Introduction to Data Management

Lecture #3
E-R Model, Continued

Instructor: Mike Carey
mjcarey@ics.uci.edu

It’s time for....

Friday Nights With Databases

Brought to you by
Today’s Reminders

- Read (and live by!) the course wiki page:
- Also follow (and live by) the Piazza page:
  - [https://piazza.com/uci/spring2018/cs122a/home](https://piazza.com/uci/spring2018/cs122a/home)
  - Everyone needs to get signed up! (~½ way there)
- The first HW assignment is now available!
  - Conceptual database design for our next-generation, cloud-based replacement for Piazza and EEE → PEEEza 😊
- And last but not least…
  - We finally landed a well-qualified Reader #3, so we may be able to open the door a bit for the discussions (upping the cap to 70 students → 420 total) if we can secure a bigger lecture hall…

More on Assignment 1

- Due date: April 13th 5PM (Friday the 13th 😊)
- Late submission: April 14th 5PM with 20% point deduction
- Use software to draw your E-R diagram in the provided template and follow the instructions carefully
- Submit your assignment to Gradescope
- Everyone should have received Gradescope (if not, join using the following code: 9ZR22G)
Another Example (Review)

(Note that we’re using M:N notation, not → s, here.)

Another Example (E’s & R’s)

Entity instance

Relationship instance
**Weak Entities**

- A *weak entity* can be identified uniquely only by considering the primary key of some other (*owner*) entity.
  - Owner entity set and weak entity set must participate in a one-to-many relationship set (one owner, many weak entities).
  - Weak entity set must have *total* participation in this *identifying* relationship set.
  - Dependent identifier is unique only *within* owner context (_ _ _ _ _ _), so its fully qualified key here is (ssn, dname)

**Ternary Relationships (and beyond)**

- A prescription is a 3-way relationship between a patient, a doctor, and a drug; with the cardinality constraints above:
  - A given patient+drug will be associated with *one* doctor (1)
  - A given patient+doctor may be associated with *several* drugs (N)
  - A given doctor+drug may be associated with *several* patients (M)

**General note:** Relationship key ≤ (entity keys)
ISA ("is a") Hierarchies

- As in Java or other PLs, ER attributes are inherited (including the key attribute).
- If we declare A ISA B, every A entity is also considered to be a B entity.
- Covering constraints: Must every Employees entity be either an Hourly_Emps or a Contract_Emps entity? (Yes or no)
  - Ex: Hourly_Emps AND Contract_Emps COVER Employees (pick 1 of 2 vs. 1 of 3)
- Overlap constraints: Can some Employees entity be an Hourly_Emps as well as a Contract_Emps entity? (Allowed or disallowed)
  - Ex: Hourly_Emps OVERLAPS Contract_Emps (else pick 1 of the 3 types)
- Reasons for using ISA:
  - To add descriptive attributes specific to a subclass.
  - To identify subclasses that participate in a relationship.
- Design: specialization (top-down), generalization (bottom-up)

Aggregation

- Used when we have to model a relationship involving (entity sets and) a relationship set.
  - Aggregation allows us to treat a relationship set as an entity set for purposes of participating in (other) relationships.
- Aggregation vs. ternary relationship:
  - Monitors is a distinct relationship; even has its own attribute here.
  - Each sponsorship can monitored by zero or more employees (as above).
Additional Advanced ER Features

- Multi-valued (vs. single-valued) attributes
  - Note: Can model (two of) these using additional entity and relationship types in vanilla E-R tools.

- Derived (vs. base/stored) attributes

- Composite (vs. atomic) attributes