Today

- Quiz 8
- Exam 2 overview
- More on the ER model

Assignments

- HW 9 and PROJ 2 due Thur
Exam 2 Overview

Basics ...

- Closed book, notes, etc.
- 4 multi-part questions
- Worth 15% of final grade

Possible Topics (What we’ve covered):

- SQL
  - Group By and Having
  - Subqueries (various forms and operators)
  - Rewriting subqueries into joins (when possible)
  - Outer joins
  - Some SQL possible from previous exam (joins, union, aggregates, etc.)

- Normalization
  - Redundancy and anomalies (insert, update, delete)
  - Functional dependencies (different types, spotting them)
  - FDs and redundancy
  - FDs and inference (closures)
  - Finding keys from a set of FDs
  - BCNF (normal form and algorithm)
  - Good decompositions: lossless and dependency preserving
  - Canonical covers and 3NF (normal form and algorithm)

- Basic ER modeling
  - Entity sets, Relationship sets, Entities, Relationships, Attributes, Unique Attributes (Keys), Cardinality Constraints
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></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Maximum cardinalities only

- 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>
<tr>
<td></td>
<td>*</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Maximum cardinalities only
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

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

Role names help in more complex relationships

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