

DB분야의 역사와 기술전망

서울대학교 김형주*

DBMS는 방대한 양의 데이터를 편리하고 효율적으로 저장, 검색할 수 있는 환경을 제공해주는 시스템 소프트웨어이다. 화일시스템에 비해 효율적인 저장시스템 관리와 여러 사용자가 동시에 사용했을 때 동시성 제어 및 고장회복 기능을 가지고 있다. 또한 사용자가 원하는 데이터를 절차적으로 기술할 필요 없이 선언적으로 기술하면 시스템이 효율적인 질의 계획을 세워 최적화하여 처리하는 기능을 가지고 있다.

종래의 자료처리 시스템에서는 응용 프로그램머가 직접 각각의 데이터 파일을 접근하고 관리하기 위해 검색, 삽입, 삭제 및 갱신을 할 수 있는 루틴을 작성해야 했다. 이러한 화일 시스템에서는 응용 프로그램과 데이터 파일간의 상호 의존관계로 인한 데이터의 중복성 및 데이터의 중복과 같은 문제가 나타나게 되었다. 데이터의 중복이 생기게 되면 이 중복된 데이터들 사이의 일관성 및 무결성을 유지하기가 힘들어진다.

관계형 데이터베이스가 나타나기 전에는 계층형, 네트워크 형 데이터베이스 관리시스템이 많이 사용되었다. 계층 데이터베이스 시스템(hierarchical database system)은 트리형태의 데이터 모델을 기반으로 하였다. 계층 데이터베이스 관리시스템의 가장 대표적인 예는 1968년 IBM에서 발표한 IMS(information management system) 데이터베이스 시스템이다. 네트워크 데이터 모델은 1971년 4월의 CODASYL(Conference on Data Systems Languages) DBTG(Database Task Group)의 보고서에서

공식적으로 제안되었다. 이 DBTG 데이터 모델을 기초로 하여 구현된 상용 시스템은 IDMS(Computer Associates), IDS II(Honeywell Information System), DMS/1100(UNIVAC), TOTAL(Cincorn Systems) 등이 있다. 이 계층형, 네트워크 DBMS는 링크(pointer)를 통한 빠른 데이터 접근 측면에서 장점을 지니고 있으나, 데이터들 사이의 물리적인 데이터 중복성이 있게 되는 단점을 가지고 있다.

1970년 E.F. Codd는 CACM에 "The Relation Data Model"이라는 논문을 통해 관계형 모델을 제안하였다. 관계형 모델에서는 비절차적인 선언적 질의어를 이용하여 데이터를 검색하였다. 이 비절차적인 질의어는 질의 처리기에서 질의 최적화(query optimization)를 통해 질의를 처리하게 된다. 1976년 최초의 상업용 관계형 DBMS로 IBM San Jose에서 System/R이 만들어졌다. 또한 같은 시기에 최초의 연구용 관계형 DBMS로 UC Berkley에서 Ingres가 만들어졌다. 관계형 DBMS는 관계형 데이터모델이라는 수학적 기반과, 사용의 편이성이라는 측면에서 장점을 지니고 있다. 반면 평면적인 릴레이션만을 지원함으로써 계층형 및 네트워크 데이터 구조를 직접적으로 표현하는게 불가능하고, 데이터의 값을 기반으로 하는 연산과 조인으로 인해 성능이 저하되는 문제가 있었다. 1970년대에서 1980년대 말까지 관계형 DBMS 모델에 대한 여러 연구들이 있어 왔다. 최적의 관계형 데이터베이스 스키마를 구하기 위한 함수 중복성연구와 질의어 처리의 성능 향상을 위한 SQL문 질의 최적화 알고리즘의 연구, 관계형 데이터 모델의

* 증신회원

표현력을 높이기 위한 의미기반 데이터모델 (semantic data model)들에 대한 연구가 이 기간에 중점적으로 연구되었다.

