Today …

- ER Modeling

Homework

- HW 3 out
Conceptual Data Modeling Approaches

Design typically involves multiple steps prior to creating tables!

- We’ll focus on the **Entity-Relationship Model** (ER)
- Similar to UML diagrams in software design
- Sometimes called ER diagrams, or ERDs

![Entity-Relationship Diagram](image_url)

**Key:**
- Entity Set
- Relationship Set
- Attribute

```
Employee

Department

Project
```
ER Terminology

An “Entity”

- an object distinguishable from other objects
- e.g., the employee “John Smith”
- described using a set of attribute-value pairs
- designated attribute (or composite attributes) serves as an “identifier” (key)

An “Entity Set”

- a collection of similar entities
- defined by attributes and relationships that characterize the entities
- sometimes just called “Entity” (when it is clear we’re talking about the set)
- sometimes referred to as the Entity Type
- an entity is an instance (or member) of an entity set
A “Relationship”

- an association among 2 or more entities
- e.g., John Smith’s home department is Pharmacy 2

A “Relationship Set”

- a collection of similar relationships
- e.g., the set of home department relationships
- defined by the participating entity types and other constraints
- just called a “Relationship” (when context clear)
- also sometimes called a “Relationship Type”
- a relationship is an instance (or member) of a relationship set
**ER versus the Relational Model**

ER is a different data model than the relational model

- different *constructs* for modeling schemas and instances
- they are pretty close though

The *relational model* has ...

- tables (relations) with attributes, keys and foreign keys, rows, values

The *ER model* has ...

- entity & entity sets with attributes and entity identifiers (like keys)
- relationship & relationship sets with cardinality constraints, roles, attributes

Usually start with ER models, then map them to relational schemas
**ER Cardinality Constraints (Example)**

- An Employee can have **0 or 1** home Departments
- A Department can have **0 or many** Employees
- A Department must have **exactly one** Manager
- etc.

**Note on Notation ...**

- we’re using more of a “UML” style notation
- other notations as well (e.g., “crow’s foot”)
How do these differ?

Q: Which of these is correct?

• Based on the application requirements!

Constraints are expressed over Entity and Relationship Sets

• Constrain the members of the corresponding sets

Q: Does this satisfy the cardinality constraints?
Q: What about now?

Q: And now?
Continuing with ER modeling

There are different notations for writing cardinality constraints ...

- Examples of “one to many” constraints

<table>
<thead>
<tr>
<th>one</th>
<th>many</th>
<th>zero..one</th>
<th>one..many</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>n</td>
<td>0:1</td>
<td>1:n</td>
</tr>
<tr>
<td>1</td>
<td>*</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

Maximum cardinalities only

Minimum and maximum cardinalities

- Examples of “many to many” constraints

<table>
<thead>
<tr>
<th>many</th>
<th>many</th>
<th>one..many</th>
<th>one..many</th>
</tr>
</thead>
<tbody>
<tr>
<td>m</td>
<td>n</td>
<td>1:n</td>
<td>1:n</td>
</tr>
<tr>
<td>*</td>
<td>*</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Maximum cardinalities only

Minimum and maximum cardinalities
Relationship Attributes

Similar to Entities, Relationships can have attributes

- Each instance of the relationship set has a value for the attribute

Try all three locations ... where does the attribute make sense?

- Because Employees have zero or one home department
  - start date will work as an Employee or home attribute
- Not clear what it would mean at Department

Q: What if Employee could have multiple home departments?
  - Instead of 0..1, we’d have 0..*
  - In this case, it would need to be a relationship attribute


**Role names**

Relationships can have role names

![Diagram](image_url)

- An employee “manages” zero or one department
- A department is “managed by” exactly one employee

**Role names help in more complex relationships**

![Diagram](image_url)

- Here Employee participates in different roles for the same relationship set