

## Sistemas Distribuidos vs Sistemas Paralelos

- Un sistema paralelo consiste en múltiples procesadores que se comunican entre ellos compartiendo memoria
- Un sistema distribuido puede correr software paralelo y viceversa
  - Con mensajes u otras soluciones (BD) se puede simular la memoria compartida
  - Con memoria compartida se pueden simular mensajes

## Sistemas Distribuidos vs. Sistemas Paralelos

- ¿Por qué no hacerlo paralelo?
  - Escalabilidad: El compartir memoria es un cuello de botella en la escalabilidad
  - Modularidad y heterogeneidad: Los sistemas distribuidos muestran mayor flexibilidad
  - Compartición de Datos: Los sistemas distribuidos proveen datos distribuidos
  - Compartición de Recursos: Un nodo puede compartirse para varias organizaciones
  - Estructura Geográfica: La estructura geográfica de una aplicación puede ser inherentemente distribuida

## Sistemas Distribuidos vs. Sistemas Paralelos

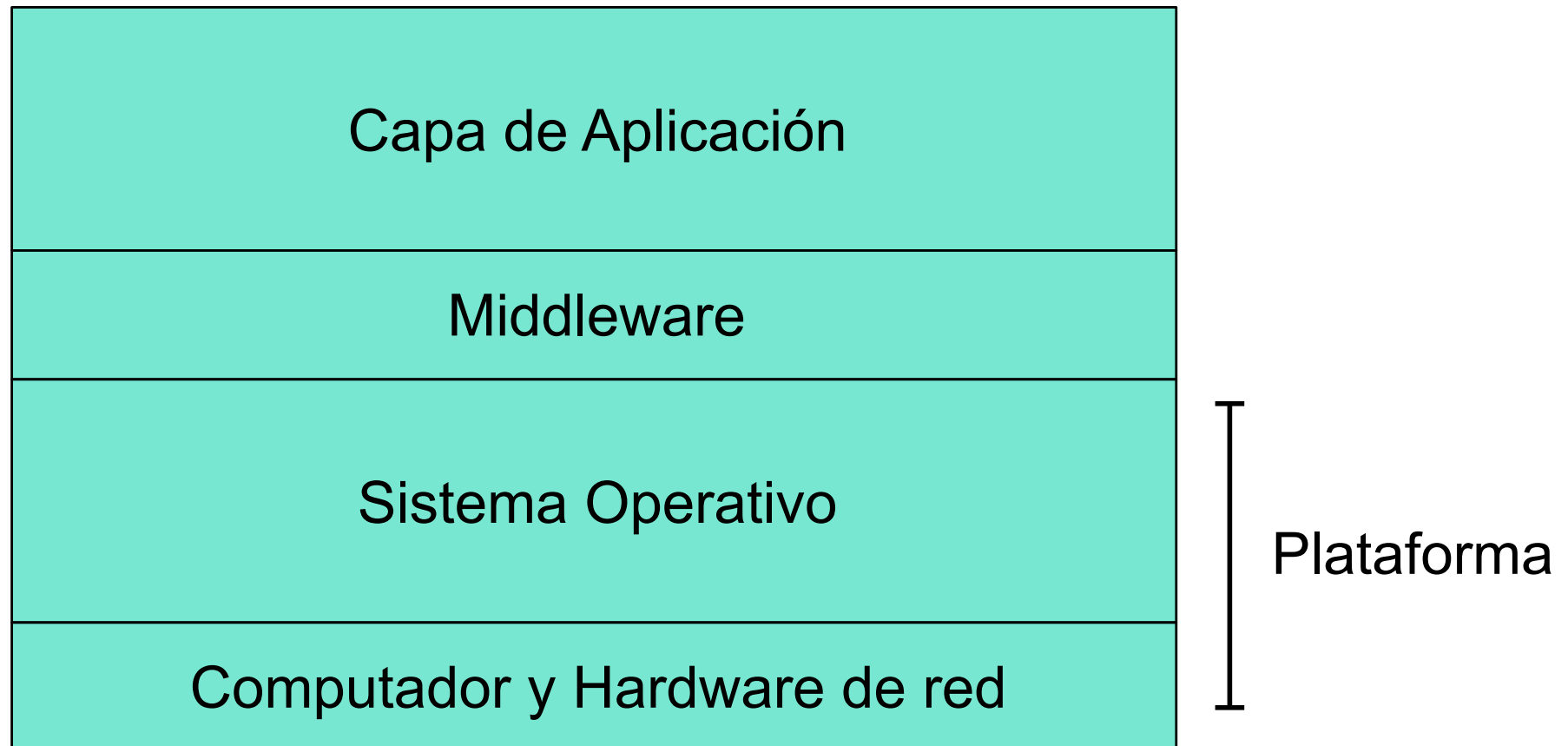
- Fiabilidad: Los sistemas distribuidos son más confiables porque un fallo en un nodo no tiene porque representar un fallo en todo el sistema
- Bajo coste: El precio de los ordenadores de consumo y la disponibilidad de redes hacen que crear un sistema distribuido sea bastante más barato

