

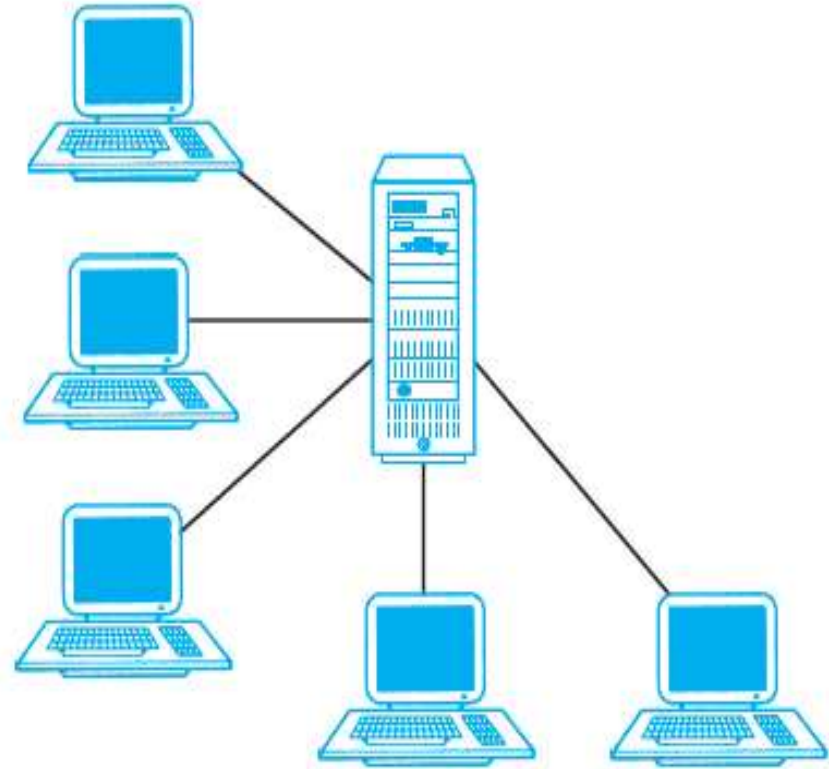
Multi-User DBMS Architecture

- **Types**
 - Teleprocessing
 - File-Server
 - Client-Server

Multi-User DBMS Architecture

- Major components in a data-intensive environment:
 - User Interface
 - Business Logic
 - DBMS & Database

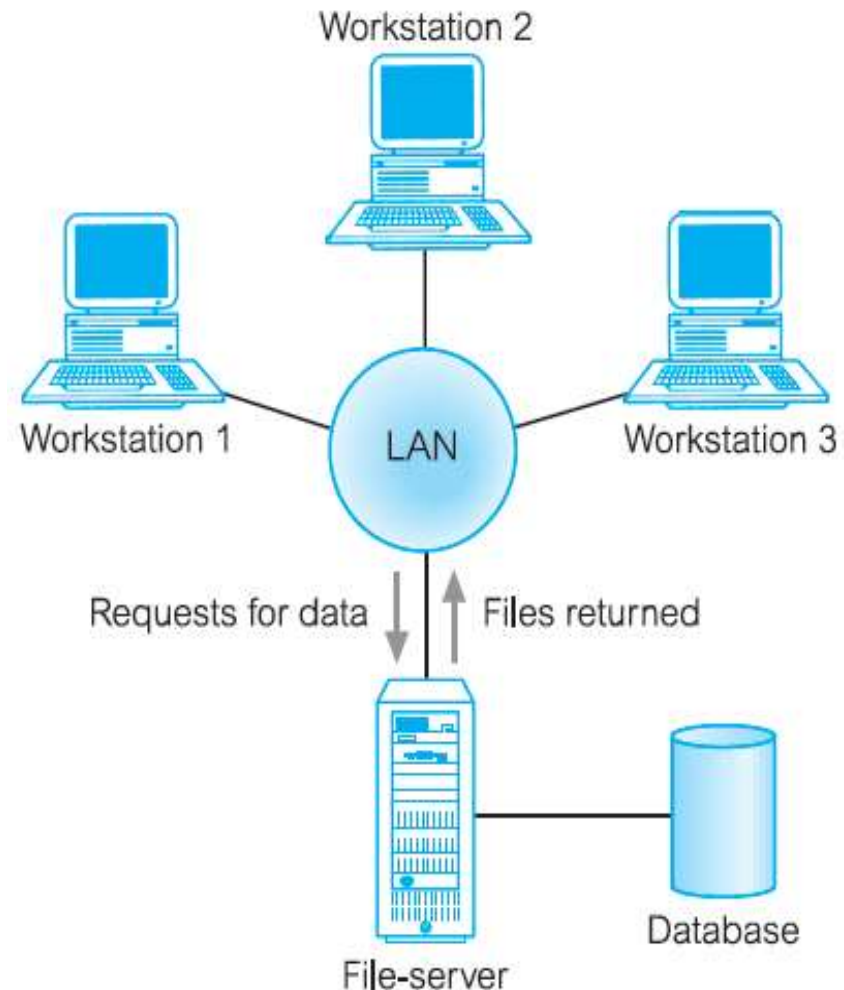
Teleprocessing (Single-Tier)



