Database Environment
ANSI-SPARC Three-level Architecture

External view 1

<table>
<thead>
<tr>
<th>Sno</th>
<th>FName</th>
<th>LName</th>
<th>Age</th>
<th>Salary</th>
</tr>
</thead>
</table>

External view 2

<table>
<thead>
<tr>
<th>Staff_No</th>
<th>FName</th>
<th>LName</th>
<th>Bno</th>
</tr>
</thead>
</table>

Conceptual level

<table>
<thead>
<tr>
<th>Staff_No</th>
<th>FName</th>
<th>LName</th>
<th>DOB</th>
<th>Salary</th>
<th>Branch_No</th>
</tr>
</thead>
</table>

struct STAFF {
    int Staff_No;
    int Branch_No;
    char FName [15];
    char LName [15];
    struct date Date_of_Birth;
    float Salary;
    struct STAFF *next; /* pointer to next Staff record */
};

index Staff_No; index Branch_No; /* define indexes for staff */
Database Design Methodology

Three main phases
- Conceptual database design
- Logical database design
- Physical database design

Conceptual Database Design
- The process of constructing a model of the information used in an enterprise, independent of all physical considerations.

Logical Database Design
- The process of constructing a model of the information used in an enterprise based on a specific data model (e.g. relational), but independent of a particular DBMS and other physical considerations.

Physical Database Design
- The process of producing a description of the implementation of the database on secondary storage; it describes the storage structures and access methods used to achieve efficient access to the data.

Critical Success Factors in Database Design
- Work interactively with the users as much as possible.
- Follow a structured methodology throughout the data modeling process.
- Employ a data-driven approach.
- Incorporate structural and integrity considerations into the data models.
- Combine conceptualization, normalization, and transaction validation techniques into the data modeling methodology.
- Use diagrams to represent as much of the data models as possible.
- Use a Database Design Language (DBDL) to represent additional data semantics.
- Build a data dictionary to supplement the data model diagrams.
- Be willing to repeat steps.
Methodology Overview - Conceptual Database Design

- Step 1. Build local conceptual data model for each user view
  - Step 1.1 Identify entity types
  - Step 1.2 Identify relationship types
  - Step 1.3 Identify and associate attributes with entity or relationship types
  - Step 1.4 Determine attribute domains
  - Step 1.5 Determine candidate and primary key attributes
  - Step 1.6 Specialize/generalize entity types (optional step)
  - Step 1.7 Raw Entity–Relationship diagram
  - Step 1.8 Review Local Conceptual Data Model with User

Methodology Overview - Logical Database Design for Relational Model

- Step 2. Build and Validate Local Logical Data Model
  - Step 2.1 Map Local Conceptual Data Model to Local Logical Data Model
  - Step 2.2 Derive Relations from Local Logical Data Model
  - Step 2.3 Validate Model using Normalization
  - Step 2.4 Validate Model against User Transactions
  - Step 2.5 Draw Entity-Relationship Diagram
  - Step 2.6 Define Integrity Constraints
  - Step 2.7 Review Local Logical Data Model with User

Methodology Overview - Logical Database Design for Relational Model

- Step 3. Build and Validate Global Logical Data Model
  - Step 3.1 Merge Local Logical Data Models into Global Model
  - Step 3.2 Validate Global Logical Data Model
  - Step 3.3 Check for Future Growth
  - Step 3.4 Draw Final Entity-Relationship Diagram
  - Step 3.5 Review Global Logical Data Model with Users
Methodology Overview - Physical Database Design for Relational Databases

- Step 4 Translate Global Logical Data Model for Target DBMS
  - Step 4.1 Design Base Relations for Target DBMS
  - Step 4.2 Design Integrity Rules for Target DBMS

- Step 5 Design Physical Representation
  - Step 5.1 Analyze Transactions
  - Step 5.2 Choose File Organization
  - Step 5.3 Choose Secondary Indexes
  - Step 5.4 Consider the Introduction of Controlled Redundancy
  - Step 5.5 Estimate Disk Space

- Step 6 Design and Implement Security Mechanisms
  - Step 6.1 Design User Views
  - Step 6.2 Design Access Rules

- Step 7 Monitor and Tune the Operational System