



Chap 1. Database and Database user



Introduction

개요

- 데이터베이스 : 컴퓨터의 활용 범위가 확대됨에 따라 모든 분야에서 사용
(상업 , 공학, 의학 , 법학, 교육, 도서관 ,....)

정의

- 데이터베이스 : 연관된 데이터의 모임
(collection of related data)
- 데이터 : 의미를 갖는 사실 (known facts)
 - [전화번호, 주소, 주민등록번호]
- 책의 한 페이지 : 관련된 데이터의 모임
=> 내포된 성질 정의 필요



Introduction

- 성질 : Implicit properties
 - Universe of Discourse (UoD) : mini-world
 - 고유의 의미를 갖고 논리적으로 결합된 데이터의 모임
 - 특정한 목적을 위해 설계, 구현됨



Introduction

- 데이터베이스 관리시스템(DataBase Management System : DBMS) : 데이터베이스를 정의, 생성, 유지하는 프로그램들의 집합
 - 데이터베이스 정의 : 데이터의 형, 구조, 제약조건 정의
 - 데이터베이스 구축 : 저장 매체에 데이터를 저장함
 - 데이터베이스 조작 : 특정한 데이터의 질의 및 검색
- 데이터베이스 시스템
 - DBMS + Application Program + Stored Database + Stored Database Definition

