What is “Data”?

- ANSI definition of data:
  - A representation of facts, concepts, or instructions in a formalized manner suitable for communication, interpretation, or processing by humans or by automatic means.
  - Any representation such as characters or analog quantities to which meaning is or might be assigned. Generally, we perform operations on data or data items to supply some information about an entity.

- Volatile vs persistent data
  - Our concern is primarily with persistent data

Representation of Data

**Data Model**
Formalism that defines how data is structured and how it can be accessed and manipulated

Examples:
- Hierarchical; network
- Relational
- Object-oriented (more recently object-relational)

**Schema**
Definition of how a particular collection of data is organized, using a given data model

**Instance**
A particular collection of data
- Instances conform to a schema.
- A schema can have many instances.
Representation of Data (Example)

- **Data model:** Relational
- **Schema:**
  - EMP(ENO, ENAME, TITLE)
  - PROJ(PNO, PNAME, BUDGET)
  - WORKS(ENO, PNO, RESP, DUR)
- **Instance:**

```
<table>
<thead>
<tr>
<th>EMP</th>
<th>WORKS</th>
</tr>
</thead>
<tbody>
<tr>
<td>ENO</td>
<td>PNO</td>
</tr>
<tr>
<td>E1</td>
<td>P1</td>
</tr>
<tr>
<td>E2</td>
<td>P1</td>
</tr>
<tr>
<td>E3</td>
<td>P2</td>
</tr>
<tr>
<td>E4</td>
<td>P3</td>
</tr>
<tr>
<td>E5</td>
<td>P4</td>
</tr>
<tr>
<td>E6</td>
<td>P5</td>
</tr>
<tr>
<td>E7</td>
<td>P6</td>
</tr>
<tr>
<td>E8</td>
<td>P7</td>
</tr>
<tr>
<td>E9</td>
<td>P8</td>
</tr>
</tbody>
</table>
```

Early Data Management – Ancient History

- Data are not stored on disk
- One data set per program. High data redundancy

```
PROGRAM 1

Data Management

DATA SET 1
```

```
PROGRAM 2

Data Management

DATA SET 2
```

```
PROGRAM 3

Data Management

DATA SET 3
```

File Processing – More Recent History

- Data are stored in files with interface between programs and files.
- Various access methods exist (e.g., sequential, indexed, random).
- One file corresponds to one or several programs.
Database Approach

Definition (Database)
A large and persistent collection of (more-or-less similar) pieces of information organized in a way that facilitates efficient retrieval and modification.

Definition (Database Management System (DBMS))
A software system designed to store and manage databases.

Database Management System

Idea
Abstracts common functions and creates a uniform well defined interface for applications accessing data.

- Data model
  all data stored in a well defined way
- Access control
  only authorized people get to see/modify it
- Concurrency control
  multiple concurrent applications access data
- Database recovery
  nothing gets accidentally lost
- Database maintenance
  structures are modified to accommodate evolving needs
Data Independence

**Idea**

Applications do not access data directly but, rather through an abstract data model provided by the DBMS.

Two kinds of data independence:
- **Physical**: applications immune to changes in storage structures
- **Logical**: applications immune to changes in data organization

**Note**

One of the most important reasons to use a DBMS!

---

Three Levels of Abstraction

**External Schema (View)**

What an application program or user sees (may differ for different users of the same database)

**Conceptual Schema**

Description of the logical structure of all data in the database

**Physical Schema**

Description of physical aspects of how data is structured (selection of files, devices, storage structures, etc.)

---

Three Levels of Abstraction (cont’d)

[Diagram showing three levels of abstraction]

---

Notes
Types of Database Users

**End user:**
- Accesses the database indirectly through forms or other query-generating applications, or
- Generates ad-hoc queries using a dedicated language.

**Application developer:**
- Designs and implements applications that access the database.

**Database administrator (DBA):**
- Manages conceptual schema.
- Assists with application view integration.
- Monitors and tunes DBMS performance.
- Defines internal schema.
- Loads and reformats database.
- Is responsible for security and reliability.

Primary Interface to a DBMS

**Data Manipulation Language (DML) for specifying queries and modifying the data**
- navigational (procedural)
- non-navigational (declarative)

**Data Definition Language (DDL) for specifying schemas**
- may have different DDLs for external schema, conceptual schema, internal schema
- information is stored in the data dictionary, or catalog

Summary

Using a DBMS to manage data helps:
- to remove common code from applications
- to provide uniform access to data
- to guarantee data integrity
- to manage concurrent access
- to protect against system failure
- to set access policies for data