

# UF 3.3

## Programación De Comunicaciones En Red

**MODELOS DE COMUNICACIÓN**



**Universidad  
Europea de Madrid**

**LAUREATE** INTERNATIONAL UNIVERSITIES

# CONTENIDOS

## 1. Modelos de Comunicación

- Modelo Cliente/Servidor
- Modelo de comunicaciones en grupo
- Modelo híbridos y P2P





## MODELOS DE COMUNICACIÓN

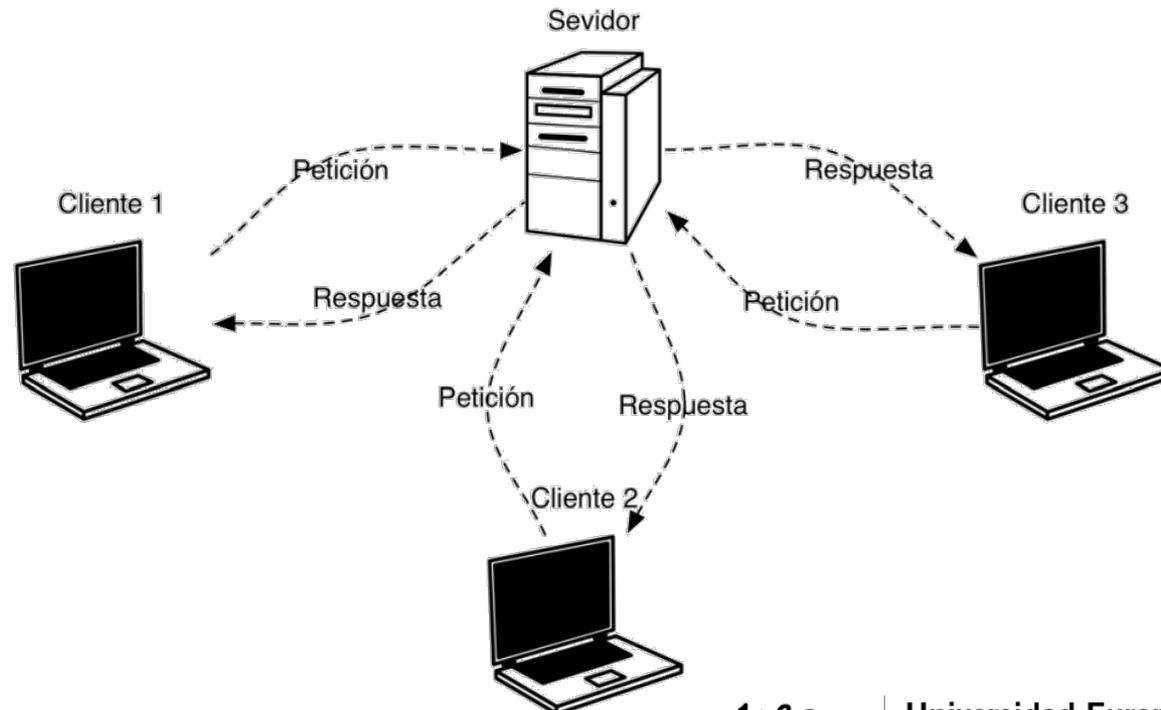
- Los sockets son una herramienta básica para enviar y recibir mensajes. Sobre ellos se desarrollan casi todas las aplicaciones distribuidas que existen en la actualidad.
- Los sockets son sólo la herramienta básica, pero a la hora de desarrollar aplicaciones distribuidas debemos tener en cuenta aspectos de más alto nivel.
- Dependiendo de cuál sea el propósito de nuestra aplicación, y cómo vaya a funcionar internamente, deberemos escoger un modelo de comunicaciones distinto.
- Un modelo de comunicaciones es una arquitectura general que especifica cómo se comunican entre sí los diferentes elementos de una aplicación distribuida.
- Un modelo de comunicaciones normalmente define aspectos como cuántos elementos tiene el sistema, qué función realiza cada uno, etc.
- Los modelos más usados en la actualidad son:
  - Cliente/servidor.
  - Comunicación en grupo
  - Modelos híbridos y P2P (redes peer to peer)



