



Client/Server - Introduction

Contents:



- Origins
- Open Systems
- Client/Server: Definition & Characteristics
- Why C/S?
- Infrastructure
- Architectures
- The Block Model
- Clients, Servers and the Operating Systems

Mariano Cilia – mcilia@gmail.com

1



Origins

- Only big companies had computers (mainframes)
- Choosing the right vendor/provider
 - Only few providers:
 - IBM, NCR, UNISYS, DIGITAL, ...
- Absolute everything depending on the provider
 - hardware
 - communications
 - software
 - people!!!!
- Known as the **Closed Systems Era**



Mainframes

2



Open Systems

- Freedom to choose/select components
 - Every component can be bought to different providers
- Flexibility
- Extensibility
- Scalability
- But, now many open issues (menú à la carte)
 - Which platform?
 - Which communication protocols?
 - Which DBMS?
 - Which middleware?
 - Which technology? (db servers, TP monitors, groupware, distributed objects, intranets)

3



Open Systems (cont.)

- Still other issues
 - Now everything depends on the IT Manager
 - Each component of a system (possibly coming from different providers) need to be integrated
 - The integration of those components needs to be done
- There is no recipe!

4



Why Client/Server?

- The main reasons:
 - Share (expensive) resources
 - print servers, file servers, ...
 - Specialization
 - compute servers, data servers, ...
 - Scalability
 - Extensibility

5



What is Client/Server?

- Clients and Servers are logical entities that cooperate in order to carry out a determined task
- Loosely coupling between Client and Server
 - Logically separated

6



C/S - Characteristics

- Service
 - The server provides services
 - The client uses them
- Operations are based on messages
 - Interaction by using the message passing mechanism (request/reply)
- Encapsulation of services
 - The server is a specialist on carrying out a determined task
 - The message specifies the service
 - As long as the protocol (interface) is maintained its implementation can be replaced
 - This without affecting clients

7



C/S – Characteristics (cont.)

- Asymmetric Protocols
 - 1:N relationship between server and clients
 - Clients initiate the dialog by requesting a service
 - Servers are passive
- Shared Resources
 - Server can manage several requests of clients concurrently
 - Access to shared resources is controlled/administered
- Scalability
 - Incorporation of additional servers (transparently) imply more computing power

8



C/S – Characteristics (cont.)

- Location Transparency
 - The server is a process that can run in a local machine or in any computer of the network
- “Mix & match”
 - The ideal C/S software should be independent of hardware and OSs

9



Different Viewpoints

- PCs defenders
 - Wanted to migrate applications from the mainframe to PCs (in a LAN) braking with centralization

10



Different Viewpoints

- PCs defenders
 - Wanted to migrate applications from the mainframe to PCs (in a LAN) breaking with centralization
- Mainframe defenders
 - Tight PCs to the mainframe (as terminals) without breaking with centralization

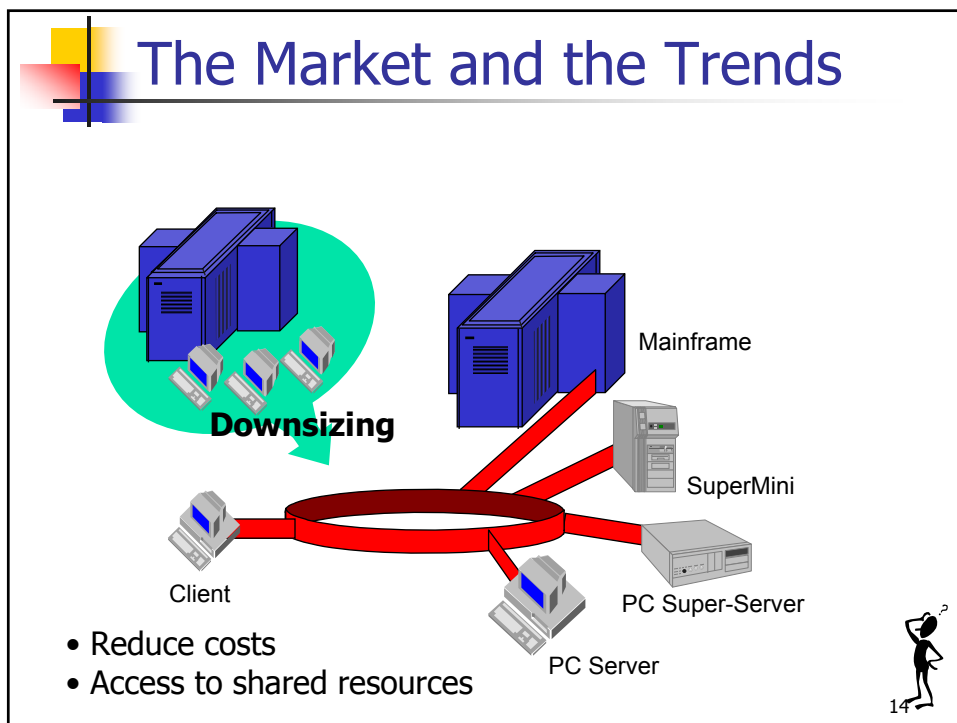
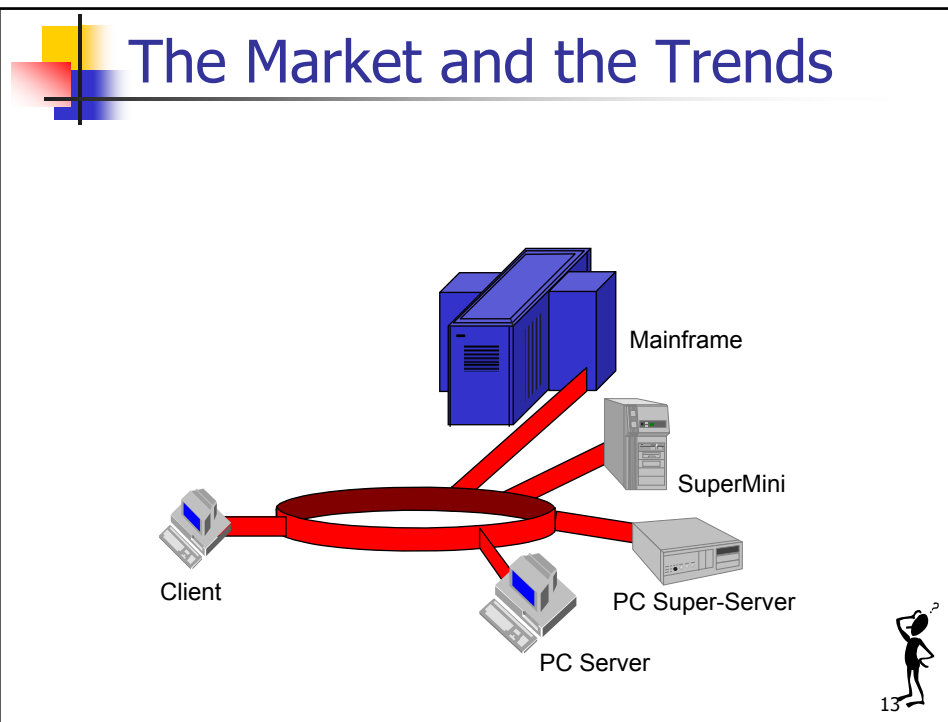
11



