

# Aplicaciones distribuidas en sistemas embebidos



Informática III

DSI – EIE - FCEIA

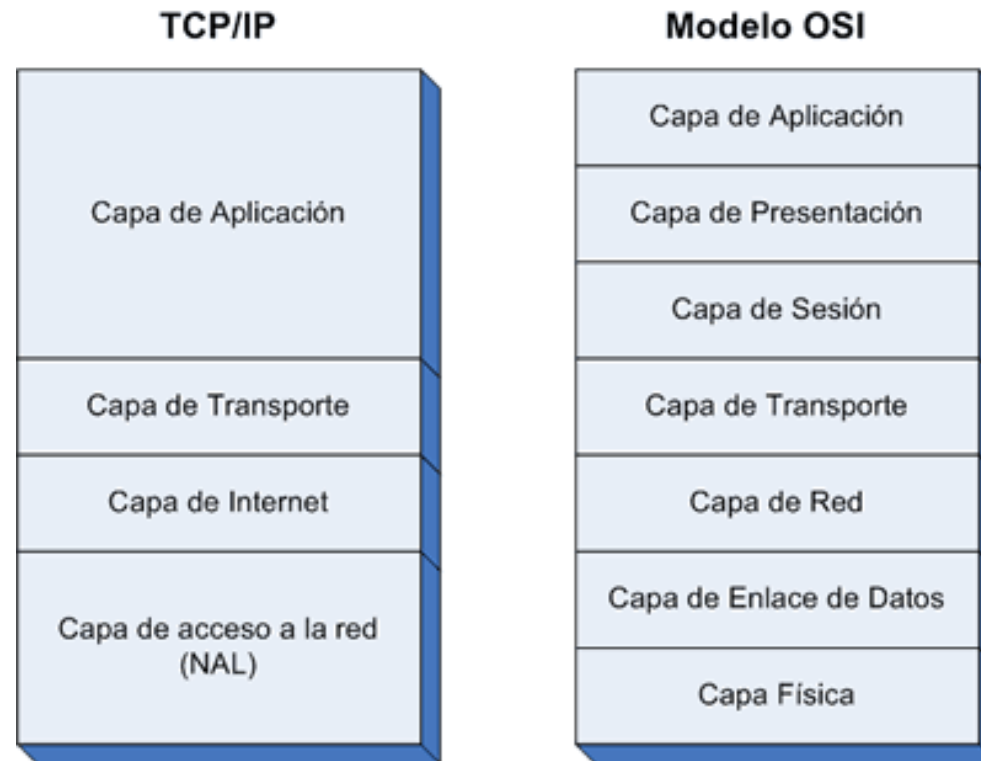


# Modelo conceptual

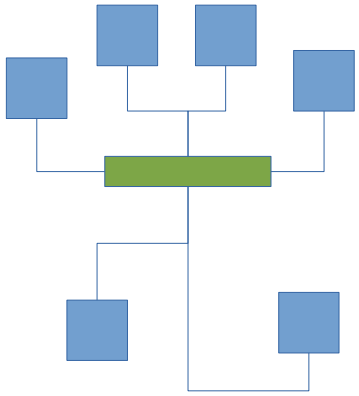
Una aplicación distribuida es un **conjunto de piezas de software** ejecutado en dos o mas dispositivos interconectados.

Este conjunto cumple su propósito mediante la cooperación de sus distintos componentes, utilizando los enlaces de datos subyacentes para el intercambio de mensajes entre los dispositivos.

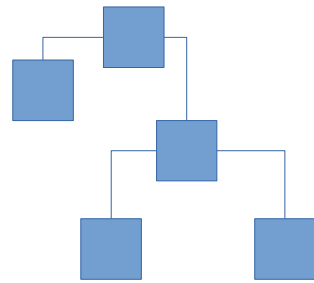
# Modelo conceptual



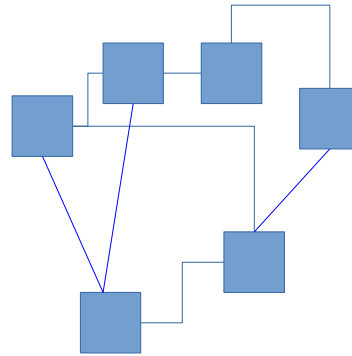
# Capa física: topologías de interconexión