# MODELOS DE COMUNICACIÓN

## Modelo Cliente/Servidor

- Es el más sencillo de los comúnmente usados en la actualidad.
- En este modelo, un proceso central, llamado servidor, ofrece una serie de servicios a uno o más procesos cliente.
- El proceso *servidor* debe estar alojado en una máquina fácilmente accesible en la red, y conocida por los clientes.
- Cuando un *cliente* requiere sus servicios, se conecta con el servidor, iniciando el proceso de comunicación.



## Modelo Cliente/Servidor - Whatsapp

- El modelo cliente/servidor se basa en un mecanismo de comunicación mediante petición y respuesta. Los clientes realizan peticiones al servidor, que las resuelve, devolviendo el resultado al cliente correspondiente en la respuesta.
- Ejemplo típico es la aplicación de mensajería instantánea de **WhatsApp**. Cuando se abre la aplicación cliente, se selecciona un destinatario, se escribe el mensaje y se pulsa en *enviar*. En ese momento la aplicación *cliente* se conecta con el *servidor* a través de internet (alojado en los ordenadores centrales de Whatsapp) y le envía un mensaje indicando el texto escrito, el destinatario y demás información. El *servidor* recibe el mensaje, lo procesa y responde al *cliente* indicando si se ha podido o no hacerlo llegar a la persona a la que iba destinado.
- Es fácil darse cuenta de que los conceptos básicos de este modelo se parecen mucho a los de los sockets stream.

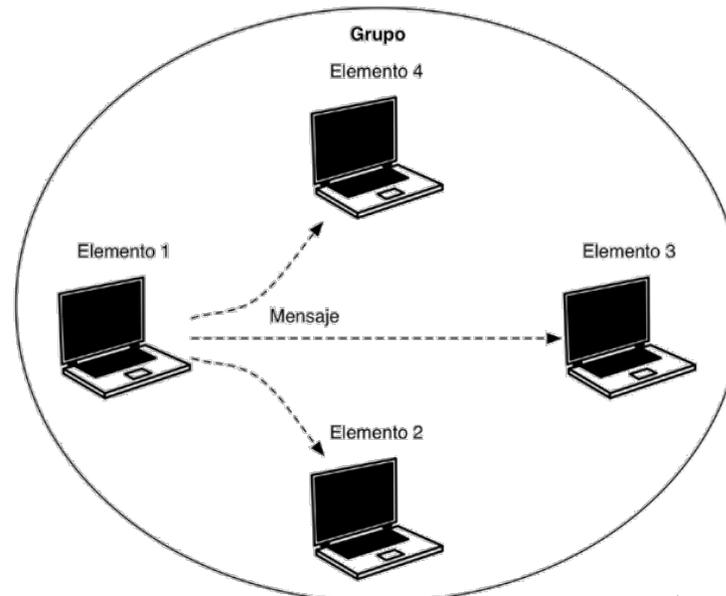




# MODELOS DE COMUNICACIÓN

## Modelo de comunicación en grupo

- Es la alternativa más común al modelo cliente/servidor.
- A diferencia del anterior, en este modelo no existen roles diferenciados.
- En la comunicación en grupo existe un conjunto de dos o más elementos (procesos, aplicaciones, etc.) que cooperan en un trabajo común.
- A este conjunto se le llama grupo, y los elementos que lo forman se consideran todos iguales, sin roles ni jerarquías definidas.
- Los mensajes se transmiten mediante radiado: Los mensajes se envían de manera simultánea a los distintos miembros del grupo, en lugar de establecer comunicaciones punto a punto (como en el caso anterior).





