Advanced Database Management System

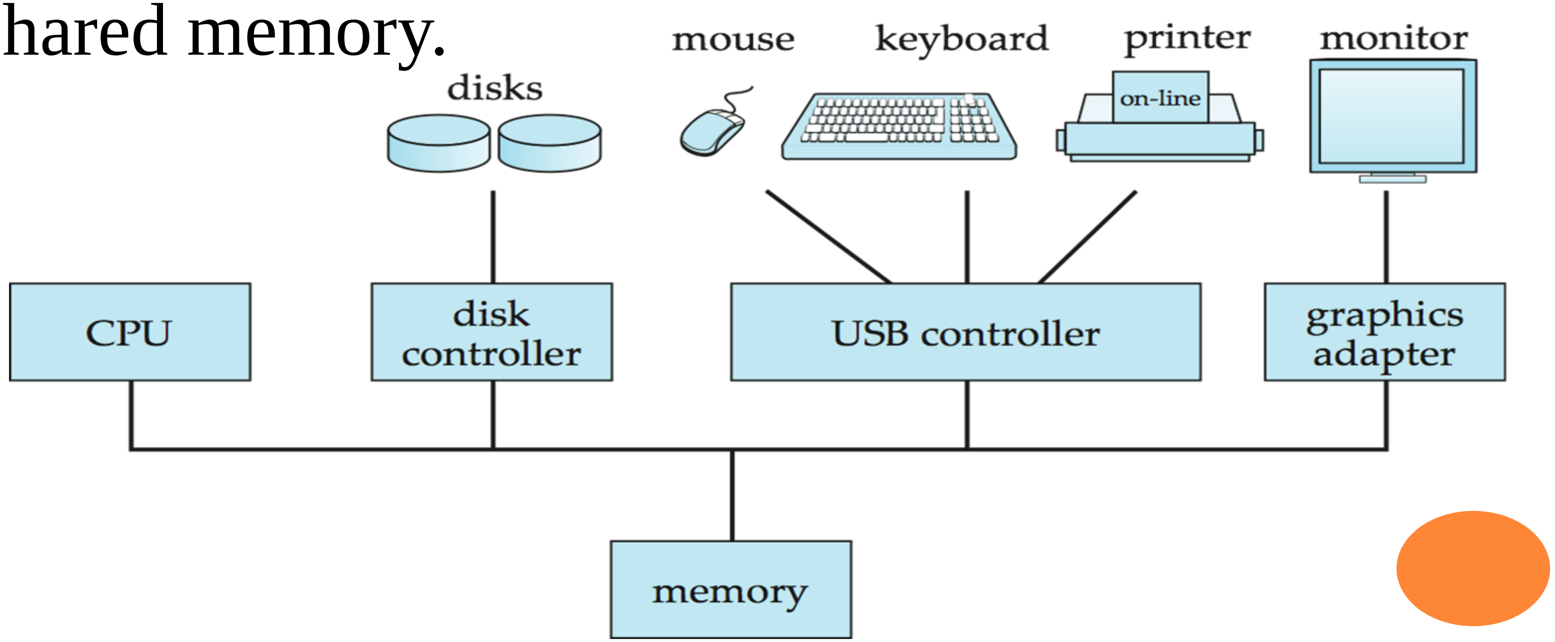
Centralized and Client/Server Architectures for DBMS

Contents

- Centralized Systems
- Centralized DBMSs Architecture
- Basic Client/Server Architectures
- Two-Tier Client/Server Architectures for DBMSs
- Three-Tier Architectures for Web Applications

Centralized Systems

- Run on a single computer system and do not interact with other computer systems.
- General-purpose computer system: one to a few CPUs and a number of device controllers that are connected through a common bus that provides access to shared memory.



Centralized Systems

- Single-user system: desk-top unit, single user, usually has only one CPU and one or two hard disks; the OS may support only one user.

- Multi-user system: more disks, more memory, multiple CPUs, and a multi-user OS. Serve a large number of users who are connected to the system vie terminals. Often called *server* systems.

