Entity Relationship Modeling

High Level Conceptual Design Language

ER Modeling

- method of analyzing requirements
- results in conceptual schema
- schema is E-R diagram
- data-model independent
- can translate into any data model
Uses of ERDs

• Verify designer’s understanding with customer.
• Basis for designing actual database

E-R diagram is a DDL

• Will describe in terms of Relational Model
• But Remember: ER is Data Model Independent
• ELEMENTS of the diagram are
  1. Entity types
  2. Relationship types
  3. Attributes
DB Design Diagrams

• DB designed from
  – Table schemas
  – Foreign keys
• DB Design Diagram just represents this graphically

Arrow: Tail is FK field, Head is target relation.

Entities, Attributes, Keys

The KEY. Underlined

Foreign Keys (DNO, SUPERSSN) not shown!
Will be represented by Relationships
Composite Keys

Key is one bubble

Two (composite) candidate keys

Note on ER Diagrams

- The Attribute Bubbles take up lots of space
- There is not a lot of space on one slide
- So this explanation will often omit some or all attributes.
Simple (1:N) Relationships

Foreign keys should not be shown. DNUM is replaced by Controls relationship.

Relationship Instances

Between one Department instance and one Project instance at most one Controls instance.
Complex (M:N) Relationships

Works_on has NO FOREIGN KEYS in ERD

Relationship Instances

No more than 1 connection between 1 emp and 1 proj
Reflexive Relationships

Neither SUPERSSN nor DNO appear in the ERD

Basic Elements: Entity Types

- Represented by Rectangles
- Have own KEYS, not derived from other types in the DB.
- Instance of Entity Type is an Entity
- Foreign Keys are NOT Attributes
Basic Elements: Relationship Types

- Represented by Diamond
- Connect Two (or more) Entity Types
- May have attributes (but not FKs)
- Instance of Relationship Type is a Relationship between two entities
- Cannot connect to other relationship types

Relationship Types (cont.)

- One Relationship Connects 2 Entities.
- Any two entities can have at most one instance of a given relationship type between them.
- That defines the relationship so there is no key.
2 entities, only one instance

- Suppose we want a history of who worked where
- Cannot use works-for relationship
- Need more complicated structure

Types and Instances

**types**
- Emp
  - Work history
  - Dept

**instances**
- Sally
  - Sect.
  - Director.
  - Research
Basic Elements: Attributes

- Represented by Bubbles
- Attach to both Entity and Relationship Types
- May also attach to other Attributes

Multivalued Attributes

- A Department has several Locations
- Represent with double bubble
Multivalued Attributes

- Each instance of the attribute may have attributes
- Which may be multivalued

Weak Entity Types

- An Employee has several Dependents
- Represent with Double Rectangle, And Defining Relationship
Weak Entity Types

- Like Regular Entity Types, BUT
- Have a defining connection with some other Entity Type,
- Get KEY from other Entity (SSN) PLUS own Partial Key (Dependent_Name)

Multi-parented Weak Entities

- May have more than one defining relationship
- Must have partial key.
Weak Entity Types and Multivalued Attributes

- Both Represented by 1:N FKs in RDB
- Weak Entities can support relationships to other entities
- Multivalued attributes cannot

Company Database as ERD