University of Central Florida
CGS 2545
Database Concepts

DEPARTMENT OF ELECTRICAL ENGINEERING & COMPUTER SCIENCE
COMPUTER SCIENCE DIVISION

Architecture

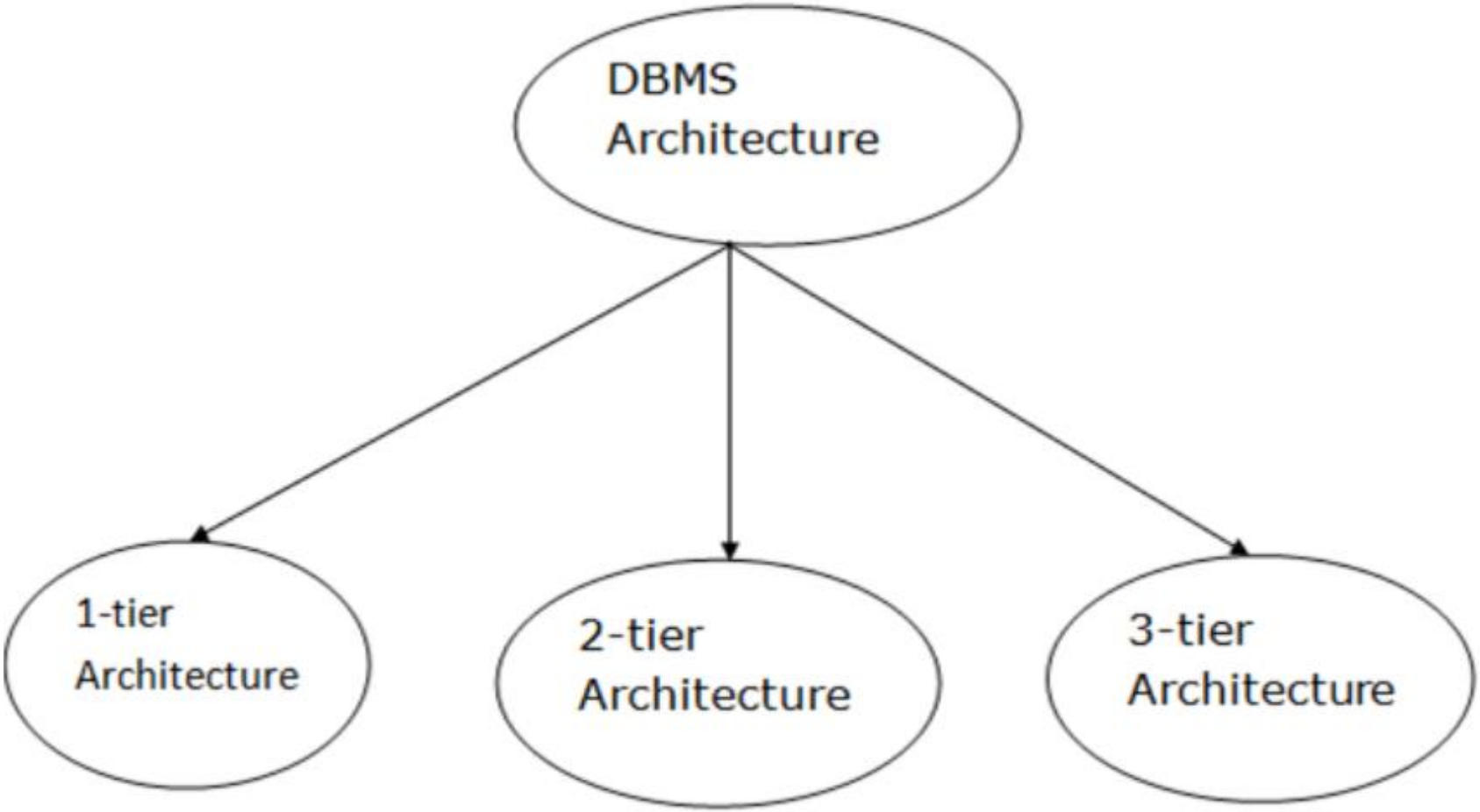
Architecture

- The design of a DBMS depends on its architecture.
- It can be centralized or decentralized or hierarchical.
- The architecture of a DBMS can be seen as either single tier or multi-tier.
- An n-tier architecture divides the whole system into related but independent **n** modules, which can be independently modified, altered, changed, or replaced.