CENTRALIZED DBMSS ARCHITECTURE

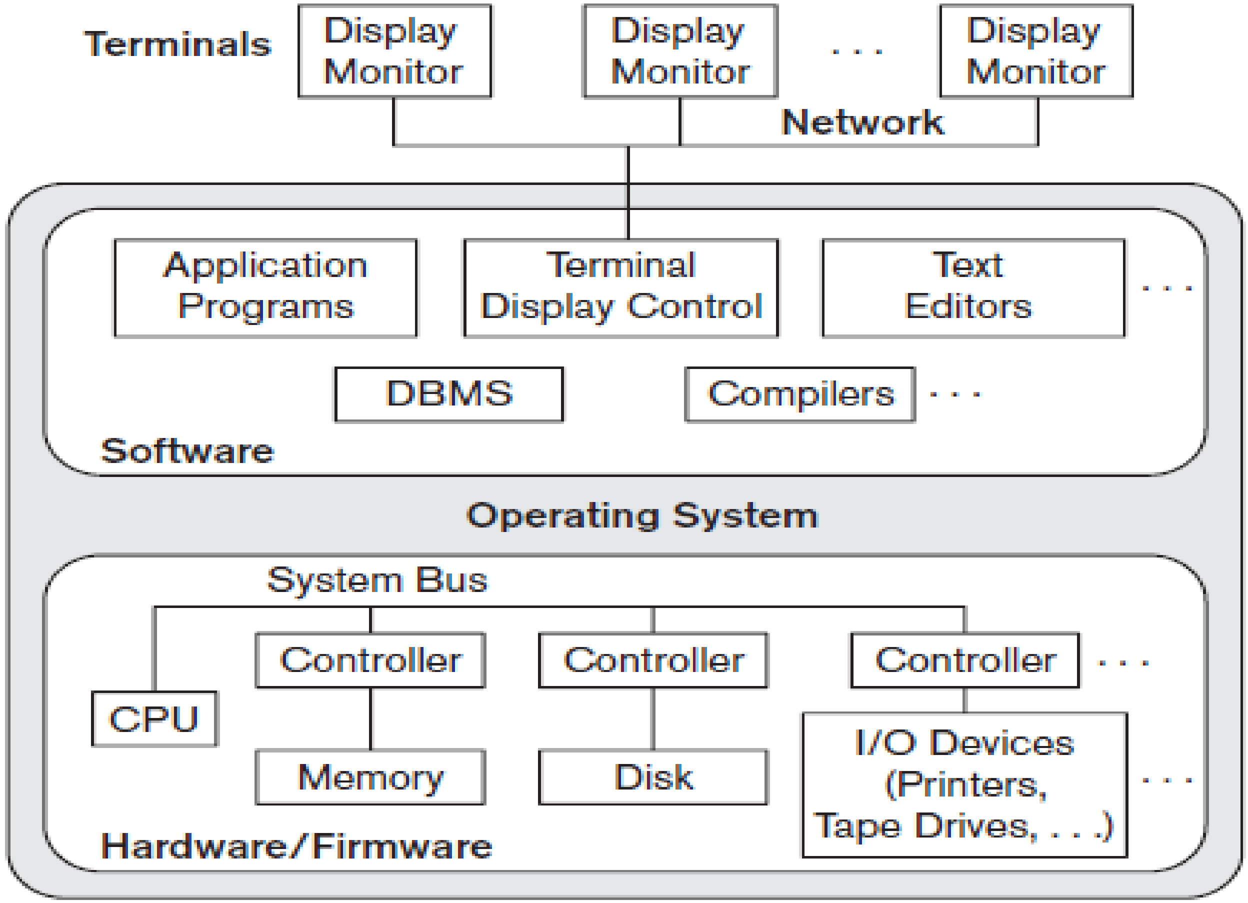
Centralized DBMS:

- Combines everything into single system including- DBMS software, hardware, application programs, and user interface processing software.

- User can still connect through a remote terminal – however, all processing is done at centralized site and user only provided display capabilities.

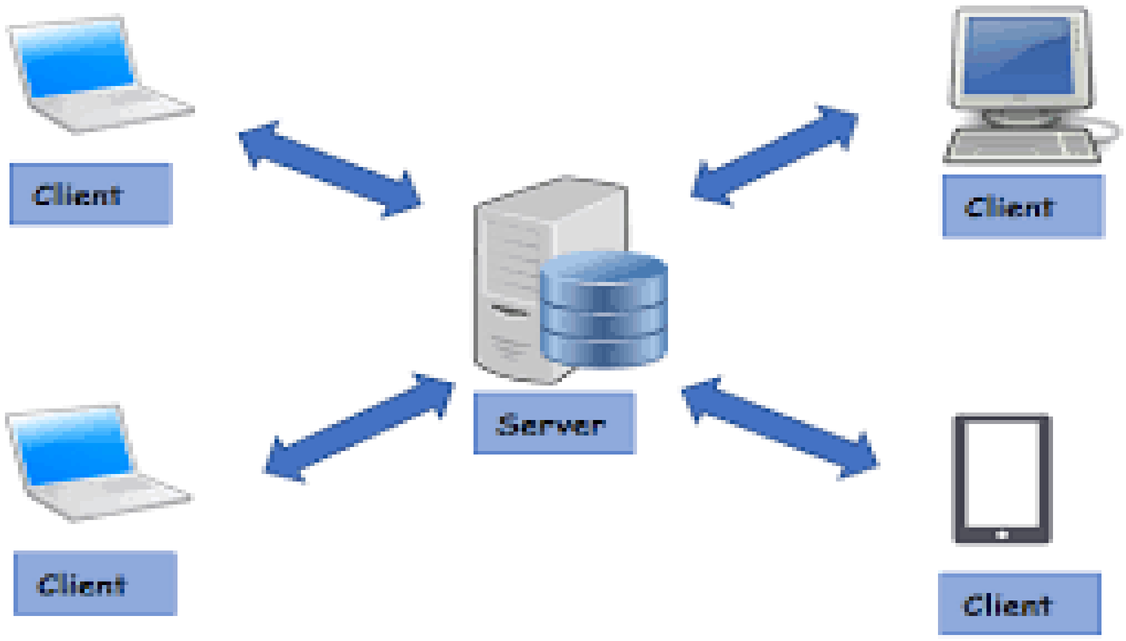
CENTRALIZED DBMSs ARCHITECTURE

Centralized DBMS:



