Introduction to Database Management System (DBMS)

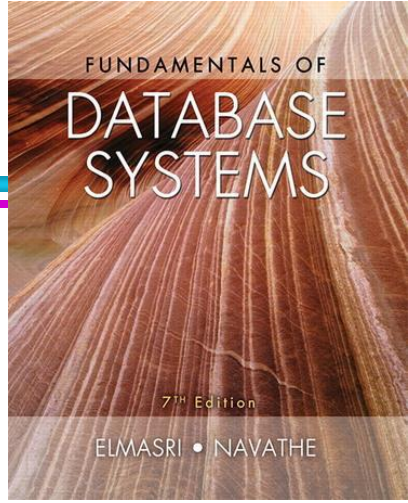
Dr. Enosha Hettiarachchi

eno@ucsc.cmb.ac.lk

Learning Outcomes:

- **ILO1:** Demonstrate a solid understanding of database management systems (DBMS), including their structure, functions, and the differences between DBMS and traditional data storage methods.
- **ILO2:** Develop proficiency in data modeling techniques, including identifying entities, relationships, and different types of relationships such as binary, unary, ternary, and special relationships.
- **ILO3:** Apply principles of entity-relationship diagrams (ERDs) and relational model to design efficient and effective databases, considering conceptual, logical, and physical design aspects.
- **ILO4:** Apply normalization techniques up to 4th normal form to ensure data integrity and minimize redundancy in database design.
- **ILO5:** Utilize relational algebra and calculus to perform operations such as selection, projection, joins, division, set operations, and renaming on relational data.
- **ILO6:** Demonstrate proficiency in SQL, including writing and executing basic to complex SQL statements, understanding aggregation operators, nested queries, handling null values, and implementing views, relations, stored procedures, and triggers.
- **ILO7:** Apply critical thinking and problem-solving skills to real-world scenarios involving database management, including identifying and resolving issues related to database design, and querying.

References



- Elmasri, R. and Navathe, S.B., 2015. *Fundamentals of database systems*. 7th Edition, Addison-Wesley.
- Ramakrishnan, R. and Gehrke, J., 2003. *Database management systems* (Vol. 3). New York: McGraw-Hill.
- DeBarros, A., 2022. Practical SQL: A Beginner's Guide to Storytelling with Data. No Starch Press.

Rubric:

- 60% Final Examination
- 40% Assignments

Structure

- Why use a Database?
- Components of Database System Environment
- File based Systems and Limitations
- Introduction to DB approach
- Data Hierarchy
- Introduction to meta data
- DB applications
- DB approach advantages & disadvantages

Why use a Database?

- Many people collect things
  - How about you?
- If you collect any thing, you probably are familiar with some of the problems of managing a collection
  - e.g. stamps, photos, paper cuttings
- One way to keep track of a collection is to create a database

Why Database Technology?

The need to manipulate large collection of data for frequent used data queries and reports.

E.g. Collection of information on library books
Queries:

- **List of books written by a particular author**
- **List of books about a particular subject**
- **Borrowing a book**
- **Reserving a book for borrowing**

Examples of Database Applications

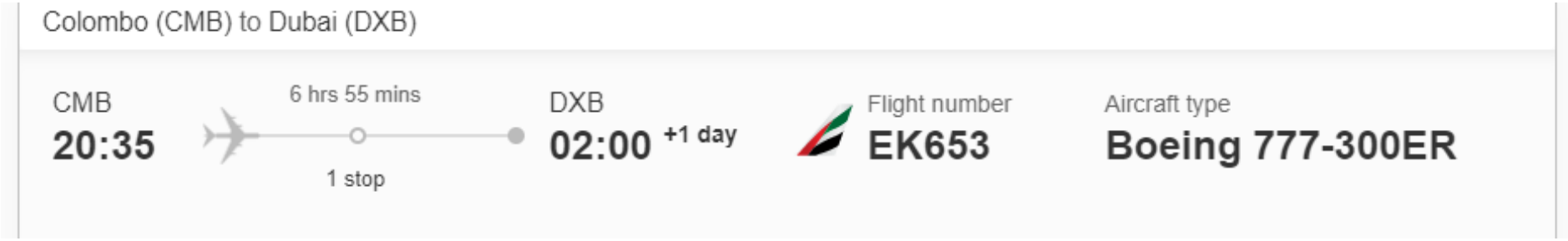
- Purchases from the supermarket
- Purchases using your credit card
- Booking a holiday at the travel agents (Air Line Reservation)
- Using the Internet
- Studying at university
- Hospital admissions
- Borrowing Books from library

