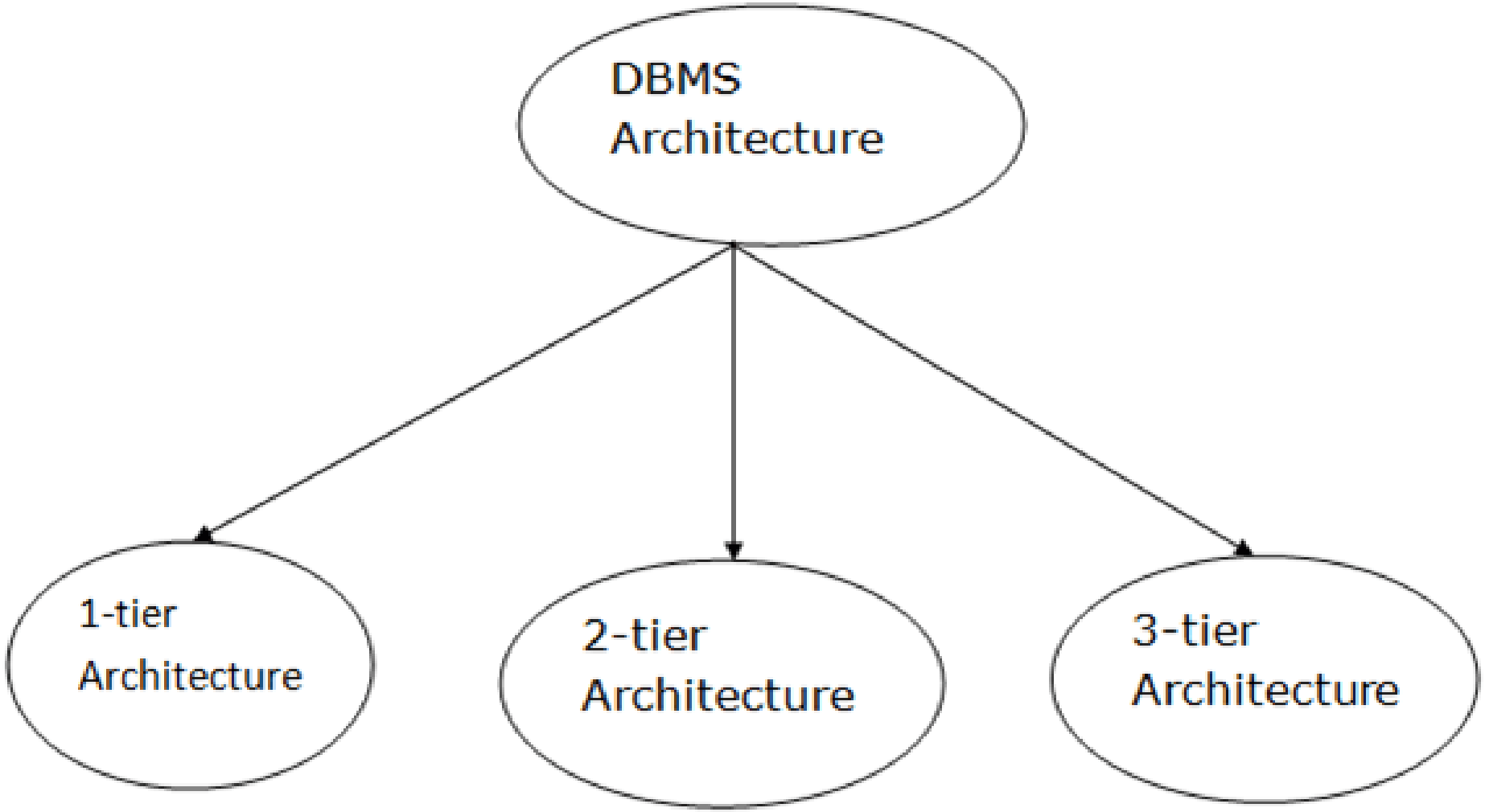
Relational Database Management System (RDBMS)

240110102

Database Architecture

- The DBMS design depends upon its 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.

