

## GRABACIÓN DE CLASE

# Sistemas distribuidos

elementos de computación autónomos  
(requiere coordinación distribuida)

- NO hay latencia 0
- La red NO es segura
- la topología cambia

## Modelos de SD

### 1) Modelo Sincronismo:

síncronico Vs Asíncronico, tiempo, concurrencia, etc.

### 2) Modelo de fallas:

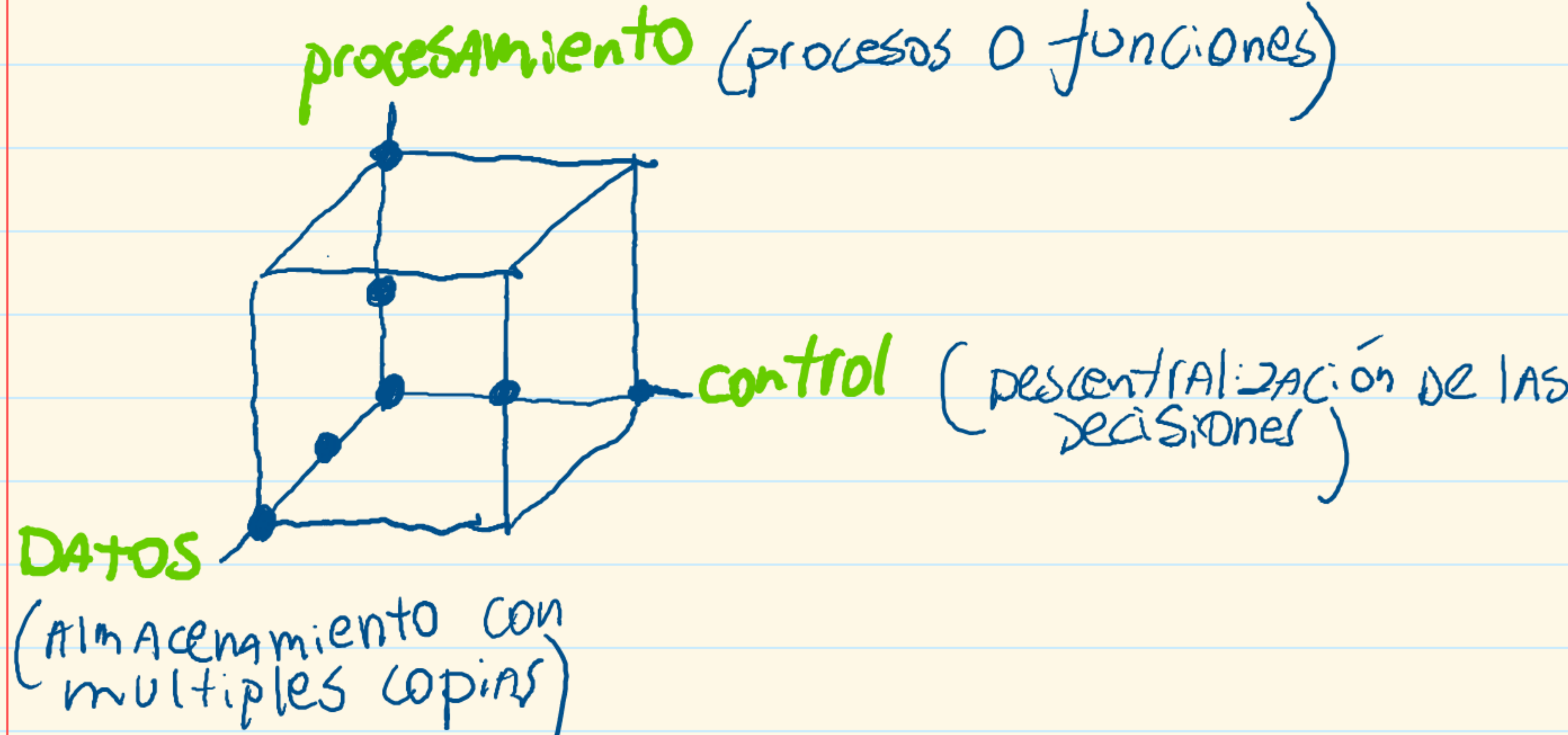
fallas de proceso, comunicación, etc.



