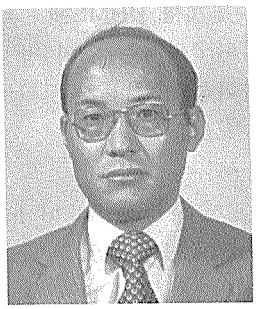
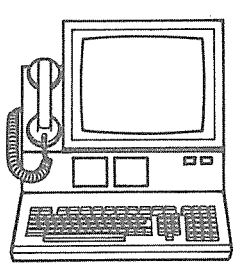


吳 吉 祿
韓國電子通信研究所
주전산기개발본부장 / 工博

데이타 베이스 관리체계에서 한글처리기능에 관한 연구



1. 머리말

데이타 베이스가 상용화되어 컴퓨터 응용에 이용되어 온 지 약 20년 정도가 지났다. 그동안 많은 양의 자료를 빠른 시간에 처리하기 위한 꾸준한 노력이 진행되어 온 결과로 데이타 베이스는 각종 컴퓨터 응용의 핵심이 되었으며, 고수준의 질의어, 빠른 처리구조, 병행수행 제어, 회복기능 등의 유용한 기능을 제공하고 있다.

데이타 베이스의 응용이 전통적 응용분야인 경영분야를 벗어나, 급속히 발전하는 정보과학(informatics)의 여러분야(소프트웨어 공학, 인공지능, 사무자동화, 전산보조설계 등등)에 걸쳐서 활용됨에 따라 새 기능을 지닌 데이타 베이스의 개발이 요구되었다. 이것은 한편으로는 데이타 베이스 기술이 어느 정도 완숙단계에 들어섰기 때문이기도 하고 또 다른 한편으로는 데이타 베이스 기술이나 도구가 이제는 여러 정보과학 분야에서 널리 사용되고 있기 때문이다.

한국형 다중처리 전산기 개발 사업의 일환으로서 다양한 응용분야의 요구조건을 충족시키기 위하여 필요한 중요한 기능 중의 하나가 사용자가 쉽게 다룰 수 있고 성능이 우수한 데이타 베이스 관리체계(DBMS)의 개발, 설치이다. 특히 행정전산망에 사용되기 위하여는 대단위(very large) 데이타 베이스, 동시에 많은 사용자를 포용할 수 있는 동시성 조정기능(concurrency control)이 보강된 데이타 베이스 그리고 분산된 컴퓨터들 위에서 데이타 베이스를 공유하여 사용할 수 있도록 하는 트랜잭션(transaction) 처리가 가능한 데이타 베이스 시스템이 요구된다.

본 소과제의 주된 목적은 위에서 열거한 조건을 충족시켜 주는 한국형 DBMS를 개발하여 주과제에서 개발되는 다중처리 전산기에 설치 가동시키는 데에 있다. 데이타 베이스 개발 경

험이 없는 국내의 사정으로는 이러한 목적을 달성하기 위하여 요구조건에 맞고 이미 세계시장에서 호평을 받고 있는 DBMS를 선정, 분석하여 기술 습득이 우선적으로 요구된다. 본고의 2장에서는 사례연구를 통하여 현재의 국내외 동향을 파악하고 앞으로 개발할 한국형 DBMS 의 개발방향을 살펴본다.

또한 앞으로 개발할 DBMS에서는 한글 자료를 주로 취급하므로, 한글처리기능 구현에 선행되어야 할 한글자료의 구분과 시스템 내에서의 효율적인 한글 자료 표현법에 대하여 3장에서 연구한다.

2. 사례연구

최근 들어 데이타 베이스를 이용하는 응용분야가 늘어나고 그 응용들의 요구가 다양해지며 컴퓨터의 기술이 발전됨에 따라 데이타 베이스를 관리하는 DBMS에 새로운 기능들이 요구되고 이에 호응하여 관련 회사들이 여러 다양한 제품들을 개발, 판매하고 있다. 본 사례연구에서는 여러 데이타 모델 중에서 관계형 모델에 한하여 세계 시장에 나와 있는 DBMS 제품들 중 널리 판매되고 있는 Informix Software Inc. 의 Informix - SQL, Oracle Corp. 의 ORACLE, Unify Corp. 의UNIFY, Relational Technology Inc. 의 INGRES 등의 관계형 DBMS에 대해 조사하였다. 국내 사례연구에서는 위의 상품을 제외하였으며 한글처리기능을 얼마나 지원하는가 하는 관점에서 살펴 보았다.