- Traditional architecture for multi-user systems
- Typically mainframe computers
- One computer with a single CPU and a number of terminals
- User terminals were 'dumb' ones, and cabled to the central computer
- Tremendous burden on the CPU in terms of implementing
 - User Interface
 - Business Logic
 - Database
- Cannot scale to thousands of users

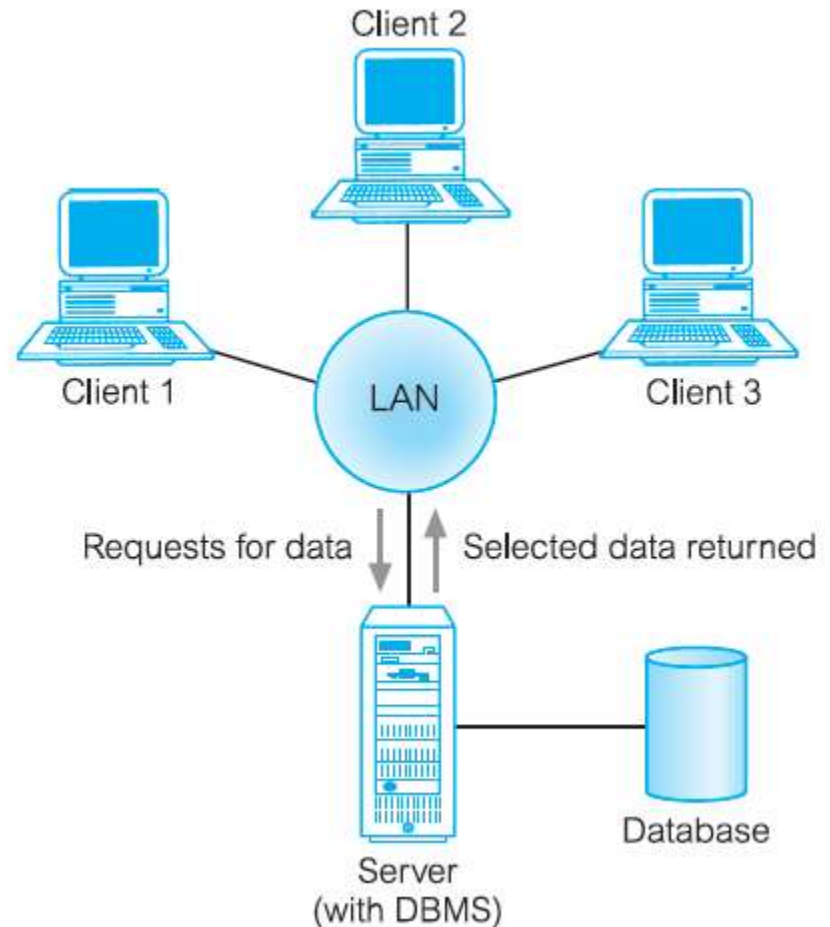
File-Server Architecture

- File-server and workstations were connected through LAN
- File-server
 - Runs the DBMS
 - Stores the Data
 - Acts simply as a shared hard disk drive
- Workstations
 - Run the applications (Interface & Business Logic)
 - Run DBMS on each workstation
 - Requesting data from the file-server when necessary
- Disadvantages:
 - (1) A full copy of all data was required on each workstation.
 - (2) There is a large amount of network traffic.
 - (3) Concurrency, security, and integrity control are more complex because there can be multiple DBMSs accessing the same files

