The Database Design Process

Design Phases:

1. Requirements collection and analysis.

The database designer interviews prospective database transaction needs users to understand and document their data and transaction needs. The outcome of this phase is a specification of data and functional requirements.
The data model designer chooses a data model.

The concepts of the data model are used to specify high-level transactions and constraints. It includes descriptions of the data types, relationships, and constraints.

The schema provides a detailed overview of the enterprise. The schema conceptualizes the enterprise into a conceptual schema of the database.

The data model operations are used to specify high-level transactions.
The conceptual schema is mapped onto the implementation data model. It includes the organization and the internal storage structure.

4. Physical design

3. Logical design
Consider part of a savings-bank enterprise.
The bank monitors the assets of each branch.
Identified by a unique name.
Each branch is located in a particular city and is
organized into branches.

The major requirements for the banking enterprise
accounts.
It keeps information about all customers and savings.

Case Study 1: A Banking Enterprise
Bank employees are also identified by their social-security number.

The bank stores the name and phone number of each employee, the names of employee’s dependents, and the employee’s start date.

It also stores the social-security number of the employee’s manager.

A customer may be associated with a loan officer.

Customers may have accounts, and can take out loans.

The bank stores each customer’s name, and the street and city where the customer lives.

Bank customers are identified by their social-security number.
The bank offers two types of accounts:

- Each savings account has an interest rate, and overdrafts are recorded for each checking account.
- Each savings account has an interest rate, and overdrafts are recorded for each checking account.

Each account was accessed by each customer holding the account, and the most recent date on which the account was accessed by each customer holding the account.

The bank maintains a record of each account’s balance.

Each account is assigned a unique account number.

Accounts can be held by more than one customer, and a customer can have more than one account.

Each savings account has an interest rate, and overdrafts are recorded for each checking account.

The bank maintains a record of each account’s balance.

Each account is assigned a unique account number.

Accounts can be held by more than one customer, and a customer can have more than one account.
A loan originates at a particular branch and can be held by one or more customers.

A loan is identified by a unique number.

For each loan, the bank keeps track of the loan amount and the loan payments.

A payment number identifies a particular payment for a specific loan.

The date and amount are recorded for each payment.
The Entity-Relationship Model facilitates database design by allowing the specification of an enterprise conceptual schema, consisting of:

- entity sets
- relationship sets
- constraints on the way entities participate in relationships
Entity Sets

An entity is a thing or object in the real world that is distinguishable from all other objects.

An entity is a set of entities of the same type that share the same properties.

An entity set is a set of entities of the same type that is distinguishable from all other objects.

An entity is represented by a set of attributes.

Attributes are descriptive properties possessed by each member of an entity.

An entity is a set of attributes.
The branch entity set with attributes:

- branch-name, branch-city, and assets.

The customer entity set with attributes:

- customer-name, social-security, customer-street, and customer-city.
The employee entity set with attributes:
- employee-name,
- social-security-number,
- telephone-number,
- salary,
- manager,
- dependent-name (multi-valued),
- start-date (base),
- employment-length (derived).

Two account entity sets savings account and checking account

Attributes:
- account-number and balance;
- account with common attributes:

The employee entity set with attributes:
An employee entity, both, or neither. A person entity may be a customer entity, an employee entity, both, or neither. Entity sets do not need to be disjoint.

Loan-number, amount, and originating branch.

The loan entity set with attributes:
Simple and composite attributes:

- Amount is a simple attribute of the entity set loan.
- Address may be composed of the attributes street, city, state, and zip-code.
- Values are natural numbers.
Single-valued and multi-valued attributes

- **Derived attributes**: Values derived from other attributes.
- **Null attributes**: Values that are not applicable or are unknown.
- **Single-valued attributes**: Attributes that contain a single value for each entity.
- **Multi-valued attributes**: Attributes that contain multiple values for each entity.
- **Loan-number**: An example of a single-valued, multi-valued attribute.
- **Dependent-name**: An example of a multi-valued attribute that may have more than one value for an employee entity.
Relationships

A relationship is an association among several entities. A relationship set is a set of relationships of the same type. Between customer and loan:

- Loan-branch, a many-to-one relationship set that indicates in which branch a loan originated.
- Loan-borrower, a many-to-many relationship set.
- Loan-payment, a one-to-many relationship set from loan to payment.