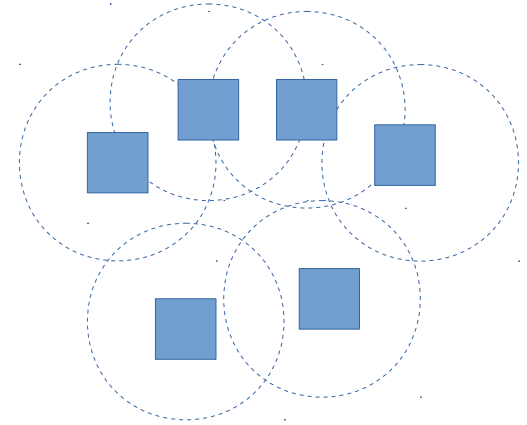
Estrella



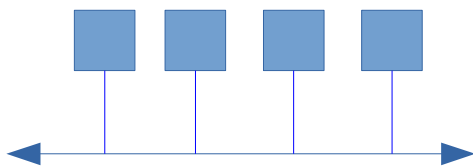
Árbol



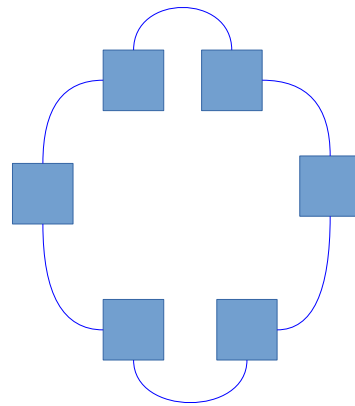
Malla



Malla inalámbrica



Bus



Anillo

Aplicaciones Distribuidas en S.E.

Ad-Hoc: combinación de dos o mas topologías básicas

# Tecnologías de capa física

- Cableado:
  - Cobre (RS-232C, USB, Ethernet UTP)
  - Óptico
- Inalámbrico:
  - RF (Wi-Fi, Bluetooth, ZigBee,...)
  - Infrarrojo
  - Óptico (Li-Fi ultravioleta)

# Factores importantes

- Cada topología tiene pros y contras
- Debemos elegir una arquitectura de red que brinde el ancho de banda y la latencia adecuadas para los propósitos del sistema a diseñar
- Mayor ancho de banda implica generalmente mayor consumo de energía
- No siempre es aplicable la tecnología inalámbrica



# Capa de enlace

- Constituída por software, tiene la responsabilidad de asegurar la transmisión de datos entre dos puntos interconectados de la red



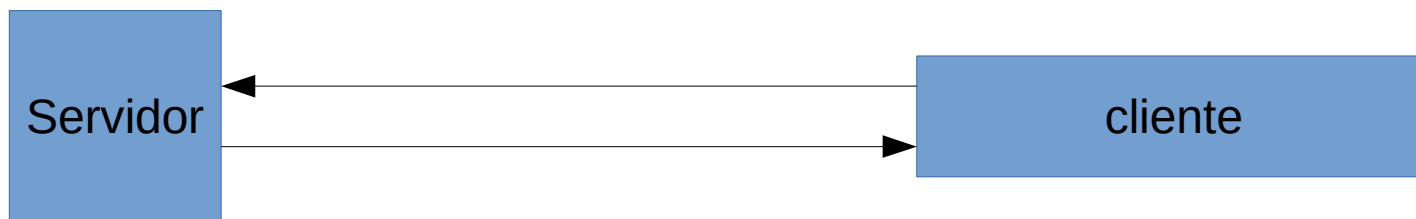
# Capa de Aplicación

- Las aplicaciones distribuidas pueden diseñarse adoptando distintos modelos arquitecturales:
  - Cliente-servidor
  - Productor-consumidor
  - Productor-distribuidor-consumidores
  - Publicador-suscriptor



# Cliente - Servidor

- El servidor ofrece información (servicios)
- El cliente pide esa información mediante requerimientos
- El servidor responde al pedido de uno o mas clientes
- La interacción es iniciada por el cliente

