CC5 - INFORMATION MANAGEMENT

- Roles of Database Administrator and Data Architect
- Data Manipulation
- Database Design Process
- Entities, Instances, Attributes

What's a Database Administrator?

Database Administrator

- **Schema** Definition
- Storage **structure** and **access method** definition
- Schema and Physical-organization **modification**
- **Security Management**
- **Routine Maintenance**.
  - Periodic backup
  - Disk Space management
  - Performance

What’s a Database Architect?

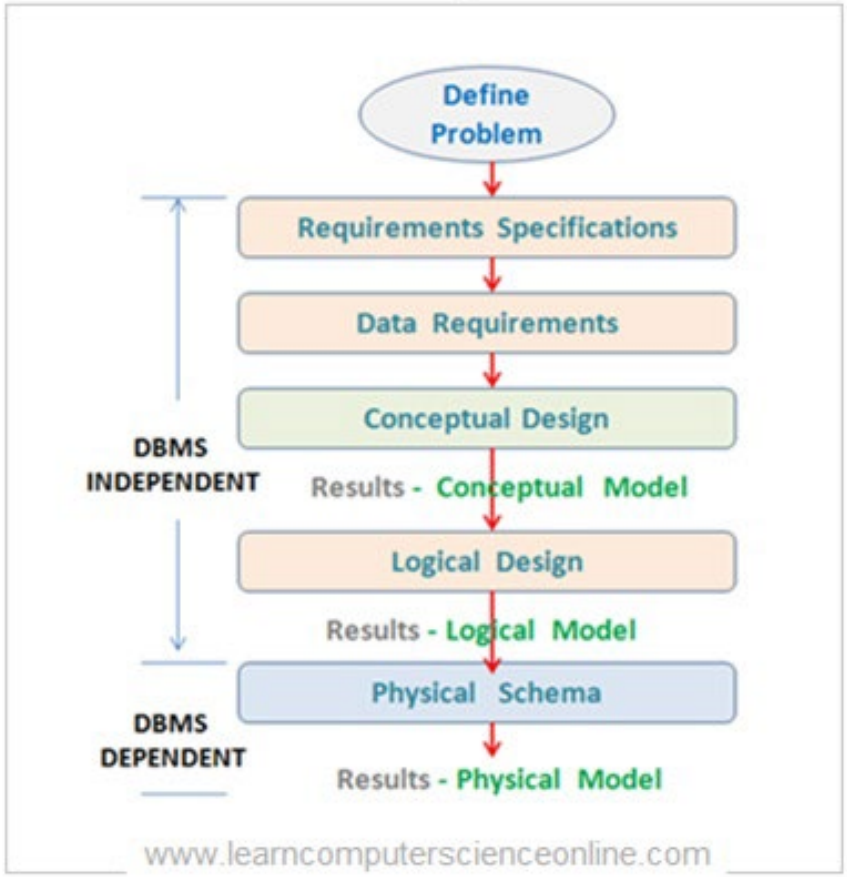
Database Architect

- Database Design – conceptual, logical, and physical designs of the database systems.
- Data Modeling - define how data is stored, accessed, and managed, ensuring alignment with business requirements.
- Integration and Scalability - integrating databases with **applications** and optimizing them for **scalability and efficiency**.
- Technical Guidance - provide **technical support** and guidance to **development teams**, ensuring **best practices** in database design and implementation

UNIT 2: DATABASE DESIGN

- PROCESS
- CONSIDERATION
- DATA MODELLING
- ERD NOTATION

Database Design - SDLC



DATABASE PROCESS

1. REQUIREMENTS COLLECTION AND ANALYSIS

- Documentation of requirements

*** FUNCTIONAL REQUIREMENTS**

- User-defined operations (retrieving and updating)

DATABASE PROCESS

2. CONCEPTUAL DESIGN

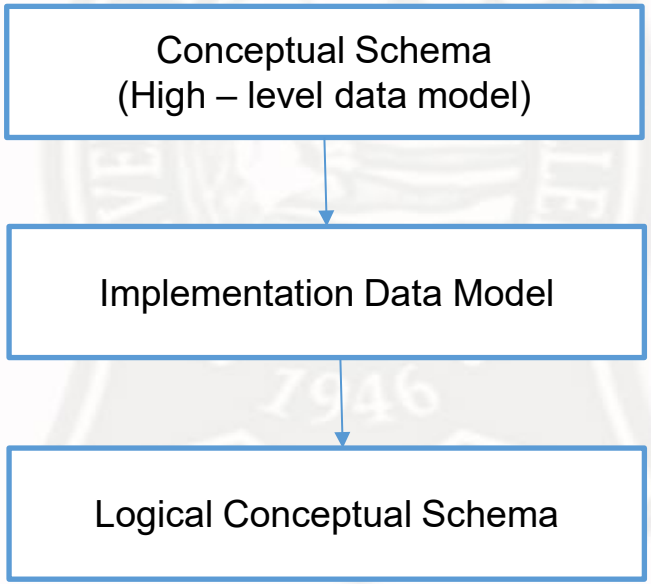
- Creating **conceptual schema (in a high-level data model)**

  - concise description of the data requirements & detailed description of the entity types, relationships & constraints.

