

Design Phase Key Ideas



- ❑ The purpose of the analysis phase is to figure out what the business needs.
- ❑ The purpose of the design phase is to figure out *how to provide it*.
- ❑ The steps in both analysis and design phases are highly *interrelated* and may require much “going back and forth”

OO Analysis and Design Foundation



- ☑ Use-case driven
- ☑ Architecture centric
- ☑ Iterative and incremental

OO Analysis and Design - Combining Three Views



- ☒ Functional
 - use case diagrams
- ☒ Static
 - class diagrams
 - object diagrams
- ☒ Dynamic
 - sequence diagrams
 - behavioural state machines (state chart diagrams)

A “Minimalist” Approach



- ❑ Planning
- ❑ Gathering requirements
- ❑ Perform a series of “builds”
- ❑ Use results of each build as feedback for design and implementation

DESIGN STRATEGIES



- ❑ Custom Development
- ❑ Packaged Software
- ❑ System Integration
- ❑ Outsourcing

Custom Development



- ❑ Allows for meeting highly specialized requirements
- ❑ Allows flexibility and creativity in solving problems
- ❑ Easier to change components
- ❑ Builds personnel skills
- ❑ May stretch company's resources
- ❑ May add **significant** risk

Packaged Software



- ☒ **Software already written**
- ☒ May be more efficient
- ☒ May be more thoroughly tested and proven
- ☒ May range from components to tools to whole enterprise systems
- ☒ Must accept functionality provided
- ☒ May require change in how the firm does business
- ☒ May require **significant “customization” or “workarounds”**

System Integration



- ❑ **The process of combining packages, legacy systems, and new software**
- ❑ Key challenge is integrating data
- ❑ Write data in the same format
- ❑ Revise existing data formats
- ❑ Develop “object wrappers”

Outsourcing



- ❑ **Hire external firm** to create system
- ❑ May have more skills
- ❑ May extend existing resources
- ❑ **Never** outsource what you don't understand
- ❑ **Carefully** choose vendor
- ❑ Prepare contract and payment style carefully

Selecting a Design Strategy



- ❑ Business need
- ❑ In-house experience
- ❑ Project skills
- ❑ Project management
- ❑ Time frame

Physical Architecture



- ❑ Software Components
 - ❑ Data Storage
 - ❑ Data Access Logic
 - ❑ Application logic
 - ❑ Presentation logic
- ❑ Hardware Components
 - ❑ Client computers
 - ❑ Servers
 - ❑ Connecting network

Server Based Architecture



- ❑ Client is a terminal
- ❑ Server has functions of
 - ❑ Presentation logic
 - ❑ Application logic
 - ❑ Data access logic
 - ❑ Data storage