Different Viewpoints

- PCs defenders
 - Wanted to migrate applications from the mainframe to PCs (in a LAN) breaking with centralization
- Mainframe defenders
 - Tight PCs to the mainframe (as terminals) without breaking with centralization
- Intermediate position
 - Open systems
 - Coexistence of PCs as clients (and possibly also as servers) and the mainframes as servers
 - Effort is necessary to **standardize protocols and platforms**

12



The Market and the Trends

- Integrate isolated PCs
- Share resources
- Improve interaction among users

Client

Upsizing

Mainframe

SuperMini

PC Super-Server

PC Server

15

The Market and the Trends

- The network is the system
- Open architecture

Client

Mainframe

SuperMini

PC Super-Server

PC Server

Rightsizing

16



New Roles

- New IT Managers
 - Wider spectrum of knowledge
 - Transaction processing, databases, communications, user interfaces, etc.
 - Know how to delegate functions/tasks
 - Two-sides strategy
 - Development and administration of departmental applications
 - Administration of company/corporate interoperability
 - Definition of standards within the company

17



New Roles (cont.)

- Competition among Providers
 - Equality of opportunities (open architecture)
 - Benefits customers, avoiding to be captive of a unique provider
 - See Hardware (open architecture)
 - (PCs, network cards, graphic cards, etc.)
 - Software tending to be open
 - DBMSs, CORBA, Web Servers, Web Services, App Servers, ...

18



Different Technologies

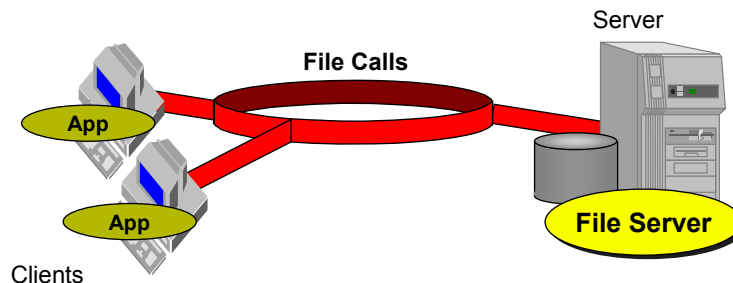
- According to the nature of the provided service
 - File Servers
 - Database Servers
 - Transaction Servers
 - Object Servers
 - Web Servers

19



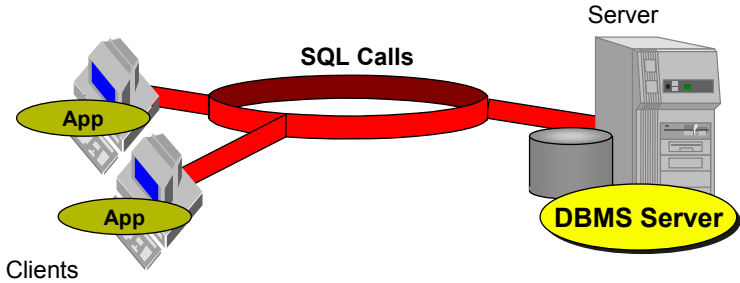
Technologies - File Servers

- Primitive model
- Big amount of messages (file level granularity)
- Client-side processing



Technologies - DB Servers

- Message = SQL sentence
- Server-side processing
- More than one message to complete a task

