Conceptual Design using the Entity-Relationship Model

Overview of Database Design
- **Conceptual design:** (ER Model is used at this stage.)
  - What are the entities and relationships in the enterprise?
  - What information about these entities and relationships should we store in the database?
  - What are the integrity constraints or business rules that hold?
  - A database ‘schema’ in the ER Model can be represented pictorially (ER diagrams).
  - Can map an ER diagram into a relational schema.
- **Schema Refinement (Normalization):** Check relational schema for redundancies and related anomalies.
- **Physical Database Design and Tuning:** Consider typical workloads and further refine the database design.

ER Model Basics
- **Entity:** Real-world object distinguishable from other objects. An entity is described (in DB) using a set of attributes.
- **Entity Set:** A collection of similar entities. E.g., all employees.
  - All entities in an entity set have the same set of attributes.
  - Each entity set has a key.
  - Each attribute has a domain.

ER Model Basics (Contd.)
- **Relationship:** Association among two or more entities. E.g., Ed works in Pharmacy department.
- **Relationship Set:** Collection of similar relationships.
  - An n-ary relationship set \( R \) relates n entity sets \( E_1 \ldots E_n \); each relationship in \( R \) involves entities \( e_1 \in E_1, \ldots, e_n \in E_n \)
  - Same entity set could participate in different relationship sets, or in different “roles” in same set.

Relationships
- **Relationships can also have attributes:**
  - “since” attribute on previous slide
  - Describe neither entity alone, but rather describe the way in which the entities are related
- **Can sometimes “migrate” these to an entity:**
  - Whether we can do so depends on relative cardinalities of entities with the relationship set....
  - Just because we can doesn’t mean we should: the semantics may become muddled

Key Constraints
- **Consider Works_In:** An employee can work in many departments; a dept can have many employees.
- In contrast, each dept has at most one manager, according to the key constraint on Manages.
Participation Constraints

- Does every department have a manager?
  - If so, this is a participation constraint: the participation of Departments in Manages is said to be total (vs. partial).
  - Every Departments entity must be related to at least one Employees entity via the Manages relationship.

Cardinality Constraints

- Key and participation constraints are special cases of cardinality constraints, which restrict the number of entities a given entity can be related to.
- Example:

Weak Entities

- A weak entity can be identified uniquely only by considering the primary key of another (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.

Conceptual Design Using the ER Model

- Design choices:
  - Should a concept be modeled as an entity or an attribute?
  - Should a concept be modeled as an entity or a relationship?
  - Identifying relationships: Binary or ternary?
- Constraints in the ER Model:
  - A lot of data semantics can (and should) be captured.
  - But some constraints cannot be captured in ER diagrams.

Entity vs. Attribute

- Should address be an attribute of Employees or an entity (connected to Employees by a relationship)?
- Depends upon the use we want to make of address information, and the semantics of the data:
  - If we have several addresses per employee, address must be an entity (since attributes cannot be set-valued).
  - If the structure (city, street, etc.) is important, e.g., we want to retrieve employees in a given city, address must be modeled as an entity (since attribute values are atomic).
**Entity vs. Relationship**

- First ER diagram OK if a manager gets a separate discretionary budget for each dept.
- What if a manager gets a discretionary budget that covers all managed depts?
  - Redundancy of `dbudget`, which is stored for each dept managed by the manager.
  - Misleading: suggests `dbudget` tied to managed dept.

**Summary of Conceptual Design**

- Conceptual design follows requirements analysis,
  - Yields a high-level description of data to be stored
- ER model popular for conceptual design
  - Constructs are expressive, close to the way people think about their applications.
  - Basic constructs: entities, relationships, and attributes (of entities and relationships).
  - Some additional constructs: weak entities, ISA hierarchies, and aggregation.
  - Note: There are many variations on ER model.

**Summary of ER (Contd.)**

- Several kinds of integrity constraints can be expressed in the ER model: key constraints, participation constraints. Some foreign key constraints (next week) are also implicit in the definition of a relationship set.
  - Some constraints (notably, functional dependencies (also next week)) cannot be expressed in the ER model.
  - Constraints play an important role in determining the best database design for an enterprise.

**Summary of ER (Contd.)**

- ER design is subjective. There are often many ways to model a given scenario! Analyzing alternatives can be tricky, especially for a large enterprise.
  - Common choices include:
    - Entity vs. attribute, entity vs. relationship, binary or n-ary relationship, whether or not to use ISA hierarchies, and whether or not to use aggregation.
  - Ensuring good database design: resulting relational schema should be analyzed and refined further. FD information and normalization techniques are especially useful....