(1) 국외사례

(표 1)은 네 회사의 관계형 DBMS들인 Informix - SQL (ISQL), ORACLE, UNIFY, 그리고 INGRES를 일반적인 항목으로 비교하였다. [4, 5] 이들 네 제품은 외국 시장에서의 높은 지명도에 반하여 국내에서는 겨우 소개 단계에 머물러 있는 실정이다. 네 상품의 우열을 가리는 것은 본 논문의 목적이 아닐 뿐더러, 현지인 미국에서도 각 제품의 우열에 관한 평가가 보는 관점에 따라 다른 형편이므로, 본고에서는 현재

표 1. ISQL, ORACLE, UNIFY, INGRES 관계형 DBMS 비교

	Informix-SQL	ORACLE	UNIFY	INGRES
Company Name	Informix Software	Oracle Corp.	Unify Corp.	Relational Tech. Inc.
O. S. Supported	UNIX PC-DOS MS-DOS	MS-DOS PC-DOS VM/CMS AOS/VS UNIX(V) AEGIS MVS VOS VMS MTS GLOS	UNIX MS-DOS	VMS UNIX
Access Method	B-trees	B-trees	Hashing B-trees links	Hashing B-trees
% of Tables	Unlimited	Unlimited	256	Unlimited
Support Raw I/O	No (Informix-Turbo)	Yes	Yes	No
Ouary Language	SQL	SQL	SQL QBF	QUEL SQL
Interactive Screen Package Name	PERFORM	FORMS	PAINT	CUOID QBF
Support Multiple Tables/Screen	Yes (14 tables)	Yes (Unlimited)	No (Only One)	Yes No
Report Writer	ACE	PLUS	PPT	RBF
Supported Language	C Cobol	C Fortran Cobol PL/1 Pascal	C Cobol	C Fortran Cobol Basic Pascal

각 소프트웨어 회사들이 연구, 개발하고 있는 주요 방향을 간략하게 살펴봄으로써, 차후 DBMS 성능 평가의 기준을 제시한다.

가. 여러 사용자들을 지원하는 DBMS의 성능 향상시 제기되는 문제점과 해결안에 대한 연구

데이타 베이스를 이용하는 여러 사용자들을 효율적으로 지원하기 위한 Informix사의 연구내용을 살펴보자.[3] 먼저 문제점들을 나열하면 다음과 같다.

- ○Single Processor Computer
- ○Software Architecture
- ○Data Redundancy
- ○Disk I/O

(가) Single Processor Computer

multi-user 컴퓨터의 사용자들이 한개의 처리기를 차지하기 위하여 서로 경쟁하기 때문에 야기되는 성능 저하를 다중 처리기 시스템으로 하는 경우 처리기에 대한 경쟁을 완화시킬 수 있다. 그러나 다중 처리기 시스템으로 하는 경우도 병렬 프로그래밍이 아닐 때는 프로그램 자체가 더 빨리 수행되는 것이 아니라 처리기로부터 빨리 서비스를 받음으로써 응답속도만 빠르게 된다.

(나) Software Architecture

Informix-SQL 계열의 제품은 요청 프로세스(requestor process)와 데이타 베이스-서버 프로세스(database-sever process)로 구성된 구조를 갖는다. 데이타 베이스-서버 프로세스는 SQL을 사용한 데이타 베이스의 업무 처리를 수행하는데 데이타 베이스 서버로 질의문을 받아 주기억 장치내의 자료나 디스크내의 자료를 이용하여 명령문을 수행한다. 요청 프로세스는 컴파일된 응용 코드를 처리하거나 보고서 작성을 포함한 모든 사용자 인터페이스를 조정하는 프로세스이다.

요청 프로세스와 데이타 베이스-서버 프로세스가 분리된 구조를 갖는 DBMS를 다중 처리기 시스템에 구현하는 방법에는 여러 개의 요청 프로세스에 하나의 데이타 베이스-서버가 존재하는 집중 방식과 각 요청 프로세스마다 한개씩 데이타 베이스-서버가 존재하는 분산 방식의 두 가지가 있는데 분산된 방식의 경우 각 데이타 베이스-서버가 서로 다른 처리기에서 수행될 수 있으므로 다중 처리기 시스템에 적합한 소프트웨어 구조는 분산된 구조이다.