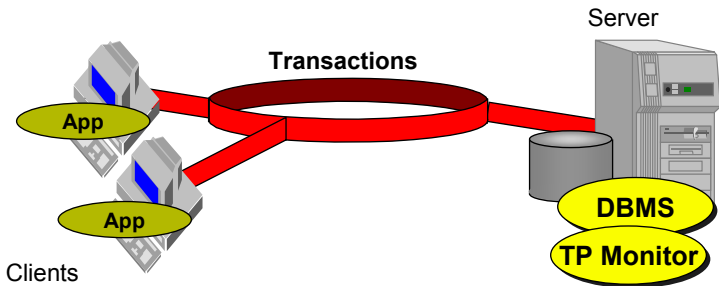


The diagram illustrates the architecture of DB Servers. On the left, under the label 'Clients', there are two computer icons, each with a yellow oval labeled 'App' attached to its base. Red lines connect these 'App' ovals to a central red ring labeled 'SQL Calls'. This ring is connected by another red line to a server icon on the right. The server is labeled 'Server' and has a yellow oval labeled 'DBMS Server' at its base. A small stick figure is in the bottom right corner.

21

Technologies – Transact. Servers

- Message = remote procedure (TX)
- Unique message for a task
- All or nothing functionality
- Server-side processing
- OLTP (critical: response time and availability)

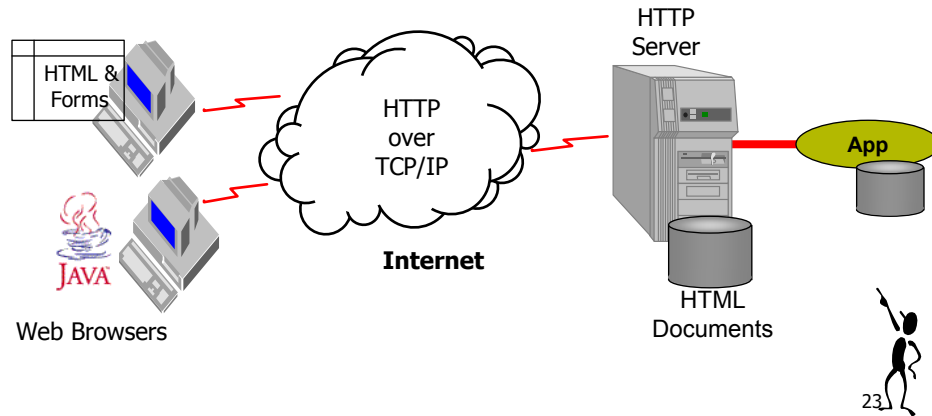


The diagram illustrates the architecture of Transact. Servers. On the left, under the label 'Clients', there are two computer icons, each with a yellow oval labeled 'App' attached to its base. Red lines connect these 'App' ovals to a central red ring labeled 'Transactions'. This ring is connected by another red line to a server icon on the right. The server is labeled 'Server' and has two yellow ovals at its base: 'DBMS' and 'TP Monitor'. A small stick figure is in the bottom right corner.