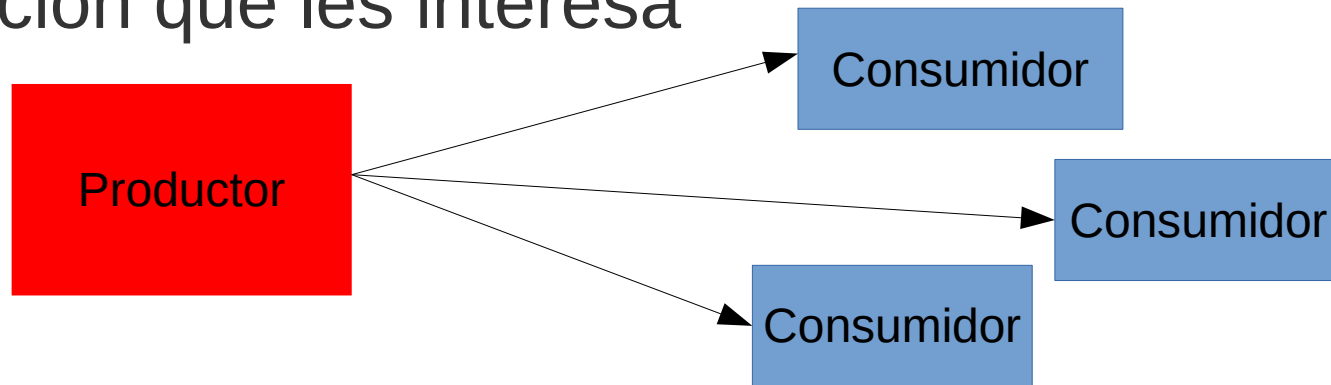


# Ciente - Servidor

- La interacción puede ser en dos modalidades:
  - **Bloqueante** (sincrónica): el cliente envía el requerimiento y detiene su procesamiento hasta la llegada de la respuesta del servidor
  - **No bloqueante** (asincrónica): el cliente envía el requerimiento y continúa su procesamiento normal. Es notificado de la llegada de la respuesta del servidor cuando esta se produce

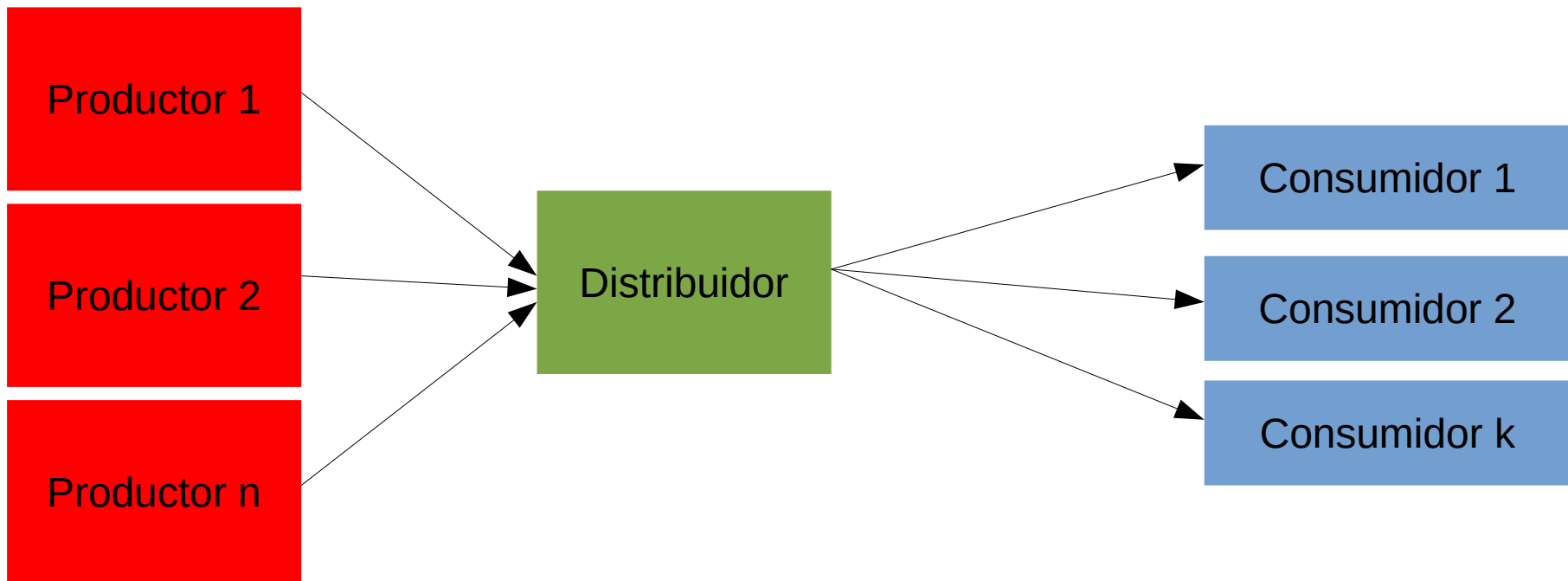
# Productor-consumidor

- Los productores diseminan la información que elaboran
- Los consumidores (suscriptores) usan esa información
- Las transacciones son iniciadas por los productores, que utilizan mecanismos de *broadcast*
- Los consumidores deben *suscribirse* a la información que les interesa



# Productores – distribuidor - consumidores

- Similar al modelo productor-consumidor, con la introducción de un distribuidor que disemina la información de múltiples productores:



# Publicador - Suscriptor

- Similar a productor-consumidores, con restricciones para quienes pueden ser suscriptores
- Utiliza difusión multicast