Library

- Membership
- Reference
- Borrow
- Return
- Order

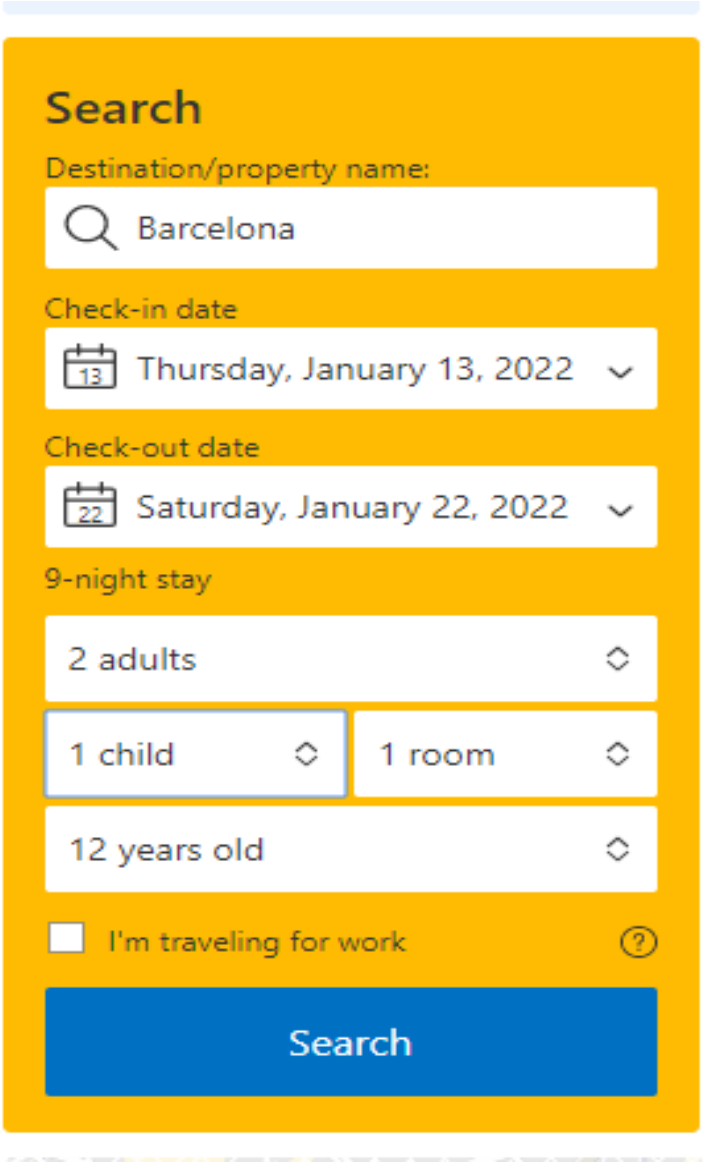
Air Line Reservations

Availability of Seats on a flight, ticket booking, issuing, reconfirmation



Hotel Reservations

Availability of Rooms, reservation



Hospital System

Doctor’s information, speciality

Ward information, no. of beds

Theatre information, facilities

Patient information, admittance information

Consultation information, booking

Listing

Read each record and print. In time sequence.

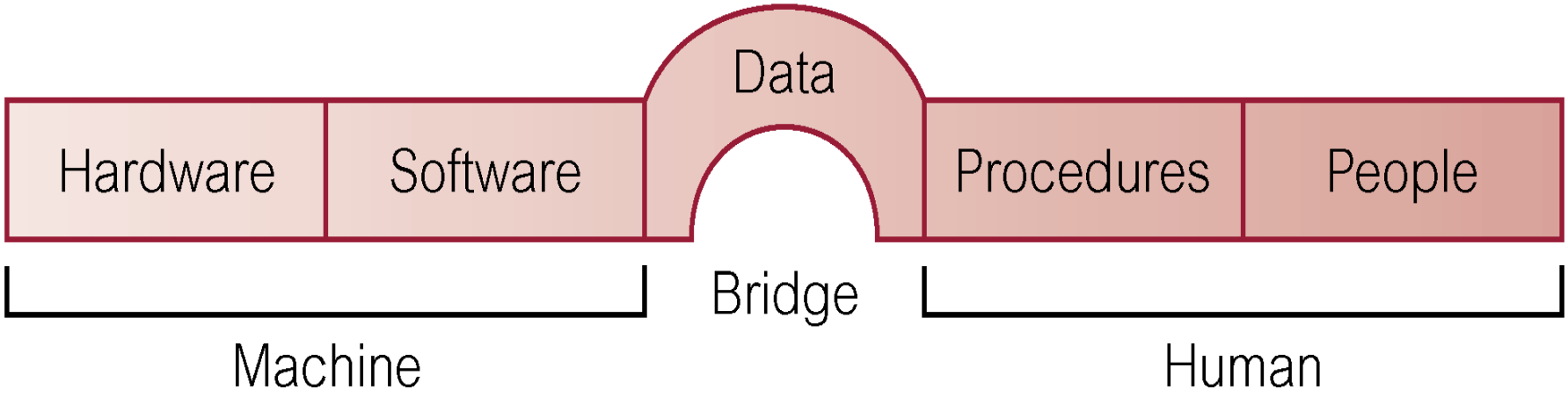
CHANNELLING DONE

Channel Details

Reference Number	: 0046764721
Appointment Number	: 9
Appointment Date	:
Time	: 09:40
Total Fee	: 3875.00 LKR
Breakdown of Fee	: Doctor Fee: 3000.00
	Hospital Fee: 600.00 Doctor/Hospital VAT: 0.00
	Booking Fee: ((Echannelling 275.00) Including VAT + NBT)
	("No-Show" option fee = 0.00)

Patient Details

Components of Database System Environment



Components of Database System Environment

- **Hardware**

  Set of physical devices on which a database resides. Can range from a PC to a network of computers.