22

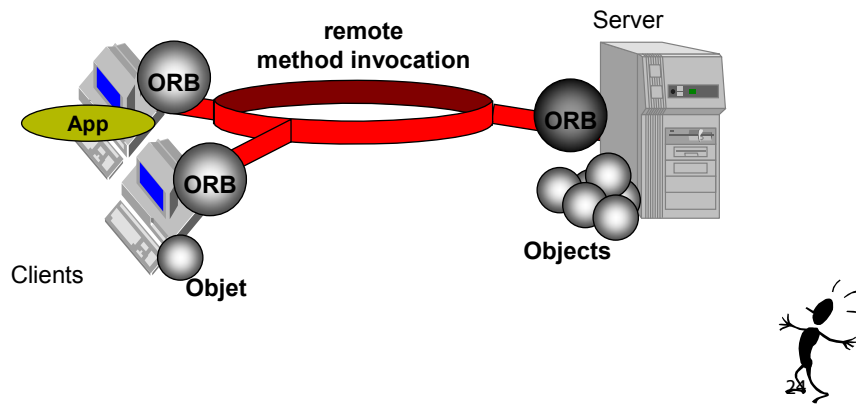
Technologies - Web Servers

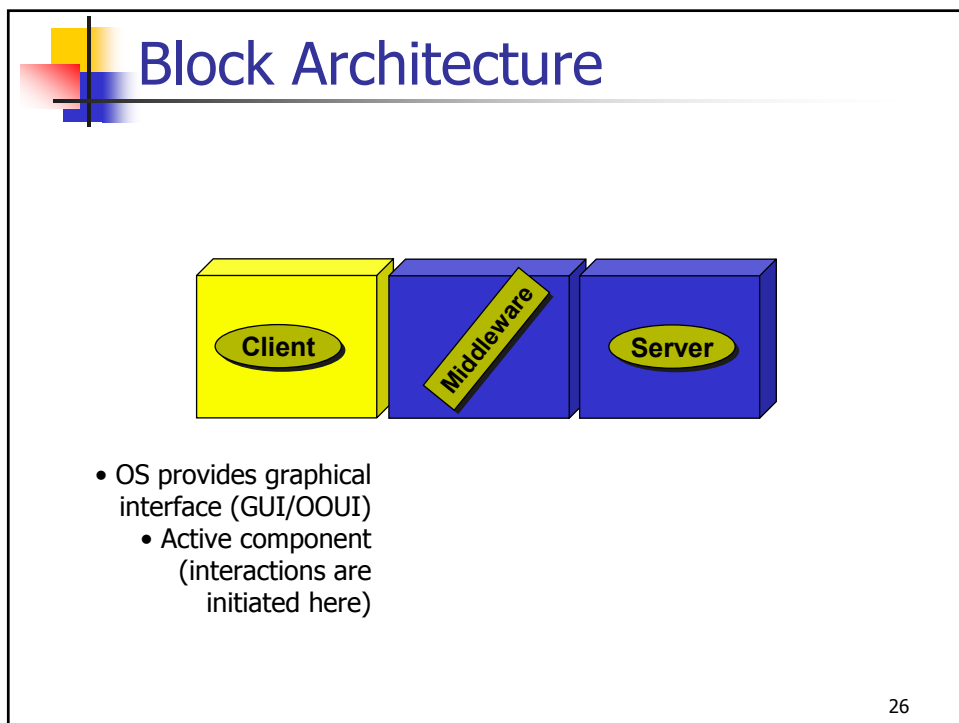
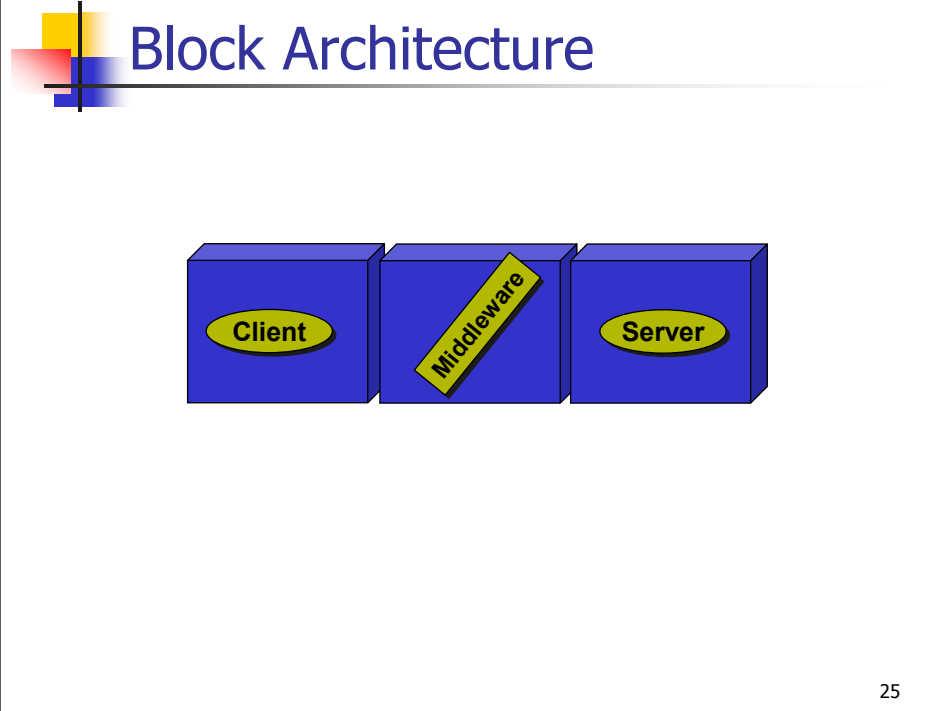
- “Universal” Clients interacting with servers
- RPC-like protocol: HTTP
- Java: first manifestation of the Object Web



Technologies - Object Servers

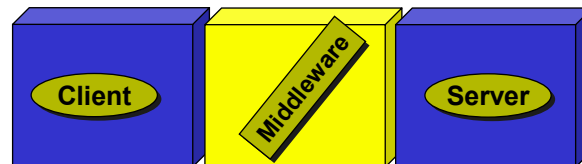
- Client invokes methods
- ORB as intermediary







Block Architecture

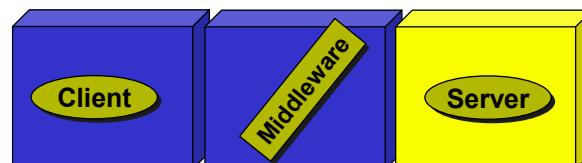


- It runs in both ends
- Divided in layers
- Allows the interaction between client and server