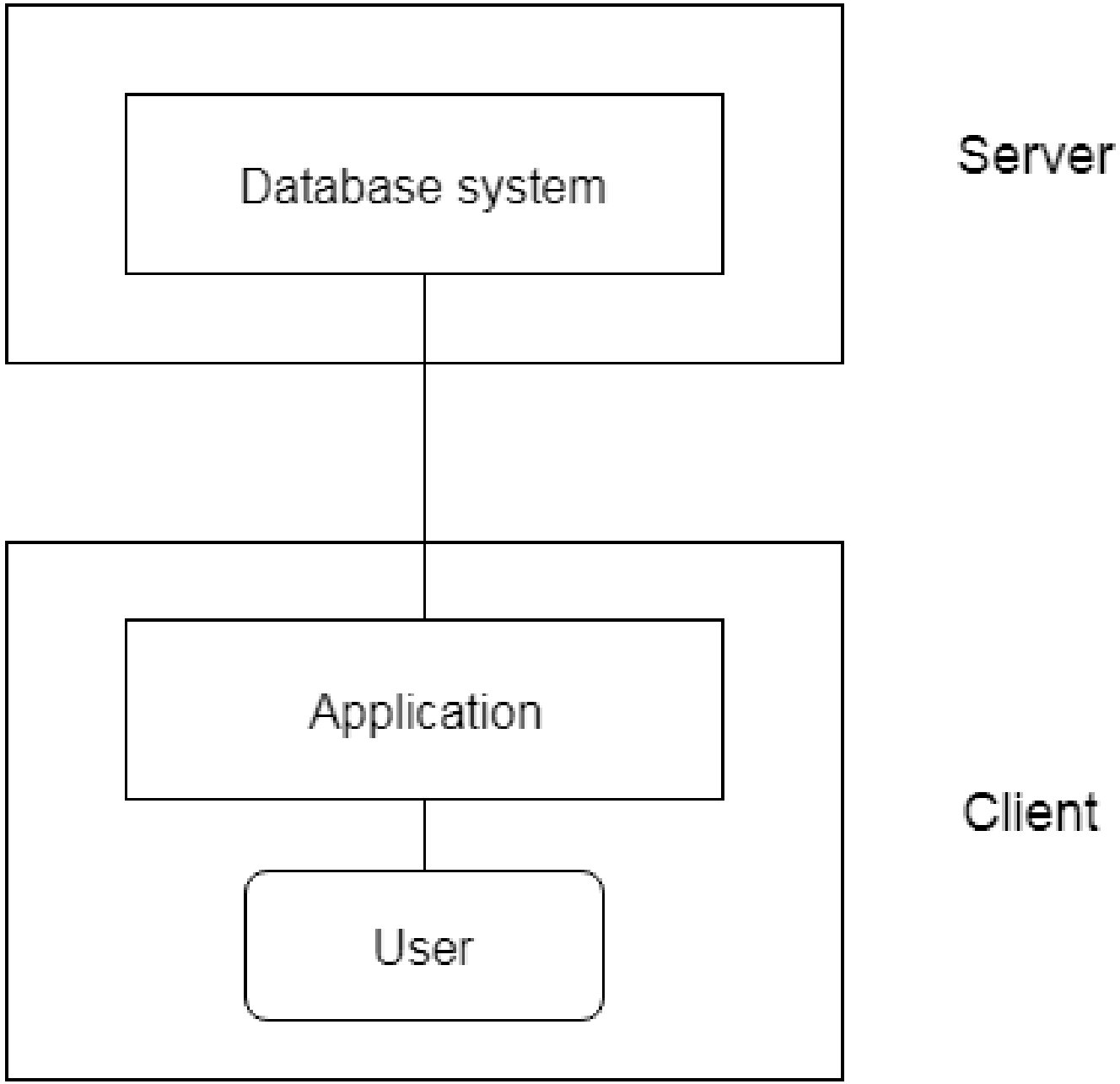

DBMS
Architecture
1-tier
Architecture
2-tier
Architecture
3-tier
Architecture

1-Tier Architecture

- In this architecture, the database is directly available to the user. It means the user can directly sit on the DBMS and uses it.
- Any changes done here will directly be done on the database itself. It doesn't provide a handy tool for end users.
- The 1-Tier architecture is used for development of the local application, where programmers can directly communicate with the database for the quick response.
- Advantages:
  - simplicity, cost-effectiveness, and ease of deployment,
  - Particularly used for small-scale or personal databases.
  - It's also known for faster data access for a single user due to the absence of network communication overhead
- Limitations:
  - lacks scalability, challenges in data sharing
  - support a single user at a time, making it unsuitable for most modern applications.