Server Based Architecture

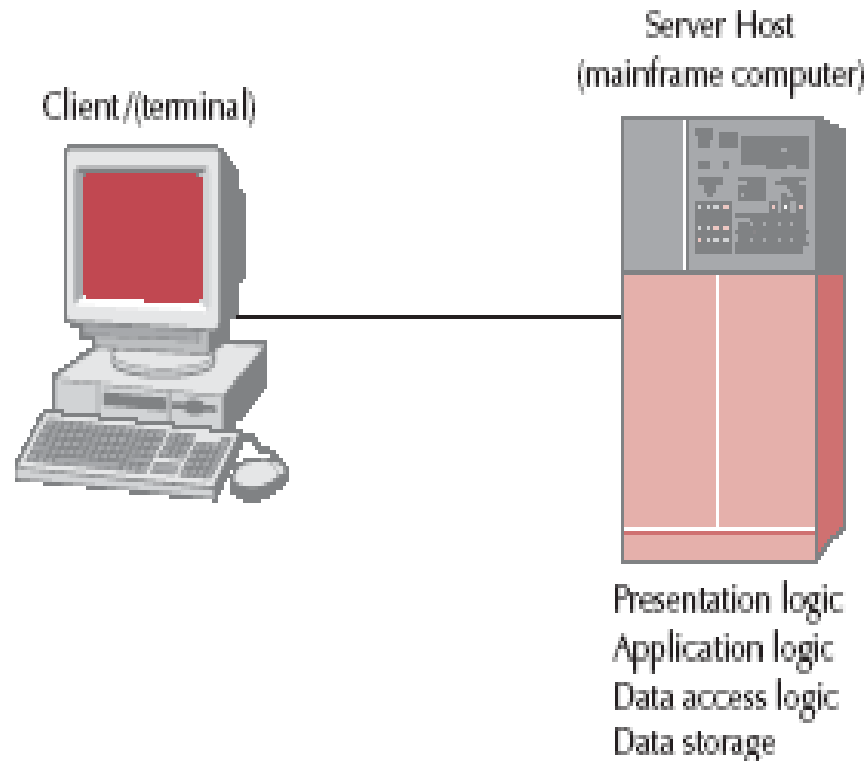


FIGURE 13-1
Server-Based
Architecture

Client-Server Architectures



- ❑ Thin Client/Fat Server
 - Client is little more than a terminal
 - Server handles all processing
- ❑ Fat Client/Thin Server
 - Client does all processing
 - Server may just store data