### 3) Modelo de Seguridad

nodos estén haciendo cosas que no deberían hacer  
(nodos maliciosos)

¿Que se distribuye?



## Modelo C-S (cliente-servidor)

Cliente invoca al servidor, servidor le responde

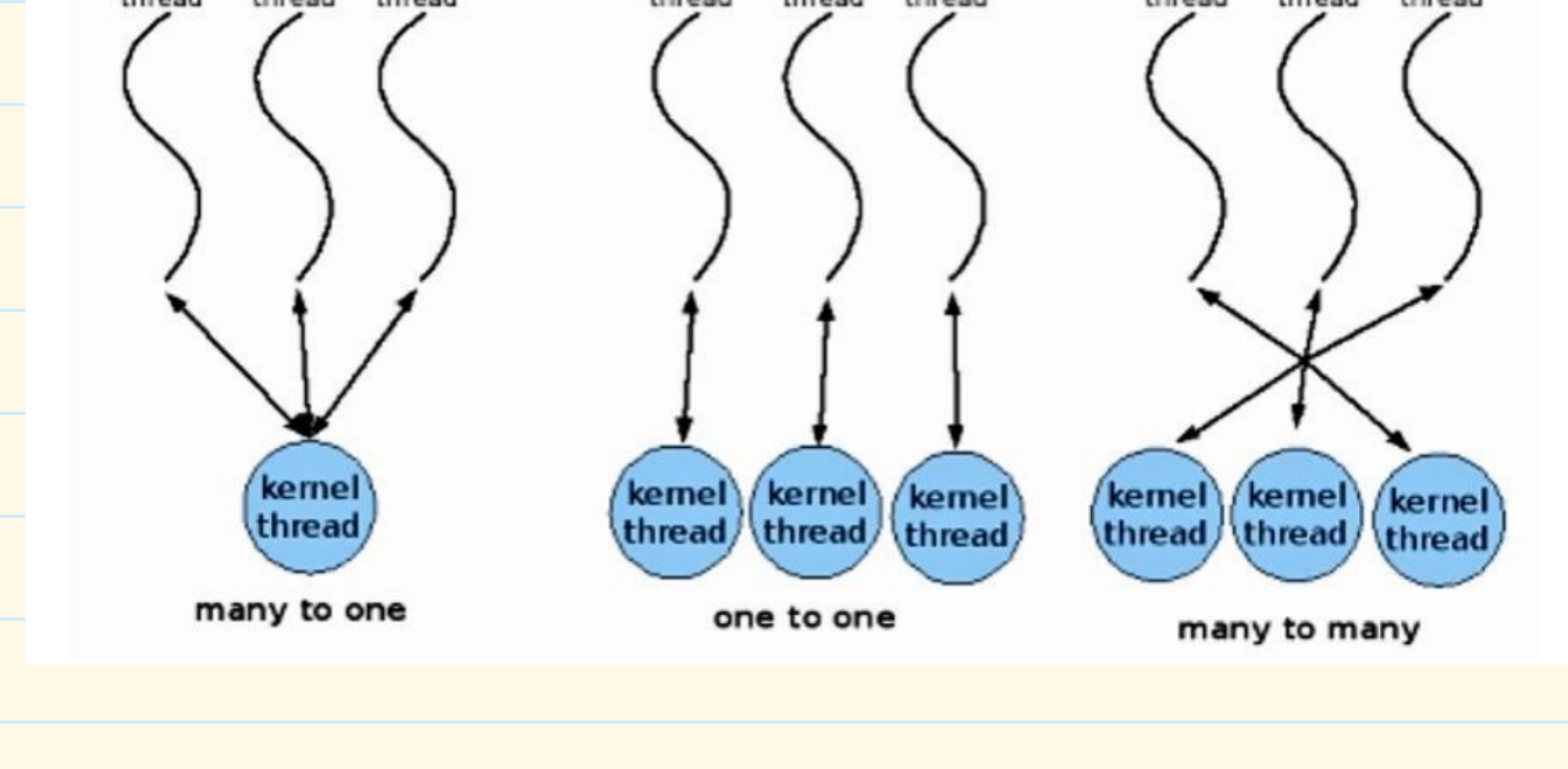
## Estructura de proceso

### proceso:

instancia de un programa en ejecución  
proceso puede realizar al mismo tiempo varias tareas.

### thread (hebra o hilo)

representan a un procesador virtual