## MODELOS DE COMUNICACIÓN

### Modelo de comunicación en grupo – Call of Duty

- En sus orígenes la mayoría de los juegos online utilizaban el modelo cliente/servidor. Todos los jugadores debían conectarse a un servidor central, que alojaba la partida y coordinaba la interacción entre ellos. El problema es que dicho servidor era un *punto único de fallo*, y cuando caía la partida se interrumpía y todos los jugadores quedaban fuera.
- En la actualidad muchos juegos online buscan alternativas basadas en la comunicación en grupo (aunque otros como World of Warcraft o Guild Wars siguen con el modelo anterior).
- Ejemplos de comunicación en grupo son Call of Duty o Battlefield, que buscan mayor tolerancia a fallos transmitiendo la información entre los jugadores de una misma partida.

CALL OF DUTY



## MODELOS DE COMUNICACIÓN

### Modelo de comunicación en grupo - Implementación

- Este modelo se puede implementar utilizando:
  - Sockets stream: Donde cada elemento del grupo establece una conexión con todos los demás. Para ello debe tener un socket distinto con cada uno de los demás miembros. Esto implica la creación de gran cantidad de sockets y conexiones, pero son muy fiables.
  - Sockets datagram: Sólo es necesario un único socket, pero es menos fiable.
- Además de estos procedimientos, el protocolo IP incorpora un mecanismo automático que se puede utilizar para realizar operaciones de comunicación en grupo, denominado **multicast** (los anteriores eran todos *unicast*). Su uso permite la comunicación en grupo de forma eficiente, aunque supone un riesgo para la seguridad y además puede saturar las redes, por lo que habitualmente está restringido a redes LAN. Su funcionamiento es el siguiente:
  - Envío: Se utiliza una dirección IP especial que automáticamente se replica y entrega a cada uno de los destinatarios. En IPv4 se reserva una serie de direcciones que van desde 224.0.0.0 – 239.255.255.255
  - Recepción: Para poder recibirlo, los sockets de todos los procesos receptores deben estar configurados utilizando la misma dirección ip de multicast



# MODELOS DE COMUNICACIÓN

## Modelos híbridos

### Introducción

- Las aplicaciones distribuidas más avanzadas suelen tener requisitos de comunicaciones muy complejos, que requieren de modelos de comunicaciones sofisticados.
- En muchos casos, los modelos de comunicaciones reales implementados en estas aplicaciones mezclan conceptos del modelo cliente/servidor y la comunicación en grupo, dando lugar a enfoques híbridos, como las redes peer-to-peer (P2P).

### Limitaciones de los modelos anteriores

- Reparto de roles fijo: La función de un elemento del sistema no cambia con el tiempo, pase lo que pase, resultando muy poco flexible y restringiendo las posibilidades de la aplicación.
- Mecanismo de comunicación único: Los elementos que forma el sistema distribuido se comunican siempre de la misma forma.

### Resumen

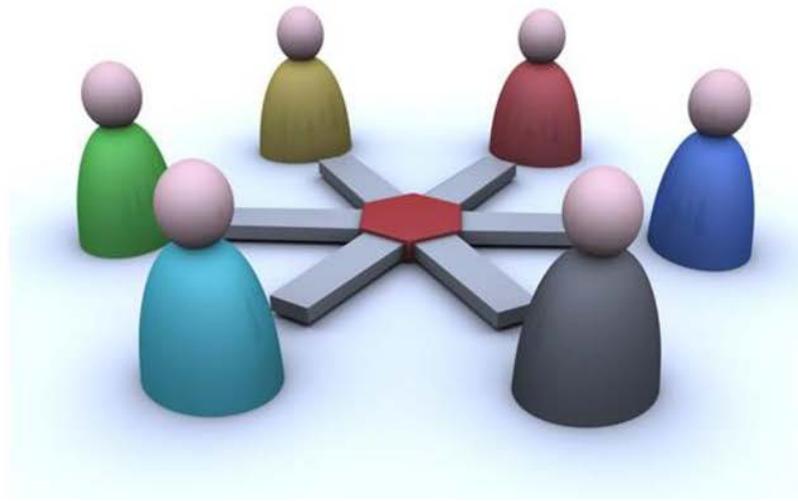
- Los modelos vistos hasta ahora son potentes pero poco flexibles.



