

### *Types of Information Systems*

*Competition needs very fast decisions and rapid development of information systems.*

Concentrate on what to do rather than how to do.

*For many companies, information systems cost 40 percent of overall costs.*

*cetiner@itu.edu.tr*

*Dr.B.G.Cetiner*

### *Types of Information Systems*

**CASE: Computer Aided Software Engineering Tools**

Software Tools used to automate Software Development Life Cycle.

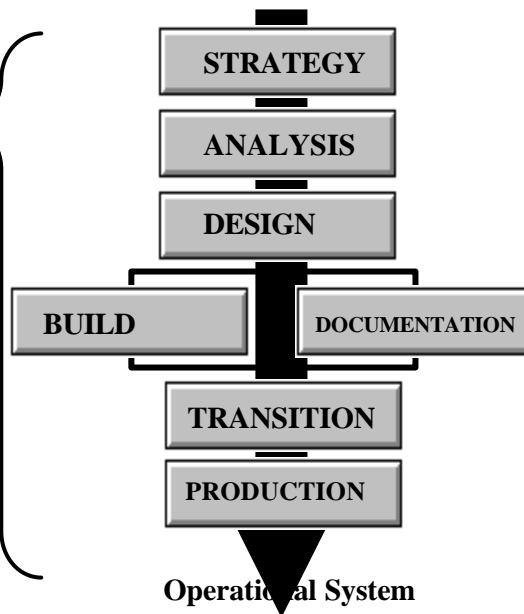
*cetiner@itu.edu.tr*

*Dr.B.G.Cetiner*

**Software Development  
Life Cycle  
(Waterfall Model)**

This type of Software Development Life Cycle is called Waterfall Model. Since it is difficult to swim up to the waterfall stream, it is costly to go back to the previous stages in life cycle.

Therefore, it is essential to finish a good data model before starting database design.

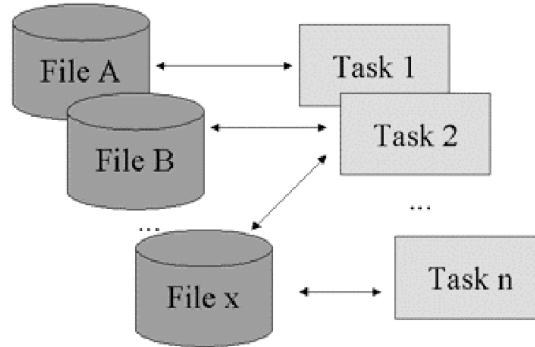


**CASE TOOLS**

**The Use of CASE in Organizations**

- Objectives of CASE
  - Improve quality of systems developed
  - Increase speed of development and design
  - Ease and improve testing process through automated checking
  - Improve integration of development activities via common methodologies
  - Improve quality and completeness of documentation
  - Help standardize the development process
  - Improve project management
  - Simplify program maintenance
  - Promote reusability
  - Improve software portability

Data Processing with Files

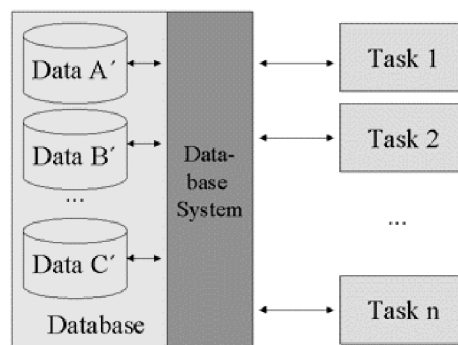


Each task uses one or more files; each file will be used by one or more tasks. In other words: The data structures of the different files are independent of each other and whenever a programmer would like to use a file, he must know the exact data structure (with respect to its implementation in the file). Access to the data is directly (using the data management facility of the operating system; without any restriction).

*cetiner@itu.edu.tr*

*Dr.B.G.Cetiner*

Data Processing with Databases



The Database System (usually called Database Management System as well; abbr. DBS or DBMS) connects Tasks with the data (no longer in file structure!). It offers views to the data (instead of direct access to file structures).

*cetiner@itu.edu.tr*

*Dr.B.G.Cetiner*

## *Database Systems*

### **Characteristics of Database Systems and Databases**

- Many data sets are stored in a single database (all data sets of a firm).
- Each data set has a unique description (stored in a data dictionary).
- Access to data can be done via Database System only (like a "Firewall").
- Relationships between data are defined in the database.
- The Database System offers tools for back-up, screen-design etc.

