

upna

Universidad Pública de Navarra  
Nafarroako Unibertsitate Publikoa

ARQUITECTURA DE REDES, SISTEMAS Y SERVICIOS  
*Área de Ingeniería Telemática*

# Arquitecturas de conmutación y protocolos

Area de Ingeniería Telemática  
<http://www.tlm.unavarra.es>

Arquitectura de Redes, Sistemas y Servicios

upna

Universidad Pública de Navarra  
Nafarroako Unibertsitate Publikoa

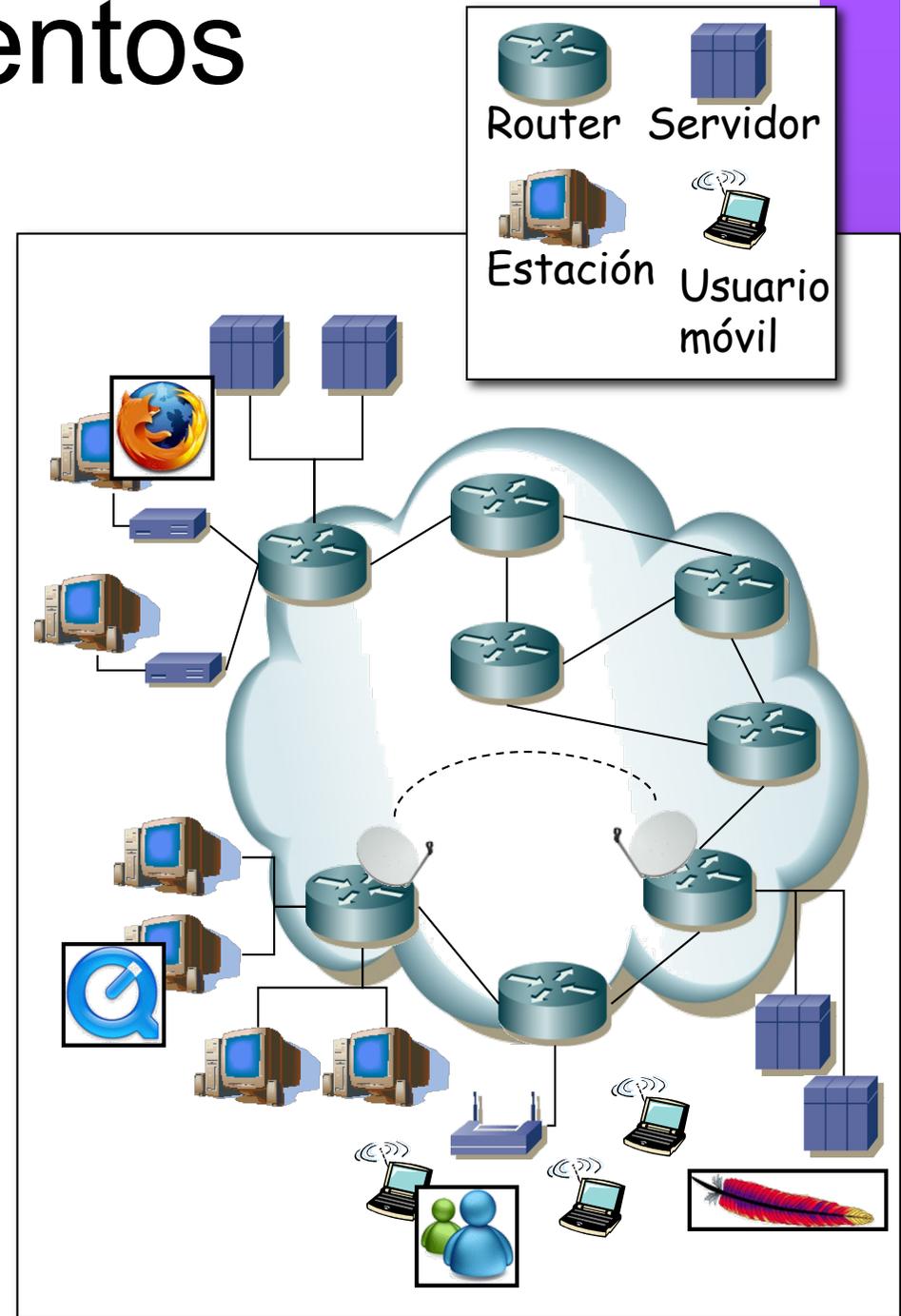
**ARQUITECTURA DE REDES, SISTEMAS Y SERVICIOS**  
*Área de Ingeniería Telemática*