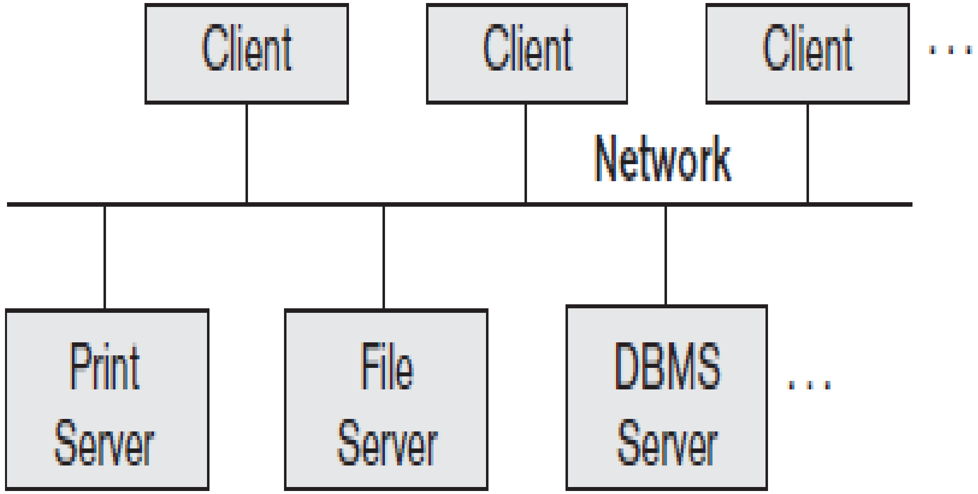
Client/Server Architectures

The **client/server architecture** was developed to deal with computing environments in which a large number of PCs, workstations, file servers, printers, data base servers, Web servers, e-mail servers, and other software and equipment are connected via a network



Client/Server Architectures

- The idea is to define specialized servers with specific functionalities
  - Print server
  - File server
  - DBMS server
  - Web server
  - Email server



- The clients provide the user with the appropriate interfaces to utilize these servers, as well as with local processing power to run local applications