- **Software**
  - Database Management System (DBMS)
  - Operating System
  - Application Programs
  - User Interface

Components of Database System Environment - Data

- **Data**

  – A representation of facts, concepts or instructions in a formalised manner suitable for communication, interpretation or processing by human beings or by automatic means.

  – Text, colours, symbols, shapes, graphics, images, temperatures, sound, video or other facts and figures are data suitable for processing.

Components of Database System Environment - Data

E.g. Person or Employee or Customer

name, address, phone, date of birth, designation, department, salary,

employee no, photograph

Information

- Knowledge derived from data
- Processed or organised or summarised data

Eg:-

- Process Date of Birth ->Age
- Process Salary (all) ->Highest paid employee
- Process all -> No of employees
- Process all -> Employees working for Sales division

Components of Database System Environment

- **Procedures**

  Instructions and rules that should be applied to the design and use of the database.

- **People**

  Two different types of people (end-users and practitioners) are concerned with the database.

  End-Users

  – are the ‘clients’ for the database, who need information from the database to carry out their duties.

  e.g. Executives, managers, staff, clerical personnel

Components of Database System Environment - People

Practitioners

- **People responsible for the database system and its associated application software.**

  e.g. Data and Database administrators, Database designers, Application developers.

People - Job Definitions

- ***Data Administration***: A high-level function that is responsible for the overall management of data resources in an organization, including maintaining corporate-wide definitions and standards.

- ***Database Administration***: A technical function that is responsible for physical database design and for dealing with technical issues such as security enforcement, database performance, and backup and recovery.

Database Administration Functions

- Often, some mixture of these duties
  - Selection of hardware and software
  - Installing/upgrading DBMS
  - Tuning database performance
  - Improving query processing performance
  - Managing data security, privacy, and integrity
  - Data backup and recovery

Manual Systems – Information on library books

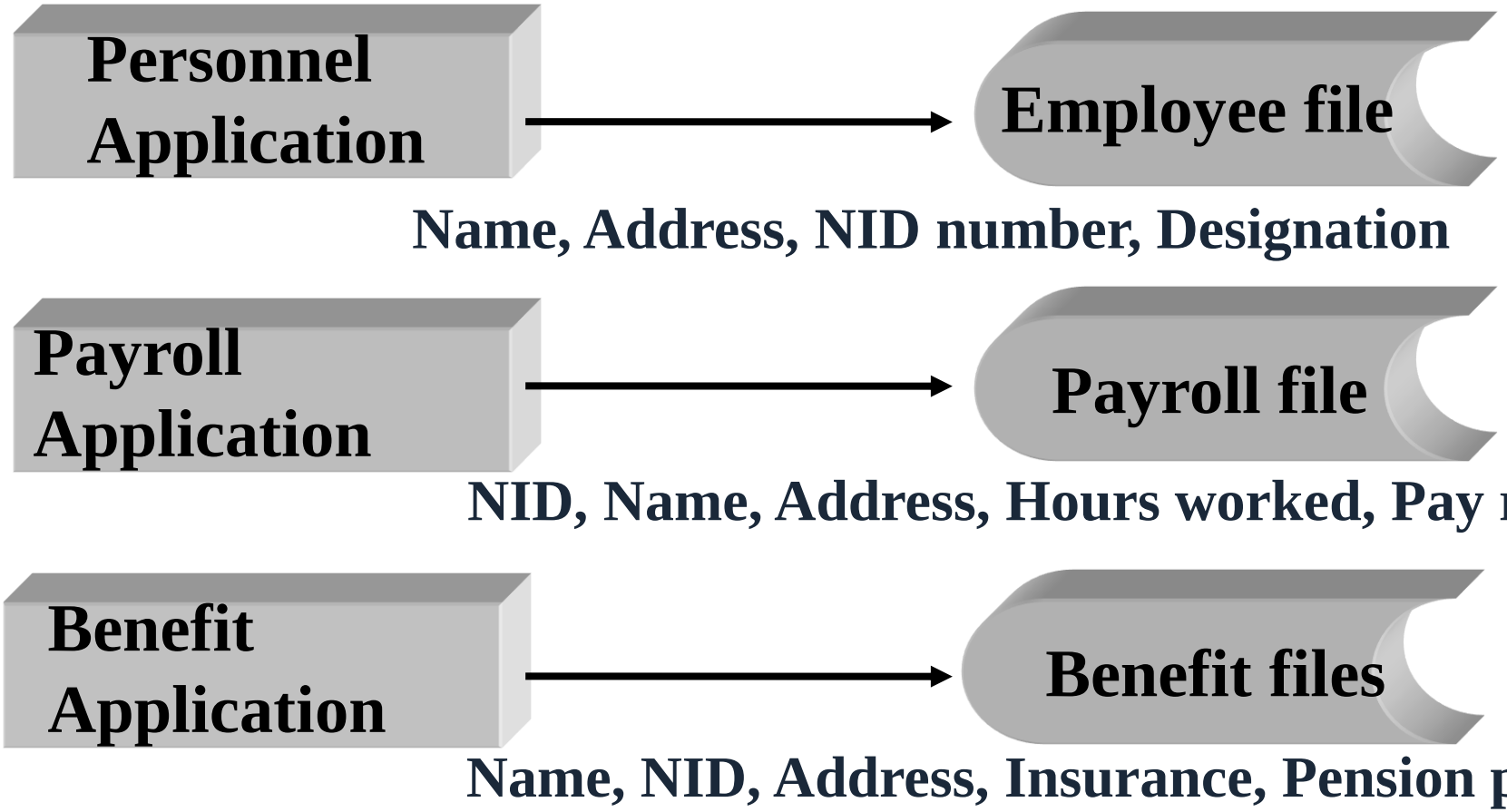
- Before and during most of the last century, libraries used card catalogues stored in drawers of special cabinets
  - cards with typed book information