# MODELOS DE COMUNICACIÓN

## Modelos híbridos - Ejemplo

- Pensemos en el caso del juego online. En él, es necesario disponer de un punto de encuentro central, donde los jugadores puedan iniciar la partida. Para esto es necesario un modelo cliente/servidor.
- Una vez creada la partida, mantenerla alojada en el servidor puede ser un problema. En lugar de eso, la partida puede pasar a al modelo de comunicaciones en grupo (puesto que los jugadores ya se conocen).
- Por lo tanto, en este modelo híbrido el servidor actúa sólo como mecanismo de enlace. Una vez que comienza la partida, aunque el servidor se cayese la partida seguiría.





## MODELOS DE COMUNICACIÓN

### Modelos híbridos – Redes peer-to-peer

- Las redes peer-to-peer son uno de los modelos de comunicación híbrida más potentes que existen en la actualidad. Deben su fama a las redes de intercambio de archivos, como el antiguo **Napster** o el más reciente **BitTorrent**, pero también se utilizan con éxito en aplicaciones como **Spotify** y **Skype**.
- Una red P2P está formada por un grupo de elementos distribuidos que colaboran con un objetivo común.

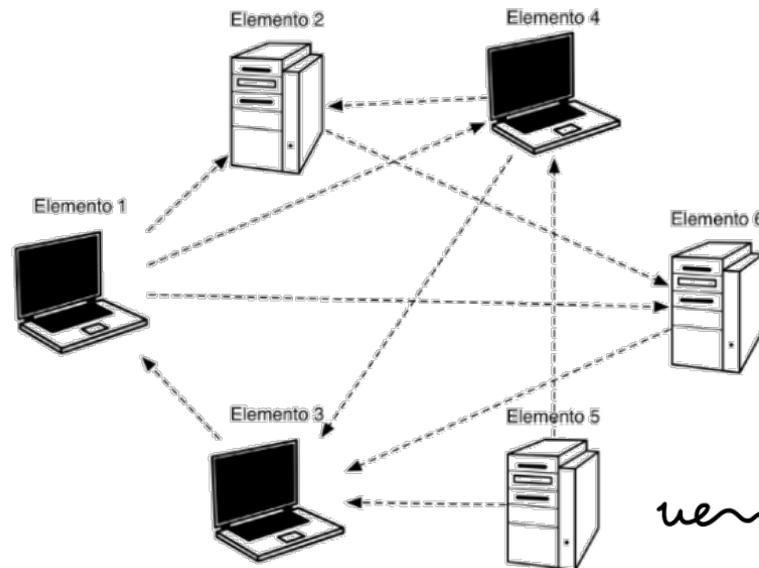




# MODELOS DE COMUNICACIÓN

## Modelos híbridos – Redes peer-to-peer

- La diferencia fundamental con el modelo de comunicación en grupo convencional es que cualquier elemento puede desempeñar los roles de servidor o cliente, como si de un modelo cliente/servidor se tratase.
- Cualquier aplicación puede conectarse a la red como un *cliente*, localizar un servidor y enviarle una petición.
- Si permanece en la red P2P, con el tiempo ese mismo cliente puede hacer a su vez de *servidor* para otros elementos de la red.
- Su ventaja fundamental es que eliminan los problemas fundamentales del modelo cliente/servidor tradicional, al no existir un único servidor. Esto hace que el sistema sea mucho más tolerante a fallos y además permite repartir la carga de forma más equilibrada.





**Universidad  
Europea**

**LAUREATE** INTERNATIONAL UNIVERSITIES

Madrid

Valencia

Canarias