# Elementos de la red

# Elementos

## *Hosts = end systems*

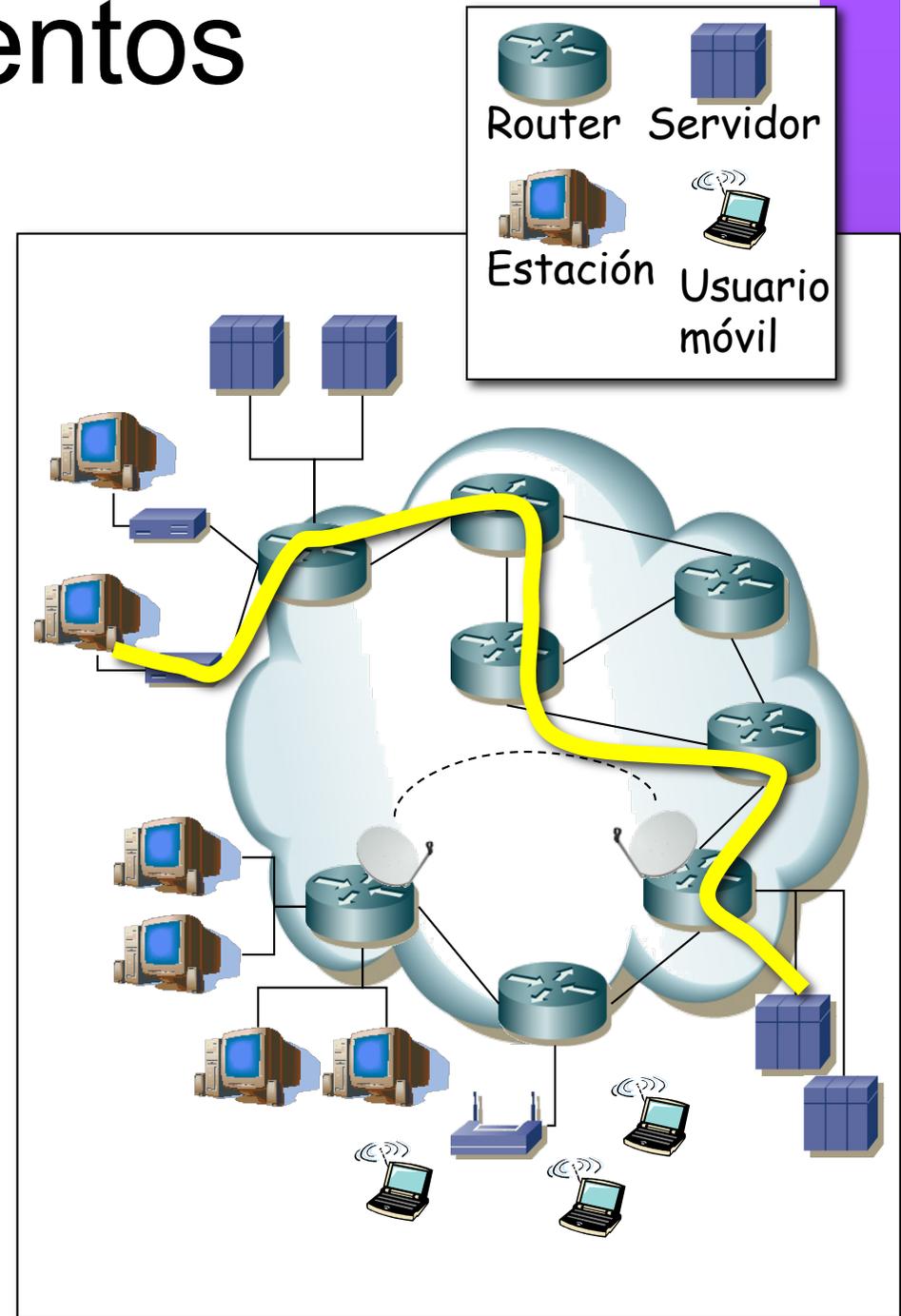
- PCs, estaciones, teléfonos, PDAs, servidores, cámaras, TVs, etc.
- Ejecutan *aplicaciones de red (...)*
- Forman el borde (*edge*) de la red
- Conectados con la red mediante *enlaces de comunicaciones*
  - Fibra, cobre, radio, satélite
  - Tasa de transmisión (b/s)  $\cong$  *ancho de banda (bandwidth)*



# Elementos

## Conmutadores

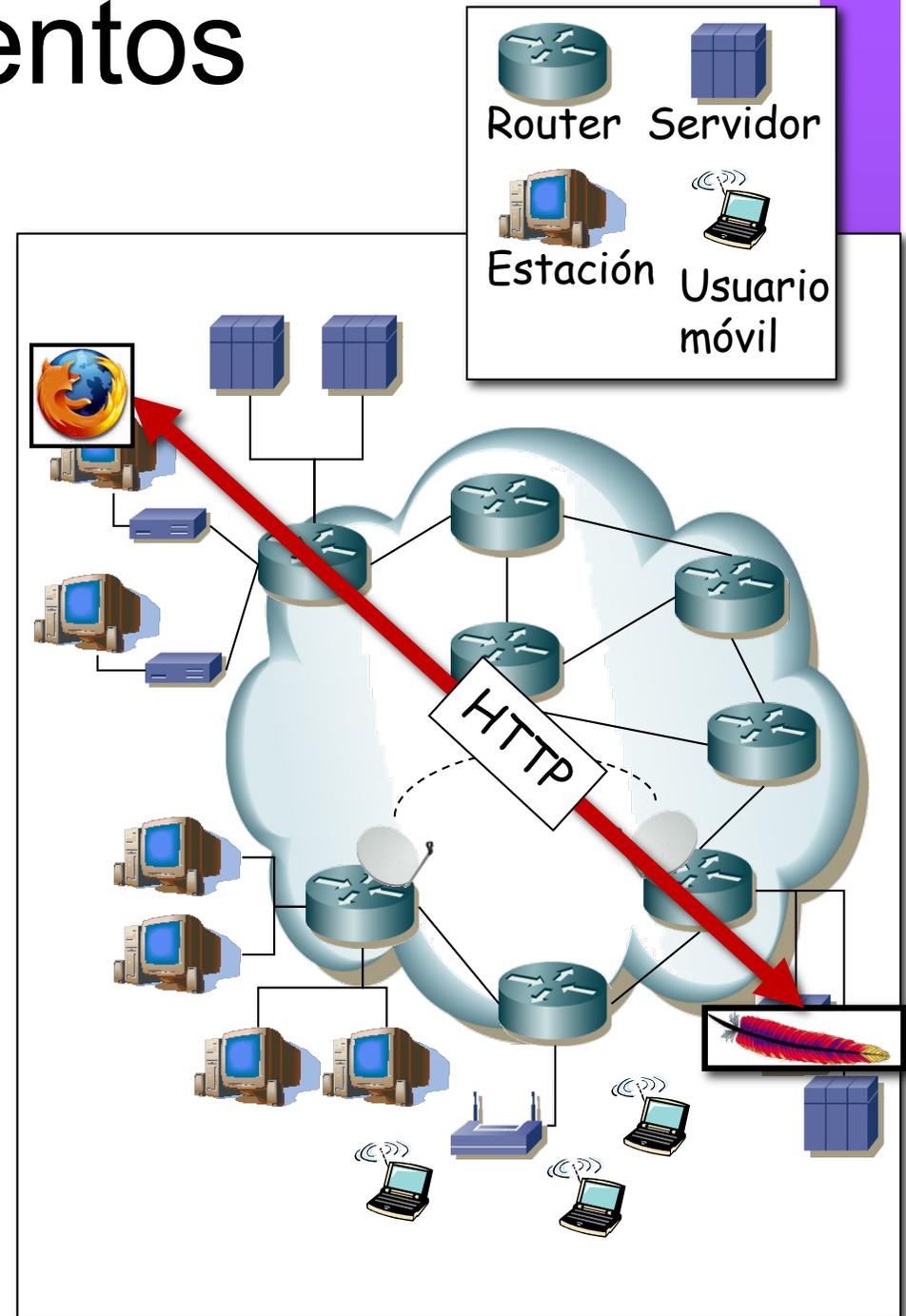
- Reenvían la información
- Hoy en día información digital
- Transparente a los datos
- Podría ser transparente a si la información es analógica o digital (conmutación óptica)
- Conmutadores telefónicos
- Routers en el caso de Internet
- Conmutadores de enlace
- Interconectados mediante enlaces de comunicaciones
- Forman el núcleo (*core*) de la red
- Emplean rutas o caminos (*paths*) dentro de la red (...)