Relationships of the banking enterprise:
employee entities with role indicator manager and
customer and account, indicating that a customer
owns an account and account, indicating that a customer
works-for, a one-to-many relationship set between
worker
that a bank employee can advise one or more cus-
tomers
a many-to-one relationship set, expressing that a
cust-banker, with attribute type,
customer and account, indicating that a customer
and account, with attribute access date.

dep ositor, withattribute accessdate,
A binary relationship set is of degree 2.

The number of entity sets that participate in a relationship set \( R \) is called the degree of the relationship set.

where \((E_1, E_2, \ldots, E_m)\) is a relationship.

\[
\{ \left( e_1 \in E_1, e_2 \in E_2, \ldots, e_m \in E_m \right) \mid e_1 \in E_1, e_2 \in E_2, \ldots, e_m \in E_m \} = R
\]

\[ R \subseteq E_1 \times E_2 \times \cdots \times E_m \]

A relationship set \( R \) is a subset of the entity sets \( E_1, E_2, \ldots, E_m \).

Formally, given the entity sets \( E_1, E_2, \ldots, E_m \),
Attributes

Keys

A primary key is a candidate key that is chosen by the database designer as the principal means of identifying entities within an entity set. A primary key is a set of one or more attributes that, taken collectively, allows one to identify uniquely an entity in an entity set. A superkey is a candidate key that is a set of one or more attributes, extraneous attributes, minimal superkeys are called candidate keys.

Superkeys

A superkey is a set of one or more attributes that, taken collectively, allows one to identify uniquely an entity in an entity set.

Keys

Superkeys
Let $R$ be a relationship involving entity sets $E_1, E_2, \ldots, E_m$. The set of attributes

\[ \text{primary.key}(E_1) \cup \text{primary.key}(E_2) \cup \cdots \cup \text{primary.key}(E_m) \]

is a superkey for the relationship set. An entity set that does not have sufficient attributes to form a primary key is termed a weak entity set. An entity set that has a primary key is termed a strong entity set.
The E-R diagram has the following components:

- Rectangles, which represent entity sets
- Ellipses, which represent attributes
- Diamonds, which represent relationship sets
- Lines, which link attributes to entity sets and entity sets to relationship sets
- Underlying lines, which denote primary keys

The E-R diagram has the following components:
Underline dashed lines, which denote partial keys.

Double lines, which indicate total participation of an entity in a relationship set.

Dashed ellipses, which denote derived attributes.

Double diamonds, which represent multi-valued attributes.

Double rectangles, which represent weak entity sets.
A relationship may have descriptive attributes.
The function that an entity plays in a relationship is called that’s entity’s role.
Mapping cardinalities express the number of entities to which another entity can be associated via a relationship set. For a binary relationship set \( R \) between entity sets \( A \) and \( B \), the mapping cardinality must be one of the following:

Mapping cardinalities express the number of entities to which another entity can be associated via a relationship set.
● One to one relationships

● One to many relationships
• Many to one relationships

• Many to many relationships
Consider the entity set account with attributes account-number, balance, account-numbe

An account is further classified as:

- checking-account with attribute interest-rate
- savings-account with attribute overdraft-amount

CIS 761 - Lecture 2

Extended E-R Features
In addition, the bank offers the following three types of checking accounts:

• A **standard checking account** with a $3.00 monthly service charge and 25 free checks each month.

• A **gold checking account** that requires a $1,000.00 minimum balance, pays 2% percent interest, and offers unlimited free check writing.

• A **standard checking account** with a $3.00 monthly service charge and 25 free checks each month.

The bank monitors the minimum balance and the number of checks written from an account each month.
A senior checking account for customers aged 65 years or older that has no monthly service charge, and that allows unlimited free check writing.

Yields the following entity sets:

The specialization of checking-account by account type.

- Senior with attribute date-of-birth
- Gold with attributes min-balance and interest-payment
- Standard with attribute num-checks

A record of the customer’s date of birth is associated with this type of account.

CIS 761 - Lecture 2
Specialization is the refinement from an initial entity set into successive levels of entity subgroups.

Under **generalization**, multiple entity sets are synthesized into a higher-level entity set, assuming that they have several attributes in common.

This is a top-down design process, while **specialization** is the refinement from an initial entity set into successive levels of entity subgrups.

The attributes created by specialization and generalization is attribute inheritance.

The design process proceeds in a bottom-up manner.

The attributes of the higher-level entity sets are said to be inherited by the lower-level entity sets.
E-R diagram for the banking enterprise
Aggregation

Relationships among relationships cannot be expressed in the E-R model.