지원하는 데이터 언어 모델의 정립도 시급하다고 볼 수 있다.

한편, 멀티미디어 DBMS의 구조 측면에서 볼 때, 우선 가장 크게 고려해야 할 점은 각 미디어를 관리하는 이질적

(heterogeneous)인 시스템 요소를 효율적으로 결합하는 방안이 관한 연구라 볼 수 있다. 이에 포함된 연구 과제로는 미디어 객체 간의 동기화, 미디어 변환과 같은 내부 관계성(intra-relationship)을 유지하며 처리하는 효율적 구성 방식에 관한 것을 들 수 있다.

또한, 멀티미디어 DBMS 구조가 가져야 될 기능들을 정의함에 있어서 운영체제의 응용 프로그램이 지녀야 할 기능과의 역할 구분을 명확히 하는 것이 중요하다. 현재 운영체제 연구 분야에서도 멀티미디어 입출력 관리 및 화일 지원에 관하여 많은 연구가 진행중이며 이에 따라 다시 DBMS와 운영체제의 상호 의존 관계에 대한 정립이 필요하게 될 것이다. 더우기 응용 프로그램과의 관계에 있어서는 멀티미디어 DBMS의 기능이 자칫 너무 비대해져서 멀티미디어 DBMS가 응용 프로그램에 대해 단지 데이터 관리 기능만을 제공하는 것이 아니라 이를 전부 포함해 버리는 오류가 생길 수도 있을 것이다.

한편 DBMS 구조에 대한 향후 연구 과제중 필수적인 고려 사항중의 하나로 분산 환경을 효율적으로 지원하는 구조에 관한 연구도 포함되어야 할 것이다.

참 고 문 헌

1. Batory, D. S., Leung, T. Y. and Wise, T. E., "Implementation Concepts for an Extensible Data Model and Data Language," ACM TODS, Vol. 13, No. 3, Sept. 1988, pp. 231~262.
2. Berra, P. B., Chen, C. Y. R., Ghafoor, A., Lin, C. C., Little, T. D. C. and Shin, D., "Architecture for distributed multimedia database systems," Computer Communication, Butterworth, Vol. 13, No. 4, pp. 217~231, 1990.
3. Chang, N. S. and Fu, K. S., "Picture Query Languages for Pictorial Database Systems," IEEE Computer, Vol. 14, No. 11, Nov. 1981.
4. Christodoulakis, S., Vandebroek, J., Li, J., Li, T., Wan, S., Wang, Y., Papa, M., and Bertino, E., "Development of a Multimedia Information System for an Office Environment," in Proc. VLDB, pp. 261~271, 1984.
5. Christodoulakis, S., "Multimedia Data Base Management: Applications and Problems-A Position Paper," in Proc. ACM SIGMOD, pp. 304~305, 1985.
6. Christodoulakis, S., Ho, F. and Theodoridou, M., "The Multimedia Object Presentation Manager of MINOS: A Symmetric Approach," in Proc. ACM