**Note: Features of a Relational Database Management System will be defined later.**

*cetiner@itu.edu.tr*

*Dr.B.G.Cetiner*

## *Database Systems*

### **So: Why Databases?**

**Redundancy** can be controlled; i.e. each given fact in the real world corresponds to one data entry in the database.

**Exception:** Controlled redundancy for security reason (e.g. shadow database).

**Inconsistency** can be avoided (to some extent); Redundancy is under control.

Data can be **shared**. New applications can operate with the same data (with only a new view).

*cetiner@itu.edu.tr*

*Dr.B.G.Cetiner*

## *Database Systems*

### **So: Why Databases?**

**Standards** can be enforced.

**Security restrictions** can be implemented. Illegal access to data can be avoided.

**Integrity** can be maintained; data correspond to facts of the real world.

Conflicting **requirements** (access speed, availability, reliability, trustworthy) can be balanced; best performance overall can be found.

*cetiner@itu.edu.tr*

*Dr.B.G.Cetiner*

## *Database Systems*

### **Please consider:**

**Database =data (in a special technical form)**

**Database System =software to work with databases**

*cetiner@itu.edu.tr*

*Dr.B.G.Cetiner*

## **History of Database Systems**

### **First Generation (1945 -1960)**

- data on punched cards or on magnetic tapes
- offline services /batch processing
- sequential access to data

## **History of Database Systems**

### **Second Generation (1960 -1970)**

- data on magnetic disks (files)
- interactive data processing
- direct access to data
- multiple access /parallel access
- Hierarchical Data Model;  
    example: IMS (Information Management System) by IBM
- Network Model;  
    example: UDS (Universelles Datenbank System) by Siemens
- Data structure diagrams introduced by Bachman

## **History of Database Systems**

### **Third Generation (1970 -1980)**

- **Data Independence**
- **Non-procedural languages (WHAT instead of HOW)**
- **Computer-independent database systems (Network DBS: CODASYL-DBTG)**
- **Relational Data Model;**  
examples: SQL /DS (Standard Query Language /Data System) or DB2 by IBM,  
Oracle by Oracle
- **Entity-Relationship Model introduced by Chen**

## **History of Database Systems**

### **Fourth Generation (1980 -1990)**

- **Database Systems developed for Personal Computers;**  
examples: dBASE, Paradox
- **Object-oriented Database Systems;**  
examples: POET by POET, O2 by O2 Technology

## History of Database Systems

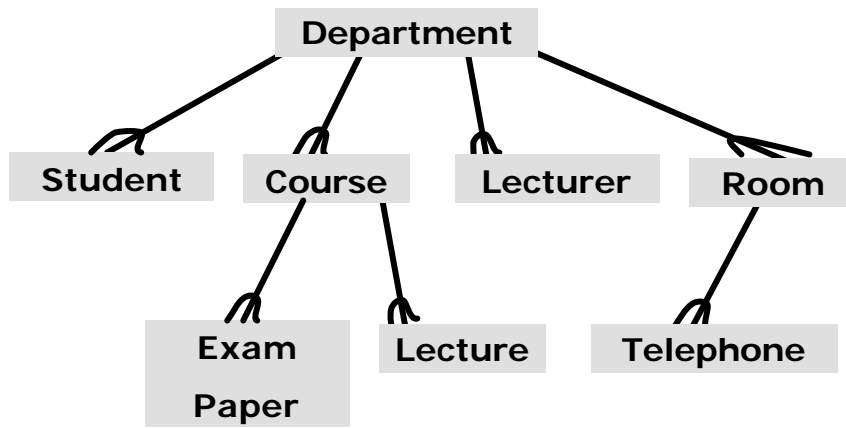
### Fifth Generation (1990 -today)

- Client-server architectures
- Massively parallel processors
- Open interfaces (esp. ODBC)
- Integration of Internet and databases (esp. JDBC)
- Object-relational Database Systems
- Deductive Databases (rules & facts)

## There are 4 important models

1. Hierarchical Model
  2. Network Model
  3. Relational Model
  4. Object-Oriented Model
- Most of the models today





You need to find a root entity (here the root entity is department)