27

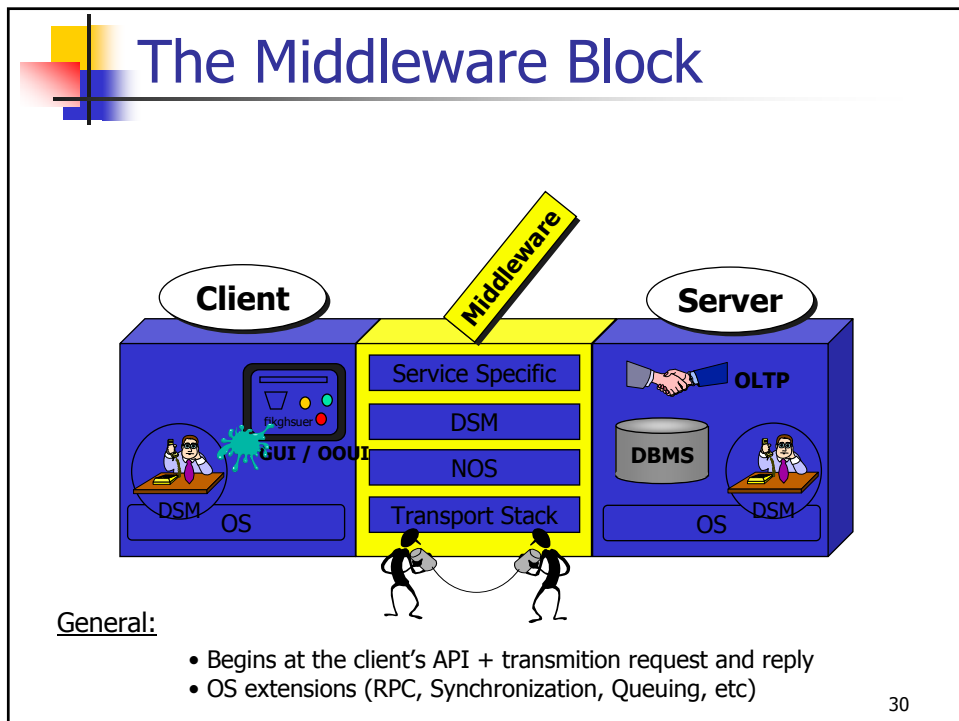
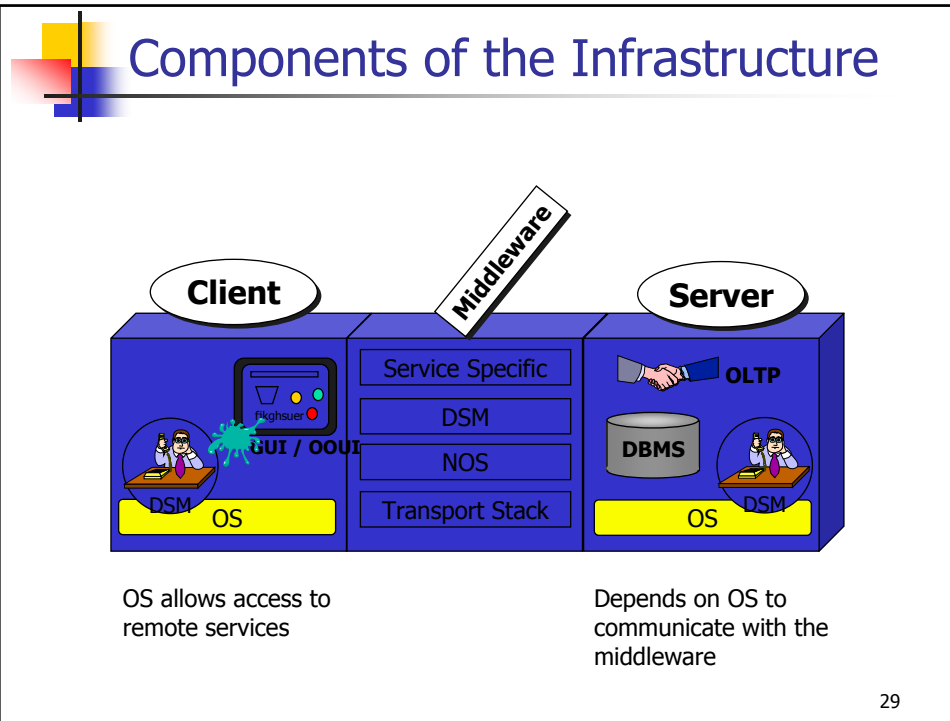


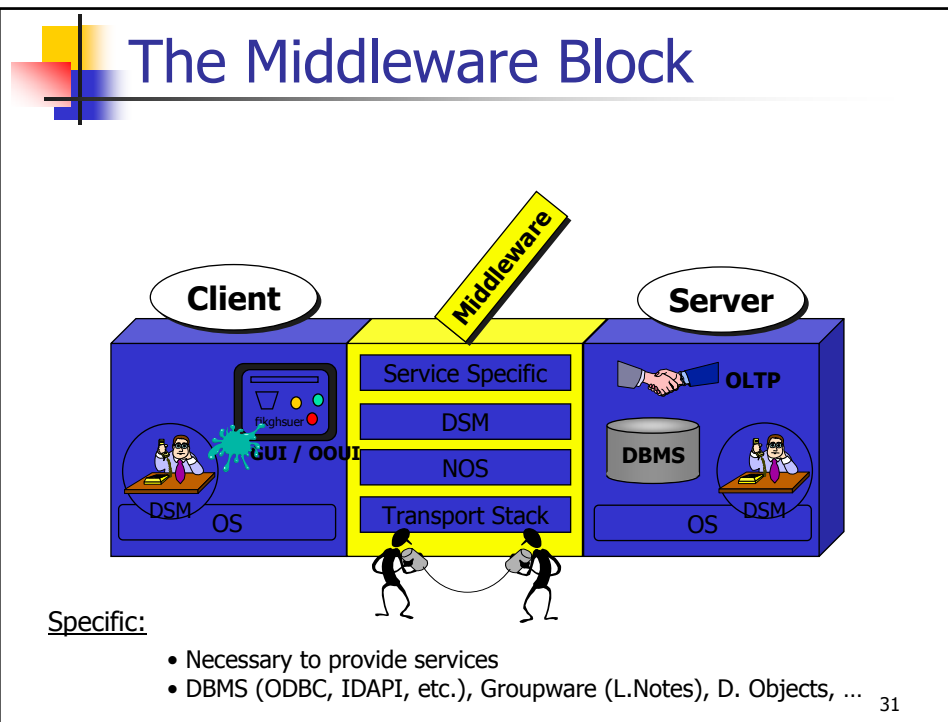
Block Architecture



- Processes requests from clients
- It is (basically) passive

28





One Model Fits All Sizes

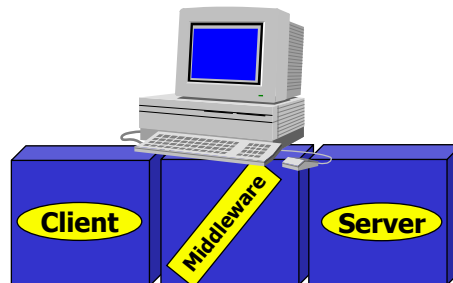
- For small or nomadic users
- For middle-size or departmental users
- For big companies
- For the future ...

