Objectives

• In this chapter, you will learn:
  – The main characteristics of entity relationship components
  – How relationships between entities are defined, refined, and incorporated into the database design process
  – How ERD components affect database design and implementation
  – That real-world database design often requires the reconciliation (調和) of conflicting goals

4.1 The Entity Relationship (ER) Model

• ER model forms the basis of an ER diagram
• ERD represents conceptual database as viewed by end user
• ERDs depict database’s main components:
  – Entities
  – Attributes
  – Relationships

Entities

• Refers to entity set and not to single entity occurrence
• Corresponds to table and not to row in relational environment
• In Chen and Crow’s Foot models, entity represented by rectangle with entity’s name
• Entity name, a noun, written in capital letters (大寫)
Attributes

- Characteristics (特性) of entities
- Chen notation: attributes represented by ovals connected to entity rectangle with a line
  - Each oval contains the name of attribute it represents
- Crow’s Foot notation: attributes written in attribute box below entity rectangle

Attributes (continued)

- **Required (必要)** attribute: must have a value, represented by bold font (粗體字)
- **Optional (選擇)** attribute: may be left empty
- **Domain**: set of possible values for an attribute
  - Attributes may share a domain
- Identifiers (識別碼): one or more attributes that uniquely identify each entity instance
- Composite (組合) identifier: primary key composed of more than one attribute
Attributes (continued)

• **Composite attribute** (組合屬性) can be subdivided (分解)
• **Simple attribute** cannot be subdivided
• **Single-value attribute** can have only a single value
• **Multi-valued attributes** (多重數值屬性) can have many values

M:N relationships and multi-valued attributes should not be implemented
– Create several new attributes for each of the original multi-valued attributes components
– Create new entity composed of original multi-valued attributes components

**Derived** (衍生) attribute: value may be calculated from other attributes
– Need not be physically stored within database

<table>
<thead>
<tr>
<th>TABLE 4.2</th>
<th>Advantages and Disadvantages of Storing Derived Attributes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>DERIVED ATTRIBUTE</strong></td>
<td><strong>STORED</strong></td>
</tr>
<tr>
<td><strong>Advantage</strong></td>
<td>Saves CPU processing cycles</td>
</tr>
<tr>
<td>Data access time</td>
<td>Computation always yields current value</td>
</tr>
<tr>
<td>Can be used to keep track of historical data</td>
<td></td>
</tr>
<tr>
<td><strong>Disadvantage</strong></td>
<td>Requires constant maintenance to ensure derived value is current, especially if any values used in the calculation change</td>
</tr>
<tr>
<td>Increases data access time</td>
<td>Adds coding complexity to queries</td>
</tr>
</tbody>
</table>
Relationships

- Association (關係) between entities
- **Participants (參與者)** are entities that participate in a relationship
- Relationships between entities always operate in both directions
- Relationship can be classified as 1:M or 1:1
- Relationship classification is difficult to establish if only one side of the relationship is known

Connectivity and Cardinality

- **Connectivity (連接性)**
  - Describes the relationship classification as 1-to-1, 1-to-many, or many-to-many
- **Cardinality (基數)**
  - Expresses minimum and maximum number of entity occurrences associated with one occurrence of related entity using the format \((x, y)\)
  - Established by very concise statements known as business rules

Existence Dependence

- **Existence dependence (存在相依)**
  - Entity exists in database only when it is associated with another related entity occurrence
- **Existence independence (存在獨立)**
  - Entity can exist apart from one or more related entities
  - Sometimes such an entity is referred to as a strong or regular entity
Relationship Strength

- **Weak (non-identifying) relationships** (弱關係，非確定性關係)
  - Exists if PK of related entity does not contain PK component of parent entity
- **Strong (identifying) relationships** (強關係，確定性關係)
  - Exists when PK of related entity contains PK component of parent entity

### Weak Entities

- **Weak entity** (弱實體) meets two conditions
  - Existence-dependent
  - Primary key partially or totally derived from parent entity in relationship
- Database designer determines whether an entity is weak based on business rules
**Relationship Participation**

- **Optional participation** (選擇性參與)
  - One entity occurrence *does not require* corresponding entity occurrence in particular relationship
- **Mandatory participation** (強制性參與)
  - One entity occurrence *requires* corresponding entity occurrence in particular relationship
Relationship Degree (關係程度)

- Indicates **number of entities** or participants associated with a relationship
- **Unary relationship** (一元關係)
  - Association is maintained within single entity
- **Binary relationship** (二元關係)
  - Two entities are associated
- **Ternary relationship** (三元關係)
  - Three entities are associated

---

Recursive (遞迴) Relationships

- Relationship can exist between occurrences of the same entity set
  - Naturally found within **unary relationship**

---

**Fig 4.16 The implementation of a ternary relationship**
Associative (Composite) Entities
(關聯實體、組合實體)

- Also known as bridge entities (橋接實體)
- Used to implement M:N relationships
- Composed of primary keys of each of the entities to be connected
- May also contain additional attributes that play no role in connective process
4.2 Developing an ER Diagram

- Database design is an iterative process (反覆式流程)
  - Create detailed narrative of organization’s description of operations
  - Identify business rules based on description of operations
  - Identify main entities and relationships from business rules
  - Develop initial ERD
  - Identify attributes and primary keys that adequately describe entities
  - Revise and review ERD
4.3 Database Design Challenges: Conflicting Goals

- Database designers must make design compromises
  - Conflicting goals: design standards, processing speed, information requirements
- Important to meet logical requirements and design conventions (慣例)
- Design of little value unless it delivers all specified query and reporting requirements
- Some design and implementation problems do not yield “clean” solutions
Summary

- Entity relationship (ER) model
  - Uses ERD to represent conceptual database as viewed by end user
  - ERM’s main components:
    - Entities
    - Relationships
    - Attributes
  - Includes connectivity and cardinality notations

Summary (continued)

- Connectivities and cardinalities are based on business rules
- M:N relationship is valid at conceptual level
  - Must be mapped to a set of 1:M relationships
- ERDs may be based on many different ERMs
- Database designers are often forced to make design compromises