    E.g. the title index has one card for every book in the library

File-Based Systems

- Collection of application programs that perform services for the end users (e.g. reports).
- Each program defines and manages its own data.

Data Redundancy



Limitations of File-Based Approach

- **Separation and isolation of data**
  - **Each program maintains its own set of data.**
  - **Users of one program may be unaware of potentially useful data held by other programs.**
- **Duplication of data**
  - **Same data is held by different programs.**
  - **Wasted space and potentially different values and/or different formats for the same item.**

Limitations of File-Based Approach

- **Data dependence**
  - **File structure is defined in the program code.**
- **Incompatible file formats**
  - **Programs are written in different languages, and so cannot easily access each other's files.**
- **Fixed Queries/Proliferation of application programs**
  - **Programs are written to satisfy particular functions.**
  - **Any new requirement needs a new program.**

Program-Data/Data Dependence

program Customer-Entry

....

```
type customer = record
        customer-id: string;
        customer-name: string;
        customer-street: string;
        customer-city: string;
    end;
```

....

program Customer-Orders

....

```
type customer = record
        customer-id: string;
        customer-name: string;
        customer-street: string;
        customer-city: string;
    end;
```

....

Database Approach

- **Arose because:**
  - **Definition of data was embedded in application programs, rather than being stored separately and independently.**
  - **No control over access and manipulation of data beyond that imposed by application programs.**

- **Result:**
  - **the database and Database Management System (DBMS).**

Database

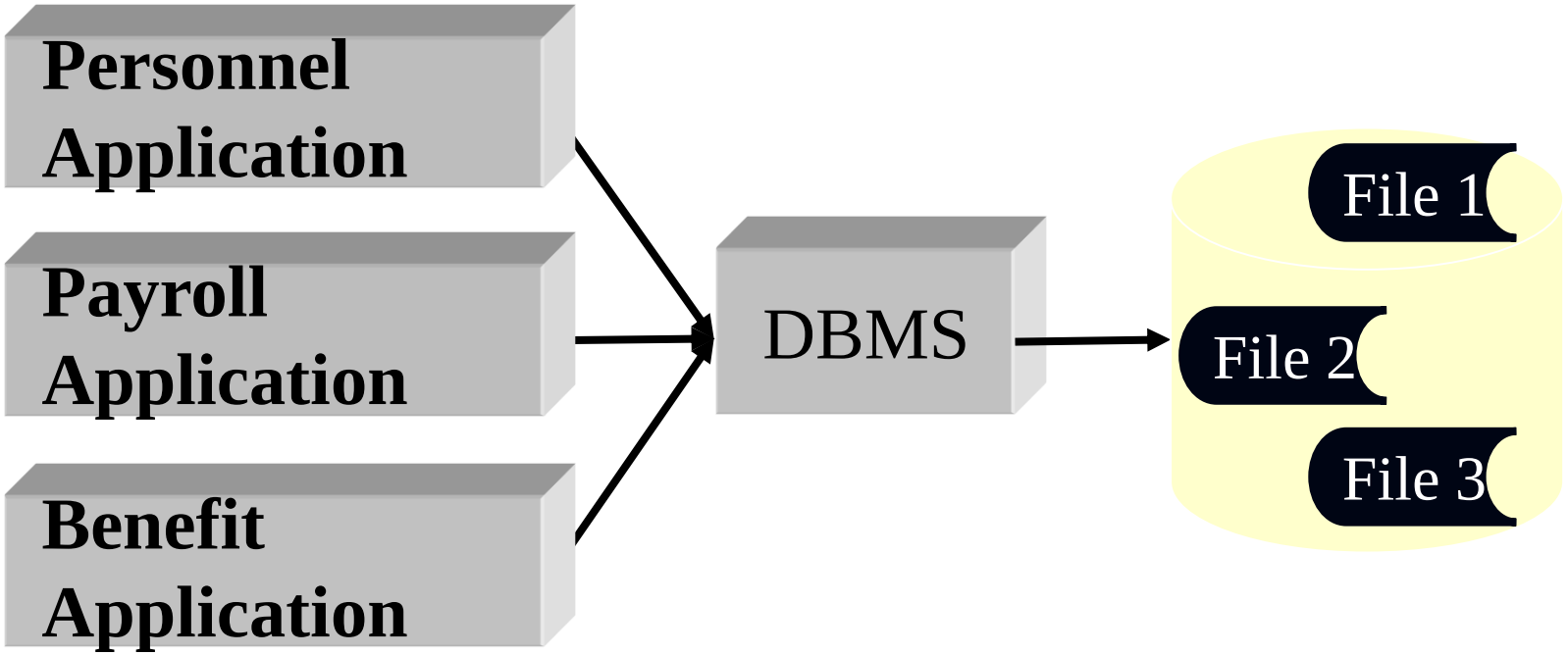
- **Shared collection of logically related data (and a description of this data), designed to meet the information needs of an organization.**
- **System catalog or data dictionary provides description of data (metadata) to enable program–data independence.**
- **Logically related data comprises entities, attributes, and relationships of an organization's information.**