# Elementos

## Protocolos

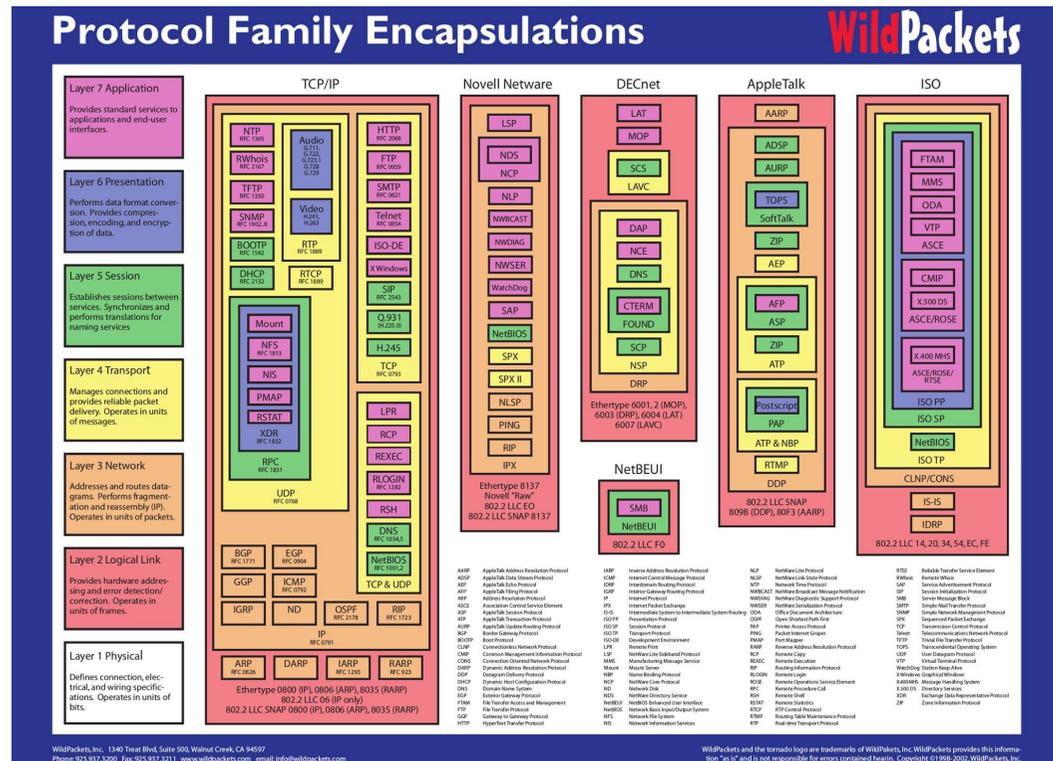
- Controlan el envío y la recepción de información
- Entre las aplicaciones (HTTP, FTP)
- Entre los hosts (UDP, TCP)
- Entre los nodos (IP, ICMP)



# En esta asignatura

- No estudiaremos los enlaces
- Sí empezaremos a ver cómo funcionan algunos tipos de conmutadores
- Y veremos algunos protocolos... porque hay muchos
- Y resuelven muchos problemas... veremos algunos de ellos

HTTP, HTTPS, FTP, SFTP, SMTP, IMAP, POP3, DNS, DHCP, IP, TCP, UDP, ICMP, ARP, RARP, BGP, OSPF, RIP, SSH, TELNET, SNMP, NTP, TFTP, LDAP, SIP, RTP, RTSP, SCTP, PPP, MPLS, GRE, L2TP, IPsec, SSL, TLS, VTP, IS-IS, IGRP, EIGRP, IGMP, PIM, STP, RSTP, MSTP, H.323, NFS, SMB, CIFS, PPPoE, PPPoA, ATM, FDDI, SLIP, GTP, MTP, XMPP, BFD, DCCP, VRRP, HSRP, LACP, LLDP, CDP, NDP, OTV, VXLAN, GVRP, LDP, DOCSIS, WAP, WPA, WEP, EAP, RADIUS, TACACS+, CHAP, PAP, MGCP, RDP, VNC, CAN, MQTT, CoAP, AMQP, DDS, TRILL, CA with SDH SONET ECoE



upna

Universidad Pública de Navarra  
Nafarroako Unibertsitate Publikoa

**ARQUITECTURA DE REDES, SISTEMAS Y SERVICIOS**  
*Área de Ingeniería Telemática*

# Protocolos

# ¿Qué es un protocolo?

Un protocolo humano:



