HOA SEN UNIVERSITY

Lecture 1
Database System Concepts

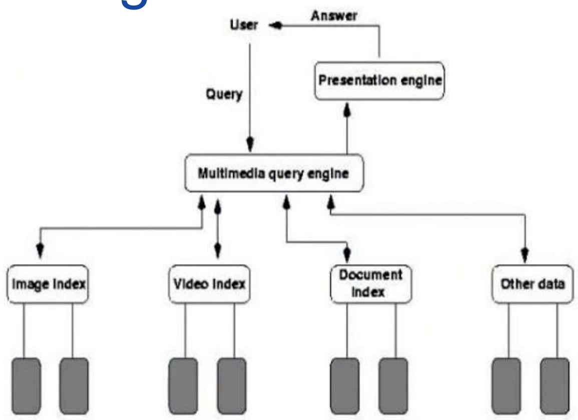
Objectives

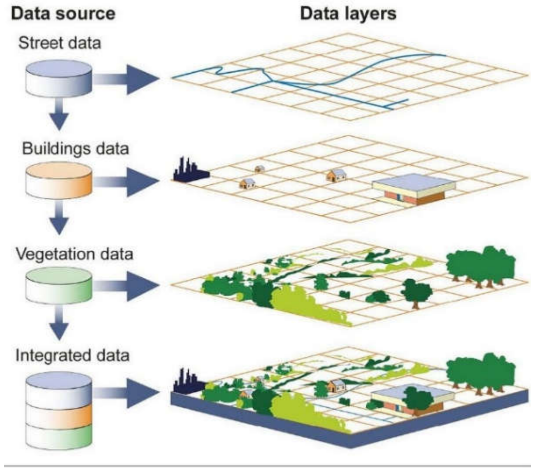
- Types of Databases and Database Applications
- Basic Definitions
- Typical DBMS Functionality
- Main Characteristics of the Database Approach
- Database and Database Users

- Ref.: Chapter 1

Types of Databases and Database Applications

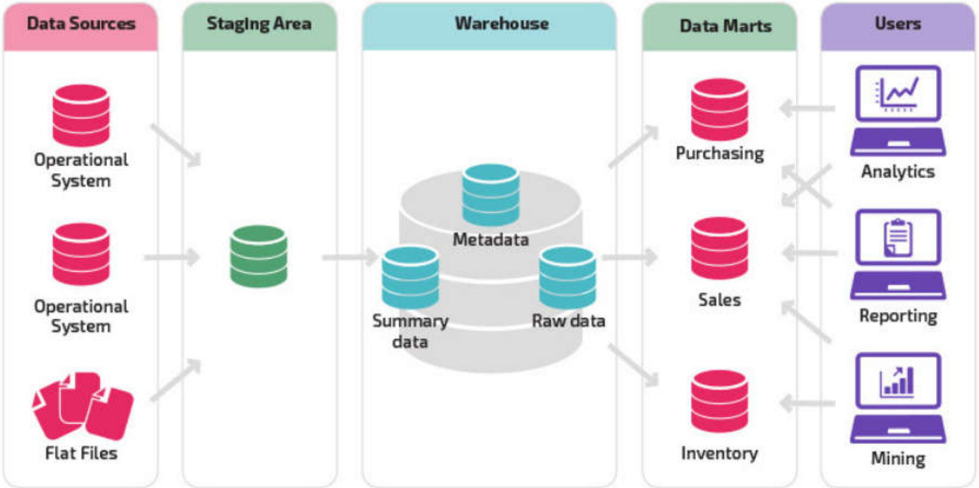
- Traditional Applications:
  - Store textual or numeric information
- More Recent Applications:
  - Multimedia Databases
    - Store images, audio clips, and video streams digitally
  - Geographic Information Systems (GIS)
    - Store and analyze maps, weather data, and satellite images





Types of Databases and Database Applications (2)