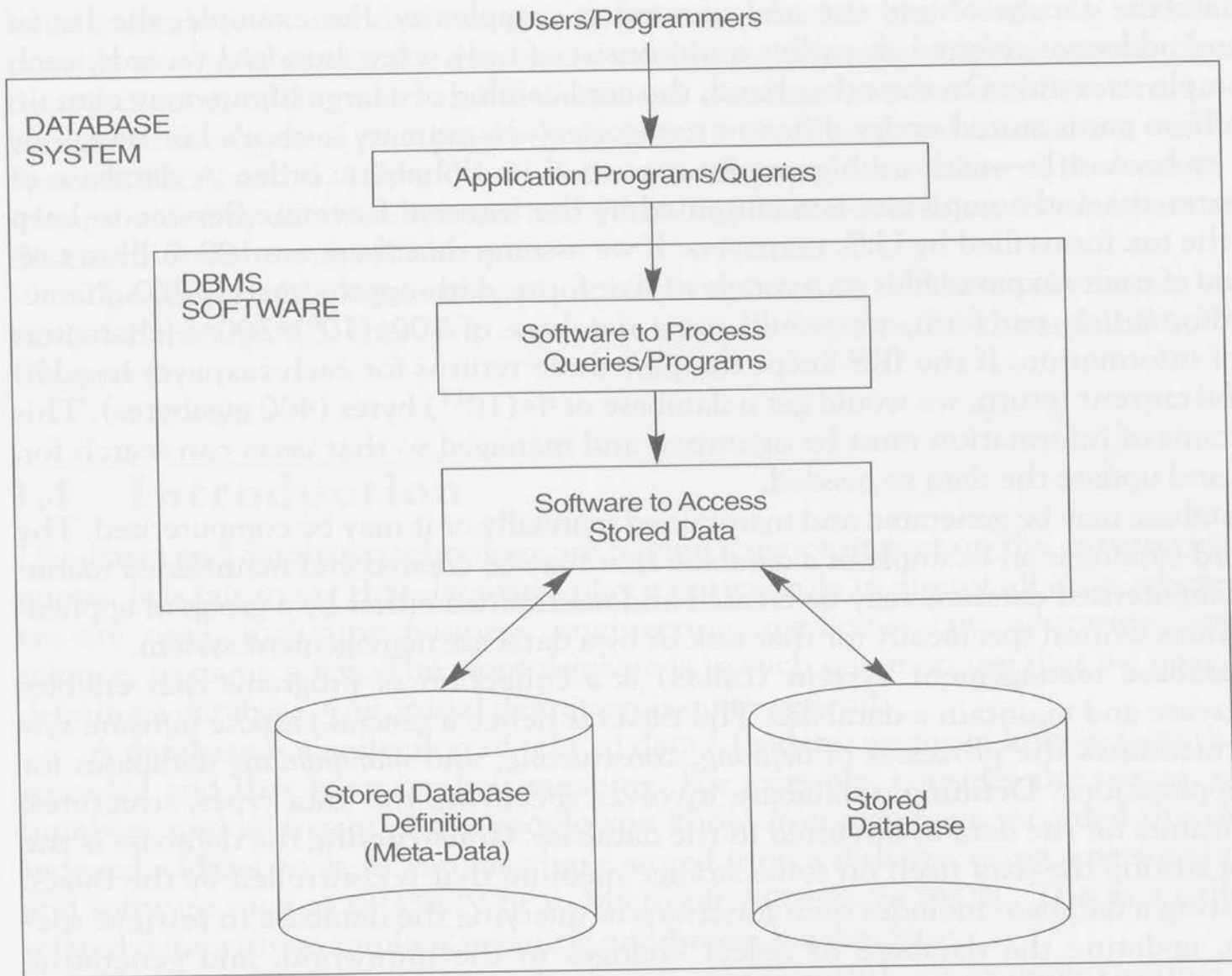


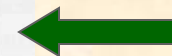
Figure 1.1 A simplified database system environment, illustrating the concepts and terminology discussed in Section 1.1.

STUDENT	Name	StudentNumber	Class	Major
	Smith	17	1	CS
	Brown	8	2	CS



학생개인정보

COURSE	CourseName	CourseNumber	CreditHours	Department
	Intro to Computer Science	CS1310	4	CS
	Data Structures	CS3320	4	CS
	Discrete Mathematics	MATH2410	3	MATH
	Database	CS3380	3	CS



과목정보

SECTION	SectionIdentifier	CourseNumber	Semester	Year	Instructor
	85	MATH2410	Fall	98	King
	92	CS1310	Fall	98	Anderson
	102	CS3320	Spring	99	Knuth
	112	MATH2410	Fall	99	Chang
	119	CS1310	Fall	99	Anderson
	135	CS3380	Fall	99	Stone



강의분반

GRADE_REPORT	StudentNumber	SectionIdentifier	Grade
	17	112	B
	17	119	C
	8	85	A
	8	92	A
	8	102	B
	8	135	A



성적정보

PREREQUISITE	CourseNumber	PrerequisiteNumber
	CS3380	CS3320
	CS3380	MATH2410
	CS3320	CS1310



선수과목정보

Figure 1.2 An example of a database that stores student records and their grades.



데이터베이스의 정의

📌 레코드 구조 정의

[학생화일 레코드 구조 - 이름, 학번, 학년, 전공]

📌 데이터 형 정의

- 이름: 문자열, 학번 : integer
- 학년: 1=>1학년, 2=>2학년, 3=>3학년,
4=>4학년, 5=>대학원생
(coding scheme)
- 성적 : 문자 {A, B, C, D, F}



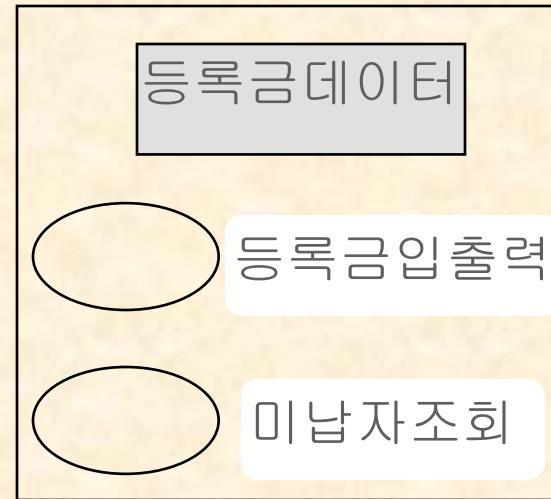
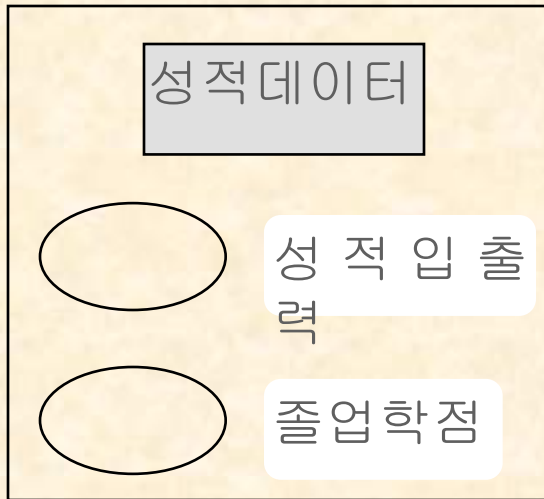
데이터베이스의 정의

