CS143
Entity-Relationship Model

Professor Junghoo “John” Cho
Entity-Relationship (E/R) Model

• Q: How should we design tables in our database?
  • Tables are not “given”
  • “Good” tables may not be easy to come up with

• E/R model: graphical, intuitive and “informal” representation of information on database
  • Used to “capture” what we learn from domain experts/database users
  • Not directly implemented by DBMS
  • Tools exist to automatically convert E/R model into tables

• Two main components
  • Entity sets and relationship sets
Entity Set

• Entity: “thing” or “object” in real world
  • E.g., I, this book, UCLA

• Entity set: a set of entities (objects). Like a class in OOP
  • Rectangle in ER
  • Consists of “name” and “attributes”

<table>
<thead>
<tr>
<th>Students</th>
<th>Classes</th>
<th>Faculty</th>
</tr>
</thead>
<tbody>
<tr>
<td>sid</td>
<td>dept</td>
<td>name</td>
</tr>
<tr>
<td>name</td>
<td>cnum</td>
<td>title</td>
</tr>
<tr>
<td>addr</td>
<td>sec</td>
<td>office</td>
</tr>
<tr>
<td>age</td>
<td>title</td>
<td>email</td>
</tr>
<tr>
<td>GPA</td>
<td>unit</td>
<td></td>
</tr>
</tbody>
</table>
Entity Set

- Entities with attributes can be thought as “tuples” (or records)
  - (301, John, 13 Hilgard, 18, 3.3), (303, James, 12 De Neve, 19, 2.5), ...

- Key: a set of attributes that uniquely identifies an entity in an entity set
  - Underline in E/R
  - **All entity sets** in E/R need a key

<table>
<thead>
<tr>
<th>Students</th>
<th>Classes</th>
<th>Faculty</th>
</tr>
</thead>
<tbody>
<tr>
<td>sid</td>
<td>dept</td>
<td>name</td>
</tr>
<tr>
<td>name</td>
<td>cnum</td>
<td>title</td>
</tr>
<tr>
<td>addr</td>
<td>sec</td>
<td>office</td>
</tr>
<tr>
<td>age</td>
<td>title</td>
<td>email</td>
</tr>
<tr>
<td>GPA</td>
<td>unit</td>
<td></td>
</tr>
</tbody>
</table>
Relationship Set

- Relationship: “connection” between entities
- Relationship set: a set of relationships of the same kind
  - Diamond in ER
  - Relationships can be thought as “edges” between entities
- Relationships can have attributes
- Not all entities have to participate in a relationship

<table>
<thead>
<tr>
<th>Students</th>
<th>Classes</th>
<th>Faculty</th>
</tr>
</thead>
<tbody>
<tr>
<td>sid</td>
<td>dept</td>
<td>name</td>
</tr>
<tr>
<td>name</td>
<td>cnum</td>
<td>title</td>
</tr>
<tr>
<td>addr</td>
<td>sec</td>
<td>office</td>
</tr>
<tr>
<td>age</td>
<td>title</td>
<td>email</td>
</tr>
<tr>
<td>GPA</td>
<td>unit</td>
<td></td>
</tr>
</tbody>
</table>

\[ (30, \text{CS}, 143, 1, A) \]
Cardinality of Relationships

• Cardinality: how many times entities participate in a relationship?
  • One-to-one
  • One-to-many
  • Many-to-many

• Cardinality: Add arrow on the “one” side

• Total participation
  • every entity participates in the relationship \textit{at least once}
  • Double line in E/R model
Meaning of Cardinality

• Q: What does it mean
  • Many-to-one in Teach?
  • One-to-one in Teach?
  • Double-line between Classes and Teach?
  • Double-line and arrow between Teach and Faculty?
  • Double lines at both sides of Teach vs one-to-one of Teach. Are they the same?
General Cardinality Notation

• Label an edge with “a..b”
  • The entity participates in the relationship between a through b times
  • * means unlimited

• Don’t get confused: for one-to-many relationship, “0..*” appears on the “one” side and “0..1” appears on the “many” side
N-ary Relationship