- More Recent Applications (2):
  - Data Warehouses and online analytical processing (OLAP) systems
    - Extract and analyze useful business information from very large databases
    - Support decision making
  - Real-time and Active Databases technology
    - Control industrial and manufacturing processes
  - Many other applications



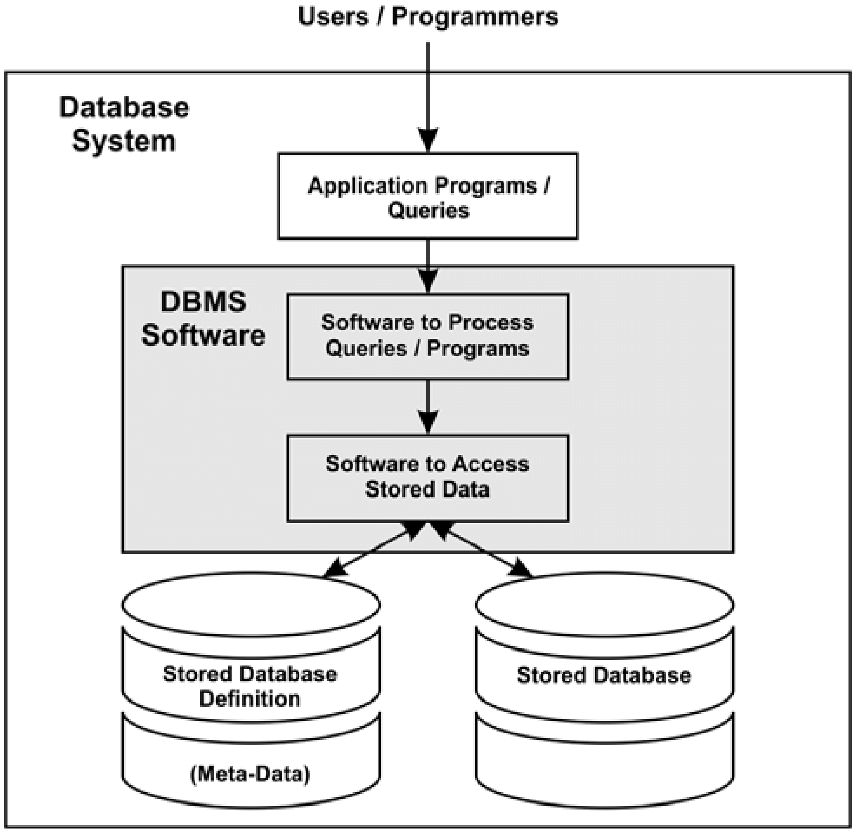
Introduction

- **Database:**
  - A collection of related data.
- **Data:**
  - Known facts that can be recorded and have an implicit meaning.
- **Mini-world:**
  - Some part of the real world about which data is stored in a database.
    - For example: student grades and transcripts at a university.

- Example of a large commercial database
  - Amazon.com

Introduction (2)

- **Database Management System (DBMS):**
  - A software package/ system to facilitate the creation and maintenance of a computerized database.
- **Database System:**
  - The DBMS software together with the data itself. Sometimes, the applications are also included.


Users / Programmers
Database System
Application Programs / Queries
DBMS Software
Software to Process Queries / Programs
Software to Access Stored Data
Stored Database Definition
(Meta-Data)
Stored Database

Introduction (3)

- Meta-data
  - Database definition or descriptive information
  - Stored by the DBMS in the form of a database catalog or dictionary

- Manipulating a database
  - Query and update the database miniworld
  - Generate reports

Introduction (4)

- Sharing a database
  - Allow multiple users and programs to access the database simultaneously
- Application program
  - Accesses database by sending queries to DBMS
- Query
  - Causes some data to be retrieved

Introduction (5)

- Transaction
  - May cause some data to be read and some data to be written into the database
- Protection includes:
  - System protection
  - Security protection
- Maintain the database system
  - Allow the system to evolve as requirements change over time