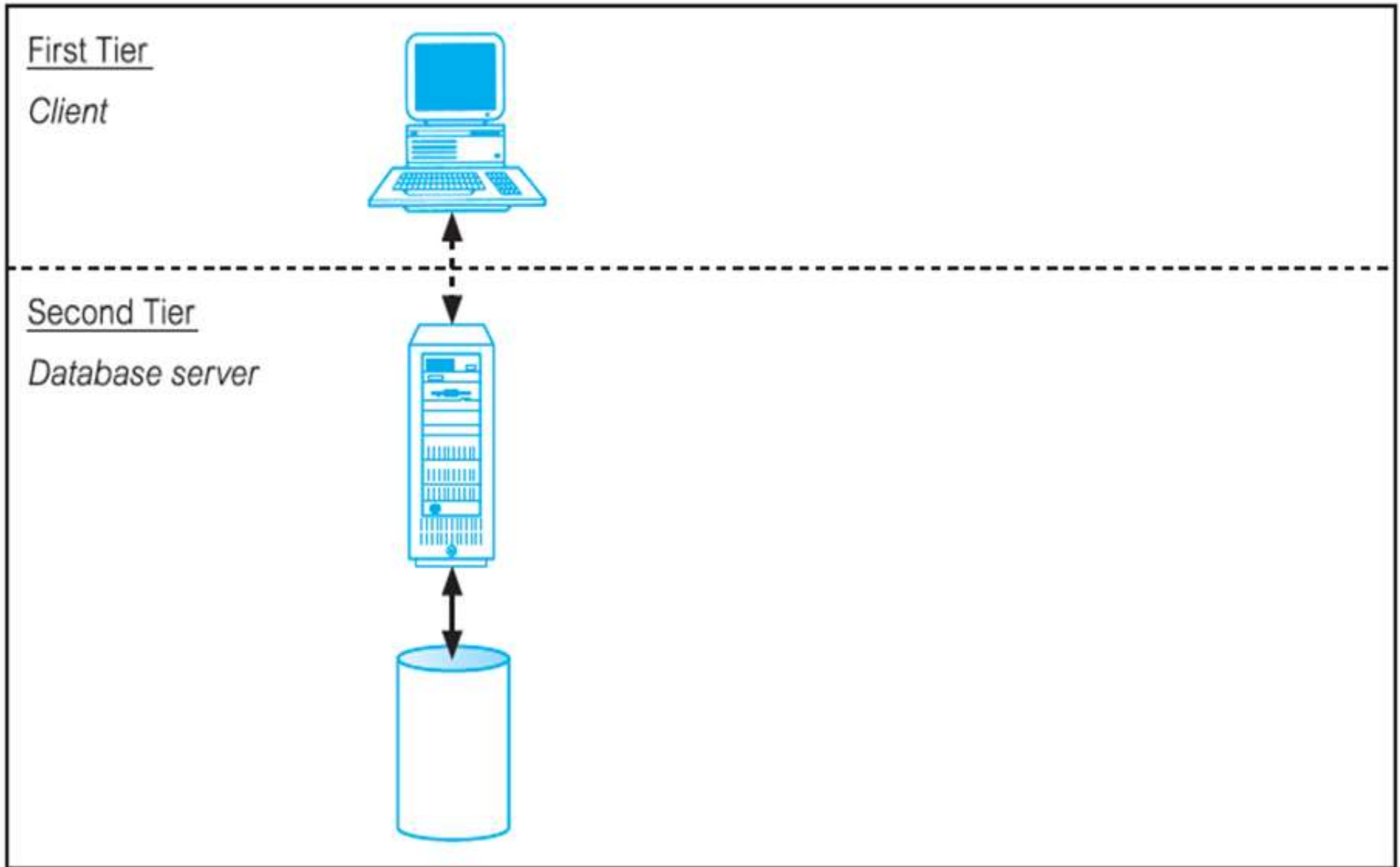


- A **client** requires some resource, and a **server** provides the resource
- Thin Clients
 - Server runs DBMS and business logic
 - Client implements interface
- Thick Clients
 - Server runs DBMS
 - Client runs business logic and implements interface