Client/Server Architectures

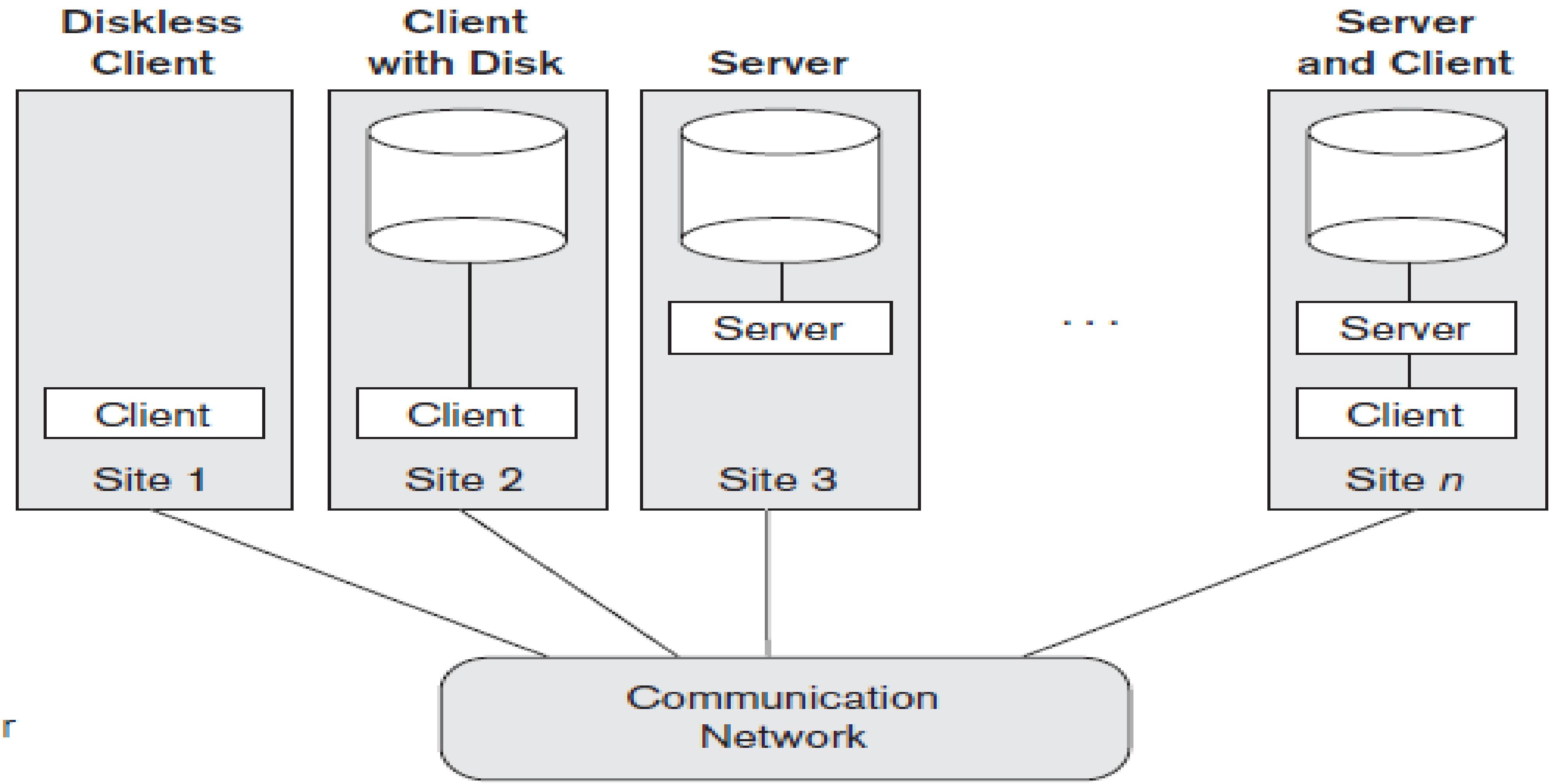
- **Clients**

- Provide appropriate interfaces through a client software module to access and utilize the various server resources.

- Clients may be diskless machines or PCs or Workstations with disks with only the client software installed.

- Connected to the servers via some form of a network.
  - (LAN: local area network, wireless network, etc.)

Client/Server Architectures



client/server

Client/Server Architectures

- **Server**

- A server is a system containing both hardware and software that can provide services to the clients, such as file access, printing, archiving, or database access.

- Some machines install only client software, others only server software, and still others may include both client and server software,

Client/Server Architectures

- **Advantages includes:**

- better functionality for the cost
- flexibility in locating resources and expanding facilities
- better user interfaces
- easier maintenance
- Increased performance
- Increased consistency

Two Tier Client-Server Architecture for DBMS

- In relational database management systems (RDBMSs), SQL created a logical dividing point between client and server

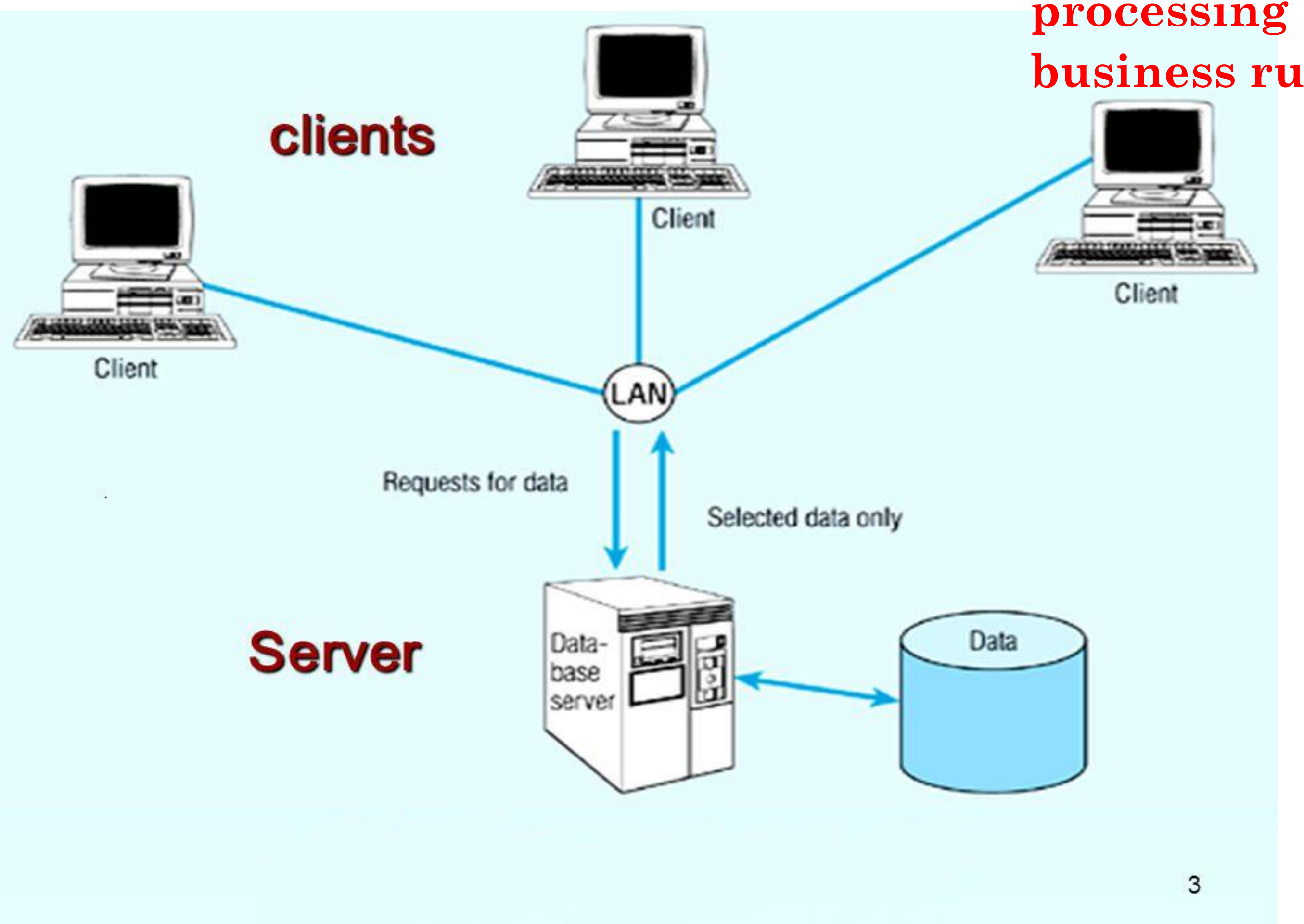
- The query and transaction functionality related to SQL processing remained on the server side