ĐẠI HỌC HOA SEN

Simplified database system environment

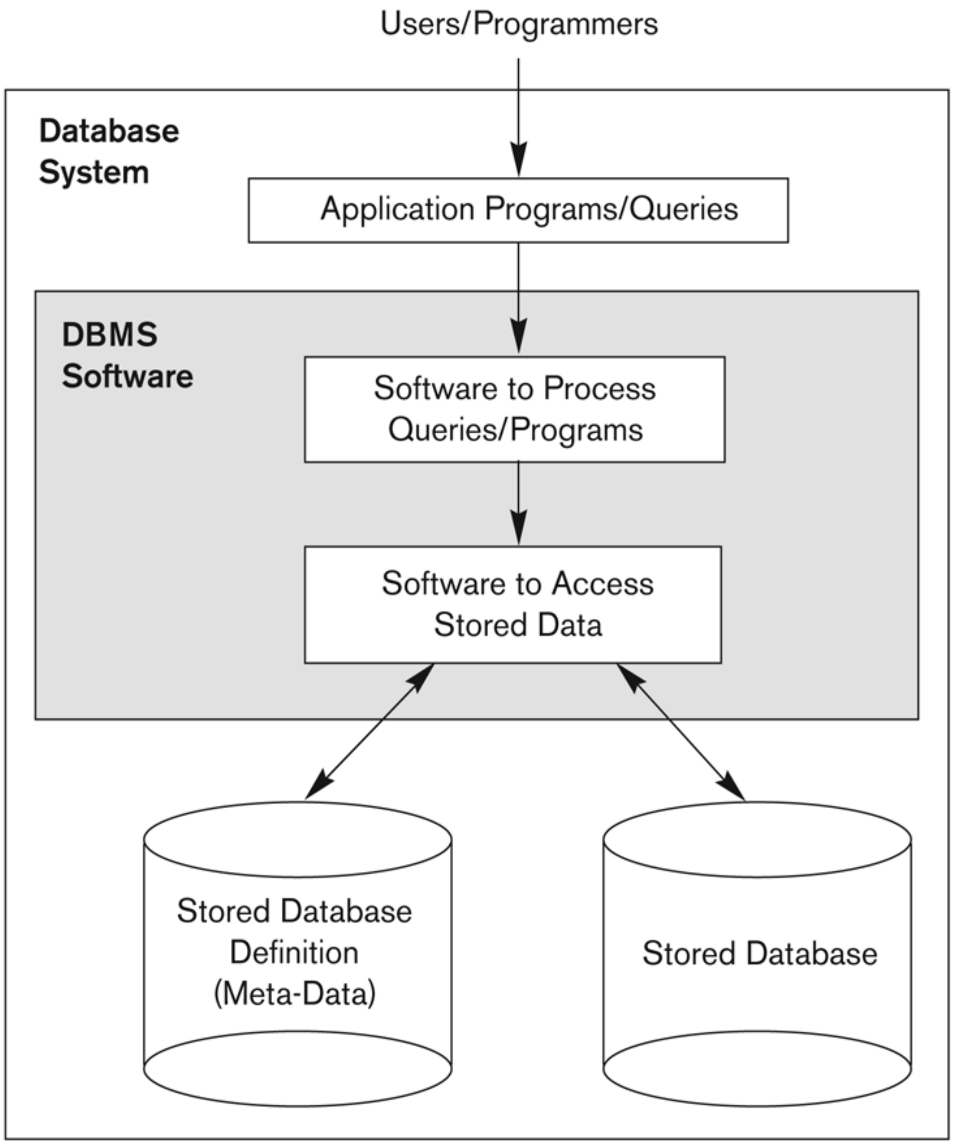
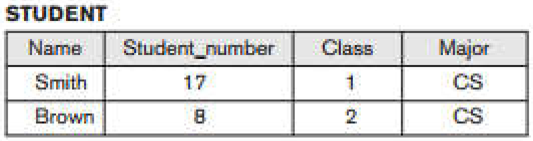