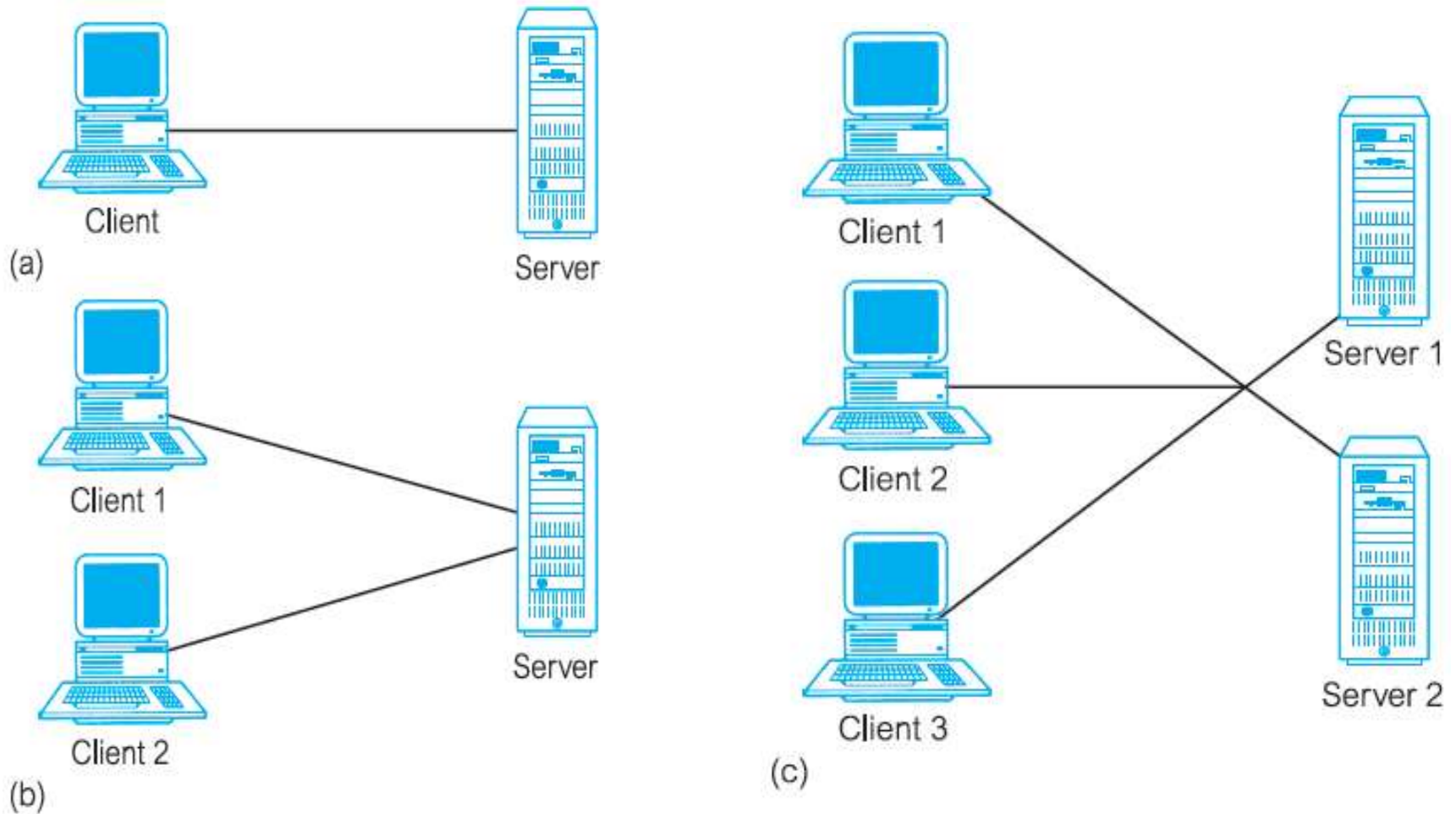
Traditional Two-Tier Client-Server Architecture