Informix-4GL를 포함한 Informix사의 제품은 각 요청 프로세스마다 서버 프로세스를 제공하여 각 데이타 베이스-서버는 한개의 처리기 위에서 일을 수행하게 하므로써 한 처리기에 과적(overloading)되는 병목현상을 제거하였다.

(다) Data Redundancy - Buffer Pool Contention

여러 개의 처리기를 둠으로써 처리기 이용에 대한 병목현상이 제거되어 다수의 사용자들이 동시에 동일한 데이타 베이스를 검색 및 갱신할 수 있다. UNIX에서 프로그램 코드는 재진입(re-entrant) 코드이므로 데이타 베이스-서버 프로그램은 메모리에 단 하나만 있으면 되지만 서버 프로세스에 필요한 지역 자료 세그먼트(local data segment)는 서버 프로세스마다 필요하다. 한 사용자에 의해 접근되는 자료의 일부는 다른 사용자의 지역 버퍼에 중복되어 저장되는 경우가 있다. 이렇게 각각의 지역 버퍼에 자료를 중복되게 저장하여 비싼 기억 장소가 낭비되고, 또한 지역 버퍼에 있는 자료의 정확성의 유지를 위하여 디스크로부터 자주 갱생(refresh)하여야 하는 번거로움이 있다. 이의 해결안으로 UNIX 시스템 V의 공유 기억장치(shared memory) 기능을 이용한 버퍼 풀을 공유하는 방식을 채택한다. 버퍼 풀을 공유하는 방식을 채택함으로써 버퍼 풀이 커질수록 디스크 입출력이 적어지며 각 서버 프로세스마다 버퍼를 가지고 있지 않아도 되므로 기억 장소의 낭비를 줄일 수 있다.

(라) Disk I/O

버퍼 풀이 데이타 베이스보다 크지 않는 한 디스크 입출력은 계속해서 발생한다. 일반적으로 UNIX의 디스크 입출력에 대한 버퍼링은 블록 단위(일반적으로 512 바이트로 고정)로 이루어지는데 응용 프로그램에서 블록의 크기를 조정할 수 없어 디스크 입출력이 많은 데이타 베이스 응용에 부적합하다. 이런 문제점을 해결하기 위하여 Informix사에서는 UNIX 화일 시스템과

인터페이스 없이 바로 디스크에 접근하는 저차원 입출력(raw I/O) 방식을 사용한 새로운 제품(Informix - Turbo)을 개발하였고, Unify사도 Express I/O라는 부분을 만들어 저차원에서 입출력을 하게 하였다.〔4〕

나. 소프트웨어 개발의 생산성 향상을 위한 접근방법(제 4 세대 프로그래밍 언어)

작성과 이해 그리고 유지보수가 쉬우며 신뢰성도 높은 소프트웨어에 대한 요구는 80년대 초부터 제 4 세대 프로그래밍 언어(4 th Generation Language : 4 GL)의 등장을 촉진하였다. 스프레드 시트나 그래픽 패키지 등 강력한 소프트웨어 기능을 갖춘 마이크로 컴퓨터는 컴퓨터를 이용하는 사람들의 기대를 모았다. 이러한 점들은 소프트웨어 개발자들이 제 3 세대 언어(우리가 흔히 말하는 고급 프로그래밍 언어들) 보다 사용하기 쉽고 강력한 언어를 개발하게 되는 계기가 되었다. 제 4 세대 언어는 비절차적인 구조를 가지며, 질의나 보고서 작성같은 반복적이고 성가신 일들을 자동화하고, 시스템 명령어와 같은 제어 언어로서도 사용하고, 제 3 세대 언어의 인스트럭션들을 포함하는 언어이다.〔2〕 이러한 제 4 세대 언어는 생산성이 높고, 사용하기에 편리하며, 적응성과 효율성이 높을 뿐만 아니라 각종 컴퓨터 시스템간에 쉽게 이식할 수 있다.