Database Management System (DBMS)

- **A software system that enables users to define, create, and maintain the database and that provides *controlled access* to this database.**

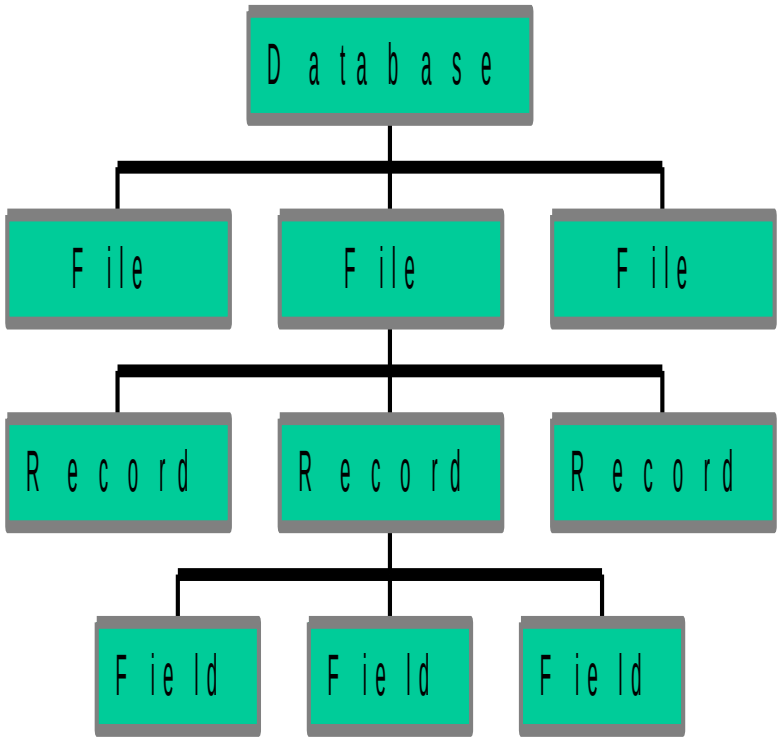
  E.g. MySQL, Oracle, Access, SQL Microsoft Server, PostgreSQL, IBM DB2, etc

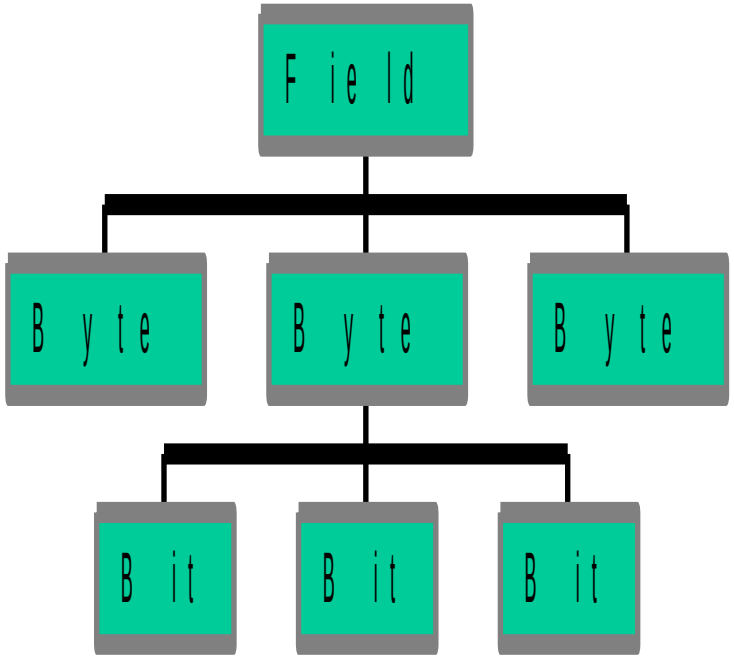
Database Approach



- e.g. Integrated human resources database
  - Personnel: *Name, Address, NID number, Designation*
  - Payroll: *Hours worked, Pay rate*
  - Benefit: *Insurance, Pension plan*