Users/Programmers
Database System
Application Programs/Queries
DBMS Software
Software to Process Queries/Programs
Software to Access Stored Data
Stored Database Definition (Meta-Data)
Stored Database

Figure 1.1
A simplified database system environment

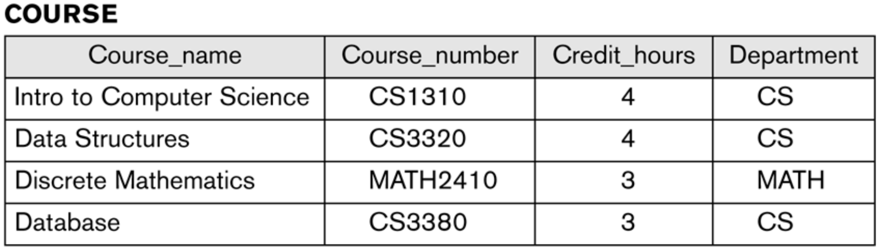
Example of a Database (with a Conceptual Data Model)

- **Mini-world for the example:**
  - Part of a UNIVERSITY environment.
- **Some mini-world *entities*:**
  - STUDENTs
  - COURSEs
  - PREREQUISITEs
  - SECTIONs (of COURSEs)
  - GRADE_REPORTSs



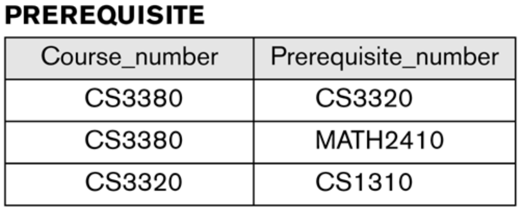
STUDENT

Name	Student_number	Class	Major
Smith	17	1	CS
Brown	8	2	CS



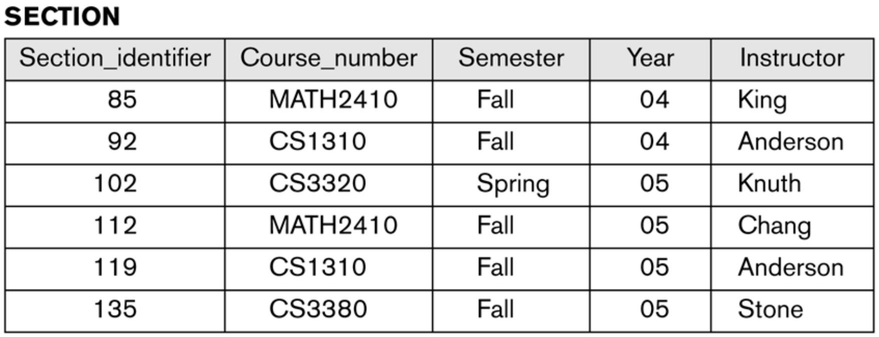
COURSE

Course_name	Course_number	Credit_hours	Department
Intro to Computer Science	CS1310	4	CS
Data Structures	CS3320	4	CS
Discrete Mathematics	MATH2410	3	MATH
Database	CS3380	3	CS



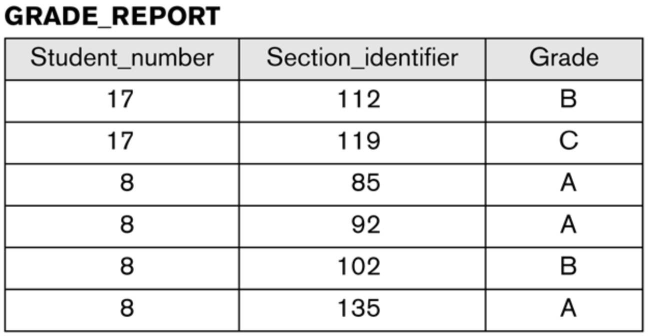
PREREQUISITE

Course_number	Prerequisite_number
CS3380	CS3320
CS3380	MATH2410
CS3320	CS1310



SECTION

Section_identifier	Course_number	Semester	Year	Instructor
85	MATH2410	Fall	04	King
92	CS1310	Fall	04	Anderson
102	CS3320	Spring	05	Knuth
112	MATH2410	Fall	05	Chang
119	CS1310	Fall	05	Anderson
135	CS3380	Fall	05	Stone