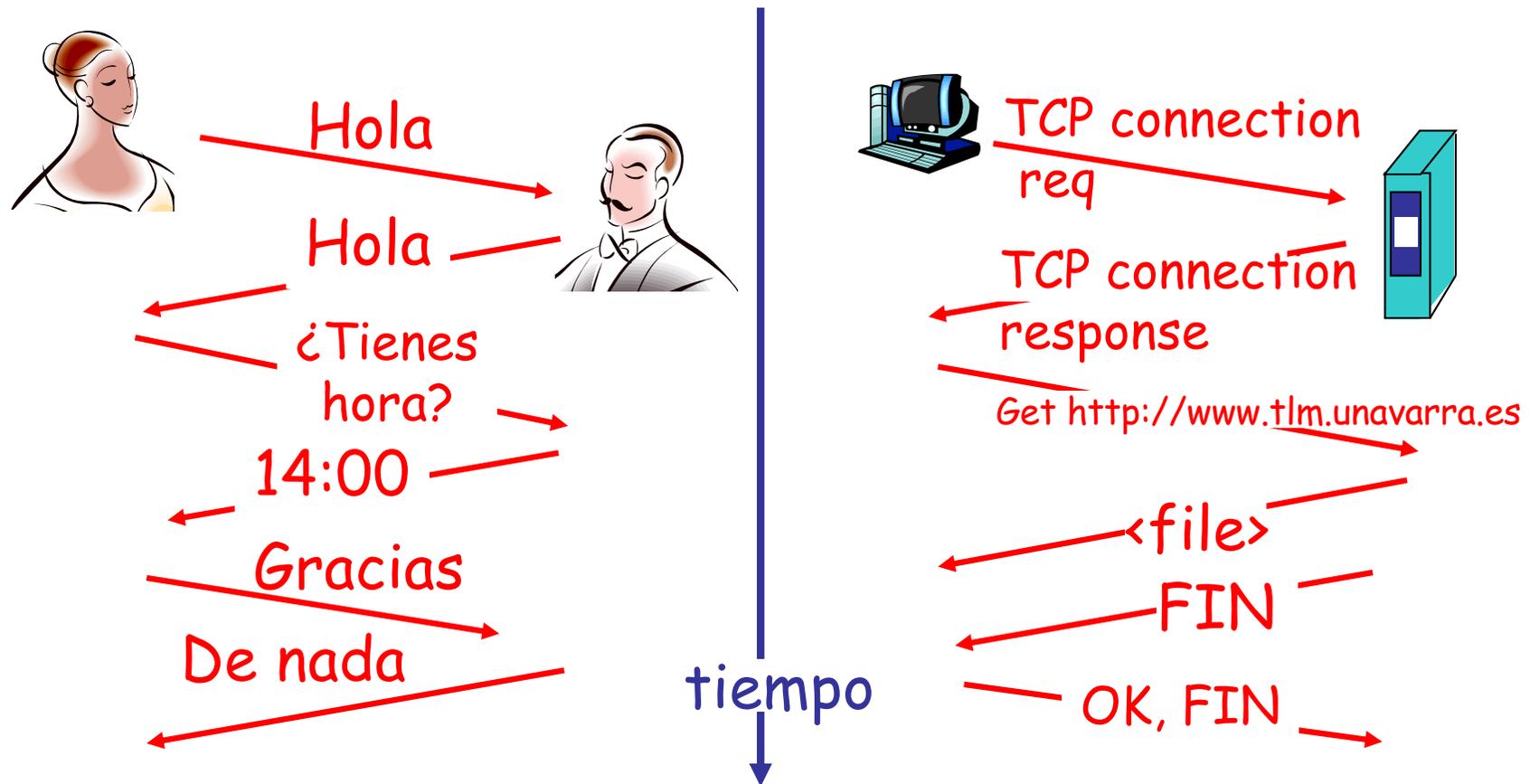
# ¿Qué es un protocolo?

- Todas las comunicaciones están gobernadas por protocolos
- Especifican:
  - Los mensajes a enviar
  - El formato de los mensajes
  - Las acciones a llevar a cabo ante ciertos mensajes o ciertos eventos



# ¿Qué es un protocolo?

Un protocolo humano y uno de redes de ordenadores:



*Aquí hay más de uno, pero ya iremos viéndolos*

# ¿Qué es un protocolo?

- Todas las comunicaciones están gobernadas por protocolos
- Especifican:
  - Los mensajes a enviar
  - El formato y orden de los mensajes
  - Las acciones a llevar a cabo ante ciertos mensajes o ciertos eventos
- Controlan por ejemplo:
  - El formato de los datos por el medio físico
  - El camino que va a seguir un paquete de origen a destino
  - La velocidad a la que se envían datos
  - Cómo se le pide una página web a un servidor
  - Etc.



*IP packet format*

0	3	7	15	31
Version	IHL	Codepoint	Checksum	
Fragment ID		D F F	Fragment offset	
Time to Live	Protocol	Checksum		
Source address				
Destination address				
Options and payload				

upna

Universidad Pública de Navarra  
Nafarroako Unibertsitate Publikoa

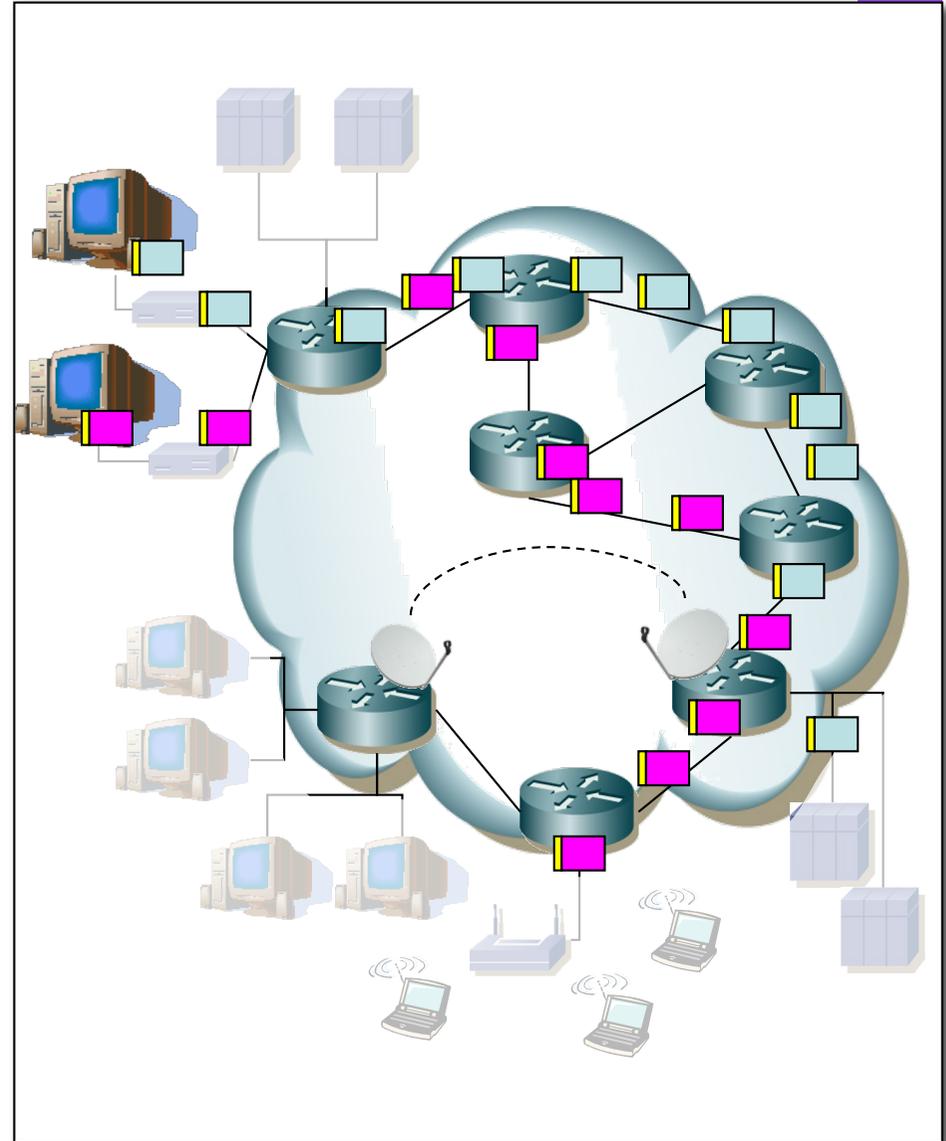
**ARQUITECTURA DE REDES, SISTEMAS Y SERVICIOS**  
*Área de Ingeniería Telemática*