Fat (Thick) Client

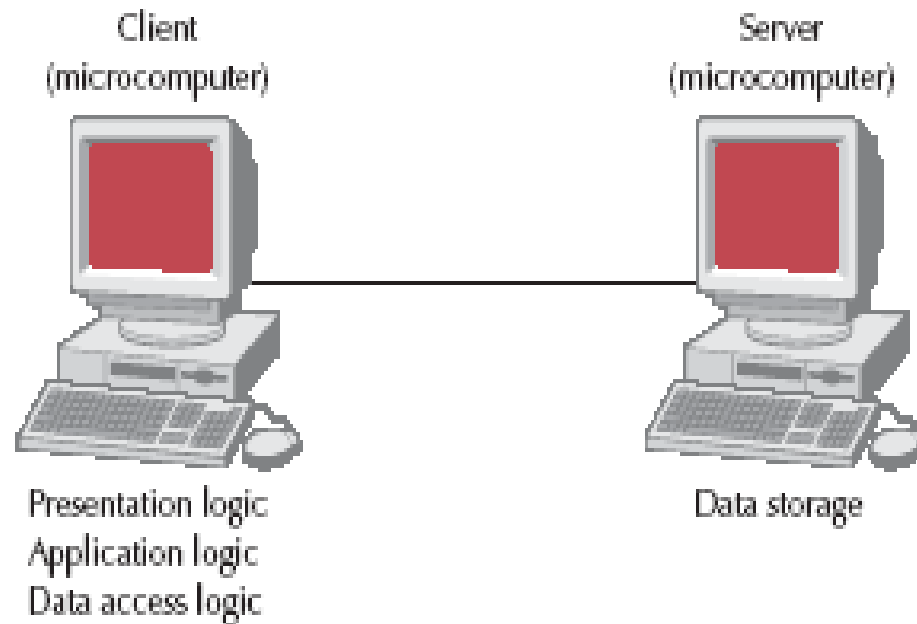


FIGURE 13-2
Client-based
Architectures

Server has Data

2-Tiered Architecture

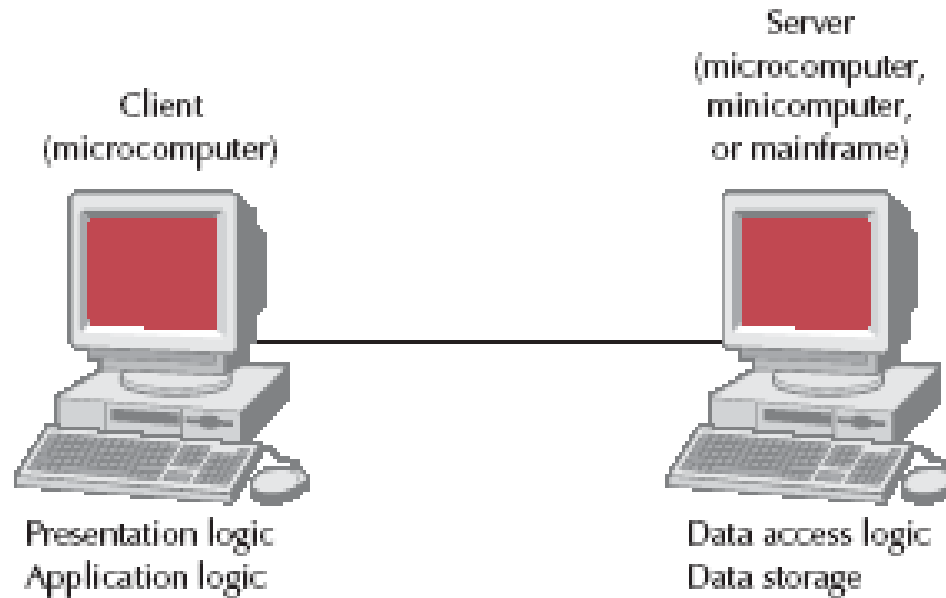


FIGURE 13-3
Client-Server
Architecture

Client has Presentation Logic

3-tiered Architecture

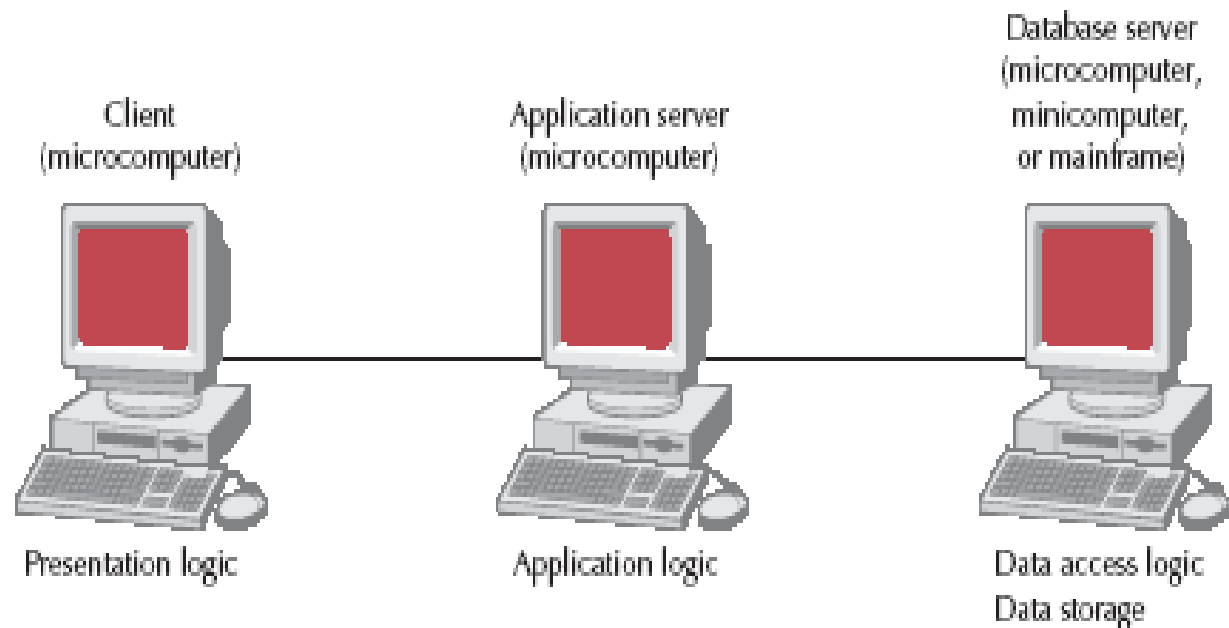


FIGURE 13-4
A Three-Tier Client-Server Architecture

N-tiered Architecture

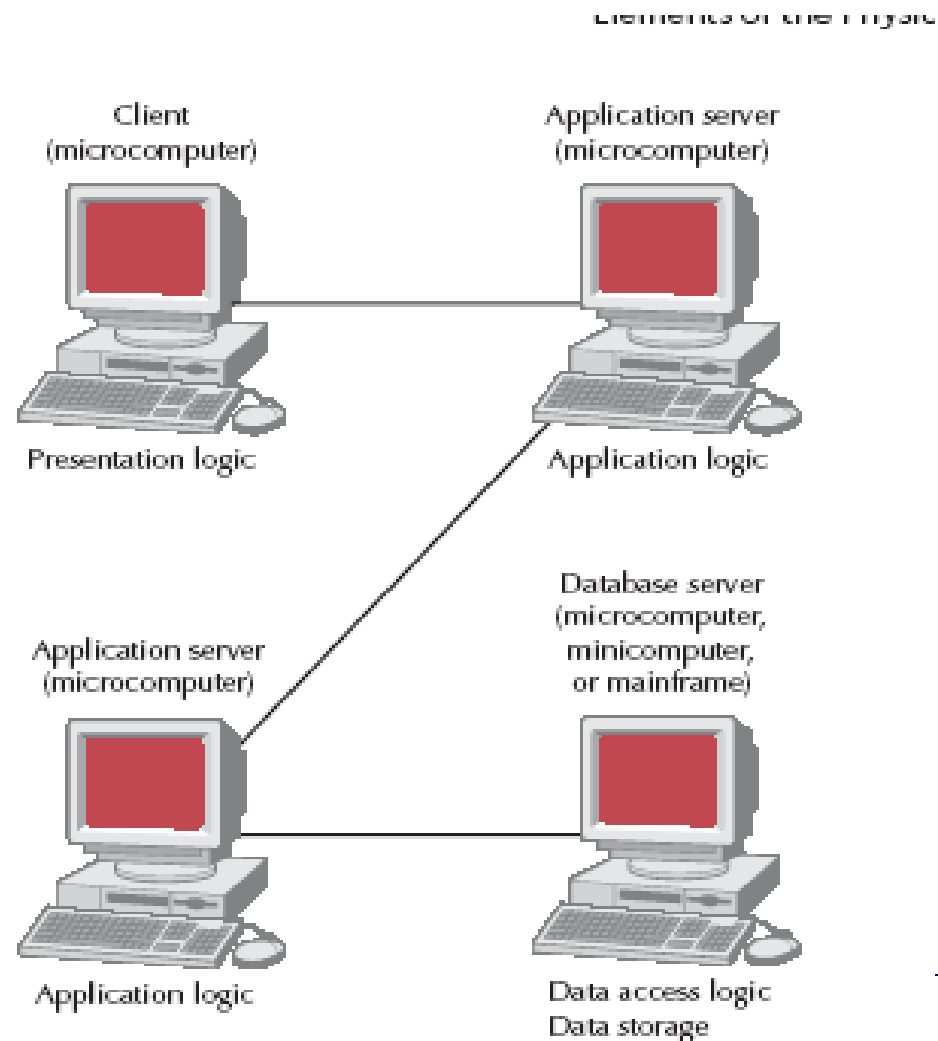


FIGURE 13-5
A Four-Tier Client-
Server Architecture

Client-Server Benefits



- ☑ **Scalable**
- ☑ Support **multiple** clients and servers
- ☑ Multiple servers make for a generally **more reliable** network

Client-Server Limitations



- ❑ **Complexity**
- ❑ Updating the network computers is more complex
- ❑ Development and Installation
 - different on client and server
 - training for developers

Selecting a Physical Architecture



- ❑ Cost of Infrastructure
- ❑ Cost of Development
- ❑ Ease of Development
- ❑ Interface Capabilities
- ❑ Control and Security
- ❑ Scalability

Characteristics



FIGURE 13-6
Characteristics of
Computing
Architectures

	Server-Based	Client-Based	Client-Server
Cost of infrastructure	Very high	Medium	Low
Cost of development	Medium	Low	High
Ease of development	Low	High	Low-medium
Interface capabilities	Low	High	High
Control and security	High	Low	Medium
Scalability	Low	Medium	High

NonFunctional Requirements



- ❑ Operational
 - Specify the operating environment
- ❑ Performance requirements
 - Speed, number of users, availability
- ❑ Security
 - Protect from disruption, data loss