The Entity-Relationship Model

Zaki Malik

September 16, 2008
Till the Midterm Examination

• Relational Data Models
  – The Entity-Relationship (ER) model
  – The relational model
  – Converting E/R diagram to relational designs.

• You should know how to
  – Identify all entities and relationships and describe them using an E/R diagram.
  – Convert the E/R model to a number of relations in a relational schema.

• Use all these ideas to design your own database application in your project.
Basic Database Terminology

• **Data model**: describes high-level conceptual structuring of data
  – Example: Data is set of student records, each with ID, name, address, and courses
  – Example: Data is a graph where nodes represent proteins and edges represent chemical bonds between proteins

• **Schema** describes how data is to be structured and stored in a database
  – Defined during creation of the database
  – Schemas rarely change

• **Data** is actual “instance” of database
  – Updated continuously
  – Changes rapidly
Why Learn About Database Modeling?

• The way in which data is stored is very important for subsequent access and manipulation by SQL.

• Properties of a good data model:
  – It is easy to write correct and easy to understand queries.
  – Minor changes in the problem domain do not change the schema.
  – Major changes in the problem domain can be handled without too much difficulty.
  – Can support efficient database access.
Purpose of E/R Model

• The E/R model allows us to sketch the design of a database informally.
  – Represent different types of data and how they relate to each other
• Designs are pictures called *entity-relationship diagrams*.
• Fairly mechanical ways to convert E/R diagrams to real implementations like relational databases exist.
Entity Sets

• *Entity* = “thing” or object.
• *Entity set* = collection of similar entities.
  – Similar to a class in object-oriented languages.
• *Attribute* = property of an entity set.
  – Generally, all entities in a set have the same properties.
  – Attributes are simple values, e.g. integers or character strings.
E/R Diagrams

• In an entity-relationship diagram, each entity set is represented by a rectangle.

• Each attribute of an entity set is represented by an oval, with a line to the rectangle representing its entity set.
Example

- Entity set Beers has two attributes, name and manf (manufacturer).
- Each Beer entity has values for these two attributes, e.g. (Bud, Anheuser-Busch)
Relationships

• A relationship connects two or more entity sets.
• It is represented by a diamond, with lines to each of the entity sets involved.
Example

Bars sell some beers.

Drinkers like some beers.

Drinkers frequent some bars.

Note: license = beer, full, none

Bars

Sells

Beers

Frequents

Likes

Drinkers
Relationship Set

• The current “value” of an entity set is the set of entities that belong to it.
  – Example: the set of all bars in our database.

• The “value” of a relationship is a set of lists of currently related entities, one from each of the related entity sets.
Example

• A relationship $R$ between entity sets $E$ and $F$ relates some entities in $E$ to some entities in $F$.
• $R$ is a set of pairs of tuples $(e, f)$ where $e$ is in $E$ and $f$ is in $F$.

• For the relationship *Sells*, we might have a relationship set like:

<table>
<thead>
<tr>
<th>Bar</th>
<th>Beer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joe’s Bar</td>
<td>Bud</td>
</tr>
<tr>
<td>Joe’s Bar</td>
<td>Miller</td>
</tr>
<tr>
<td>Sue’s Bar</td>
<td>Bud</td>
</tr>
<tr>
<td>Sue’s Bar</td>
<td>Pete’s Ale</td>
</tr>
<tr>
<td>Sue’s Bar</td>
<td>Bud Lite</td>
</tr>
</tbody>
</table>
Multiway Relationships

• Sometimes, we need a relationship that connects more than two entity sets.
• Suppose that drinkers will only drink certain beers at certain bars.
  – Our three binary relationships Likes, Sells, and Frequents do not allow us to make this distinction.
  – But a 3-way relationship would.
Example