Data Hierarchy





Data Hierarchy

Employee (Empno, Name, Designation, Salary, Depart)

1	*De Silva*	*Manager*	*50000*	*Personnel*
2	*Perera*	*Secretary*	*15000*	*Personnel*
3	*Dias*	*Salesman*	*25000*	*Sales*

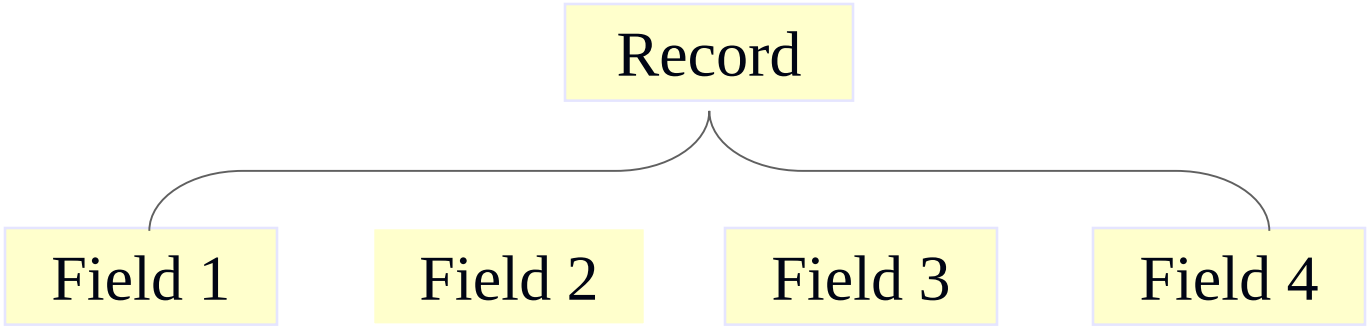
Department (Depart, Manager, Dept Addr, Dept Phone)

Personnel	*De Silva*	*Colombo*	*589123*
Sales	*Alwis*	*Kandy*	*987275*
….	….	….	….

Data Hierarchy

(Empno,name, designation, salary,department)

2 Perera Manager 35,000 Personnel



Byte

•A single character (letter, number, symbol) is represented using a group of bits, E.g. 10101010 letter J in ASCII

Bit

•The smallest unit of data, E.g. 0 or 1

Data Dictionary/System Catalog

- **A subsystem that keeps track of the definitions of data items in the database which includes:**

Elementary-level data items (fields/attributes),
Relationships that exists between various data structures.
Files or relational tables.
Indexes that are used to access data quickly.

Meta Data

- **Data that describe the properties or characteristics of other data.**

  Some of these properties include the name of the data item, data type, length, minimum and maximum allowable values (where appropriate), rules or constraints and a brief description of each data item.

- **Metadata allow database designers and users to understand what data exist, what the data mean.**
- **Data without clear meaning can be confusing, misinterpreted or erroneous.**

Meta Data

- **E.g. Employee**

Name	Type	Length	Min	Max	Description
EmpNo	Number	9			Employee No.
Name	Character	30			Employee Name
Dept	Character	10			Dept. No.
Salary	Number	8	5000	60000	Employee Salary

Employee No. (ID) unique

Table Definition

Employees : Table

Field Name	Data Type	
EmployeeID	AutoNumber	Number automatically assigned to new employee.
LastName	Text	
FirstName	Text	
Title	Text	Employee's title.
TitleOfCourtesy	Text	Title used in salutations.
BirthDate	Date/Time	
HireDate	Date/Time	
Address	Text	Street or post-office box.
City	Text	
Region	Text	State or province.
PostalCode	Text	
Country	Text	
HomePhone	Text	Phone number includes country code or area code.
Extension	Text	Internal telephone extension number.
Photo	OLE Object	Picture of employee.
Notes	Memo	General information about employee's background.
ReportsTo	Number	Employee's supervisor.

General | Lookup

Field Size	20
Format	
Input Mask	
Caption	Last Name
Default Value	
Validation Rule	
Validation Text	
Required	Yes
Allow Zero Length	No
Indexed	Yes (Duplicates OK)
Unicode Compression	Yes
IME Mode	No Control
IME Sentence Mode	None

Database Approach

- **Data Definition Language (DDL).**
  - **Permits specification of data types, structures and any data constraints.**
  - **All specifications are stored in the database.**

- **Data Manipulation Language (DML).**
  - **is used for selecting, inserting, deleting and updating data in a database.**
  - **General enquiry facility (query language) of the data.**

Database Approach

- Controlled access to database may include:
  - A security system.
  - An integrity system.
  - A concurrency control system.
  - A recovery control system.