GRADE_REPORT

Student_number	Section_identifier	Grade
17	112	B
17	119	C
8	85	A
8	92	A
8	102	B
8	135	A

Example

- Examples of queries:
  - Retrieve the transcript
  - List the names of students who took the section of the ‘Database’ course offered in fall 2008 and their grades in that section
  - List the prerequisites of the ‘Database’ course

Example

- Examples of updates:
  - Change the class of ‘Smith’ to sophomore (2nd-year)
  - Create a new section for the ‘Database’ course for this semester
- Enter a grade of ‘A’ for ‘Smith’ in the ‘Database’ section of last semester

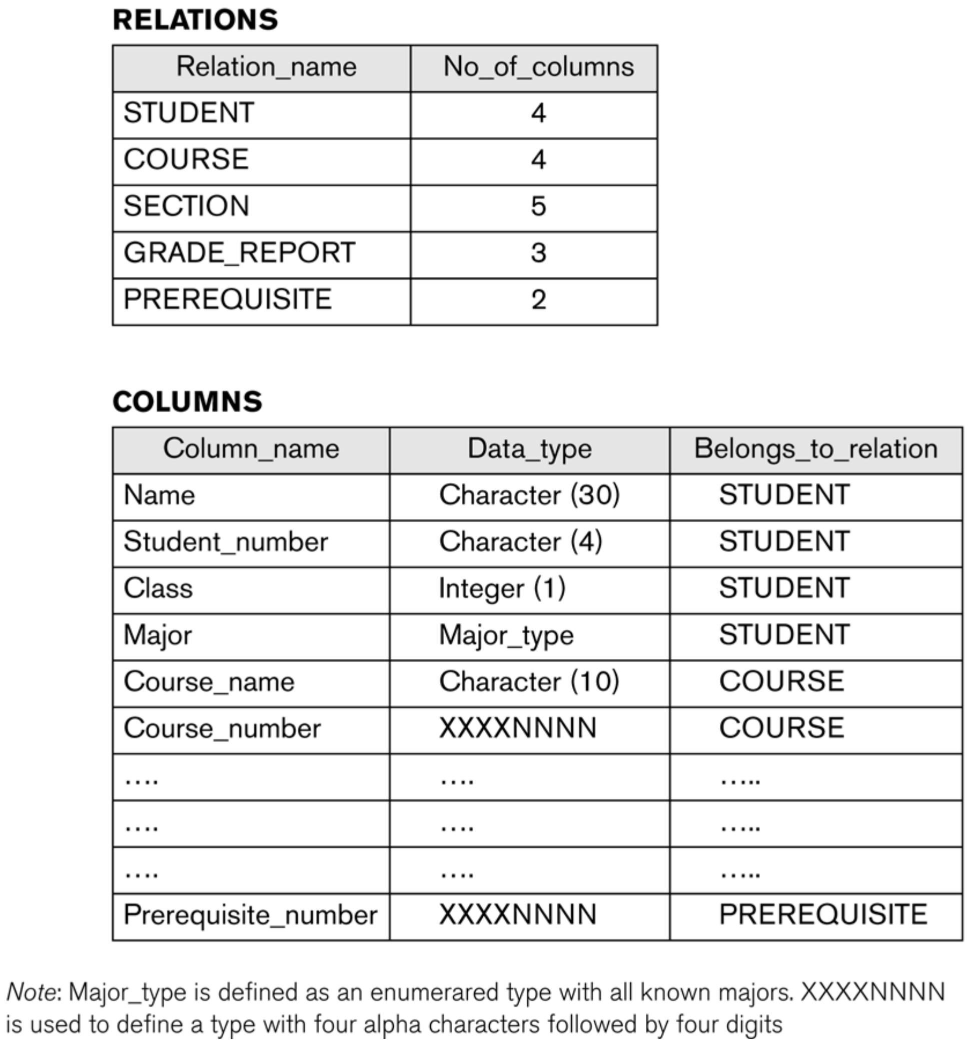
Main Characteristics of the Database Approach