- Server is often called a query server or transaction server because it provides these two functionalities.

- In an RDBMS, the server is also often called an SQL server

Two Tier Client-Server Architecture for DBMS

Responsible for I/O processing and business rules logic



Server performs all data storage and access processing →DBMS is only on server

Two Tier Client-Server Architecture for DBMS

- Clients run user interface programs and application programs

- Client program establishes a connection to the DBMS and once the connection is created, the client program can communicate with the DBMS

- Applications running on clients utilize an Application Program Interface (API) to access server databases via standard interface such as:
  - ODBC: Open Database Connectivity standard
  - JDBC: for Java programming access

- Any query results are sent back to the client program, which can process and display the results as needed

Two Tier Client-Server Architecture for DBMS

- Client and server must install appropriate client module and server module software for ODBC or JDBC

- A client program may connect to several DBMSs, sometimes called the data sources.

- In general, data sources can be files or other non-DBMS software that manages data.

Two Tier Client-Server Architecture for DBMS

- **Object-oriented DBMSs two-tier client/server architecture:**

  - Software modules of the DBMS can be divided between client and server in a more integrated way

  - Server level may include the part of the DBMS software responsible for handling data storage on disk pages, local concurrency control and recovery, buffering and caching of disk pages

Two Tier Client-Server Architecture for DBMS

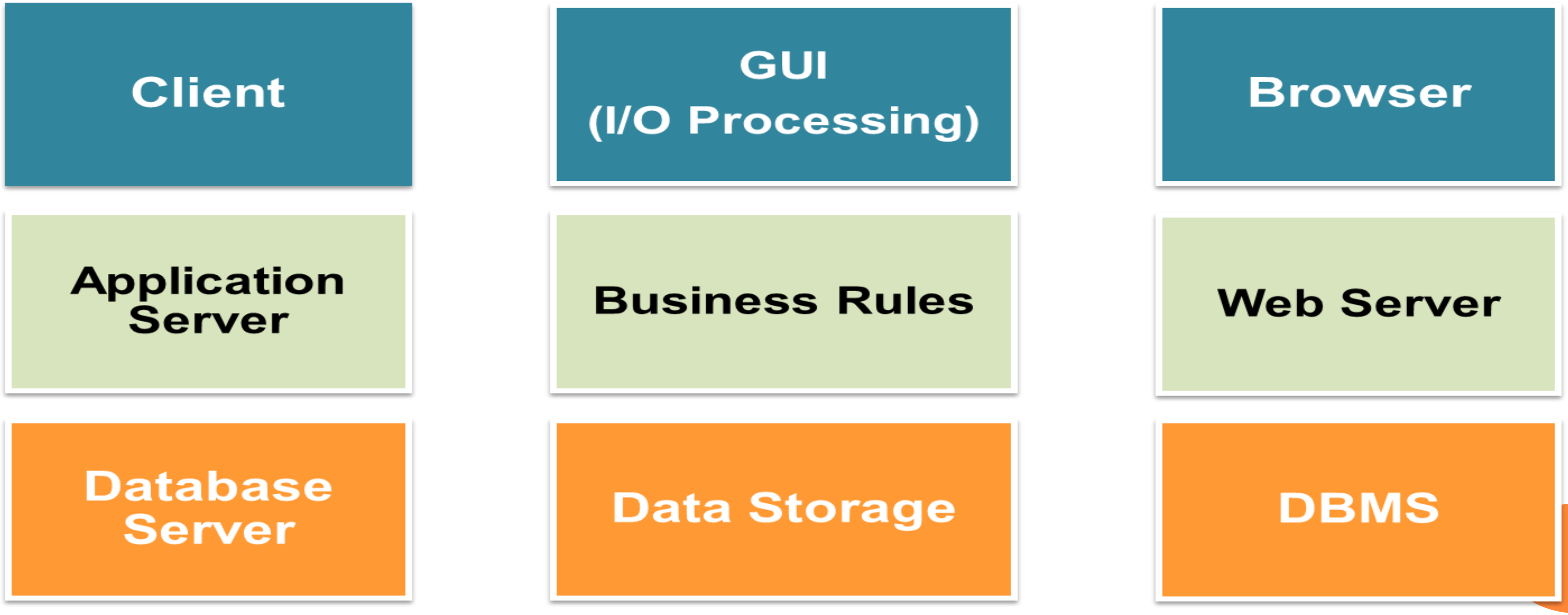
- **Object-oriented DBMSs two-tier client/server architecture:**

