Data Modeling and Data Models

• Data models
  – Relatively simple (logical or conceptual) representations of complex real-world data structures
  • Often graphical abstraction of a real-world object or event
  • Useful in understanding complexities of the real-world environment
  • Data modeling is an inherently iterative process

Review – the ER model

• Business processes need to be converted to entities and interactions
• These entities have some attributes which are of interest to the ‘organization’
• Interactions between the entities represent relationships
• The number of entities involved in a relationship gives us the degree of the relationship
• Finally, we are also interested in the cardinality of the relationship – how many instances of one entity can be involved with how many instances of the other entity
Entity Relationship (ER) Model

- ER Model: A detailed, logical representation of the data for an organization or business area expressed in terms of Entity Types, Relationships and Attributes
  - A CUSTOMER places an ORDER for a PRODUCT
- ER Diagram: A Graphical Representation of an ER Model

ER Diagram

Entity, Entity Type and Entity Instance

- **Entity**: A person, place, object, event, or concept in the user environment, about which the organization wishes to maintain data
- **Entity Type**: A collection of entities that share common properties or characteristics
- **Entity instance**: Single occurrence of an Entity type
  - Example: Suppose Susan and John are employees of a company
  - Both Susan and John are entity instances
  - They belong to entity type EMPLOYEE
Examples of Entity Types

Ex:
- **Person**: Employee, student, patient
- **Place**: Station, region, country
- **Object**: Machine, building, automobile
- **Event**: Sale, registration, renewal
- **Concept**: Account, course, workcenter

Attributes and Identifiers

**Attribute**: A named property or characteristic of an entity type that is of interest to the organization.

**Identifier/Primary Key**: An attribute (or set of attributes) that uniquely identifies individual instances of an entity type.

Strong and Weak Entity Types

**Strong entity**
- **Name**
- **Address**
- **Date_of_Birth**
- **Student_ID**

**Weak entity**
- **Has**
- **Employee_ID**
- **Employer-name**
- **Dependent-name**
- **Date-of-birth**
- **First-name**
- **Middle-name**
- **Last-name**
Simple and Composite Attributes

**Simple Attribute:** Cannot be broken down into smaller components

**Composite Attribute:** An attribute that can be broken down into its component parts.

```
STUDENT
  +------------------+
  | Student_ID       |
  +------------------+
  | Name             |
  | First            |
  +------------------+
  | Last             |
  +------------------+
  | DateOfBirth      |
  +------------------+
  | Address          |
  +------------------+
  | Street           |
  +------------------+
  | City             |
  +------------------+
  | State            |
  +------------------+
  | Zip              |
```

Single and Multi-valued attributes

**Single-valued Attribute:** Attribute with a single value

**Multi-valued Attribute:** May take on more than one value for a given entity instance.

```
STUDENT
  +------------------+
  | Student_ID       |
  +------------------+
  | Name             |
  +------------------+
  | DateOfBirth      |
  +------------------+
  | Address          |
  +------------------+
  | CoursesTaken     |
  +------------------+
```

This creates a problem

Derived Attributes

- **Derived Attribute:** An attribute whose value can be derived from related attribute values.

```
STUDENT
  +------------------+
  | Student_ID       |
  +------------------+
  | Name             |
  +------------------+
  | DateOfBirth      |
  +------------------+
  | Address          |
  +------------------+
  | Age              |
```

What do you think we do with derived attributes?
Relationships

• An association between instances of one or more entity types that is of interest to the organization (usually a VERB)

Degree of a Relationship

• The number of entity types that participate in a relationship
  – Unary: (degree 1) - Also called “Bill of Materials” or “Recursive”
  – Binary: (degree 2) - Most common
  – Ternary: (degree 3)

Examples of Relationship Degrees

Unary Relationship

Binary Relationship
Examples of Relationship Degrees

**Ternary Relationship**

```
VENDOR  Ships  WAREHOUSE
```

**Associative Entity**

- An entity that associates the instances of one or more entity types and contains attributes that are peculiar to the relationship between those entity instances.

Summary of symbols
Cardinality

- The number of instances of one entity that can (or must) be associated with each instance of another entity

**Places**

- **CUSTOMER**
- **ORDER**

Both mandatory cardinalities

**Is Assigned to**

- **EMPLOYEE**
- **COURSE**

One optional, other one mandatory cardinality

Cardinality

Maximum and Minimum Cardinality:

Optional and Mandatory:

- Optional One (0 or 1)
- Mandatory One (exactly 1)
- Optional Many (≥ 0)
- Mandatory Many (> 0)

ER Modeling Strategy/Algorithm

a. Identify the main entity types.
b. Identify the main relationship types between the entity types described in (a), and represent each relationship as an ER diagram.
c. Determine the multiplicity constraints for each relationship described in (b). Represent the multiplicity for each relationship in the ER diagrams created in (b).
d. Identify attributes and associate them with entity or relationship types. Represent each attribute in the ER diagrams created in (c).
e. Determine candidate and primary key attributes for each (strong) entity type.
Some observations

- Complex business systems can have hundreds of entities
  - Each entity can have several attributes
- The end result may be an ER diagram that is extremely confusing for the end user
- We might want to reduce the number of entities – how?
  - Do some entities have attributes in common? If so, can we generalize into “sets” of entities?
  
  **Enhanced ER = EER**
  - Borrows ideas and terminology from OO methodologies that enable us to synthesize information at an aggregate level