## SO en Ambientes distribuidos

### 1) sist. oper. de red (NOS)

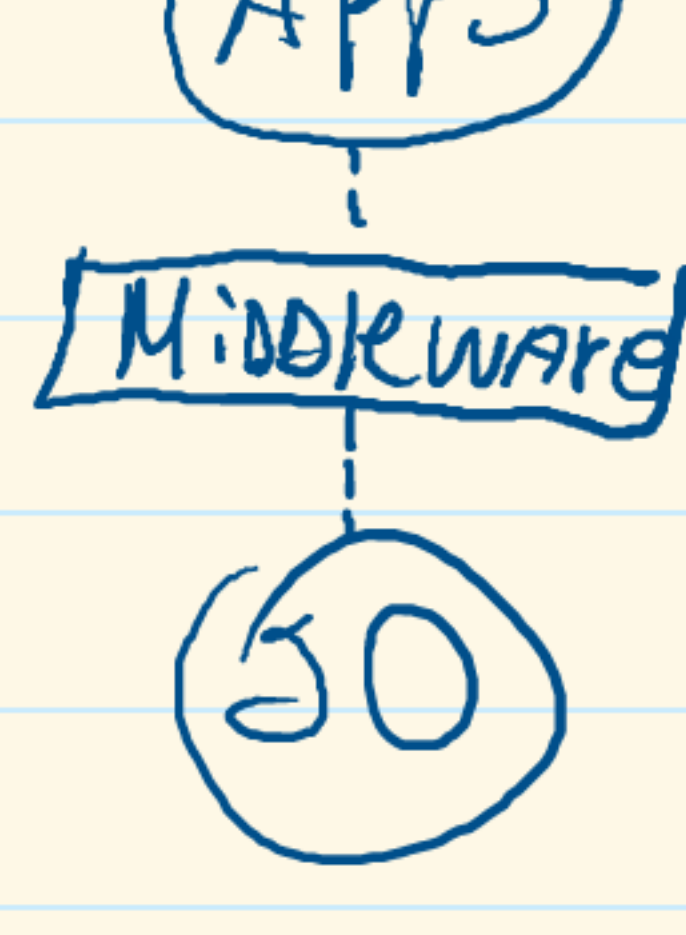
red con S.O. heterogéneo.

### 2) Sist. oper. distribuido (DOS)

S.O. gestiona muchos pcs, basados en microkernel

### 3) Middleware

basada en NOS, pero transparencia de distribución  
para la aplicación.  
CAPA de SW que se encuentra entre el S.O. y las apps. en cada sitio del sist.