32



Small or Nomadic Users

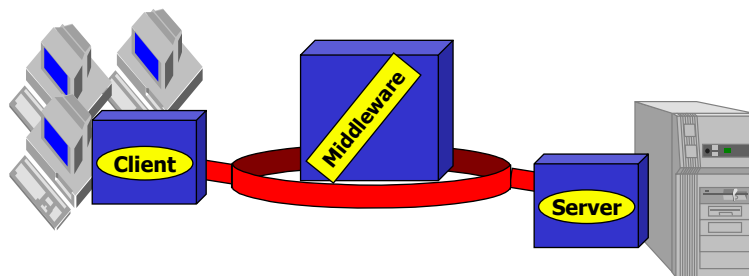
- All 3 in the same machine
- Can communicate with other servers



33



Middle-size or Departmental Users

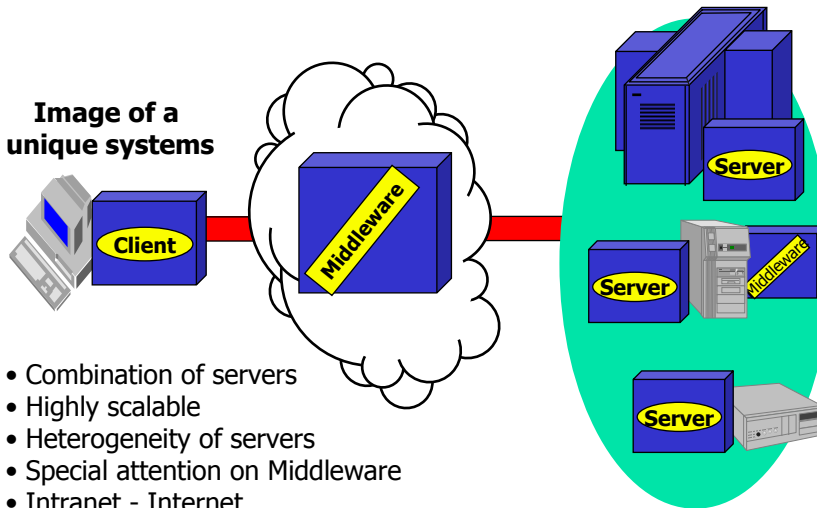


- Some clients and one server
- Typical case
- Simple middleware (homogeneity)
- Appropriate solution for departments (own server)

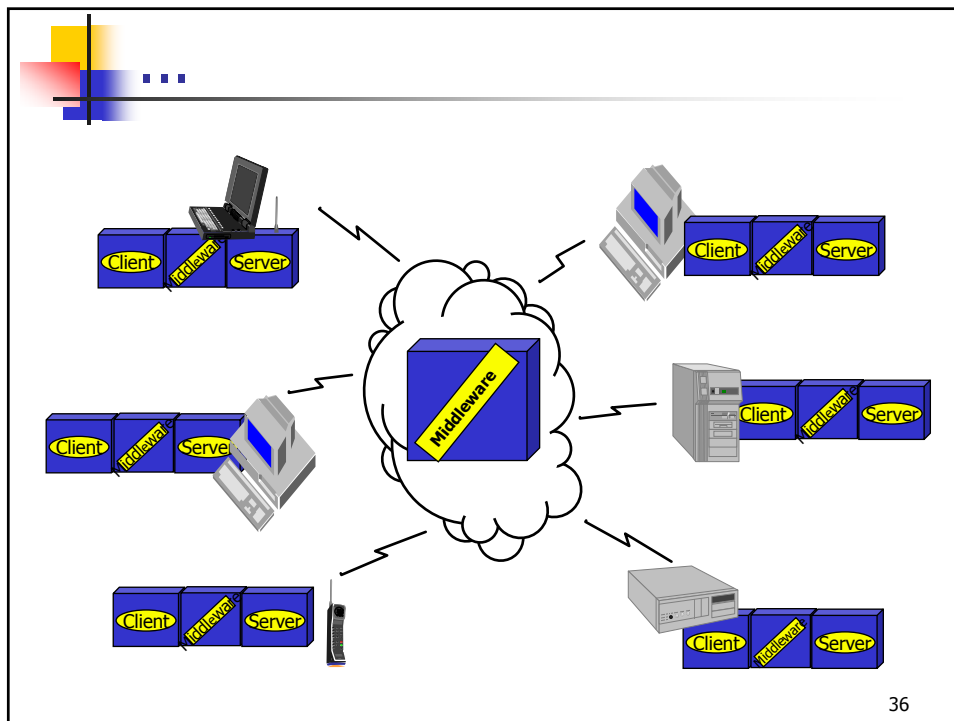
34

Big Companies

Image of a
unique systems



35



36



C, S and the OS

- Anatomy of a **Server** program
 - wait for incoming requests initiated by clients
 - manage many requests at the same time
 - First those with priority (reports vs online requests)
 - initiate and run tasks in background (e.g. backup)
 - always running
 - growth (scalability)

37



C, S and the OS (cont)

- Anatomy of a **Client** program
 - simply requests services
- Different kinds of clients based on the UI
 - clients without GUI (Graphical User Interface)
 - GUI Clients
 - OOUI Clients (Object Oriented User Interface)

38

What does a C require from the OS?