# Arquitecturas de protocolos

# Arquitecturas de protocolos

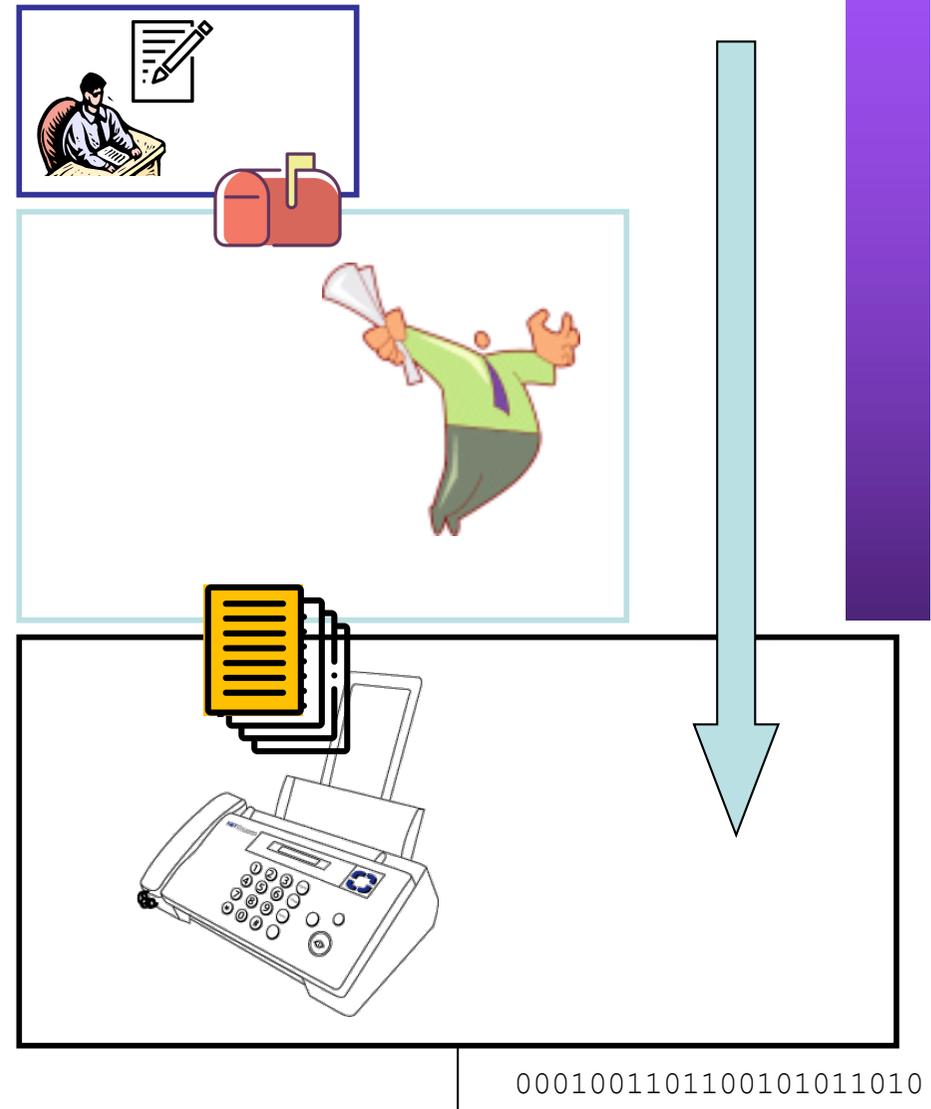
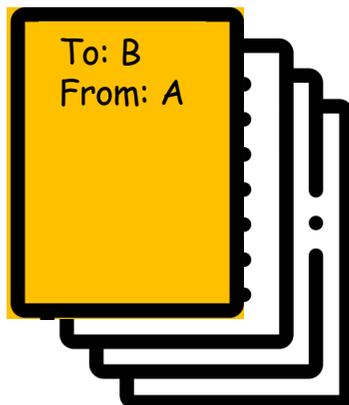
## ¡Las redes son complejas!

- Muchos elementos:
  - Hosts
  - Conmutadores
  - Enlaces de diferente tipo
  - Aplicaciones
  - Hardware, software
- ¿Hay alguna forma de organizar la estructura de la red?
- ¿O al menos la forma de explicarla?



# Analogía

- Usuario escribe una carta
- La deja en su buzón e indica a su asistente para quién es
- El asistente añade una portada indicando el remitente y destinatario
- A esto lo llamaremos “encapsulación”
- Y la envía a la oficina remota mediante un fax