Bars

name
addr
license

Preferences

name

name

Beers

manf

Drinkers

name
addr
A Typical Relationship Set

<table>
<thead>
<tr>
<th>Bar</th>
<th>Drinker</th>
<th>Beer</th>
</tr>
</thead>
<tbody>
<tr>
<td>Joe’s Bar</td>
<td>Ann</td>
<td>Miller</td>
</tr>
<tr>
<td>Sue’s Bar</td>
<td>Ann</td>
<td>Bud</td>
</tr>
<tr>
<td>Sue’s Bar</td>
<td>Ann</td>
<td>Pete’s Ale</td>
</tr>
<tr>
<td>Joe’s Bar</td>
<td>Bob</td>
<td>Bud</td>
</tr>
<tr>
<td>Joe’s Bar</td>
<td>Bob</td>
<td>Miller</td>
</tr>
<tr>
<td>Joe’s Bar</td>
<td>Cal</td>
<td>Miller</td>
</tr>
<tr>
<td>Sue’s Bar</td>
<td>Cal</td>
<td>Bud Lite</td>
</tr>
</tbody>
</table>
Many-Many Relationships

- Think of a relationship between two entity sets, such as *Sells* between *Bars* and *Beers*.
- In a *many-many* relationship, an entity of either set can be connected to many entities of the other set.
  - E.g., a bar sells many beers; a beer is sold by many bars.
Many-One Relationships

• Some binary relationships are *many-one* from one entity set to another.
• Each entity of the first set is connected to *at most* one entity of the second set.
• But an entity of the second set can be connected to *zero, one, or many* entities of the first set.
Example

• *Favorite*, from *Drinkers* to *Beers* is many-one.
• A drinker has at most one favorite beer.
• But a beer can be the favorite of any number of drinkers, including zero.
One-One Relationships

• In a one-one relationship, each entity of either entity set is related to at most one entity of the other set.
• Example: Relationship Best-seller between entity sets $\text{Manfs}$ (manufacturer) and $\text{Beers}$.
  – A beer cannot be made by more than one manufacturer, and no manufacturer can have more than one best-seller (assume no ties).
In Pictures:

- Women Give Birth Babies
- Many-many
- Many-one
- One-one

- Men Befriend Women
- Marry Women
Representing “Multiplicity”

- Show a many-one relationship by an arrow entering the “one” side.
- Show a one-one relationship by arrows entering both entity sets.
- In some situations, we can also assert “exactly one,” i.e., each entity of one set must be related to exactly one entity of the other set. To do so, we use a rounded arrow.
Example
Example

• Consider *Best-seller* between *Manfs* and *Beers*.

• Some beers are not the best-seller of any manufacturer, so a rounded arrow to *Manfs* would be inappropriate.

• But a manufacturer has to have a best-seller (we assume they are beer manufacturers).
Attributes on Relationships

- Sometimes it is useful to attach an attribute to a relationship.
- Think of this attribute as a **property of tuples** in the relationship set.

![Diagram](image)

Price is a function of both the bar and the beer, not of one alone.
Relationships

- Can the same entity set appear more than once in the same relationship?
- Prerequisite relationship between two Courses

- Which course is the pre-requisite?
Roles

- Label the edges between the relationship and the entity set with names called *roles*.
Example

Relationship Set

<table>
<thead>
<tr>
<th>Husband</th>
<th>Wife</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bob</td>
<td>Ann</td>
</tr>
<tr>
<td>Joe</td>
<td>Sue</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Example

Relationship Set

<table>
<thead>
<tr>
<th>Buddy1</th>
<th>Buddy2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Bob</td>
<td>Ann</td>
</tr>
<tr>
<td>Joe</td>
<td>Sue</td>
</tr>
<tr>
<td>Ann</td>
<td>Bob</td>
</tr>
<tr>
<td>Joe</td>
<td>Moe</td>
</tr>
<tr>
<td>...</td>
<td>...</td>
</tr>
</tbody>
</table>
Parallel Relationships

- Can there be more than one relationship between the same pair of entities?
- TA and Take relationship between Students and Classes
Subclasses

- Subclass = special case = fewer entities = more properties.
- Example: Ales are a kind of beer.
  - Not every beer is an ale, but some are.
  - Let us suppose that in addition to all the properties (attributes and relationships) of beers, ales also have the attribute color.
Subclasses in E/R Diagrams

• Assume subclasses form a tree.
  – i.e., no multiple inheritance.

• Isa triangles indicate the subclass relationship.
  – Point to the superclass.
Example

- Beers
  - name
  - manf
    - isa
      - Ales
        - color