- ❗ 데이터베이스 구축 : 데이터의 저장
 - 학생 [홍길동, 999999999, 3, CS]
- ❗ 데이터베이스 조작 : 검색, 갱신, 첨가, 삭제
 - 홍길동의 1994년도 1학기의 성적은?
 - 홍길동을 2학년으로 바꾸시오



데이터베이스 방법의 특징

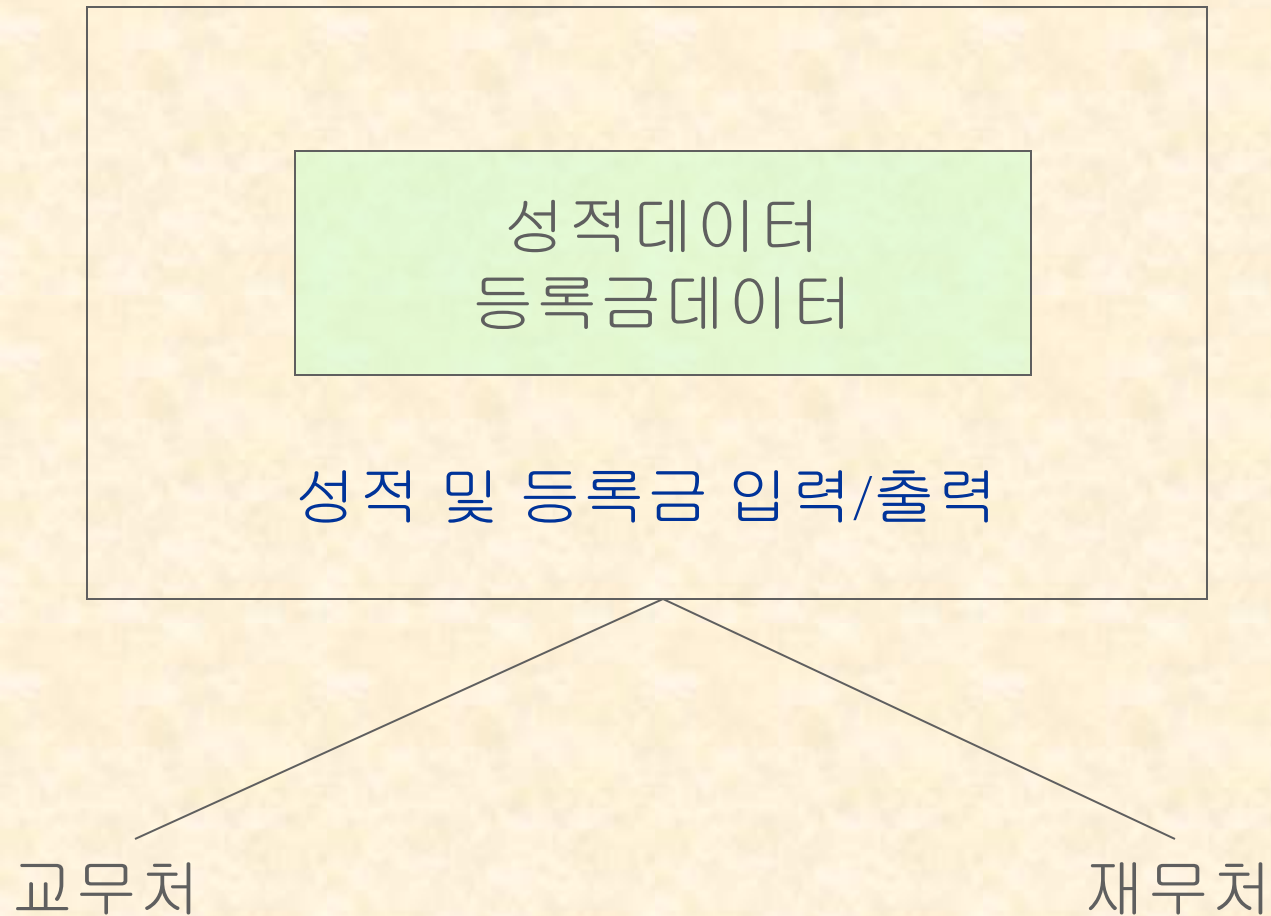
📌 File processing <=====> DB approach
(A) 교무처 (B) 재무처