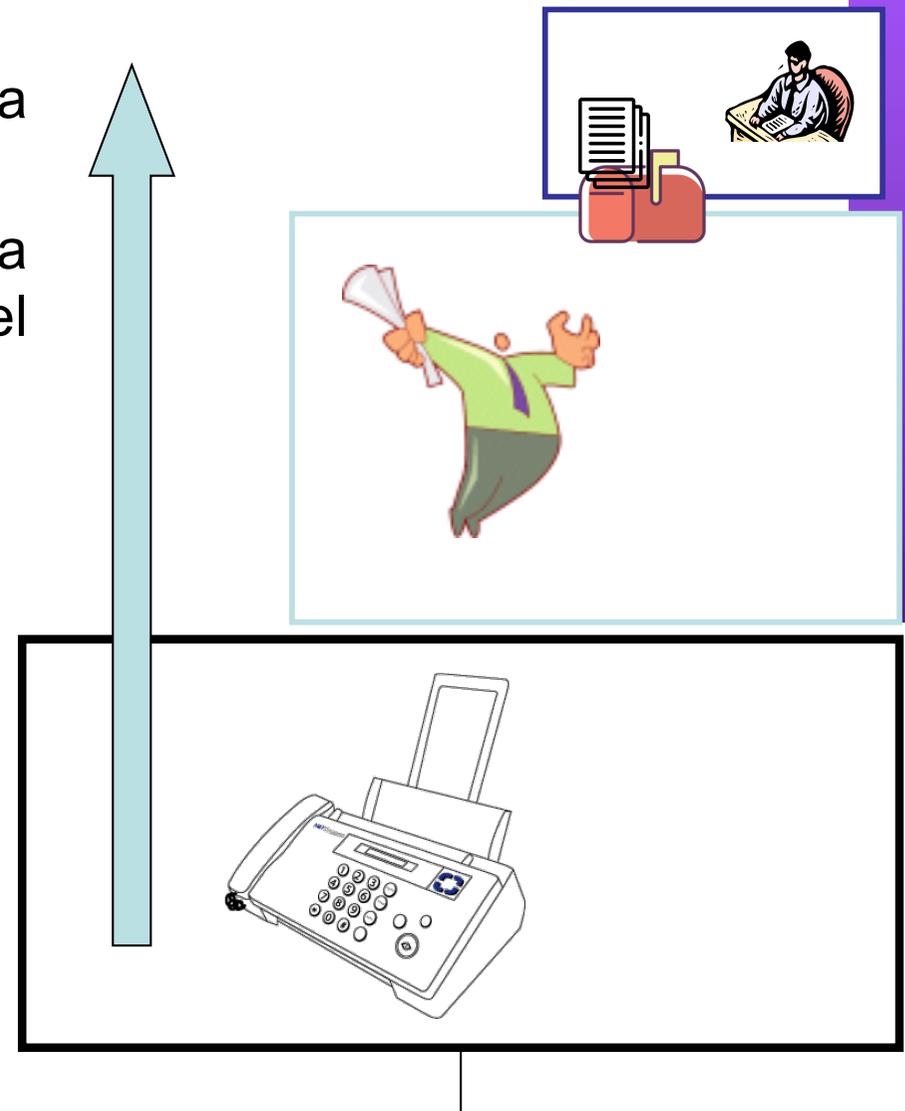
# Interludio histórico

- ¿Quién ha visto alguna vez un fax?
- Un fax escanea las páginas y las manda por una llamada telefónica (como si fuera voz)



# Analogía

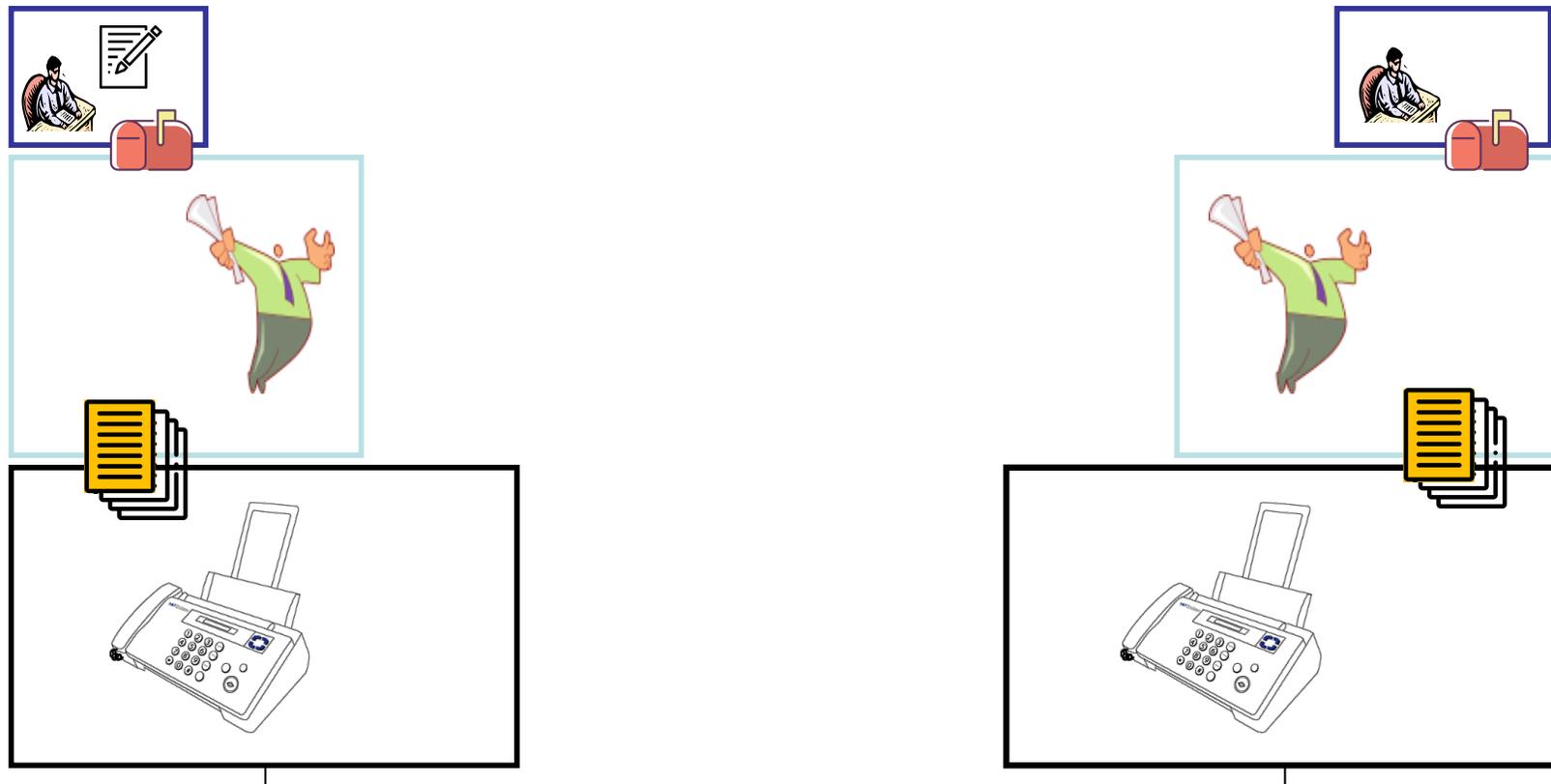
- Llega al fax de la oficina destino
- La recoge el secretario
- Retira la portada (“desencapsulación”)
- En función de esa portada la coloca en el buzón del destinatario
- La recoge el usuario
- Hemos separado las tareas