- Client level may handle the user interface; data dictionary functions; DBMS interactions with programming language compilers; global query optimization, concurrency control, and recovery across multiple servers; structuring of complex objects from the data in the buffers

Three Tier Client-Server Architecture

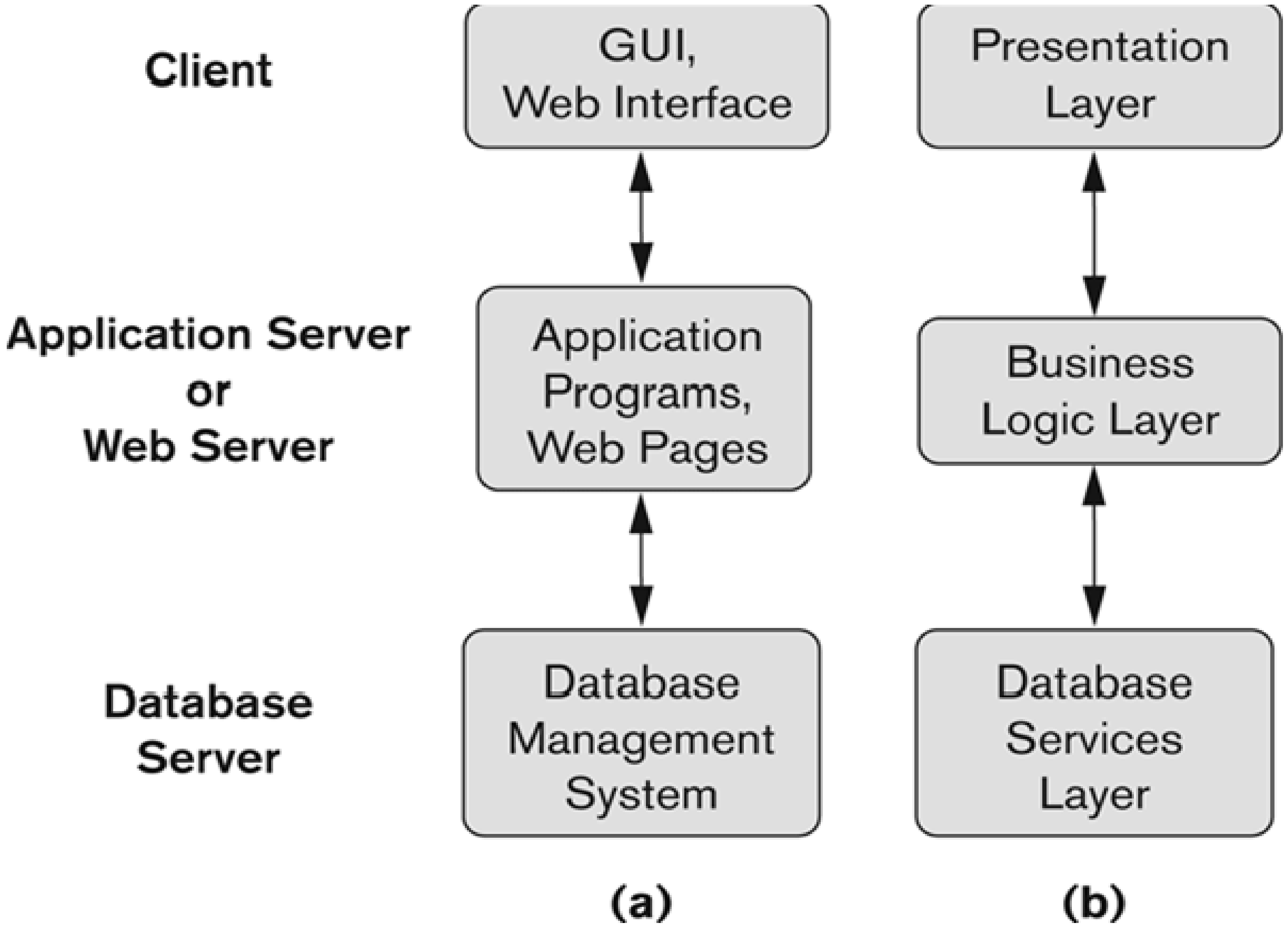
- Three-tier architecture adds an intermediate layer between the client and the database server

- This intermediate layer or middle tier is called the application server or the Web server