- **Self-describing nature of a database system:**
  - A DBMS **catalog** stores the description of a particular database (e.g. data structures, types, and constraints)
  - The description is called **meta-data**.
  - This allows the DBMS software to work with different database applications.
- **Insulation between programs and data:**
  - Called **program-data independence**.
  - Allows changing data structures and storage organization without having to change the DBMS access programs.

Main Characteristics of the Database Approach (2)

- **Data Abstraction:**
  - A **data model** is used to hide storage details and present the users with a conceptual view of the database.
  - Programs refer to the data model constructs rather than data storage details → program-data independence and program-operation independence
- **Conceptual** representation of data
  - Does not include details of how data is stored or how operations are implemented
- **Data model**
  - Type of data abstraction used to provide conceptual representation

Example of a simplified database catalog



RELATIONS

Relation_name	No_of_columns
STUDENT	4
COURSE	4
SECTION	5
GRADE_REPORT	3
PREREQUISITE	2

COLUMNS

Column_name	Data_type	Belongs_to_relation
Name	Character (30)	STUDENT
Student_number	Character (4)	STUDENT
Class	Integer (1)	STUDENT
Major	Major_type	STUDENT
Course_name	Character (10)	COURSE
Course_number	XXXXNNNN	COURSE
….	….	…..
….	….	…..
….	….	…..
Prerequisite_number	XXXXNNNN	PREREQUISITE

Note: Major_type is defined as an enumerared type with all known majors. XXXXNNNN is used to define a type with four alpha characters followed by four digits

Figure 1.3
An example of a database catalog for the database in Figure 1.2.

Main Characteristics of the Database Approach (3)

- **Sharing of data and multi-user transaction processing:**
  - Allowing a set of **concurrent users** to retrieve from and to update the database.
  - *Concurrency control* within the DBMS guarantees that each **transaction** is correctly executed or aborted
  - *Recovery* subsystem ensures each completed transaction has its effect permanently recorded in the database
  - **OLTP** (Online Transaction Processing) is a major part of database applications. This allows hundreds of concurrent transactions to execute per second.

Main Characteristics of the Database Approach (4)

- **Support of multiple views of the data:**
  - Each user may see a different view of the database, which describes **only** the data of interest to that user.
  - View
    - Subset of the database
    - Contains virtual data derived from the database files but is not explicitly stored
- **Multiuser DBMS**
  - Users have a variety of distinct applications
  - Must provide facilities for defining multiple views

Database Users

- Users may be divided into
  - Those who actually **use and control** the database content, and those who design, develop and maintain database applications (called "Actors on the Scene"), and
  - Those who **design and develop** the DBMS software and related tools, and the computer systems operators (called "Workers Behind the Scene").

Actors on the scene

- **Actors on the scene**
  - Database administrators (DBA)
  - Database Designers
  - End-users:
    - Casual
    - Naïve or Parametric
    - Sophisticated
    - Stand-alone
  - System Analysts and Application Programmers (Software Engineers)

Workers behind the Scene

- **Workers behind the Scene**
  - DBMS System Designers and Implementers
    - Design and implement the DBMS modules and interfaces as a software package
    - Tool Developers
    - Operators and Maintenance Personnel (system administration personnel)

Advantages of Using the Database Approach

- Controlling redundancy
  - Data normalization
  - Denormalization
    - Sometimes necessary to use controlled redundancy to improve the performance of queries
- Restricting unauthorized access
  - Security and authorization subsystem
  - Privileged software