0001001101100101011010

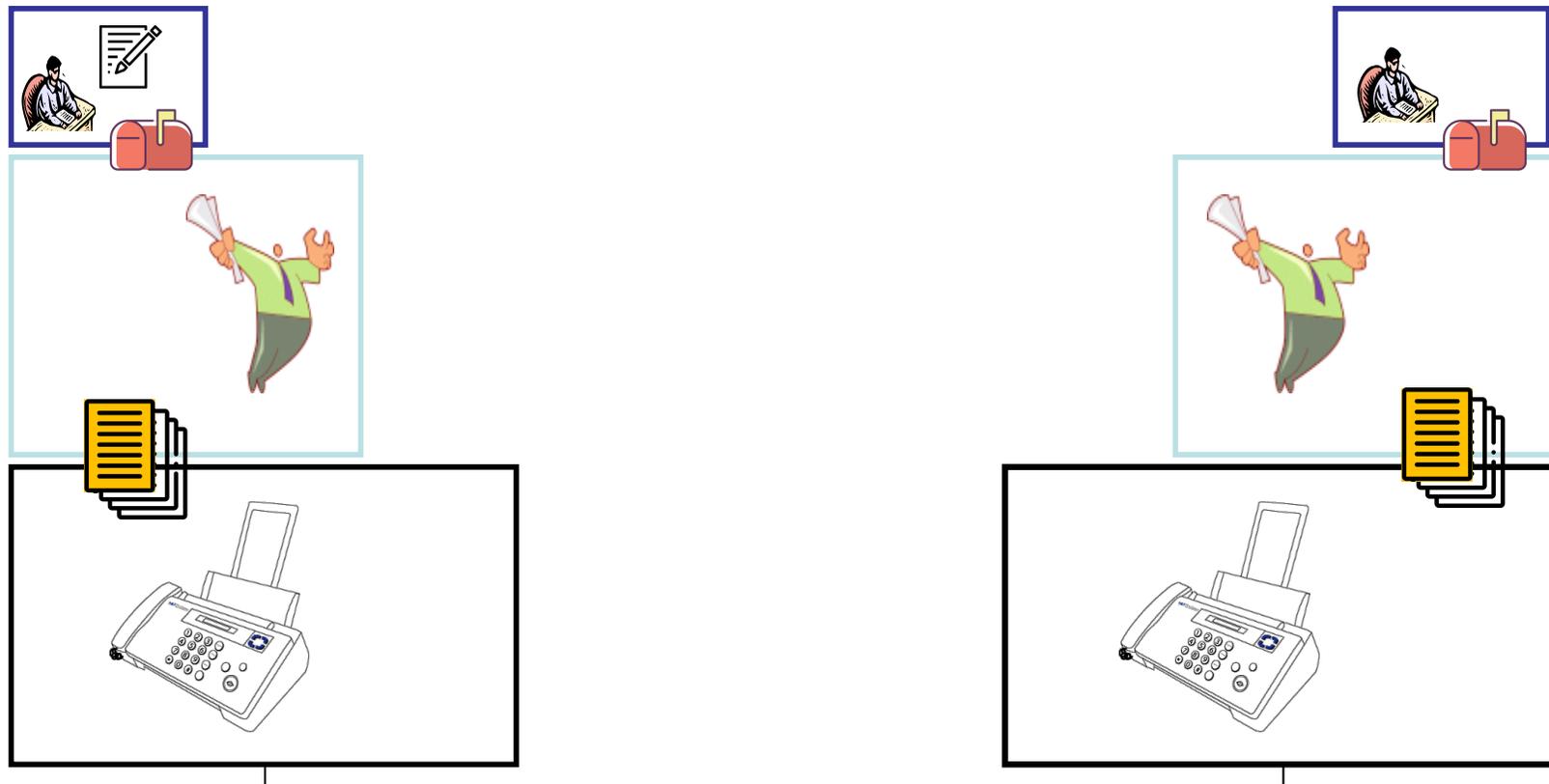
# Separación de tareas

- Los asistentes ofrecen un servicio simple realizando tareas más complicadas para ello
- Averiguan el número de teléfono de la oficina remota
- Reponen suministros de papel y tinta en el fax
- Si se corta la llamada repiten el envío
- Esperan recibir una confirmación de que el fax llegó correctamente