Architecture

- The basic client/server architecture is used to deal with a large number of PCs, web servers, database servers and other components that are connected with networks
- The client/server architecture consists of many PCs and a workstation which are connected via the network
- DBMS architecture depends upon how users are connected to the database to get their request done

Architecture



Architecture

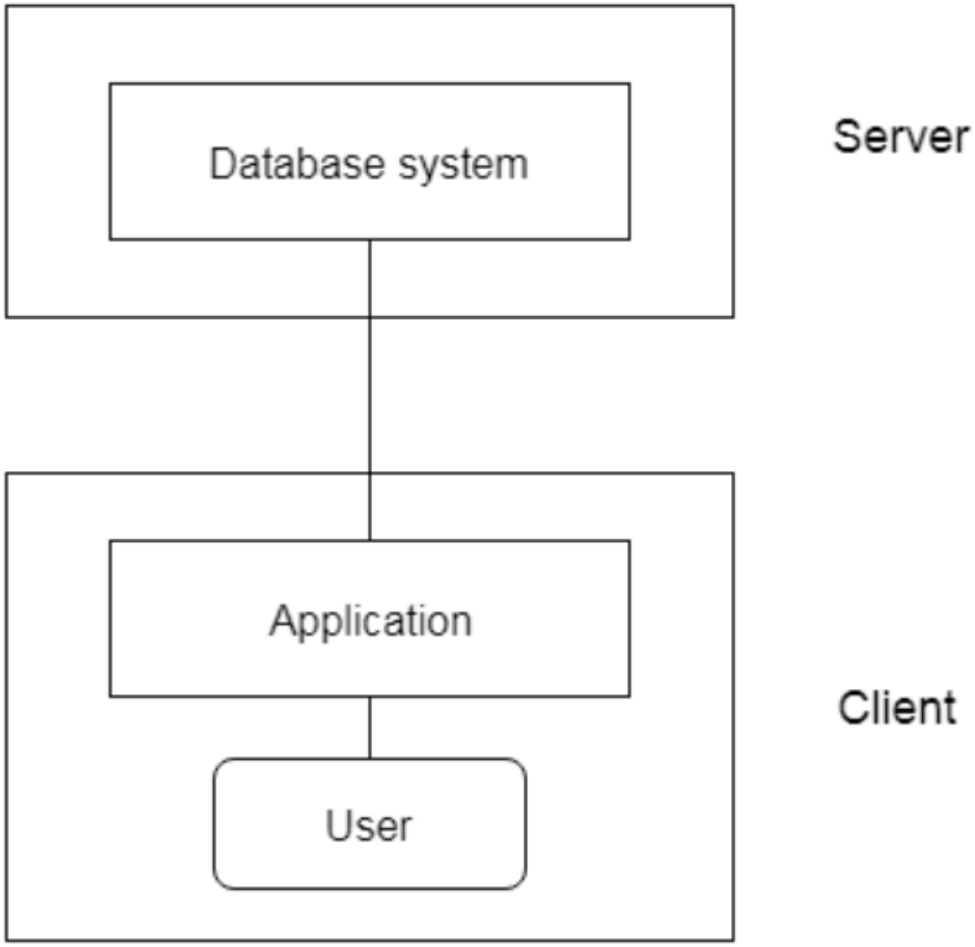
- 1-tier architecture
  - the DBMS is the only entity where the user directly sits on the DBMS and uses it.
  - Any changes done here will directly be done on the DBMS itself.
  - It does not provide handy tools for end-users.
  - Is used for development of the local application, where programmers can directly communicate with the database for the quick response
  - Database designers and programmers normally prefer to use single-tier architecture

Architecture

- 2-tier architecture
  - it must have an application through which the DBMS can be accessed
  - is the same as basic client-server architecture; applications on the client end directly communicate with the database at the server side; for this interaction, API's like ODBC or JDBC are used
  - programmers use 2-tier architecture where they access the DBMS by means of an application
  - the application tier is entirely independent of the database in terms of operation, design, and programming

