LECTURE 6

Chapter 2

Data models

OBJECTIVES

In this chapter, you will learn:

- About data modeling and why data models are important
- About the basic data-modeling building blocks
- What business rules are and how they influence database design
- How the major data models evolved
- About emerging alternative data models and the need they fulfill
- How data models can be classified by their level of abstraction

Introduction

- Designers, programmers, and end users see data in different ways
- Different views of same data lead to designs that do not reflect organization’s operation
- Data modeling reduces complexities of database design
- Various degrees of data abstraction help reconcile varying views of same data

DATA MODELING AND DATA MODELS

- Database design focuses on how the database structure will be used to store and manage end-user data
- Data modeling, the first step in designing a database, refers to the process of creating a specific data model for a determined problem domain.
- Model: an abstraction of a real-world object or event Useful in understanding complexities of the real-world environment
- Data models Relatively simple representations of complex real-world data structures Often graphical
- Data modeling is iterative and progressive

The importance of data models

- Facilitate interaction among the designer, the applications programmer, and the end user
- End users have different views and needs for data
- Data model organizes data for various users
- Data model is an abstraction
  - Cannot draw required data out of the data model

Data model basic building blocks

- Entity: anything about which data are to be collected and stored.
- An entity represents a particular type of object in the real world, which means an entity is "distinguishable"—that is, each entity occurrence is unique and distinct.
- Attribute: a characteristic of an entity
- Relationship: describes an association among entities
- One-to-many (1:M) relationship: e.g. a customer (the "one") may generate many invoices, but each invoice (the "many") is generated by only a single customer. The "CUSTOMER generates INVOICE" relationship would also be labeled 1:M.
- Many-to-many (M:N or M:M) relationship: e.g. a student can take many classes and each class can be taken by many students, thus yielding the M:N label for the relationship expressed by "STUDENT takes CLASS."
- One-to-one (1:1) relationship: e.g. retail company's management structure may require that each of its stores be managed by a single employee. In turn, each store manager, who is an employee, manages only a single store. Therefore, the relationship "EMPLOYEE manages STORE" is labeled 1:1.
- Constraint: a restriction placed on the data help to ensure data integrity, expressed in the form of rules