

CS 445  
Introduction to Database Design  
E-R Diagrams

Chapter 2

August 29, 2019

# Design Steps

- How do we model the data?
  - what do we need to identify?

# Design Steps

- Requirement Analysis
  - talk to the user!
- Conceptual Database Design
  - E-R Diagram
- Logical Database Design
  - logical schema
- Schema Refinement
  - normalization
- Physical Database Design
  - performance tuning
- Application and Security Design
  - GUI / end user software

# Data

- An employee has a name, birthday, and email
- A department has a name and budget
- Each employee is either salaried or hourly
- An employee works in a department
- Each employee has exactly one manager (who is also an employee)
- Each department has exactly one Headquarters at exactly one location
- Each department has exactly one department manager
- Each HQ has exactly one office manager (an employee can only be an office manager for one HQ).
- Each department can be present at many locations.
- Each location for a department has exactly one office manager.

# Bits of Data

- Entity
  - some particular object in the real world
- Entity Set
- Attribute Employee
  - domain
  - key
  - candidate key
  - primary key

# Doing interesting things with data

- Relationship
  - association among two or more entities
- Relationship Set
- Descriptive attribute
- Roles

# Constraints

- What limits are placed on how entities are involved in a relationship
  - Key Constraints
    - One to many
    - Many to many
    - One to one
  - Participation Constraints

# Weak Entities

- Entities without keys!
- Identifying owner
  
  
  
  
  
  
  
  
  
  
- Identifying relationship



# Class Hierarchy

- Some entities may be related
  - similar to Object Oriented class hierarchy
  - C++/Java
  - superclass
  - specialized subclasses
- Inheritance
  - ISA
- Overlap constraints

# Aggregation

- View a set of entities/relationships as one big entity
  - meta-entity

# How do we use all this?

- When do we use an **entity** vs an **attribute** to represent data?
  - it all depends on how you want to **use** the data
  - how many other bits of data will reference it?
  - how will they reference it?
    - **will our model allow that?**
- Example: Name and Address

# How do we use all this?

- When do we use an **entity** vs a **relationship**?

# Tool Support

- E-R diagram builders
    - Umlet
  - Unified Modeling Language (UML)
    - used to model all kinds of data interactions
    - Object Oriented code design (Class diagrams)
    - database design
      - think of entities and relationships as classes
    - Use cases (process flow)
    - UMLet
    - <http://argouml.tigris.org>
    - <http://argouml-sql.tigris.org>
- Some tools generate database tables/queries for you.

# Key Constraints

- emp MANAGES dept
- each emp can manage more than one dept
- each dept is managed by only one emp
  - Each dept key appears in ONE MANAGES relationship
  - ONE TO MANY
  - one employee can be associated with MANY depts
  - each dept associated with ONE emp
  - what if each emp ONLY managed ONE dept? (ONE TO ONE)
- emp WORKSIN dept
  - each emp can work in several depts
  - each dept has several emp
  - MANY TO MANY
  - what is each emp worked in only one DEPT?

# Ramakrishnan/Gehrke: Page 52

do this in pairs, produce one UMLet diagram

**Exercise 2.3** Consider the following information about a university database:

- Professors have an SSN, a name, an age, a rank, and a research specialty.
- Projects have a project number, a sponsor name (e.g., NSF), a starting date, an ending date, and a budget.
- Graduate students have an SSN, a name, an age, and a degree program (e.g., M.S. or Ph.D.).
- Each project is managed by one professor (known as the project's principal investigator).
- Each project is worked on by one or more professors (known as the project's co-investigators).
- Professors can manage and/or work on multiple projects.
- Each project is worked on by one or more graduate students (known as the project's research assistants).
- When graduate students work on a project, a professor must supervise their work on the project. Graduate students can work on multiple projects, in which case they will have a (potentially different) supervisor for each one.

# Car Dealership

- Work in pairs again
- Build the ER diagram in UMLet