=> 많은 동일한 데이터를 별도로 관리함
저장 공간 및 관리 노력 : 비효율적



DB System Approach





데이터베이스 방법의 특징

- ✚ Self-describing nature
 - program-data independence
 - program-operation independence
(Object-oriented DBMS)
- ✚ Insulation between Programs and Data, and Data Abstraction
- ✚ Multiple View
- ✚ 데이터 공유 및 다사용자 트랜잭션 처리



Self-describing nature

- definition of DB + DB
- catalog(meta-data) : 파일들의 구조, 데이터의 형, 제약조건 (성적 : E, G, H,... 없음)
- DBMS와 사용자 참조
 - file processing : 데이터의 정의를 항상 응용프로그램에서 정의함 (application-dependent)
 - 특정한 응용프로그램에 종속적, 프로그램간 공유 불가능



Insulation between Programs and Data, and Data Abstraction

- ✚ program-data independence
 - file processing : 데이터의 구조가 프로그램에 종속적으로 정의됨.
 - 데이터의 구조 변경 => 프로그램의 변경
 - DB approach : 데이터 구조가 카탈로그에 저장 되어 독립적으로 관리됨.
- ✚ program-operation independence
 - file processing : impossible
 - DB system : object-oriented DBMS
 - operation : interface + implementation
(hide implementation details)



Insulation between Programs and Data, and Data Abstraction

📌 Data Abstraction

- Abstracted data type + class Hierarchy
- Data structure & operations
- Abstraction operation
 - calculate_GPA(성적의 평균값 계산) --> student object : user do not need to know how



Support of Multiple Views of the Data

- 📌 Multiple Views
 - view : 데이터베이스의 부분 또는 데이터베이스에서 유도된 가상 데이터
 - 각 사용자가 사용하는 데이터만을 가공하여 보여줌
- 📌 Sharing of Data and Multi-user Transaction Processing
 - Concurrency control: control multiple simultaneous updates on the same data
 - 비행기 예약, 호텔 예약

(a)

TRANSCRIPT	StudentName	Student Transcript				
		CourseNumber	Grade	Semester	Year	SectionId
	Smith	CS1310	C	Fall	99	119
		MATH2410	B	Fall	99	112
	Brown	MATH2410	A	Fall	98	85
		CS1310	A	Fall	98	92
		CS3320	B	Spring	99	102
		CS3380	A	Fall	99	135

(b)

PREREQUISITES	CourseName	CourseNumber	Prerequisites
	Database	CS3380	CS3320
			MATH2410
	Data Structures	CS3320	CS1310

Figure 1.4 Two views derived from the example database shown in Figure 1.2. (a) The student transcript view. (b) The course prerequisite view.