- SIGMOD, pp. 295~310, 1986
- 7 Christodoulakis, S., Theodoridou, M., Ho, F. and Papa, M., "Multimedia Document Presentation. Information Extraction, and Document Formation in MINOS: A Model and a System," ACM TOOLS, Vol. 4, No. 4, Oct. 1986.
 8. Constantopoulos, P., Yergaroudakis, Y., Rabitti, F., et al., "Office Document Retrieval in MULTOS," in Proc. 3rd ESPRIT Technical Week, Brussels, Sept. 1986.
 9. Conti, P. and Rabitti, F., "Retrieval of Multi-Media Document Images in MULTOS," ESPRIT '87, Vol. 2, pp. 1389~1412.
 10. Duchêne, H., Kaul, M. and Turau, V., "Vodak Kernel Data Model," in Proc. 2nd Int. Workshop on Object-Oriented Database Systems, (Springer-Verlag LNCS 334), pp. 242-261, 1988.
 11. Herrtwich, R. G., "Time Capsules: An Abstraction for Access to Continuous-Media Data," in Proc. Reliable Time Sharing Systems, pp. 22~20. 1990.
 12. Kim, W., Bertino, E. and Garza, J. F., "Composite Objects Revisited." in Proc. ACM SIGMOD, pp. 337~347, 1989.
 13. Klas, W., Neuhold, E. J. and Schrefl, M., "Visual Databases need Data Models for Multimedia Data," in Proc. IFIP TC2/WG2.6 Working Conf. on Visual Database Systems, pp. 433~462, 1989.
 14. Klas, W., Neuhold, E. J. and Schrefl, M., "Using an object-oriented approach to model multimedia data," Computer Communication, Butterworth, Vol. 13, No. 4, pp. 204~216, 1990.
 15. Little, T. D. C. and Ghafoor, A., "Multimedia Object Models for Synchronization and Databases," in Proc. IEEE Int. Conf on Data Engineering, pp. 20~27, 1990.
 16. Little, T. D. C. and Ghafoor A., "Spatio-Temporal Composition of Distributed Multimedia Objects for Value-Added Networks." IEEE Computer, Vol. 24, No. 10, pp. 42~50, 1991.
 17. Lockermann, P. C., Multimedia Databases: Paradigm, Architecture, Survey and Issues, Fakultät fuer Informatik, Universitaet Karlsruhe, 1989.
 18. Masunaga, Y., "Multimedia Databases: A Formal Framework." in Proc. IEEE Computer Society Office Automation Sympo., pp. 36~45. 1987.
 19. Masunaga, Y., "An Object-Oriented Approach to Multimedia Database Organization and Management," in Proc. Int. Sympo. on Database Systems for Advanced Applications(DASFAA), pp. 190~200, 1989.
 20. Meghini, C., Rabitti, F. and Thanos, C., "Conceptual Modeling of Multimedia documents." IEEE Computer, Vol. 24, No. 10, pp. 23~30, 1991.
 21. ISO/IEC JTC1/SC29/WG12, MHEG Draft Document S.6, April. 1992.
 22. Nah, Y. and Lee, S., "Two-level Modeling Schemes for Temporal-Spatial Multimedia Data Representation," in Proc. of Int. Conf. on Database and Expert Systems Applications(DEXA), Springer-Verlag, Valencia, Spain, pp. 102~107, 1992.
 23. Nah, Y. and Lee, S., "Object-Relationship Model for Conceptual Modeling of Multimedia Data," in Proc. the 2nd Far-East workshop on Future Database Systems, (Future Databases '92, Advanced Database Research and Development Series-Vol. 3, World Scientific), Kyoto, Japan, pp. 125~132, 1992.
 24. Rabitti, F., "A Model for Multimedia Documents," Office Automation, Springer Verlag, pp. 227~250, 1985.
 25. Roussopoulos, N. and Leifker, D., "Direct Spatial Search on Pictorial Databases using Packed R-trees." in Proc ACM SIGMOD, 1985.
 26. Shimojo, S., Matsuura, T., Fujikawa, K. Nishio, S. and Miyahara, H., "A New Hyperobject System Harmony: Its Design and Implementation," in Proc. Int. Conf. on Multimedia Information systems, pp. 243~257. 1991.
 27. Stonebraker, M., et al., "Document Processing in a Relational Database Systems." ACM TOOLS, pp. 143~158, 1983.
 28. Thanos, C.(ed.), *Multimedia Office Filing-The MULTOS Approach*, North-Holland, 1990.
 29. Tsichritzis, D., Christodoulakis, S., Economopoulos, P., Faloutsos, C., Lee, A., Lee, D., Vandebroek, J. and Woo, C., "A Multimedia Office Filing system," in Proc. VLDB, pp. 2~7, 1983.
 30. Woelk, D., Kim, W. and Luther, W., "An Object-Oriented Approach to Multimedia Databases," in Proc. ACM SIGMOD, pp. 592~606, 1986.
 31. Woelk, D. and Kim, W., "Multimedia Information Management In an Object-Oriented Database System," in Proc. VLDB, pp. 319~329, 1987.
 32. Woelk, D., Luther, W. and Kim, W., "Multimedia Applications and Database Requirements," in Proc. IEEE Computer Society Office Automation Sympo., pp. 180~189, 1987.

- 33. WiSS Implementation Team, "Implementation Notes for the Wisconsin Storage System," Computer Science Department, Univ. of Wisconsin Madison, 1983.
- 34. Yoon, B. D., Suzuki, F., Ishikawa, H. and Makinouchi, A., "Experimental Multimedia DBMS using an Object-Oriented Approach," in Proc. IEEE COMP-SAC, pp. 632~641, 1987.

나 연 목



- 1986 서울대학교 컴퓨터공학과 졸업
- 1988 서울대학교 컴퓨터공학과에 석사학위 취득
- 1990 서울대학교 컴퓨터공학과 박사학위 취득
- 1991 IBM Watson연구소에서 비즈니스 프로세스 모델링에 관해 연구함
- 현재 서울대학교 컴퓨터공학과 박사과정 재학중
- 관심 분야: 데이터베이스, 객체지향 시스템, 멀티미디어 시스템, CAD/CAM 시스템, 소프트웨어 공학

황 수 찬



- 1984 서울대학교 컴퓨터공학과 졸업
- 1986 서울대학교 컴퓨터공학과에 석사학위 취득
- 1991 서울대학교 컴퓨터공학과에 석사학위 취득
- 1991 ~ 현재 한국 항공대학교 진신학과 책임강사로 재직중
- 관심 분야: 데이터베이스, 인공지능, 객체지향 시스템

이 규 철



- 1984 서울대학교 컴퓨터공학과 졸업
- 1986 서울대학교 컴퓨터공학과에 석사학위 취득
- 1990 서울대학교 컴퓨터공학과에 박사학위 취득
- 1989 ~ 현재 충남대학교 컴퓨터공학과 조교수로 재직중
- 관심 분야: 연역 및 객체 지향 데이터베이스, 멀티미디어 시스템, 지식표현, 소프트웨어 공학

이 석 호



- 1964 연세대학교 정치외교학과 졸업
- 1979 미국 텍사스대학교 전산학 석사 및 박사학위 취득
- 1979 ~ 1982 한국과학기술원 김산학과 조교수
- 1982 ~ 1986 한국정보과학회 논문지 편집위원장
- 1986 ~ 1988 한국정보과학회 부회장
- 1988 ~ 1989 IBM Watson 연구소 객원교수
- 1989 ~ 1991 서울대학교 중앙교육연구실 선임 원장
- 1982 ~ 현재 서울대학교 컴퓨터공학과 교수
- 관심 분야: 한글 DBMS, 멀티미디어 데이터베이스