소프트웨어 개발의 생산성 향상을 노리고 시판되고 있는 4 세대 언어의 제품으로 INFORMIX사의 Informix - 4 GL, UNIFY사의 ACCELL이 있다. ACCELL은 4 세대 언어 뿐만 아니라 관계형 데이타 베이스 시스템, 응용 생성기, 그리고 윈도우 인터페이스를 통합한 종합 응용 개발체계이다.〔5〕

다. 분산 관계 데이타 베이스 시스템으로의 접근

ORACLE 회사에서는 SQL* Star라는 분산형 관계 데이타 베이스 시스템을 발표하였다. SQL* Star 시스템은 기계의 종류나 DBMS 로부터 독립적인 환경을 사용자에게 제공하고 모든 질의는 자료의 지리적인 위치에 무관하게 분산 DBMS에 의해 자동적으로 처리된다.

현재 SQL* Star 시스템은 IBM의 VMS/CMS와 IBM PC - DOS 등 IBM 계열 뿐만 아니라 UNIX 계열기기 사이에서도 호환성 있게 운용되는 제품을 시판하기 시작하였다.

(2) 국내사례

앞으로 개발할 DBMS의 한글처리기능을 정립하기 위하여 현재까지 국내에서 사용중이거나 판매 및 개발된 DBMS들 중 한글처리가 가능하다고 주장하는 시스템에 대하여 다음과 같은 조사를 하였다.

가. 한글처리 가능한 DBMS에 관한 조사

국내에서 한글자료처리가 가능하다고 주장하는 DBMS를 제공하는 업체는 17개 회사로 조사되었다. 그러나 여기에서는 DBMS를 개발하는데 있어 일차적으로 중요하다고 판단된 아래와 같은 DBMS들에 대하여만 방문조사를 통하여 한글처리기능 즉, 한글 자료의 내부 표현 형태, 릴레이션 또는 속성 이름을 한글로 사용할 수 있는가, DBMS에서 출력되는 전달사항(message)이 한글로 되는가, 한글 질의어 가능 여부 등에 대한 조사를 시행하였다.〔6〕 (표 2)는 국내의 대표적인 DBMS 비교 표이다.

나. 국내사례 연구의 분석 결과

이번 한글 처리 가능한 DBMS 조사를 통하여 살펴본 결과 처음부터 DBMS를 독자적으로 개발한 경우는 없고 대부분이 외국에서 개발된 DBMS를 바탕으로 하여 한글처리 기능을 첨가하는 방식으로 개발하였다.

이들의 개발 방식은 첫째, 한글처리기능 개발자가 DBMS의 원천 코드를 다룰 수 있느냐 하는 사실에 따라 아래와 같이 구분된다.

○원천 코드를 다룰 수 없는 경우

이러한 경우 DBMS의 한글처리기능은 한글자료 입출력만이 가능하고, 일반적으로 n - 바이트 한글 코드를 사용한다. (예 : ORACLE, FOCUS, RAMIS - Ⅱ)

○원천 코드의 전부 혹은 입출력 부분을 다룰 수 있는 경우

이러한 경우 DBMS의 한글처리기능은 일반적으로 한글자료 입출력, DBMS의 한글 메시지 등의 제공이 가능하고, 2 바이트(조합형) 한글 코드를 사용하고 질의어의 일부분을 한글화하고

표 2. 국내외 대표적인 DBMS 비교

	samsung	goldstar	k. c. c	ssangyong com.
DBMS	IDMSR	10-BASE	INFORMATION	RAMIS-II
data model	network, relation	relation	relation	relation
source lang.	assembly	C, assembly	assembly	assembly
source available	no	yes	i/o part	i/o part
Machine (O. S)	IBM (MOS, MVS/ SP, VM, DOS/VSE)	IBM (hangul DOS)	PRIME (PRIMEOS)	IBM
size	main	pc	mini	main
hangul processing ability :				
internal representation	DBCS (double byte control system)	2 byte	2 byte	2 byte
relation name	hangul	English	English	English
field name	hangul	hangul	hangul	hangul
system message	English	hangul	hangul	English
query lang.	English	English	English	English
chinese cha.	no	no	no	no
the others		multi-user through LAN report generator	ROAM : file manager	

있다. (예 : 샛별DB)

둘째, DBMS를 사용하는 시스템의 운영체제에서의 한글기능 지원 문제이다. 즉 현재 대부분의 DBMS는 릴레이션 이름이 화일 체계에서의 화일 이름과 일치하는데, 일반적으로 운영체제에서 한글 화일 이름을 허용치 않음으로서 (예외 : 한글 MS - DOS) 완전하게 한글처리가 가능한 DBMS 개발이 불가능하다.