## Introducción

- Un modelo arquitectónico distribuido trata sobre
  - La ubicación de los componentes en la red de computadoras
  - Las interrelaciones entre los componentes
    - Roles funcionales
    - Patrones de comunicación entre ellos
- Simplificación en cuanto a roles
  - Servidores
  - Clientes
  - Iguales

## Capas de Software

- La arquitectura de un software se refiere a su estructuración como capas o módulos en un único computador
- Plataforma: El nivel de hardware y las capas más bajas del software
- Middleware: Capa de software cuyo propósito es enmascarar la heterogeneidad y proporcionar un modelo de programación para los programadores de la aplicación



## Capas de Software

### – Middleware

- Sun RPC
- CORBA
- RMI
- DCOM
- RM-ODP
- Etc.

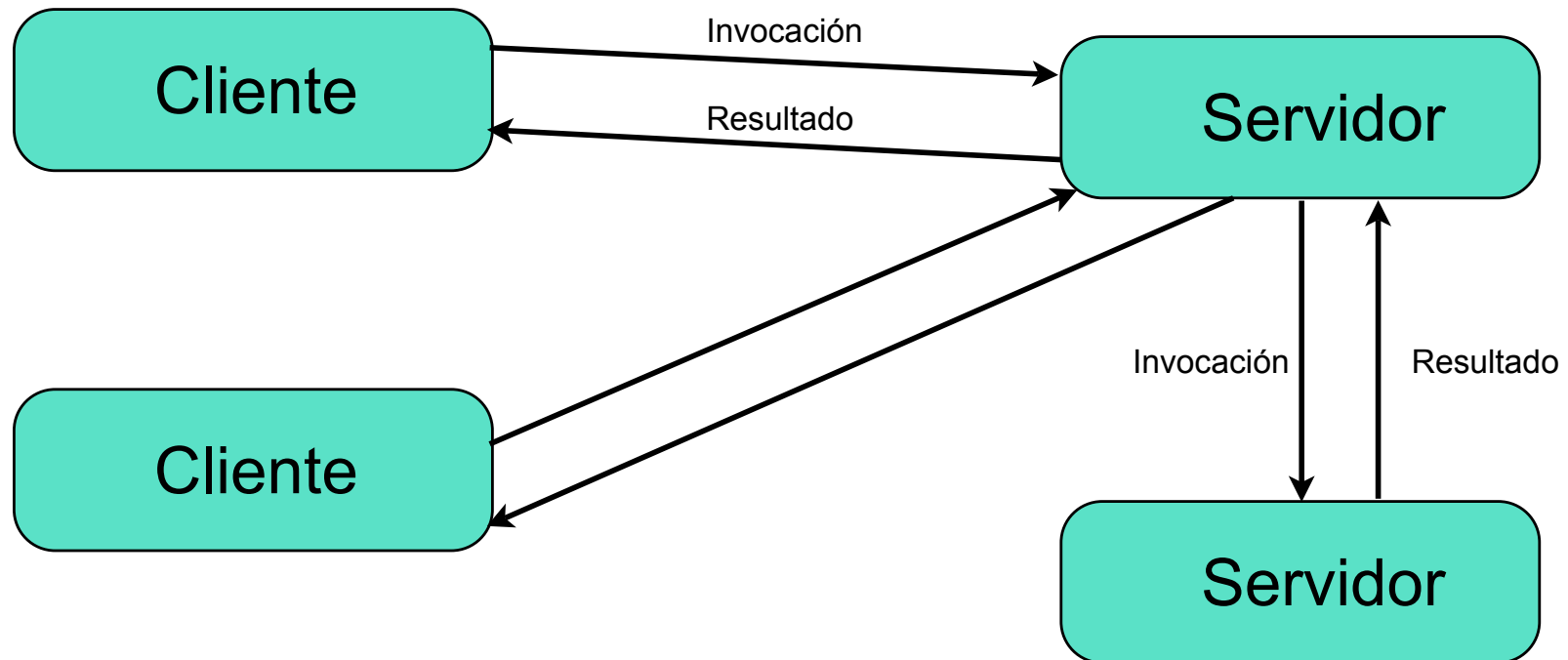
## Arquitecturas de Sistemas

- Modelo cliente-servidor
- Servicios proporcionados por múltiples servidores
- Servidores proxy y cachés
- Procesos peer to peer (de igual a igual)