DATABASE PROCESS

3. LOGICAL DESIGN (Data Model Mapping)

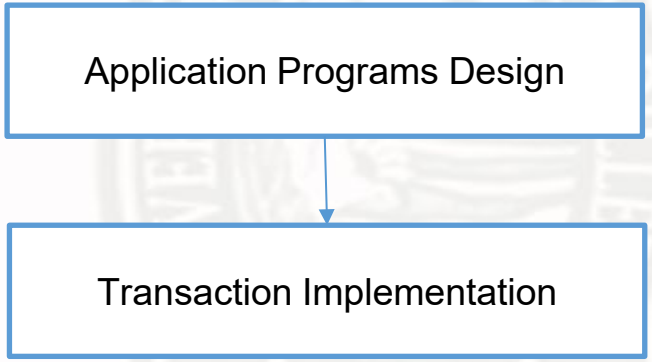
- **Actual Implementation of the database**, using commercial DBMS.



DATABASE PROCESS

4. Physical Design

- Internal storage structures, indexes, access paths -> specified



* Corresponds to the high level transaction specification

OBJECTIVES:

01 **Entities, Instances, Attributes, and Identifiers**

1.1: Entities

1.2: Attributes

1.3: UID

02 **Conceptual and Physical Models**

03 **Entity Relationship Modeling and ERDs**

3.1: Definition

3.2: Goals

3.3: ERD Introduction

Entities

- “Something” of significance must be known to the business about which data must be known
- A name for a set of similar things that you can list

Entities

- Usually a noun
- Entities have instances.

Instances

- A more specific example of an entity

Entities with Instances

Entities	Instances
PERSON	Mahatma Gandhi, George Washington
PRODUCT	Nike Air Jordan, Gibson Les Paul Custom
PRODUCT TYPE	Shoe, Video Game
JOB	Electrician, IT Technician
SKILL LEVEL	Beginner, Expert
CONCERT	U2 at the Palladium, Beyoncé at the Greek Theatre L.A.
ANIMAL	Dog, Cat
CAR	Volkswagen Beetle, Toyota Corolla

Entities with Instances

- If a dog is an instance of ANIMAL, can a dalmation, golden retriever, or pug be an instance of DOG?

Attribute

- Like an entity, an attribute represents something of significance to the business.
- Attributes have values.

Attribute

- An attribute value can be a number, character string, a date, an image, a sound, etc.
- These are called "data types" or "formats." Every attribute stores one piece of data of one specific data type.

Attribute

Entities	Attributes
CUSTOMER	family name, age, shoe size, town of residence, email
CAR	model, weight, catalog price
ORDER	order date, ship date
JOB	title, description
TRANSACTION	amount, transaction,
EMPLOYMENT CONTRACT	start date, salary

Attribute

- Some attributes (such as age) have values that constantly change. These are called volatile attributes.

Attribute

- Other attributes (such as order date) will rarely change, ever. These are nonvolatile attributes.

Attribute

- If given a choice, select the nonvolatile attribute.
- For example, use birth date instead of age.

Attribute

- Some attributes must contain a value—these are mandatory attributes.
- Other attributes may either contain a value or be left null— these are optional attributes.

UID

- A UID is either a single attribute or a combination of multiple attributes that distinguishes one instance from another.

Physical Model

- Is a product based from the conceptual model
- Addresses the implementation of a blueprint

Conceptual Model

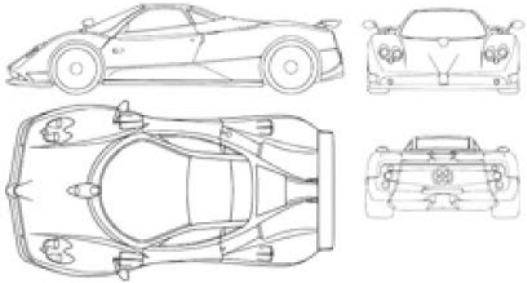
- Captures the functional and informational needs of a business
- Is based on current needs but it may reflect future needs

Conceptual Model

- Answers the needs of a business but does not address its implementation

2.1 Conceptual and Physical Models

Conceptual Model



Physical Model

Entity Relationship Modeling

- Is a conceptual model
- Is an "Implementation-Free Model" - stays the same regardless of the type of database the system implemented on

Entity Relationship Modeling

- Is a list of all entities and attributes as well as all relationships between the entities that are of importance.
- Provides background information such as entity descriptions, data types, and constraints.

ERM Goals

- Capture all required information
- Ensure that information appears only once
- Model no information that is derivable from other information already modeled
- Locate information in a predictable, logical place

Business Scenario

- Our company is divided into departments. Each employee reports to a department -- for example, accounting, sales, or development. We need to know the department responsible for each employee and the department location. Each department has a unique number. Some of the employees are managers. We need to know each employee's manager and all of the employees that are managed by each manager.

Business Scenario

1. Define the possible Entities
2. For each entity, list down their possible attributes