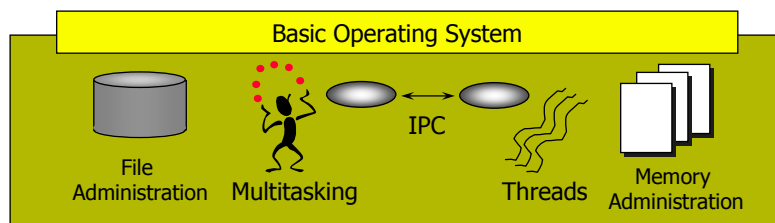
- Request/Reply Mechanism
- Facility to transfer files
- Preemptive multitasking
- Tasks with priorities
- Inter-process communication
- Thread support to enable background communication with the server

39

What does a S require from the OS?

■ Basic Services

- Preemptive multitasking
- Threads
- Tasks w/priorities
- Semaphores
- Inter-Process communication (IPC)
- Memory protection (task level)
- High performance multi-user file systems
- Efficient management of main memory
- Dynamic extension of services at run-time



40

What does a S require from the OS?

■ Extended Services

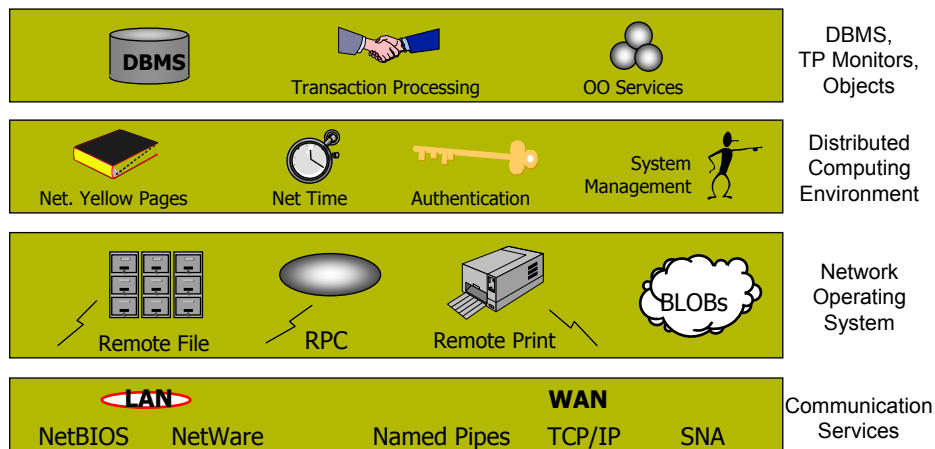
- Diversity of communication platform
- Extensions at NOS level (transparency)
- Yellow pages (to search for services)
- Authentication and authorization services
- Tools to monitor and manage (performance, install, update, pay-per-use, etc)
- Network Time (NTP)
- Transaction service
- DBMS
- Support for Object-Orientation

41

What does a S require from the OS?

Extensions

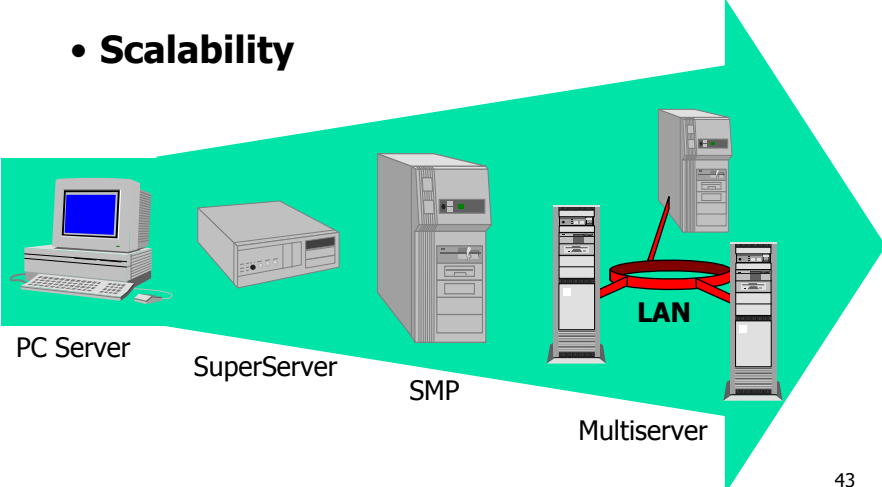
- exploit distribution
- flexibility of access to shared resources
- facilitate administration and maintenance



42

S – Some considerations

- **Scalability**



PC Server SuperServer SMP Multiserver

LAN

43

S - Some considerations

- **Scalability**
- **Superserver con Multiprocesamiento (SMP)**
- **Massive Parallel Processing (MPP)**
- **RISC vs. INTEL**

45



The OS War

Which one for the Client ?

Which one for the Server ?

46



The OS War (cont)



Client

SuperClient


Server

Which one for the Client?

The OS should be ROBUST!!