그러므로 관계형 DBMS에서 한글처리기능에 대한 현 국내 상황은 소형 컴퓨터 이상 계열의 경우보다는, 원천 코드와 한글기능을 지원받을 수 있는 운영체제에 비교적 쉽게 접근할 수 있는 개인용 컴퓨터 계열에서 DBMS의 한글처리화가 잘 되어 있으며 한글의 내부 표현은 한자 표현도 고려하여 2 바이트 조합형의 사용이 일반화되고 있다.

3. 한글처리기능을 위한 제안

사례 연구한 결과를 바탕으로 하여 DBMS 에서 지원해야 할 한글처리기능의 종류를 나열하면 다음과 같다.

○데이타 베이스, 릴레이션, 속성에의 한글 이름 사용
○한글 자료태 정의
○한글 자료의 입출력
○한글 차림표를 포함하는 한글 명령어 사용
○시스템 메시지의 한글화
○한글 사용자 지침서

이와 같은 한글처리기능들을 실현하기 위해선 먼저 한글 자료의 명확한 정의와 한글 자료의 시스템 내부 표현 방법에 대해 선행 연구되어야 한다.

(1) 한글 자료의 종류

가. 시스템 메시지 : 데이타 베이스를 사용하는 사용자를 돕기 위해 시스템에서 제공하는 모든 전달 사항들로 시스템을 구현할 때 결정되어 사용시 변경되지 않는다.

나. 한글 자료태 : 한글 자료를 입출력하기 위해선 기본 자료태 내에 한글자료태를 첨가하여야 한다. 새로운 자료태의 지원은 자료 입력시 Integrity 검사, 내부 표현, 터미날이나 프린터를 위한 변환, 그리고 값의 비교와 sorting 등

의 기능을 모두 지원하여야 한다.

다. 한글 이름 : 데이타 베이스 구축할 때와 구축된 데이타 베이스를 수정, 질의할 경우 자료 정의어 혹은 자료 취급어 내에서 데이타 베이스, 릴레이션, 속성들에 대해 한글 이름들을 사용할 수 있어야 한다.

라. 한글 자료태로 정의된 속성의 값으로서 한글 자료

위의 모든 한글에 관계된 것을 취급하는 정책과 기술이 전체 DBMS의 입장에서는 물론 운영체제의 입장에서 검토되고 결정되어야 한글처리 기능 설계를 시작할 수 있다.

(2) 한글 표현 방법

가. 한글 자료 표현을 위한 한글 코드 표현 방식

가장 기본적으로 제공되어야 할 부분으로 사용자가 사용하는 한글 자료의 입출력을 가능하게 한다. 이 부분에서 중요하게 고려되어야 할 점은 내부적인 한글 표현을 위한 코드의 선택이다. 현재 DBMS에서의 한글 코드 표현 방식은 2바이트 조합형으로 모아지는 경향을 보이고 있고 또한 다른방식에 비하여 많은 장점을 갖고 있으므로 2바이트 조합형 코드를 사용할 것을 제안한다.〔7〕 참고로 1986년도에 한국표준연구소에서 시행한 한글부호표준시안 작성을 위한 연구 결과로 정보교환용 부호로 2바이트 완성형을 채택하였고, 내부 처리용으론 2바이트 조합형과 n-바이트 방식 모두를 권장안으로 채택하였다.〔9〕

나. 시스템 내부의 한글 자료 표현 방법

일반적으로 컴퓨터 시스템에서 한글을 표현하는 방법은 한글 자료의 존재 장소에 따라서 구분될 수 있다. 즉 정보교환용, DBMS내, 화일에서의 한글 표현 방법으로 아래와 같이 구분된다.

(가) 정보 교환용 한글 자료 표현 방법

CRT, 프린터와 같은 단말기에서 한글 표현 방법은 단말기 종류에 따라 다르며 현재 일반적으로 사용되는 한글단말기들은 대부분 과학기술처 7비트 한글 표준 코드를 제공하고 있다. 이러한 단말기에서 한글 자료는 7비트 n바이트의 열이며 한글 시작과 영문 시작을 나타내는 구분 코드가 한글 자료의 앞뒤에 붙는다.