Traditional Two-Tier Client-Server



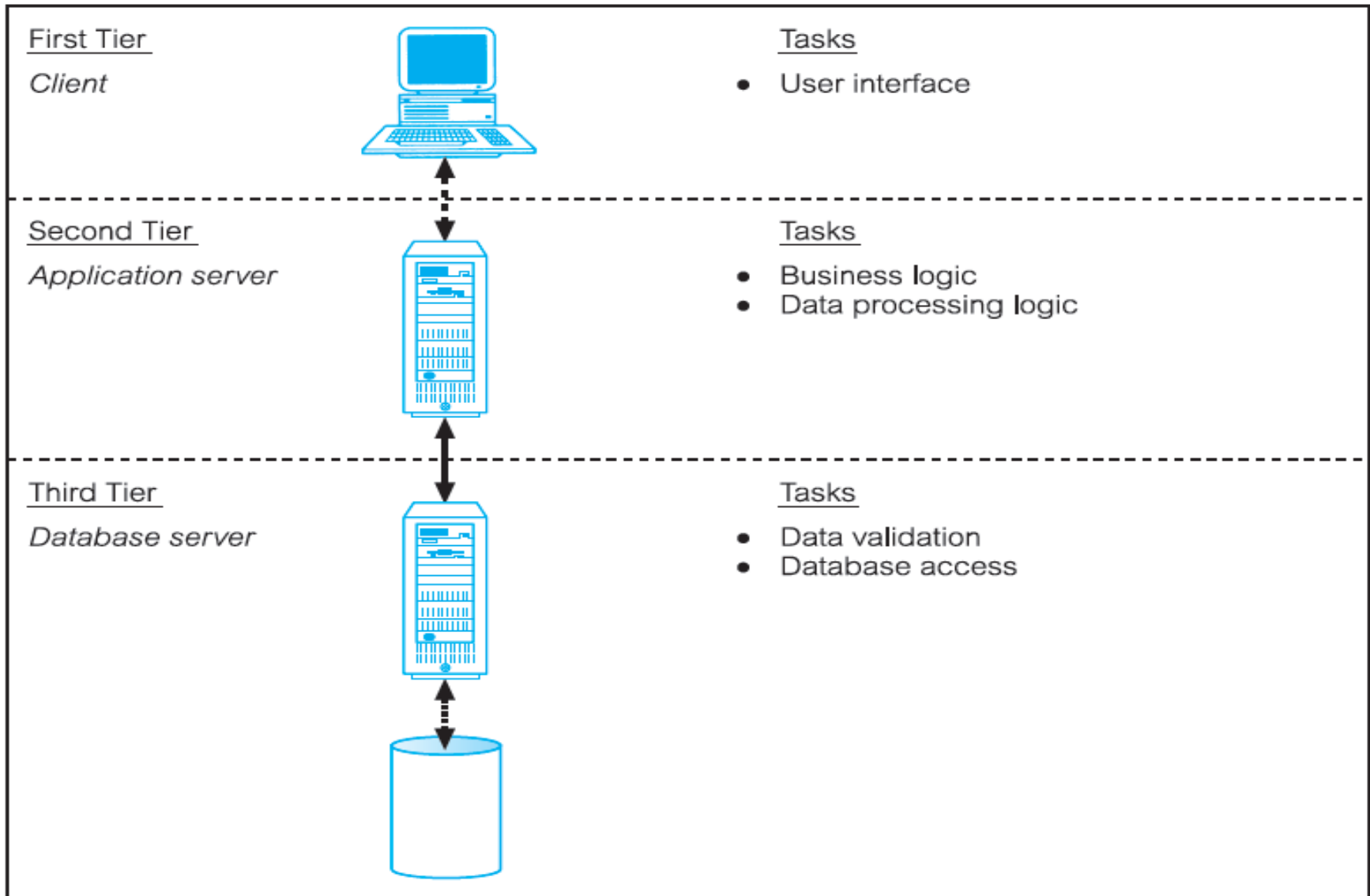
Alternative client–server topologies: (a) single client, single server; (b) multiple clients, single server; (c) multiple clients, multiple servers



Three-Tier Client Server Architecture

- **Problems with 2-tier Architecture:**
 - ‘Fat’ client, requiring considerable resources on client’s computer to run effectively
 - Significant client side administration overhead
 - Preventing true scalability
- **In 1995, three layers were proposed, each potentially running on a different platform**

Three-Tier C-S Architecture



Three-Tier Client-Server Architecture

- User interface layer - runs on client
- Business logic layer - middle tier runs on a server (*application server*)
- DBMS - stores data required by the middle tier and runs on a separate server (*database server*)

Three-Tier C-S Architecture

- Better overall performance
- ‘Thin’ client, requiring less expensive hardware
- Easier to modify or replace one tier without affecting others
- The three-tier architecture can be extended to n -tiers, with additional tiers added to provide more scalability
- For example, the middle tier of the three-tier architecture could be split into two, with one tier for the Web server and another for the application server