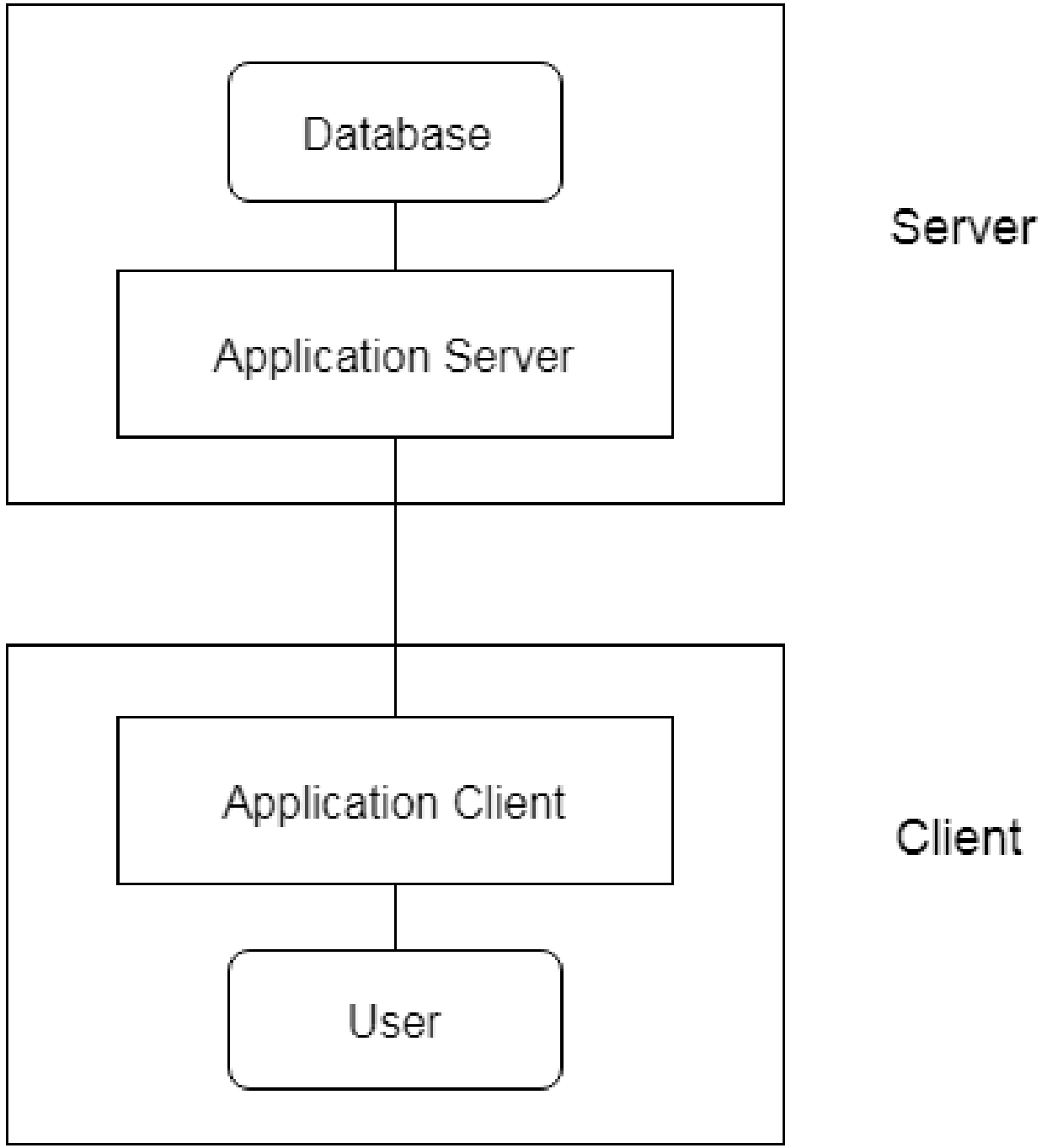
2-Tier Architecture

- The 2-Tier architecture is same as basic client-server. In the two-tier architecture, applications on the client end can directly communicate with the database at the server side. For this interaction, API's like: ODBC, JDBC are used.
- The user interfaces and application programs are run on the client-side.
- The server side is responsible to provide the functionalities like: query processing and transaction management.
- To communicate with the DBMS, client-side application establishes a connection with the server side.
- Advantages:
  - simplicity and direct client-server communication.
  - Key benefits include ease of access, scalability, cost-effectiveness, and simple deployment
- Limitations:
  - scalability, security, and maintainability.
  - It's generally not suitable for large, complex applications with many users due to performance bottlenecks and security vulnerabilities.


Database system
Server
Application
Client
User

3-Tier Architecture

- The 3-Tier architecture contains another layer between the client and server. In this architecture, 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 also has no idea about any other user beyond the application.
- The 3-Tier architecture is used in case of large web application.
- Advantages:
  - By separating the application into three distinct layers (presentation, application, and data), it allows for independent scaling, enhanced security protocols, and easier updates and modifycations
- Limitations:
  - increased complexity, potential performance bottlenecks due to increased network traffic,
  - higher development and maintenance costs, and reduced agility compared to simpler architectures


Database
Application Server
Server
Application Client
User
Client