• We may need more than binary relationship sometimes
• Example: Students, TA’s, and Classes
  • All TA’s help all students
  • Each student is assigned to a particular TA
Roles

• We can designate a “role” to each entity set that participate in a relationship set
  • Labels on edges of a relationship in E/R model
  • Useful if an entity set participates more than once in a relationship
Superclass and Subclass

• ISA relationship in E/R connects superclass and subclass

• Notes
  • Specialization: superclass → subclass, generalization: subclass → superclass
  • Subclass inherits all attributes of its superclass
  • Subclass participates in the relationships of its superclass
  • Subclass may participate in its own relationship
  • Disjoint specialization vs overlapping specialization
    • Either-or vs multiple specialization
    • single hollow arrow vs multiple hollow arrows
Weak Entity Set

- Weak Entity Set: An entity set without a unique key
  - Double rectangle in E/R model
- Part of its key comes from one or more entity sets it is linked to
  - Owner entity set: entity set providing part of the key
  - Identifying relationship: relationship between a weak entity set and owner entity set
    - Double diamond in E/R model
    - Discriminator: attributes in a weak entity set that become part of the key
      - Dashed underline
E/R for Stores and Products

• All products are either “private-label products” (like Kirkland shoes at Costco) or “national-brand products” (like Kleenex Tissue)

• Every product is manufactured by exactly one manufacturer (like 7up by Coke company, etc.)

• Every private-label product is carried by exactly one chain store (eg, Kirkland shoes by Costco)

• Some national-brand products may not be carried by any chain store
E/R for Stores and Products

• All products are either “private-label products” or “national-brand products”
• Every product is manufactured by exactly one manufacturer
• Every private-label product is carried by exactly one chain store
• Some national-brand products may not be carried by any chain store
E/R Design Principles

• Often it is not clear what choices to make
  • One gigantic entity set with many attributes vs many smaller entity sets?
  • Attribute vs Entity set?

• General rule of thumb for good design: avoid redundancy
  • Saying the same thing more than once
  • Space waste and potential inconsistency
E/R Design Example

• Faculty(name, addr) are instructors of Class(dept, cnum, title)

• Things to consider for entity set vs attribute
  • Do we need more attributes than keys?
  • Is it one-to-one relationship?
    • Create multiple entity sets for many-to-many or many-to-one relationships
E/R to Relation

• Converting E/R diagram to tables is mostly straightforward
  • Automatic conversion tools exists
• (Strong) entity set: one table with all attributes
Example Conversion from E/R to Relation

Student (name, addr)
TA (name, addr)
Class (cnum, title)
Faculty (name, addr)
E/R to Relation

• Converting E/R diagram to tables is mostly straightforward
  • Automatic conversion tools exists

• (Strong) entity set: one table with all attributes

• Relationship set: one table with keys from the linked entity sets and its own attributes
  • If attribute names conflict, prefix them with entity set name
Example Conversion from E/R to Relation

- Student(name, addr)
- Class(cnum, title)
- TA(name, addr)
- Faculty(name, addr)
E/R to Relation

• Converting E/R diagram to tables is mostly straightforward
  • Automatic conversion tools exists

• (Strong) entity set: one table with all attributes

• Relationship set: one table with keys from the linked entity sets and its own attributes
  • If attribute names conflict, prefix them with entity set name

• Weak entity set: one table with its own attributes and keys from owner entity set
  • No table for identifying relationship set
Example Conversion from E/R to Relation

- Student(name, addr)
  - Class(cnum, title)
  - TA(name, addr)
  - Faculty(name, addr)

- Teach(name, cnum)
  - Take(Student.name, cnum, TA.name, quarter)
  - Partner(coder, tester)

ProjectReport(name, num, grade)
Submit(name, num)
Conversion of Subclass(es)

• Two popular approaches
  1. One table for each subclass with its own attributes and the key of its superclass
     • Student(name, addr)
       ForeignStudent(name, country)
       HonorStudent(name, fellowship)
  2. One gigantic table for the super class that includes all attributes
     • Student(name, addr, country, fellowship)
       • NULL values for missing attributes