데이터베이스 사용자군

- 📌 DBA : DataBase Administrator
 - 데이터베이스 사용 허가 => 보안 유지(security)
 - 데이터베이스 사용 현황 조절 및 DB 성능 감시
=> 성능 관리
 - S/W & H/W 관리

- 📌 DB Designer
 - 데이터 정의 및 저장구조 설계
=> application domain expert



데이터베이스 사용자군

- End user : 데이터베이스를 사용하는 사람
 - casual end user : DB 사용 빈도수가 적지만 다양하고 복잡한 정보를 원하는 사용자 (중급/고급 관리자)
 - naive, parametric end user : 정형화된 질의/갱신 작업을 계속적으로 수행하는 사용자 [은행, 예약]
 - canned transaction : 정형화된 질의 갱신 작업
 - sophisticated end user : 데이터에 복잡한 작업을 수행하는 사용자 (엔지니어, 과학자, 사업 분석자) [증권]
 - stand-alone user : 개인 데이터베이스 사용자
- 시스템 분석가 및 응용프로그램 개발자
 - 시스템 분석가 : 사용자의 요구사항을 분석
=> canned transaction 설계
 - 응용프로그램 개발자 : canned transaction 프로그래밍, 디버깅, 시험



데이터베이스 관리시스템의 사용 목적

❗ Redundancy

- file processing : 모든 화일 을 독립적으로 보관
=> 동일한 데이터가 여러 데이터 화일에 저장됨
 - 저장공간 낭비, 복사 및 관리 노력
 - 동일 데이터의 비일치 (두 파일에 저장된 데이터 중 한 개만 갱신할 때 데이터 값이 일치하지 않음)
- DB System : 데이터 저장 공간 공유
 - 저장공간절약, 데이터일치, 복사 및 관리노력의 감소

❗ Security : 데이터의 접근 허가 및 형태 관리 (read/write/privileged)



데이터베이스 관리시스템의 사용 목적

- ✚ Multiple User Interface :
 - 다양한 사용자 유형 및 숙련도 지원
 - casual user : query language
 - application programmer : programming language interface
 - naive user : form
 - stand-alone user : menu-driven, natural language
- ✚ Represent complex relationship among data



데이터베이스 관리시스템의 사용 목적

- ✚ Enforcing integrity constraint \leq semantics of DB
 - 예제 : 학년: 1 -4, 과목 번호: unique, 모든 분반 \leftrightarrow 과목
 - DBMS에 의한 자동 점검 또는 데이터의 갱신 시 점검
(한계 : 성적 $A \twoheadrightarrow X$ (거부), $A \twoheadrightarrow C$ (허용))
- ✚ Backup & Recovery
 - H/W 또는 S/W 고장 시 DBMS가 올바른 데이터로 유지
- ✚ 프로그램과 데이터 구조의 영구적 보관
 - 파일처리
 - 프로그램이 파일과 메모리 사이의 데이터를 변환시킴
 - 프로그램 실행 후 : 데이터는 파일에 저장되지만 변환필요,
➔ DBMS에 의한 자동 변환.



데이터베이스 시스템 사용의 장점

- 표준화
 - 명칭, 데이터 포맷, 출력 포맷 등
 - 파일처리 : 각 파일의 총체적인 관리 불가능
- 응용프로그램 개발 시간의 단축
 - DBMS에서 데이터베이스가 구축된 후 응용프로그램 개발 시간은 1/6-1/4로 감소,
- 유동성
 - 새로운 변화에 대처 용이
 - 기존의 데이터/프로그램에 영향 없음
- 최신 데이터 이용 가능
 - 다사용자 환경하에서의 concurrency control (은행, 예약)
- 경제성 : 중복투자, 데이터 관리 인력 감소



DBMS 사용의 범위

- DBMS 사용 : overhead
 - 최초 투자 규모 (H/W, S/W, 교육)
 - DBMS가 지원하는 데이터형의 일반성 :
=> 특정 목적과 불일치
 - overhead for security, concurrency control, recovery, integrity constraint checking
- When not to use DBMS
 - simple, well defined, not expected to change
 - real-time requirement : overhead
 - no multiple user access