Data Security

- The database is a valuable resource needing protection.
- The DBMS provides database security by limiting access to the database to authorised personnel.
- Authorised users will generally be restricted as to the particular data they can access and whether they can update it.
- Access is often controlled by passwords and by data views.

Data Integrity

- **The integrity and consistency of the database are protected via constraints on values that data items can have.**

- **Data constraint definitions are maintained in the data dictionary.**

Concurrency control

- **Concurrency control in database management systems (DBMS) ensures that database transactions are performed concurrently without the concurrency violating the data integrity of a database.**

Recovery Control

- **Backup and recovery are supported by software that automatically logs changes to the database and provides for a means of recovering the current state of the database in case of system failure.**

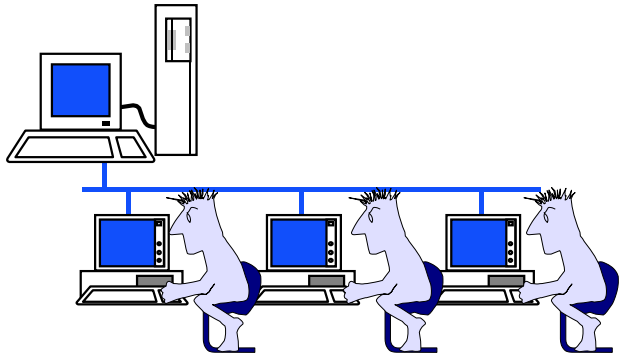
  Power Failure in a Bank?

  Day's transactions?

  Bomb or Floods destroy your computer system?

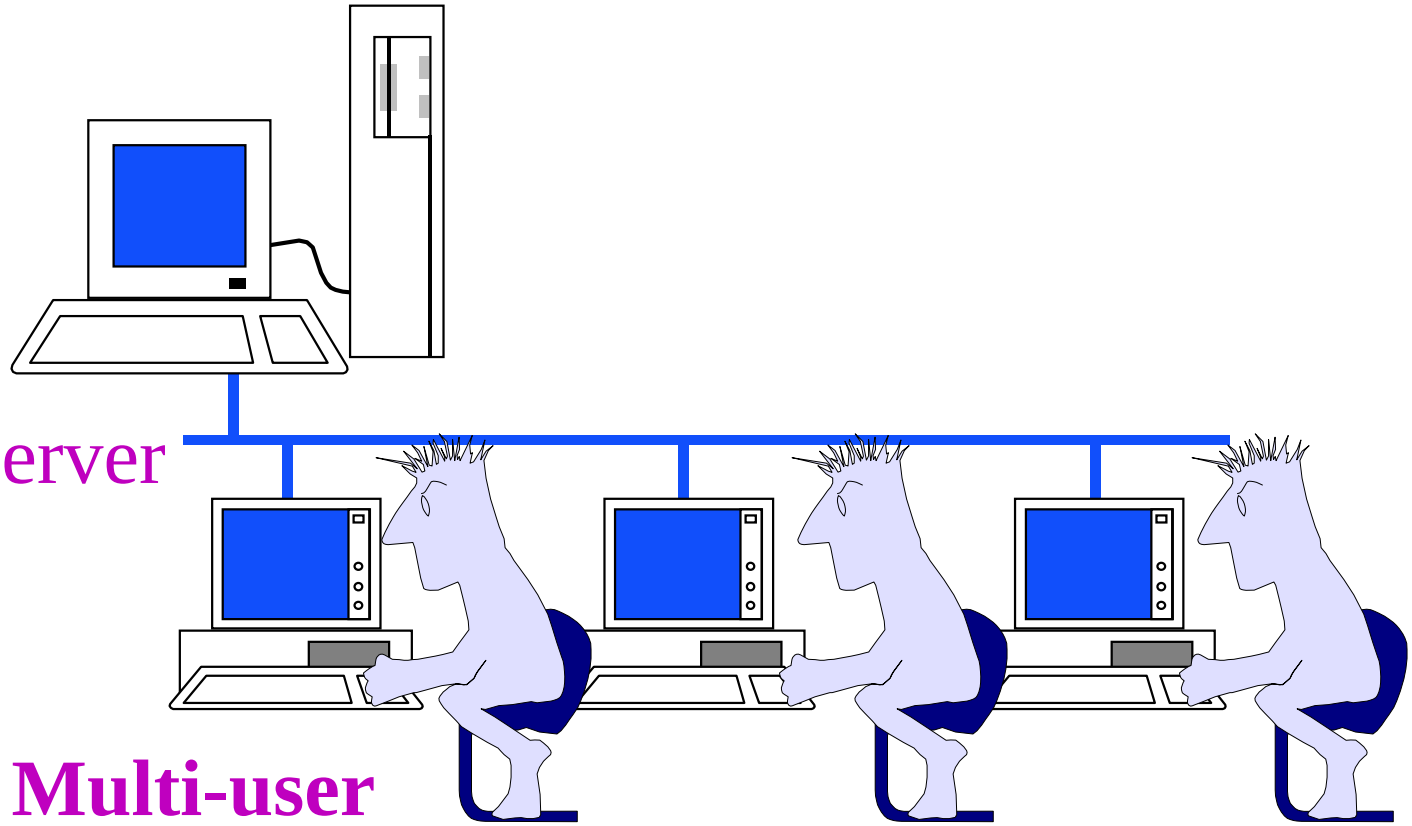
Database Server Architecture

- **Client/server platform**
  - A local area network consisting of client computers which receive services from a server computer.
- **Database server**
  - A program running on server hardware to provide database services to client machines.