47

The OS War (cont)



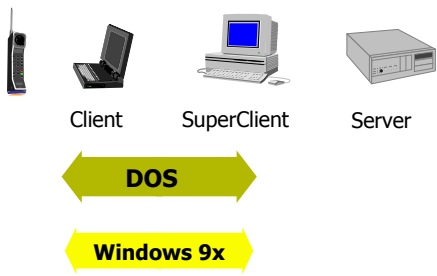
Client SuperClient Server

DOS

- It is no robust
- The task scheduler does not support preemptive multitasking

48

The OS War (cont)



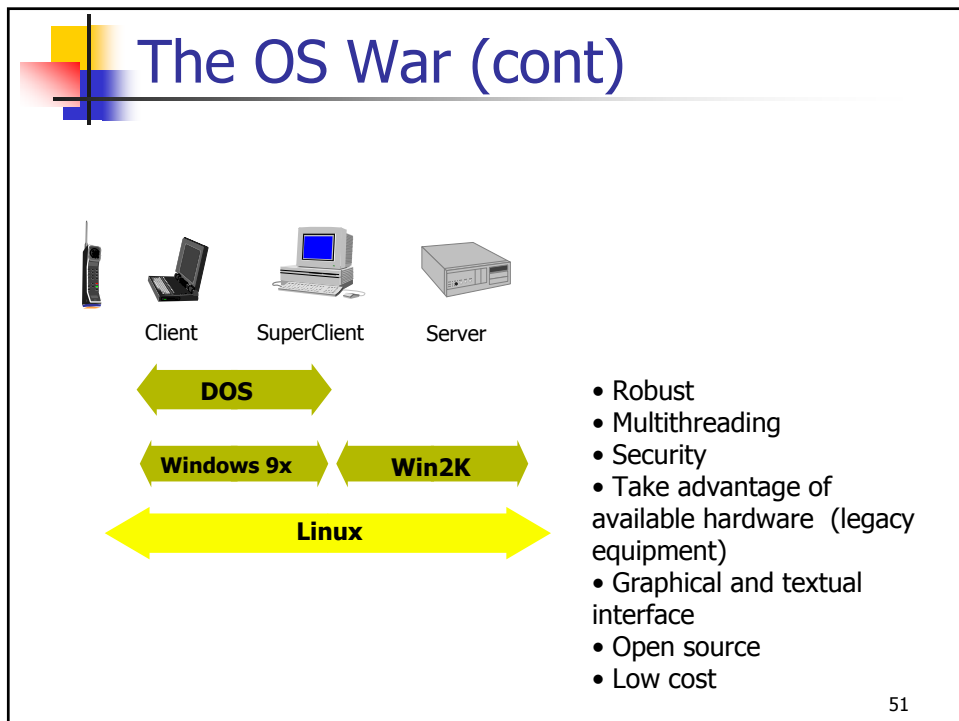
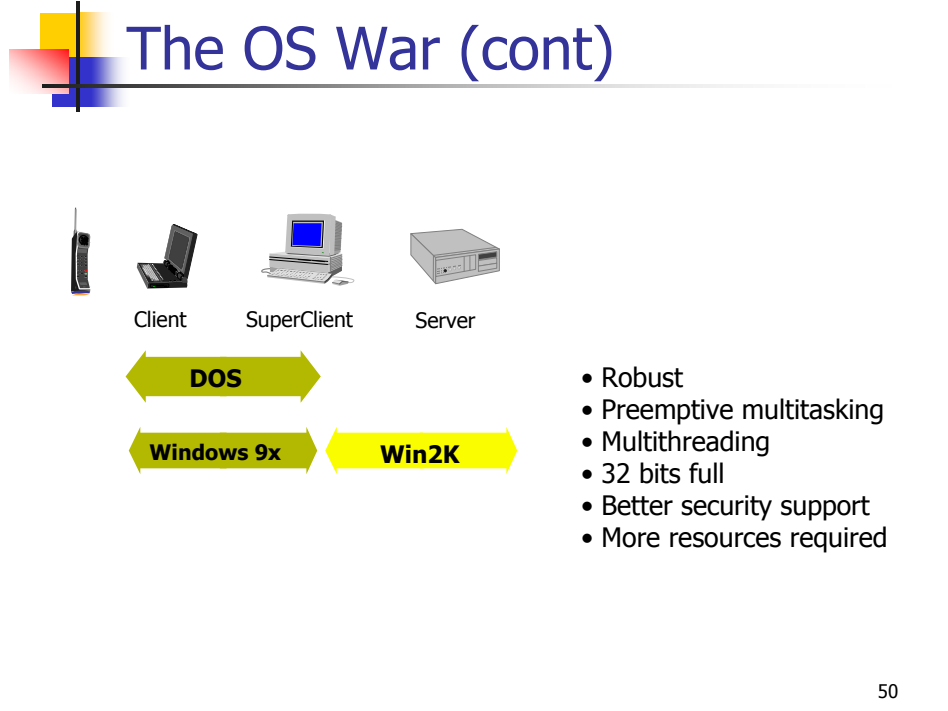
Client SuperClient Server

DOS


Windows 9x

- Robustness: better than DOS
- Hardware auto-detection
- Plug and play
- Multithreading
- Network Ready
- OOUI inconsistent (GUI)
- not fully 32 bits
- Preemptive multitasking

49



The OS War (cont)




Client SuperClient Server Servidor SMP MegaCluster

Which one for the Server?

52

The OS War (cont)



Client SuperClient Server SMP Server MegaCluster

Win2K Server

- Big amount of resources are needed
- Restricted command language
- Friendly interface
- Secure and "scalable"
- It is not portable
- Natural server for M\$ applications

53

