Announcements
- Subscribe to the mailing list: cs23500
  - Details are on the course web page
- Office hours
  - Svetlozar: 3-4pm Tue, Thu
  - Xuehai: 5-6pm Mon, Wed
- As always, details are on the web!

The Big Picture
- Stages of building DB application: data tier
  - Real-world domain.
  - Understand client needs.
- Design data model:
  - Using entity-relationship (E/R) model.
- Database data model:
  - Using relational model.
- Create schema in DBMS, load data.

Last Time
- Entity-Relationship Model

Outline
- More design issues:
  - Subclasses,
  - Keys,
  - Weak entity sets.
- Exercise problems

Subclasses
- Subclass:
  - Special case
  - Fewer entities
  - More properties.
- Example: Ales are a kind of beer.
  - In addition to the properties (= attributes and relationships) of beers, there is a color attribute for ales.
**E/R Subclasses**

- *isa* triangles indicate the subclass relation.

![E/R Subclasses Diagram](image)

**Different Subclass Viewpoints**

- **E/R viewpoint**: An entity has a component in each entity set to which it logically belongs.
  - Its properties are the union of the properties of these E.S.
- **Object-oriented viewpoint**: An object (entity) belongs to exactly one class.
  - It inherits properties of its superclasses.

**Subclasses Example**

![Subclasses Example Diagram](image)

**Multiple Inheritance**

- Theoretically, an E.S. could be a subclass of several other entity sets.

![Multiple Inheritance Diagram](image)

**Problems**

- How should conflicts be resolved?
- Example: *manf* means grower for wines, bottler for beers. What does *manf* mean for “grape beers”?
- Need ad-hoc notation to resolve meanings.
- In practice, we shall assume a tree of entity sets connected by *isa*, with all “isas” pointing from child to parent.

**Keys**

- A *key* is a set of attributes whose values can belong to at most one entity.
  - The value of a key is *unique*.
- In E/R model, every E.S. must have a key.
  - It could have more than one key, but one set of attributes is the *designated key*.
- In E/R diagrams, you should underline all attributes of the designated key.
Example

- Suppose *name* is key for *Beers*.

  ![Beers Diagram](beers_diagram.png)

- *Beer name* is also key for *ales*.
  - In general, key at root is key for all.

Example: A Multiattribute Key

- What is the key?

  ![Courses Diagram](courses_diagram.png)

Weak Entity Sets

- Sometimes an E.S. *E*'s key comes not (completely) from its own attributes, but from the keys of one or more E.S.'s to which *E* is linked by a *supporting* many-one relationship.
- Called a *weak E.S.*
- Represented by putting double rectangle around *E* and a double diamond around each supporting relationship.
- Many-one-ness of supporting relationship (includes 1-1) essential.
  - With many-many, we wouldn't know which entity provided the key value.

Example: Email Addresses

- Email address = user name + host name, e.g., *evtimov@cs.uchicago.edu*.
- Email address corresponds to a user name on a particular host.
- Once on a host, you only need user name, e.g., *evtimov*
- Key for an email = the user name at the host (which is unique for that host only) + the IP address of the host (which is unique globally).

Example: Email Addresses

- Design issue: Under what circumstances could we simply make user-name and host-name be attributes of email, and dispense with the weak E.S.?

Example: Chain of Weakness

- Consider IP addresses consisting of a primary domain (e.g., *edu*), subdomain (e.g., *uchicago*), and host (e.g., *cs*).
Chain of Keys

- Key for primary domain = its name.
- Key for sub-domain = its name + name of primary domain.
- Key for host = its name + key of sub-domain = its name + name of sub-domain + name of primary domain.

All Connecting Entity Sets Are Weak

- In this special case, where bar and beer determine a price, we can omit price from the key, and remove the double diamond from ThePrice.
- Better: price is an attribute of BBP.

Design Principles

- Faithfulness to requirements.
  - Remember the design schema should enforce as many constraints as possible. Don't rely on future data to follow assumptions.
  - Example: If registrar wants to associate only one instructor with a course, don't allow sets of instructors and count on departments to enter only one instructor per course.

Good and Bad Design

Good?

- Beers
  - ManfBy
    - Manfs

Bad?

- Beers
  - manf
    - addr
Question: Why is it OK to have *Beers* with just its key as attribute? Why not make set of beers an attribute of manufacturers?