Client/Server Platform

- Clients interacting with a database server



Database Applications

Databases range from those for a single user with a desktop computer to those on mainframe computers with thousands of users.

- **Personal databases**
- **Workgroup databases**
- **Departmental databases**
- **Enterprise databases**

Personal databases

Designed to support one user with a standard alone PC.

E.g. a sales person keeping track of this customer information with contact details.

Workgroup databases

A relatively small team of people (less than 25) who collaborate on the same project or application.

E.g. a team of engineering designers maintain versions of the artifact that they design.

Departmental databases

A department is a functional unit of an organisation. It is larger than a workgroup.

Department databases are designed to support the various functions and activities of a department.

E.g. a personnel database that is designed to track data concerning employees, jobs, skills and job assignments.

Enterprise databases

An enterprise is one whose scope is the entire organisation or enterprise.

Such databases are intended to support organisation-wide operations and decision making.

E.g. a large health care organisation that operates a group of medical centre's including hospitals, clinics and nursing homes.

Enterprise databases

An enterprise database does support information needs from many departments. The most important type of enterprise database today is called a data warehouse.

- **Data warehouse**
  - **An integrated decision support database whose content is derived from the various operational databases.**

Database Approach -Advantages

- Improved maintenance through program-data independence
- Minimal data redundancy
- Improved data consistency
- Improved data sharing
- Increased productivity

Database Approach -Advantages

- Enforcement of standards
- Improved data integrity
- Improved data accessibility and responsiveness
- Improved security
- Increased concurrency

Improved maintenance through Program-Data/Data Independence

- The separation of data descriptions (metadata) from the application programs that use the data.

- This simplifies database application maintenance.

Improved maintenance through Program-Data/Data Independence

- In the database approach data descriptions are stored in a central location called the data dictionary.

- This property allows an organisation's data to change and evolve (within limits) without changing the application program that process the data.

Minimal Data Redundancy

- Data files are integrated into a single, logical structure. Each primary fact is recorded (ideally) in only one place in the database.
- E.g. Employee data not with the payroll and benefit files.

Note: Data redundancy is not eliminated entirely. Some data items will appear in more than one place (e.g. employee no.) to represent the relationship with others.

Improved Data Consistency

- By eliminating (or controlling) data redundancy, we greatly reduce the opportunities for inconsistency.

  E.g. employee address is stored only once and hence we cannot have disagreement on the stored values.

- Updating data values is greatly simplified and have avoided the wasted storage space.

Improved Data Sharing

- A database is designed as a shared corporate resource and can be shared by all authorised users. In this way more users share more of the data.

  E.g. employee data common to payroll, benefit applications will be shared among different users.

  New applications can be built on the existing data in the database.

Increased Productivity

- A major advantage of the database approach is that it greatly reduces the cost and time for developing new business applications.

- Programmer could concentrate on the specific functions required for the new application, without having to worry about design or low-level implementation details; as related data has already been designed and implemented.

Increased Productivity

- DBMS provides many of the standard functions (e.g. forms and report generations) that the programmer would normally have to write in a file-based application DBMS.

Enforcement of Standards

- When the database approach is implemented with full management support, the database administration function should be granted single-point authority and responsibility for establishing and enforcing data standards.

- Standards include naming conventions, data quality standards and uniform procedures for accessing, updating and protecting data.

Improved Data Integrity

- Integrity can be expressed in terms of *constraints*, which are consistency rules that the database is not permitted to violate.

Eg: A member of staff's salary cannot be greater than 60,000.

Improved Data Accessibility and Responsiveness

- With relational database, end users without programming experience can often retrieve and display data, even when it crosses traditional departmental boundaries.

- English-like query language SQL and query tools such as Query-By-Example provide such facilities.

Improved Security

- DBMS can be used to enforce database security. This may take the form of user names and passwords to identify people authorised to use the database.

- The access that the authorised user is allowed on the data can also be restricted by the operation type (retrieval, delete, update, insert).

Increased concurrency

- Many DBMSs allow users to undertake simultaneous operations on the database. The DBMS implements a *concurrency control* mechanism that prevents database accesses from interfering with one another.

Disadvantages of DBMS

- Complexity
- Size
- Cost of DBMS
- Additional hardware costs
- Cost of conversion
- Performance
- Higher impact of a failure

-END-