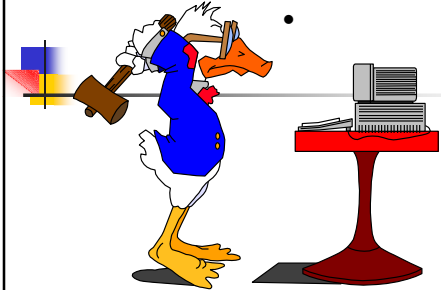


Data Modeling Using the Entity-Relationship Model



Introduction

- Once all the requirements have been collected and analyzed,
- the next step is to create a conceptual schema for the database, using a high-level conceptual data model.
- This step is called conceptual design.

Introduction

- The conceptual schema is a concise description of the data requirements of the users and
- includes detailed descriptions of the entity types, relationships, and constraints;
- these are expressed using the concepts provided by the high-level data model.
- Because these concepts do not include implementation details, they are usually easier to understand and can be used to communicate with nontechnical users.

Introduction

- The high-level conceptual schema can also be used as a reference to ensure that
- all users' data requirements are met and that the requirements do not conflict.
- This approach enables the database designers to concentrate on specifying the properties of the data, without being concerned with storage details. Consequently,
- it is easier for them to come up with a good conceptual database design

- ER modeling process is independent of the development platform (or software)
 - Enables unambiguous, accurate communication of understanding of the data resource in an *abstract level*
- It is used for communications between database designer & users during system analysis & design process

Top-down Approach

- identify data entities
- determine attributes of the entities
- determine the nature of the relationships

Top-down Approach

- usually results in a data model that is well organized
- but details can be easily overlooked.

E-R Modeling

- E/R model consists of:
 - Entity type
 - Attribute type
 - Relationship type

Entity Type

- A collection of objects (or a concept) that is identified by the enterprise as having an independent existence & share common properties (characteristics).
-

Entity Type

- Most entities are recognizable business concepts, either concrete or abstract, about which various data are stored.
- .

Entity Type

- In our usage, the term 'entity' will be synonymous with the terms 'entity type' or 'entity class'.
- it refers to the generalization of occurrences

Entity Occurrence or an instance

- An object or concept that is uniquely identifiable.
- Entity occurrence: Mr Mufungulwa is a lecturer at the Copperbelt University
- Entity type: Lecturer

Weak & Strong Entity

- Weak Entity Type
- An entity type that is existence-dependent on some other entity type.
- Children of an employee in a company DB.

Strong Entity Type

- An entity type that is not existence-dependent on some other entity type.
- An employee entity type in a company DB.

Attributes

- An attribute is a piece of information at the atomic level - that is it cannot be subdivided into meaningful component pieces
- Property of an entity that is of interest to the organization
-

Attributes

- We are usually only interested in a subset of an entity's attributes which is directly related to the application
- it is a good practice to have for each attribute a brief description
-

Attributes

- Registration Number: "The unique identifier assigned by Driver & Vehicle Licensing Agency to a vehicle driven on the public roads in Zambia."

Attributes

- Attributes take on particular values in occurrences
- E.g. VehicleID of Car
- Each entity will have one (or more) attribute that distinguishes it from all other entities, called an *Identifier* or a *Primary Key*
-

Attributes

Where two or more attributes comprise the identifier it is called a *composite identifier*

- E.g. Patient_Id (Patient_Name, Date_Of_Birth)
- An entity type may have more than one key (alternative keys)
- E.g. Car Registration No., Vehicle Identification No

Types of Attributes

- Simple:
 - Each entity has a single atomic value for the attribute, for example SSN, CourseNo.
- Composite:
 - The attribute may be composed of several components
-

Types of Attributes

- Multivalued:
 - An entity may have multiple values for that attribute; for example Color of a Car or PreviousDegrees of a Student
- Derived:
 - The domain value of attribute can be determined from one or more

Nested attribute

- In general, composite and multiple-valued attributes may be nested arbitrarily to any number of levels
- although this is rare.

Relationship

- Relationship = An association among entities
- STUDENT Taught by LECTURER

Unary or Recursive Relationships

- **A relationship where the same entity participates more than once in a different roles.**
- PERSON Married to PERSON
- Staff **Supervises** Staff

Participation & Structural Constraints

- Two types:
 - cardinality and participation constraints.
- Cardinality Constraints (Ratio)
 - Determines the number of possible relationships for each participating entity.

Participation & Structural Constraints

- The number of allowed instances of entity B that can (or must) be associated with each instance entity A.
- Most common degree for relationships is binary with cardinality ratios of one-to-one (1:1), one-to-many (1:M) or many-to-many (M:N).





Participation Constraints





- is about the importance of the instances' *participation* in a specific relationship.
 - If it is applied to every instance of an entity, then it is called a *total* or a *mandatory participation*
 - (it is a *must* for each instance belongs to that entity type).

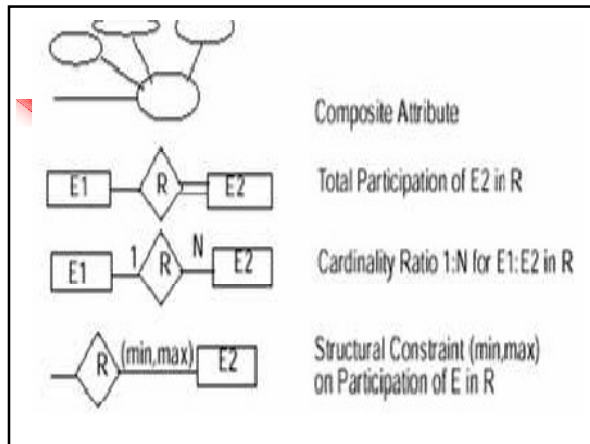
Participation Constraints

- If only parts of the instances participate in a relationship or in other words, an instance may or may not participate in that relationship, then this is called a *partial* or *optional participation*.

ER-diagram Chen Notation

	Entity
	Weak Entity
	Relationship Type
	Identifying Relationship Type

	Attribute
	Key Attribute
	Multivalued Attribute
	Derived Attribute



AN EXAMPLE DATABASE APPLICATION

- Lets consider an example database application, called COMPANY,
- it serves to illustrate the basic ER model concepts and their use in schema design.
- We first list the data requirements for the database here, and then create its conceptual schema step by step as
- we introduce the modeling concepts of the ER model