(나) DBMS내의 한글자료 표현 방법

효율적인 DBMS를 구현하는데 있어 가장 중요한 요인중의 하나가 주기억 장치에서의 한글 자료 표현 방법이다. 이 표현 방법은 가능하면 (a)와 (c)의 표현과 호환적이어야 하고 DBMS에서 처리하기 쉬운 형태를 갖추어야 한다. 그러나 한글 자료 표현 방법의 완벽한 통일성을 주기에는 많은 무리가 따르므로 가능하면 (c)와 같은 코드 형태를 채택하기로 한다.

(다) 화일에서 한글자료 표현 방법

한글 자료를 디스크나 테이프의 화일로 저장했을 때의 표현 방법으로 가능한 한글 자료를 축소시켜 보조기억장치에서의 저장량을 최소한으로 줄인다. 본고에서는 저장할 화일에서 한글 자료태로 정의된 속성의 자료만 2바이트로 저장하고 그외의 자료태로 정의된 속성들의 자료는 1바이트로 저장하는 방법을 제안한다.

4. 맺음말(향후 추진 계획)

DBMS는 단위가 매우 큰 소프트웨어로 그 개발 기간도 장기간(본 사업이 제안됐을 때는 3년 계획이었음)에 걸친다. 개발 첫해인 올해엔 본격적인 개발 업무의 준비 작업을 포함한 다음과 같은 일을 수행하였다.

- ○차기년도 연구개발을 위한 환경조성과 필요한 도구개발
- ○기존의 외국의 DBMS들의 기능 분석과 대상체계 선정
- ○한글처리기능 확충을 위한 제안
- ○DBMS의 기본 부분인 자료 접근 방법에 관한 연구
- ○목표 DBMS 설계를 위한 기초연구

제2차연도엔, 선정한 대상체계를 도입하여 그 원천코드를 분석하고, 1차연도에 제안한 한글처리기능을 구현하여 데이타 베이스 관리체계 개발기술을 축적한다. 그리고 자체 데이타 베이스 관리체계를 개발하기 위한 요구 조건을 분석, 기능적 모습을 결정한다. 제3차연도엔,

자체 데이타 베이스 관리체계를 주어진 개발환경 위에서 설계, 구현한다. 이 기간 동안에는 데이타 베이스의 새로운 기능들을 연구하여 이들을 목표 데이타 베이스 관리체계에 포함시켜야 할 것이다. 그 이후엔 구현된 결과를 시험 운전한 후, 국내에서 개발된 컴퓨터 시스템에 이식 설치하여 한국형 DBMS를 개발 완성하여 보급한다.

기대 효과로는, 현재 전적으로 외국산 DBMS를 사용하는 국내사정을 고려할 때 본 과제가 끝나면서 얻을 수 있는 수입대체 효과도 클 것이다. 그리고 과제 수행 기간동안 얻어지는 데이타 베이스 기반기술의 추적은 소프트웨어 저작권법 실시 등으로 선진국이 고급 기술을 보호하는 추세로 보아, 우리가 마지막 기회를 활용하게 되는 것이다.

참고문헌

1. Ullman, Jeffrey D., Principles of Database Systems, Computer Science Press, 1982.
2. Chorafas, D. N., Fourth and Fifth Generation Programming Language, McGraw-Hill Book Company, 1986.
3. INFORMIX Software Inc., "SQL-Based Products for Multiprocessor Machines", Technical Paper, 1986.
4. Oracle Corp., "ORACLE versus INGRES, INFORMIX and UNIFY", 1985.
5. Unify Corp., "A Comparison between ACCELL Integrated Development System and INFORMIX-4GL", Technical Paper, 1986.
6. 김문자, 허대영, "한-DBMS의 한글처리기능 개발을 위한 국내 사례조사", ETRI 컴퓨터 개발부 메모, 1986.
7. 김문자, 허대영, "제 1 차년도 한-DBMS의 한글 처리기능을 위한 제안", ETRI 컴퓨터개발부 TD-86-1200-06, 1986.
8. 한국전자통신연구소, Multiprocessor 컴퓨터 개발에 관한 연구", 최종 보고서, 1987.
9. 한국표준연구소, "한글/한자 코드 표준화 연구 주요내용 설명자료", 1986.