Client	GUI (I/O Processing)	Browser
Application Server	Business Rules	Web Server
Database Server	Data Storage	DBMS

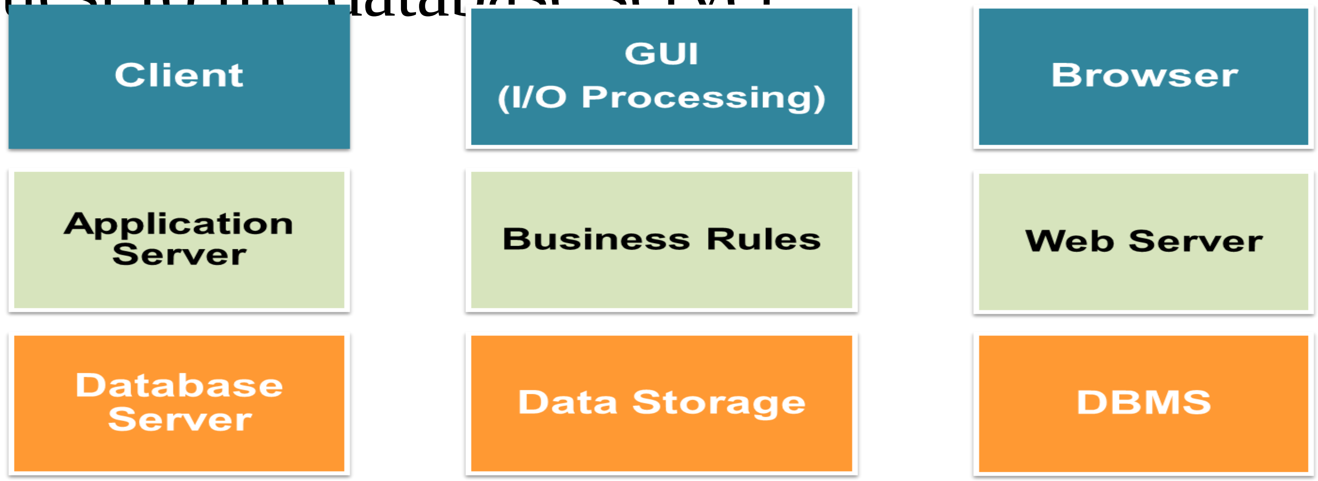
THREE TIER CLIENT-SERVER ARCHITECTURE



Three Tier Client-Server Architecture

- Web/Application server plays an intermediary role by running application programs and storing business rules (procedures or constraints) that are used to access data from the database server

- It can also improve database security by checking a client's credentials before forwarding a request to the database server

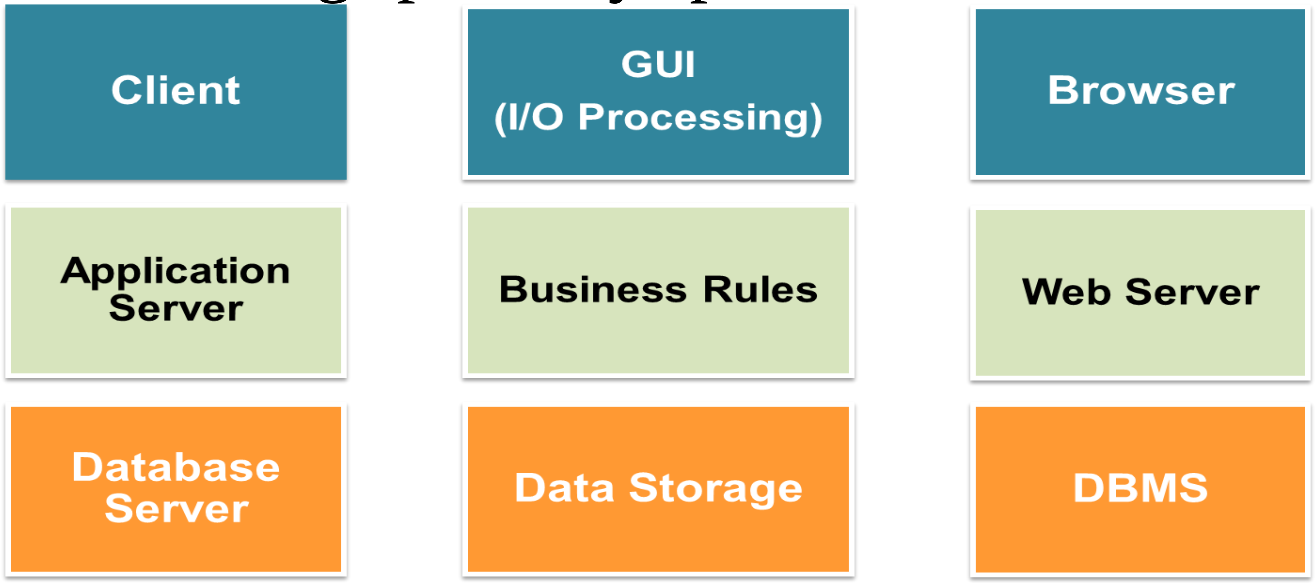


Client	GUI (I/O Processing)	Browser
Application Server	Business Rules	Web Server
Database Server	Data Storage	DBMS

