Logical Design

• “During logical design you transform the E-R diagrams that were developed during conceptual design into relational database schemas.”  p. 218

• Inputs:

• Outputs:
**Logical Design**
Transforming ER Diagrams into Relations - 1

**LOGICAL DESIGN:** “During logical design you transform the E-R diagrams that were developed during conceptual design into relational database schemas.” p. 218

- **Step 1: Map Regular Entities**
  - Regular entities become relation
  - Entity name typically becomes relation name
  - Entity identifier becomes primary key
- **Step 1a: Composite attribute**
  - Only include simple component values
- **Step 1b: Multi-valued attributes**
  - Create two new relations if entity has multi-valued attributes
    - Relation 1 – all attributes except Multi-valued attribute
    - Relation 2 – two attributes as key
      - Attribute 1 = primary key of relation 1 (as foreign key)
      - Attribute 2 = multi-valued attribute
- **Step 2: Map Weak Entities**
  - Create new relation
  - Include all simple attributes
  - Include primary key of owner relation, as foreign key
  - Primary key of this relation is
    - primary key of owner
    - Partial identifier
- **Step 3: Map Binary Relationships**
  - Map One-to-Many
    - Create relation for each entity
    - Include the primary key attribute (and attributes) of the entity on the one-side of the relationship as a foreign key in the relation that is on the many-side of the relationship
  - Map Many-to-Many
    - Create relations for both entities
    - Create additional new relation
    - Include primary key of both original relations as foreign keys in new relation
    - Primary key of new relation to be both foreign keys of original relation (plus additional attributes to ensure that key will be unique)
  - Map One-to-One
    - Create relations for both entities
    - Include the primary key of one relation as the foreign key of the other relation
Logical Design
Transforming ER Diagrams into Relations - 2

• **Step 4: Map Associative Entities**
  – Proceed as with Many to Many Binary relationship
    • If Associative Entity does not have Meaningful Unique Identifier (as in Many-to-Many relationship), primary key for new relation is the combination of the primary keys of the original relations (which are included as foreign keys)
    • If Associative Entity has Meaningful Unique Identifier – primary key of new entity is the identifier. The primary keys of related entities are included as foreign keys in new relation.

• **Step 5: Map Unary Relationships**
  – Unary One-to-Many
    • Foreign key attribute is added within the same relation that references the primary key values. **Recursive foreign key** is a foreign key in a relation that references the primary key values of that same relation.
  – Unary Many-to-Many
    • Create two relations
    • Relation 1 represents the entity type
    • Relation 2, an associative relation, represents the M:N relationship itself
    • Primary key of associative relation has two attributes, where both attributes are primary key of other relation

• **Step 6: Map Ternary (and n-ary) Relationships**
  – Create 3 relations to represent participating entities
  – Create additional associative relation.
  – Default primary key of associative relation has three primary key attributes for participating entity types
Logical Design
Transforming ER Diagrams into Relations

• **Step 1: Map Regular Entities**
  – Step 1a: Composite attribute
  – Step 1b: Multi-valued attributes

• **Step 2: Map Weak Entities**

• **Step 3: Map Binary Relationships**
  – Map One-to-Many
  – Map Many-to-Many
  – Map One-to-One

• **Step 4: Map Associative Entities**
  – Proceed as with Many-Many Binary relationship

• **Step 5: Map Unary Relationships**
  – Map Unary One-to-Many
  – Map Unary Many-to-Many

• **Step 6: Map Ternary (and n-ary) Relationships**