1980년대 초 CAD/CASE/CAM, 인공지능, 그리고 멀티미디어 등과 같이 DBMS를 사용하기 위한 새로운 응용분야가 출현함에 따라서 데이터 모델의 확장성이 요구되었다. 이에 따라 1985년부터 1995년에 걸쳐서 객체지향 데이터베이스 시스템이 연구되었다. 이 기간의 연구용 프로토타입 시스템으로 ORION, POSTGRES, Encore/ObServer, AVANCE 등이 있고, 상업용 제품으로는 O2, ObjectStore, Objectivity, Versant 등이 있다. 객체지향 데이터베이스 시스템은 객체(object), 객체 식별자(object identity), 클래스, 클래스 계층구조, 계승(inheritance) 등을 요소로 갖는 객체지향 데이터 모델을 지원한다. 그리고 의미기반 데이터모델을 확장하여 복합개체라든지 버전 등의 기능을 지원하고 있다. 새로운 응용분야에서 요구하는 여러 기능들을 지원하기 위해 persistent 프로그래밍 언어, 장기 트랜잭션, 대용량 객체 등에 대한 연구와 지원이 되었다. 객체지향 데이터베이스에 대한 표준은 ODMG-93에서 명문화 되었다. 객체지향 DBMS는 객체 캐쉬의 지원으로 빈번한 동일객체 접근시 빠른 접근을 제공하고 복잡한 데이터모델을 지원하는 장점을 지니고 있는 반면, 효율적인 질의처리가 부족하고 시스템의 안정성 측면에서 관계형 DBMS에 비해 부족한 면이 있다.

객체지향 DBMS의 여러 장점들을 관계형 데이터베이스에 통합 흡수하여 객체 기능을 갖춘 상용 객체 관계형 DBMS(ORDBMS)가 1996년 등장하였다. ORDBMS의 주요 기능으로는 대용량 객체(LOB)의 지원, 사용자 정의 데이터 타입 지원(객체지원), 타입 계승 지원, 사용자 정의 함수 및 저장 프로시저 지원, 응용프로그램 도메인에 특화된 확장 함수 지원, SQL 프로시저 확장, 그리고 롤/트리거 시스템의 지원등을 들 수 있다.

국내에서도 '88 올림픽 이후 컴퓨터 소프트웨어 분야의 활성화와 한국형 DBMS 개발의 필요, SI 분야의 활성화로 해외 DBMS 회사에 지불하는 로열티 비용의 증가 등의 이유로 DBMS 개발을 시도하게 되었다. 이 결과 1989년 인하디에서 개발한 관계형 데이터베이스 KORED를 시발

로 하여 KAIST의 IM, ETRI에서 개발하여 대우통신에서 상품화한 바다-I, 삼성전자에서 개발한 코다, 서울대 OOPSLA 연구실에서 개발한 관계형 DBMS인 SRP, 객체지향 DBMS SOP, KAIST에서 개발한 Odysseus, GIS 전용 OODBMS인 OMEGA 등 여러 DBMS가 개발되었다. 하지만 개발된 DBMS의 안정성 및 성능 등의 문제로 인하여 여러 기관들에서 많이 사용되어지는 않고 있다. 외국의 상용 DBMS 회사들은 SQL 3 지원이나, 롤/트리거 등의 지원 등 여러 부분에서 국내 개발 DBMS를 앞서고 있는 형편이다.

ORDBMS 시장의 형성도 Oracle, Informix, Sybase와 같은 관계형 DBMS 업체의 주도에 의해 이끌려가고 있다. 아직도 선진 major vendor들의 벽은 높다고 할 수 있다. Oracle은 전 세계적으로 직원이 4만 명이 넘고, 1년 총 수익이 85억불 정도 된다고 한다. 향후 국내 소프트웨어 분야가 발전되기 위해서는 성공한 소프트웨어 제품과 회사의 주요 성공 요인들을 분석하고, 아웃소싱과 기술 제휴 등을 적극적으로 검토해야 할 것이다. 또한 경쟁력 있는 소프트웨어 기술자를 개발하고, 벤처기업이나 대학, 연구소 등도 실리콘 벨리에 진출하여 여러 선진 기술 습득이나 동향 파악에 주력해야 할 것이다.