THREE TIER CLIENT-SERVER ARCHITECTURE

- Stores the web connectivity software and the business logic part of the application used to access the corresponding data from the database server

- Acts like a conduit for sending partially processed data between the database server and the



Client	GUI (I/O Processing)	Browser
Application Server	Business Rules	Web Server
Database Server	Data Storage	DBMS

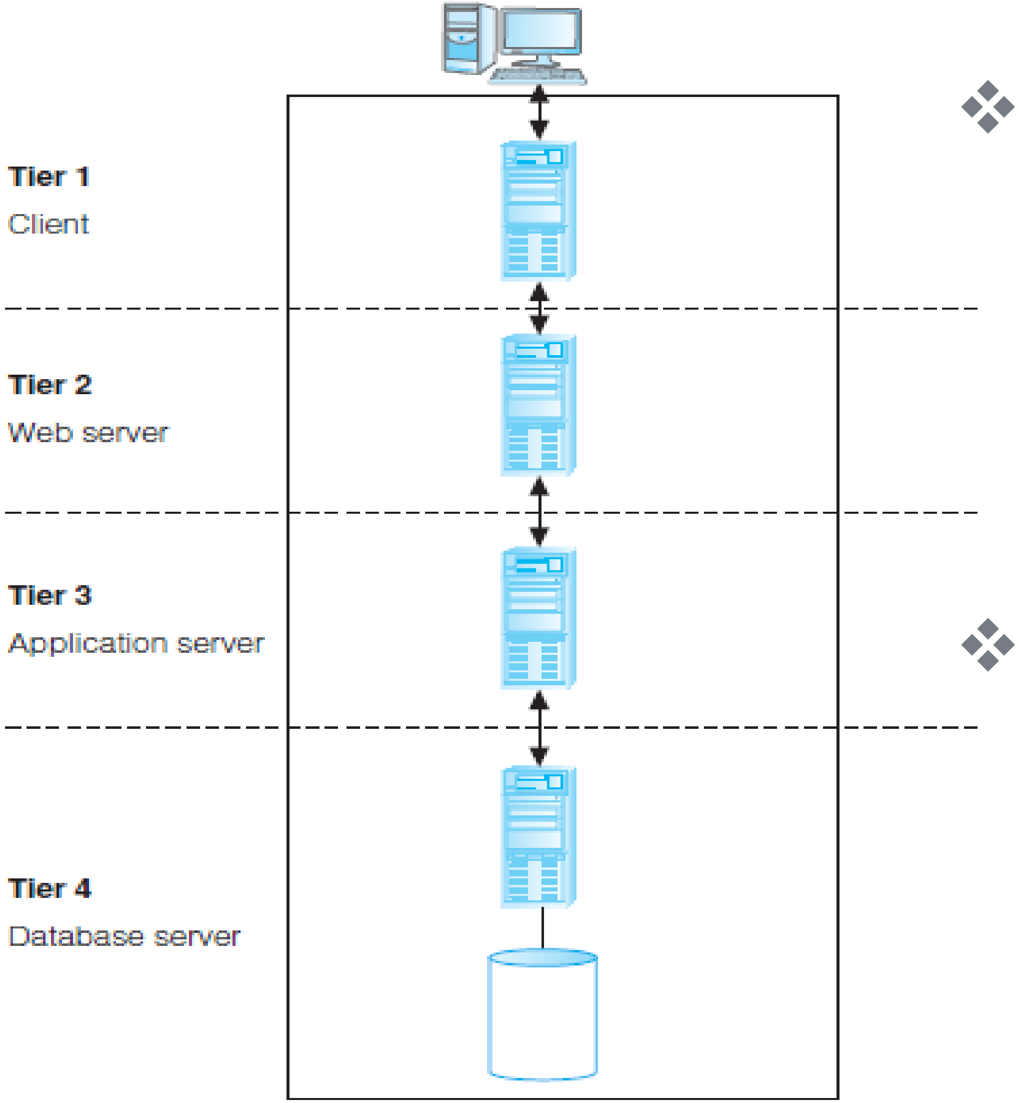
THREE TIER CLIENT-SERVER ARCHITECTURE

- **Three-tier Architecture Can Enhance Security:**
- Database server only accessible via middle tier
- Clients cannot directly access database server
- Clients contain user interfaces and Web browsers
- The client is typically a PC or a mobile device connected to the Web

THREE TIER CLIENT-SERVER ARCHITECTURE

- **Advantages:**
  - Requiring less expensive hardware
  - Application maintenance centralized
  - Easier to modify or replace one tier without affecting others
  - Separating business logic from database functions makes it easier to implement load balancing
  - Maps quite naturally to Web environment

N-TIER CLIENT-SERVER ARCHITECTURE



- The three-tier architecture can be expanded to *n* tiers, with additional tiers providing more flexibility and scalability.

- Applications servers host API to expose business logic and business processes for use by other applications.