ĐẠI HỌC HOA SEN

Advantages of Using the Database Approach (2)

- Providing backup and recovery services.
- Providing multiple interfaces to different classes of users.
- Representing complex relationships among data.
- Enforcing integrity constraints on the database.
- Drawing inferences and actions from the stored data using deductive and active rules

Advantages of Using the Database Approach (3)

- Providing persistent storage for program objects
  - Complex object in C++ can be stored permanently in an object-oriented DBMS
  - Impedance mismatch problem
    - Object-oriented database systems typically offer data structure compatibility

Advantages of Using the Database Approach (4)

- Providing storage structures and search techniques for efficient query processing
  - Indexes
  - Buffering and caching
  - Query processing and optimization

Advantages of Using the Database Approach (5)

- Providing backup and recovery
  - Backup and recovery subsystem of the DBMS is responsible for recovery
- Providing multiple user interfaces
  - Graphical user interfaces (GUIs)
- Representing complex relationships among data
  - May include numerous varieties of data that are interrelated in many ways

Advantages of Using the Database Approach (6)

- Enforcing integrity constraints
  - Referential integrity constraint
    - Every section record must be related to a course record
  - Key or uniqueness constraint
    - Every course record must have a unique value for Course_number
  - Business rules
  - Inherent rules of the data model

Advantages of Using the Database Approach (7)

- Permitting inferencing and actions using rules
  - Deductive database systems
    - Provide capabilities for defining deduction rules
    - Inferencing new information from the stored database facts
  - Trigger
    - Rule activated by updates to the table
  - Stored procedures
    - More involved procedures to enforce rules

Advantages of Using the Database Approach (8)

- Additional implications of using the database approach
  - Reduced application development time
  - Flexibility
  - Availability of up-to-date information
  - Economies of scale

Extending Database Capabilities

- New functionality is being added to DBMSs in the following areas:
  - Scientific Applications
  - XML (eXtensible Markup Language)
  - Image Storage and Management
  - Audio and Video Data Management
  - Data Warehousing and Data Mining
  - Spatial Data Management
  - Time Series and Historical Data Management
- The above gives rise to *new research and development* in incorporating new data types, complex data structures, new operations and storage and indexing schemes in database systems.

A Brief History of Database Applications

- Early database applications using hierarchical and network systems
  - Large numbers of records of similar structure
- Providing data abstraction and application flexibility with relational databases
  - Separates physical storage of data from its conceptual representation
  - Provides a mathematical foundation for data representation and querying

A Brief History of Database Applications (2)

- Object-oriented applications and the need for more complex databases
  - Used in specialized applications: engineering design, multimedia publishing, and manufacturing systems
- Interchanging data on the Web for e-commerce using XML
  - Extended markup language (XML) primary standard for interchanging data among various types of databases and Web pages

A Brief History of Database Applications (3)

- Extending database capabilities for new applications
  - Extensions to better support specialized requirements for applications
  - Enterprise resource planning (ERP)
  - Customer relationship management (CRM)
  - Supply Chain Management (SCM)
- Databases versus information retrieval
  - Information retrieval (IR)
    - Deals with books, manuscripts, and various forms of library-based articles

When not to use a DBMS

- Main inhibitors (costs) of using a DBMS:
  - High initial investment and possible need for additional hardware.
  - Overhead for providing generality, security, concurrency control, recovery, and integrity functions.
- When a DBMS may be unnecessary:
  - If the database and applications are simple, well defined, and not expected to change.
  - If there are stringent real-time requirements that may not be met because of DBMS overhead.
  - If access to data by multiple users is not required.
- When no DBMS may suffice:
  - If the database system is not able to handle the complexity of data because of modeling limitations
  - If the database users need special operations not supported by the DBMS.

ĐẠI HỌC
HOA SEN

Q & A