1990년대에 들어서 World-Wide-Web이 모든 종류의 정보를 공유하고 저장하는 데의 주된 용도로 사용되게 되었다. Web 상의 데이터들은 일부는 일정한 구조를 지니기도 하지만, 비정형적인 꼴을 갖는 경우도 많아졌다. 전통적인 데이터베이스 관리 시스템에서는 데이터들이 미리 정의된 스키마를 가지고 있고, 모든 데이터들은 이 스키마에 맞는 구조를 가져야만 하였다. 관계형 데이터 모델보다 좀 더 풍부한 모델 표현이 가능한 객체지향 데이터베이스 시스템에서조차도 모든 데이터들은 미리 정의된 스키마에 따라야 했다. Web의 출현과 멀티미디어 데이터의 등장으로 인하여 이러한 이질적인 데이터 소스로부터 정보를 추출하고 통합하는 일이 필요하게 되었다. WWW에서 데이터에 대한 표현과 Web상의 데이터 교환의 새로운 표준으로 XML(Extended Markup Language)를 채용하게 되었다. XML은 데이터 마크업 언어로서뿐만 아니라

semistructured data를 위한 데이터 모델링 언어로서도 사용될 수 있다. 이에 따라 semi-structured data에 대한 많은 연구들이 XML에도 적용되고 있다. 이질적인 정보소스로부터 정보를 통합하여 사용자에게 보여주기 위한 수단으로 크게 두가지 접근 방법이 있다. 하나는 사용자가 통합된 뷰에 질의를 하기 전에 미리 데이터를 모아서 통합하는 웨어하우징(warehousing)방법이다. 다른 하나는 사용자가 질의할 때 필요한 웹 데이터를 접근하여 처리하는 방법이다. 이를 가상 접근 방법(virtual approach)이라고 한다. 가상 접근 방법에서는 데이터는 웹 소스에 있고 사용자의 질의가 실행시 여러개의 서버 질의문으로 나뉘어져 처리되게 된다. 가상 접근 방법은 데이터 소스 양이 방대하고 데이터가 자주 바뀌는 상황에서 적절하다고 할 수 있다. 이러한 가상 접근 방법에는 미들웨어 기술이나 중재자(mediator)기술이 많이 사용되고 있다. Mediator는 상위 응용 프로그램을 위해 정보를 생성시키기 위해 어떠한 데이터 셋에 대하여 정보를 추출 가공하는 모듈을 말한다. Mediator가 가지는 특징으로는 생성 또는 관리를 위해 특정한 도메인에 맞도록 데이터의 분할, 손쉬운 재조정을 위해 모듈 구조화 되어 있을 것. 비동기적인 데이터

접근을 위한 데이터 캐싱, 성능을 위한 데이터 복제 등을 들 수 있다.

앞으로는 데이터베이스 자체에 대한 연구뿐만 아니라 Internet/Web 환경을 위한 데이터베이스 연구, 이질적인 데이터의 통합을 위한 데이터 웨어하우징 기법이나 미들웨어 기법에 대한 연구들이 2000년대에 활발히 진행될 연구 주제로 남아 있다.

김 형 주



1982 서울대 컴퓨터공학과(학사)
 1985 Univ of Texas at Austin
 전자계산학과(석사)
 1988 Univ. of Texas at Austin
 전자계산학과(박사)
 1988~1990 Georgia Institute of
 Technology 조교수
 1991~현재 서울대학교 컴퓨터공학과
 부교수
 관심분야: 객체지향시스템, 사용자
 인터페이스, 데이터베이스
 E-mail:hjk@oopsia.snu.ac.kr

● HCI 2000 ●

- 일 자 : 2000년 1월 24 ~ 26일
- 장 소 : 피닉스 파크 컨벤션 센터
- 주 최 : HCI·컴퓨터그래픽스연구회, 한국가상현실협회
- 문 의 처 : 한국과학기술원 HCI 2000 사무국
 Tel. 042-869-5572