Architecture

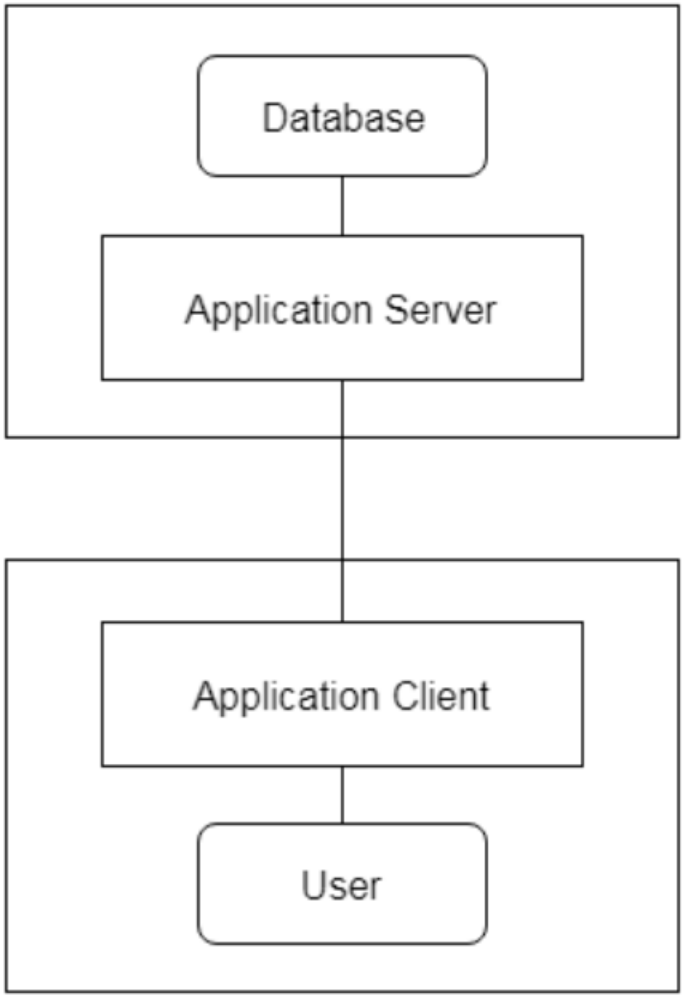
- 2-tier architecture



Architecture

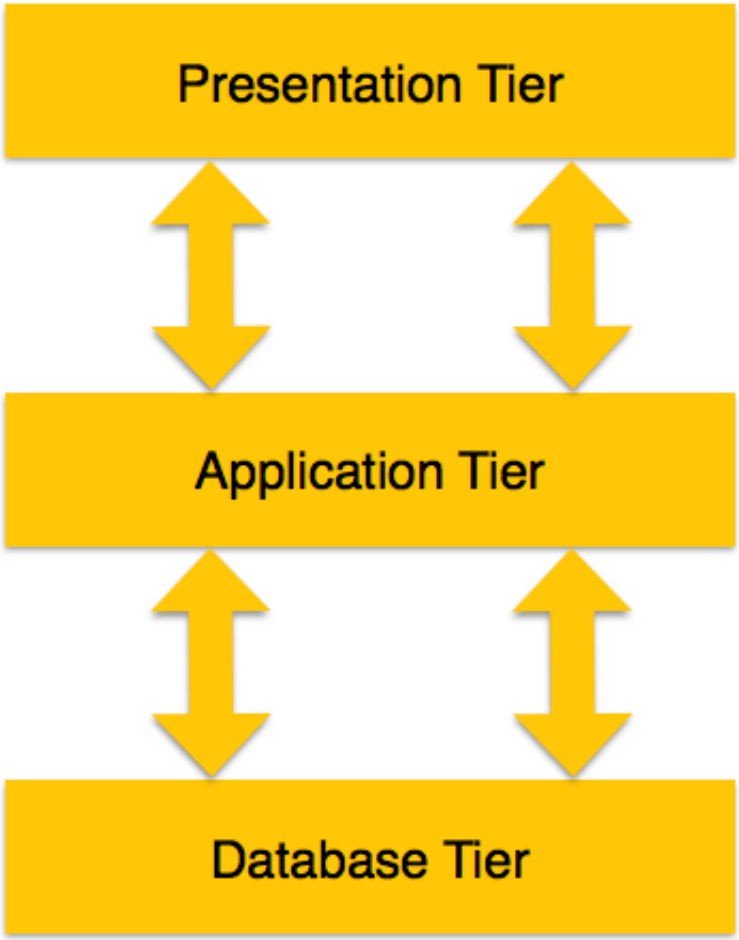
- 3-tier Architecture
  - separates its tiers from each other based on the complexity of the users and how they use the data present in the database
  - contains another layer between the client and server
  - client can't directly communicate with the server
  - the application on the client-end interacts with an application server which further communicates with the database system
  - end user has no idea about the existence of the database beyond the application server
  - the database has no idea about any other user beyond the application
  - is the most widely used architecture to design a DBMS
  - commonly used in case of large web application

Architecture



Server

Client



Architecture

- 3-tier architecture
  - **Database (Data) Tier**
    - At this tier, the database resides along with its query processing languages.
    - We also have the relations that define the data and their constraints at this level.

Architecture

- 3-tier Architecture
  - **Application (Middle) Tier**
    - At this tier reside the application server and the programs that access the database.
    - For a user, this application tier presents an abstracted view of the database.
    - End-users are unaware of any existence of the database beyond the application.
    - At the other end, the database tier is not aware of any other user beyond the application tier.
    - Hence, the application layer sits in the middle and acts as a mediator between the end-user and the database.

Architecture

- 3-tier Architecture
  - **User (Presentation) Tier**
    - End-users operate on this tier and they know nothing about any existence of the database beyond this layer.
    - At this layer, multiple views of the database can be provided by the application.
    - All views are generated by applications that reside in the application tier.

Architecture

- Multiple-tier database architecture
  - is highly modifiable
  - as almost all its components are independent and can be changed independently.