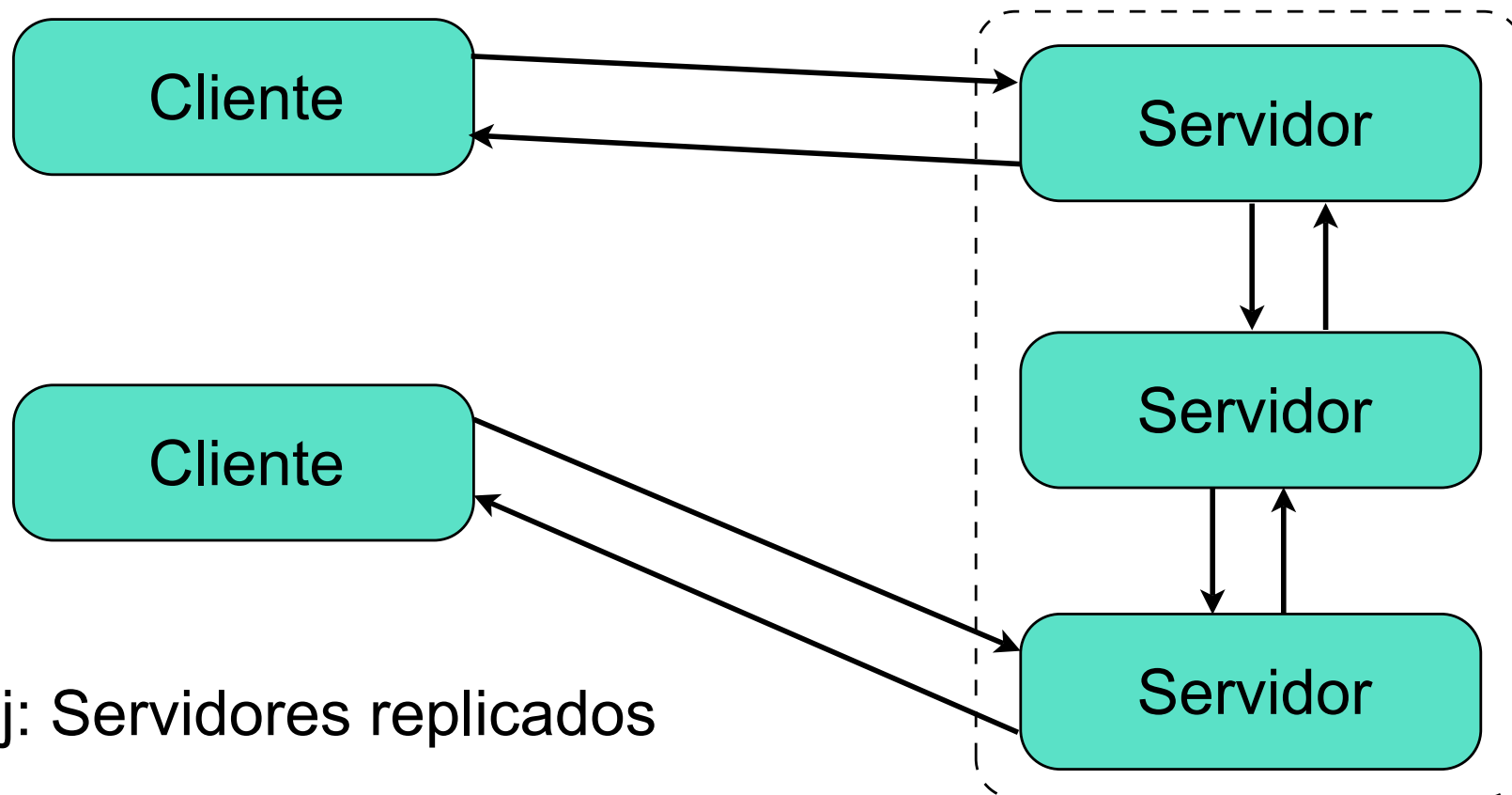
## Arquitecturas de Sistemas

### – Cliente-Servidor



## Arquitecturas de Sistemas

- Servicios proporcionados por múltiples servidores



Ej: Servidores replicados

## Arquitecturas de Sistemas

### – Servidores Proxys y Cachés