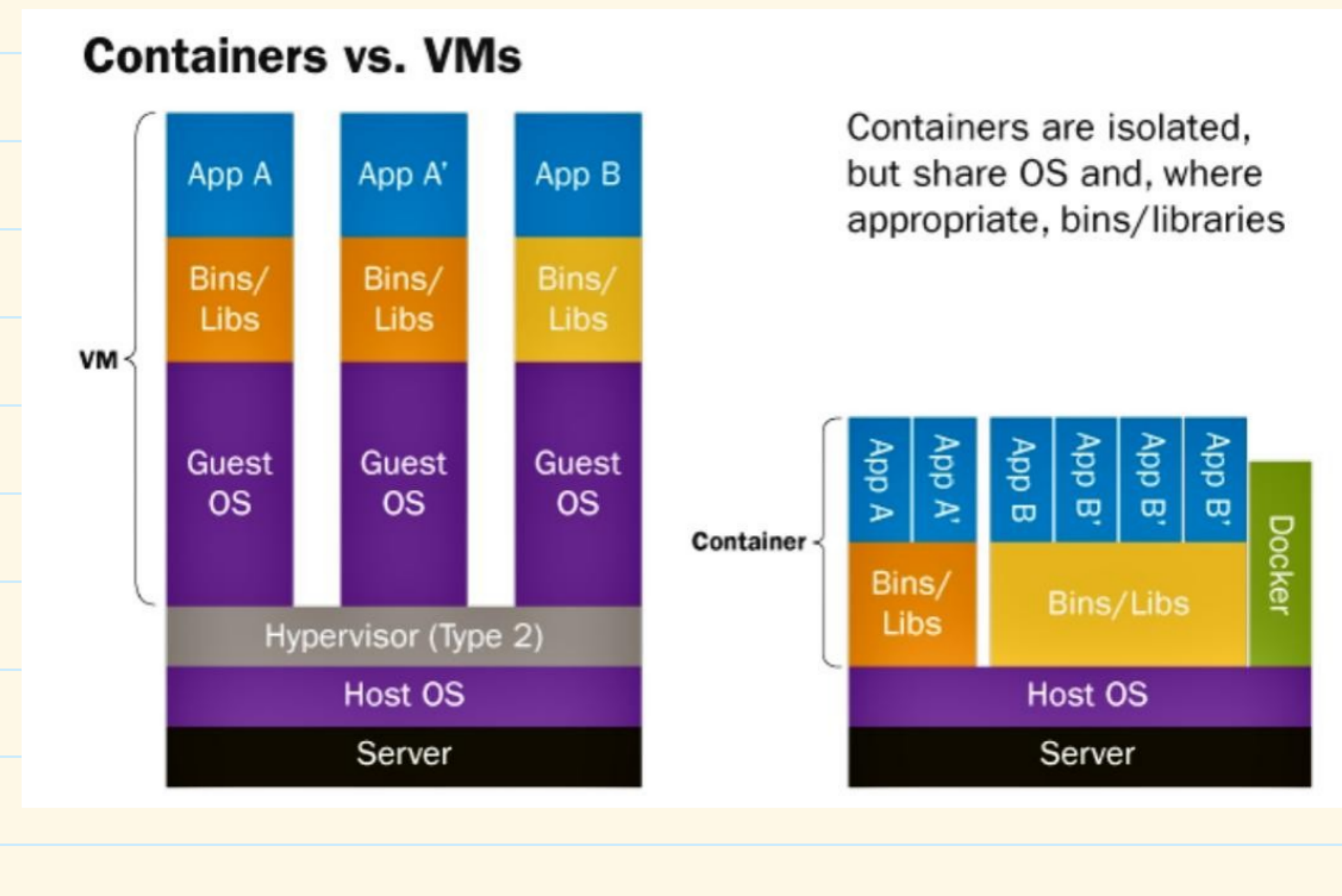
## Virtualización

crear una versión virtual de un recurso.

**Snapshot:** Grabar el estado de un sistema en un determinado tiempo, para volver ahí en caso de cualquier problema.

**contenedores:** Agrupar procesos y sobre eso compartir recursos.

## VMs V/S Contenedores



## C-S V/S P2P

- |  |   |
|--|---|
| <ul style="list-style-type: none"> <li>• estructura simple</li> <li>• Distribuir roles y responsabilidades</li> <li>• Controlar Acceso a recursos o servicios</li> </ul> | <ul style="list-style-type: none"> <li>• roles y privilegios similares</li> <li>• NO existe coordinación central</li> <li>• Mjr tiempo de respuesta.</li> </ul> |
|--|---|
- Estructurada      NO estructurada