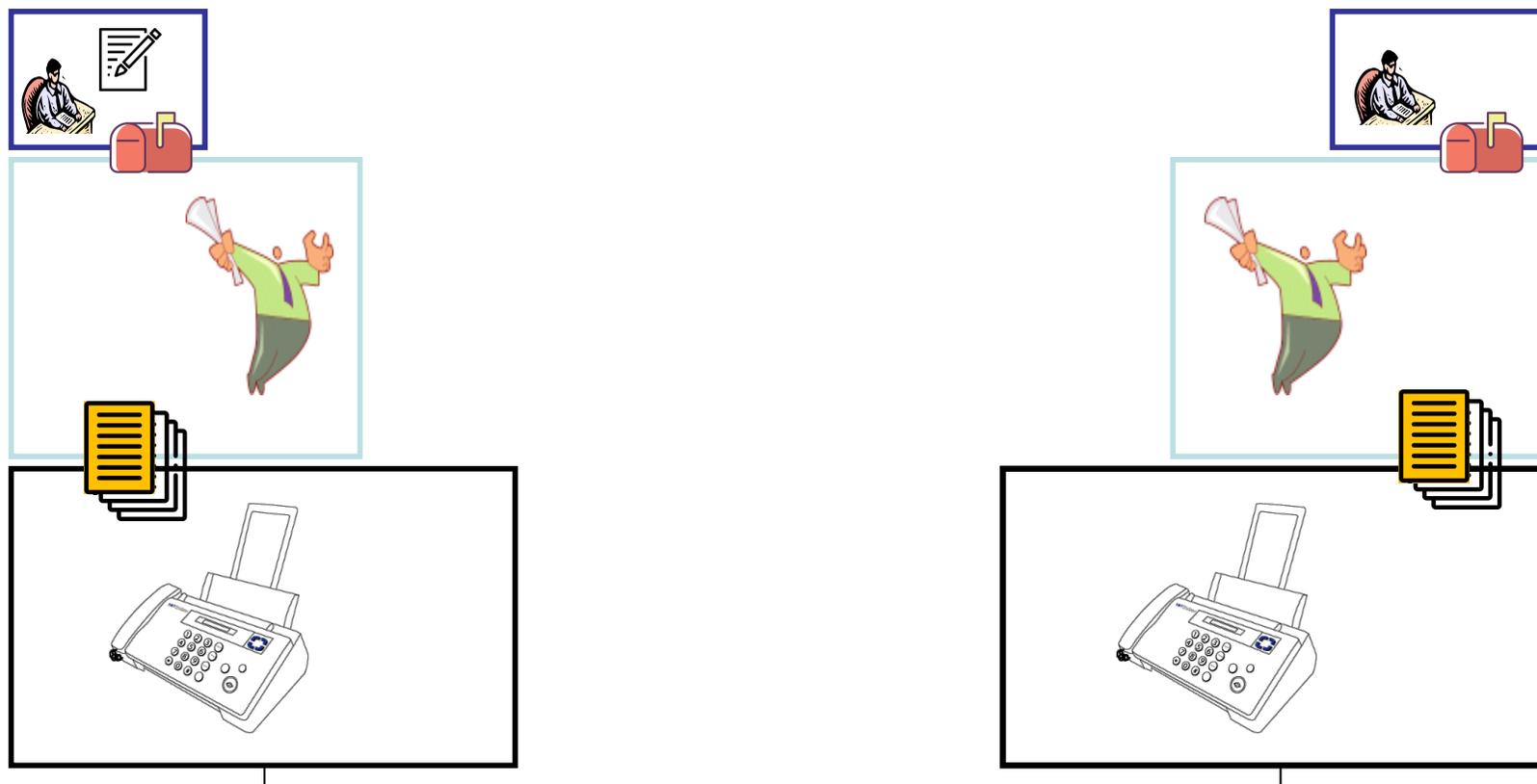
# Separación de tareas

- Las máquinas de fax hacen todo el trabajo de escanear e imprimir el documento
- Se comunican con la red telefónica para establecer la llamada



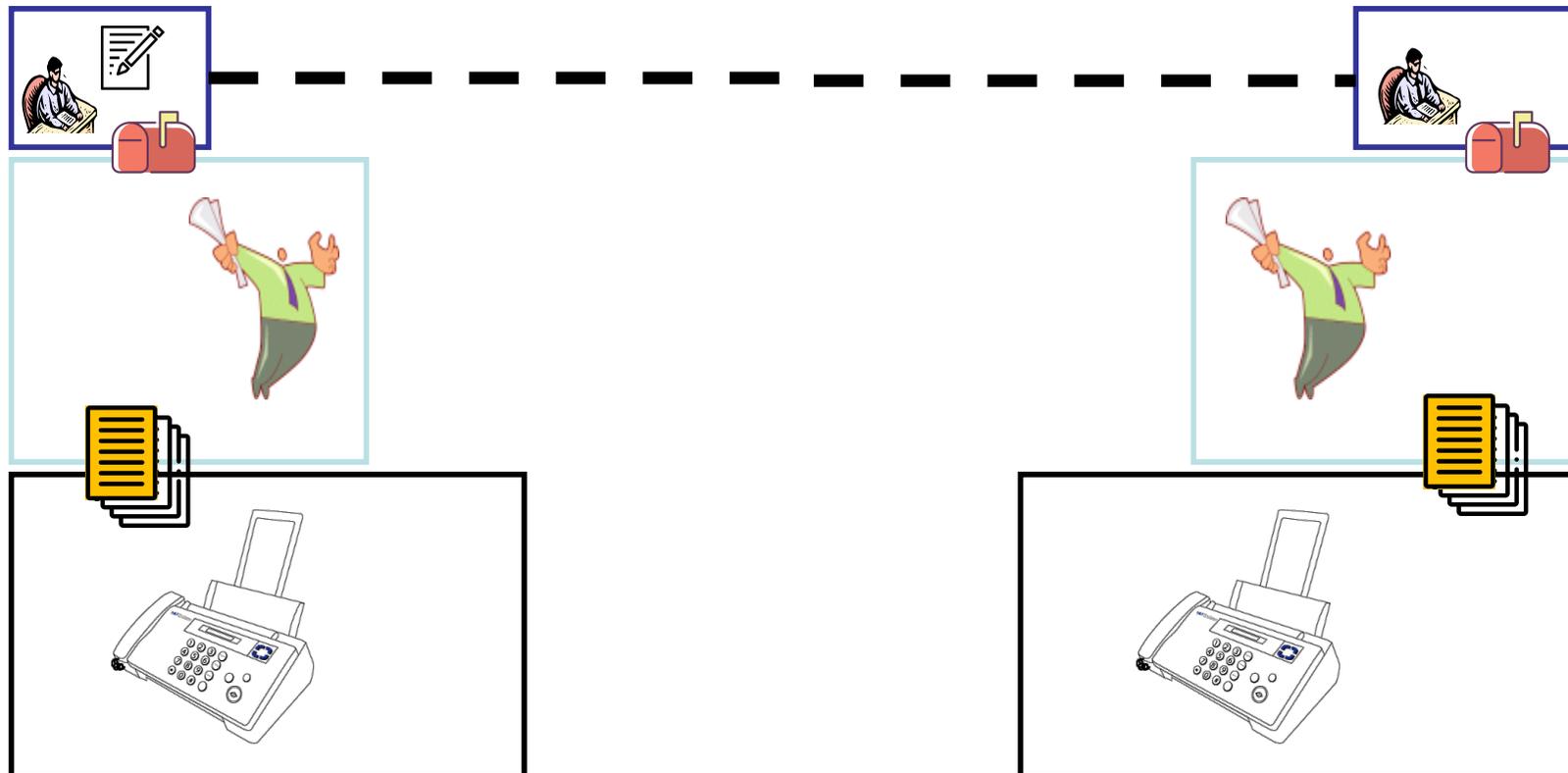
# Separación de tareas

- Hay tareas que podrían hacer uno u otro
- Por ejemplo, si se corta la llamada telefónica, en vez de encargarse el asistente de volver a marcar podría hacerlo automáticamente el fax
- Diferentes formas de repartir las responsabilidades



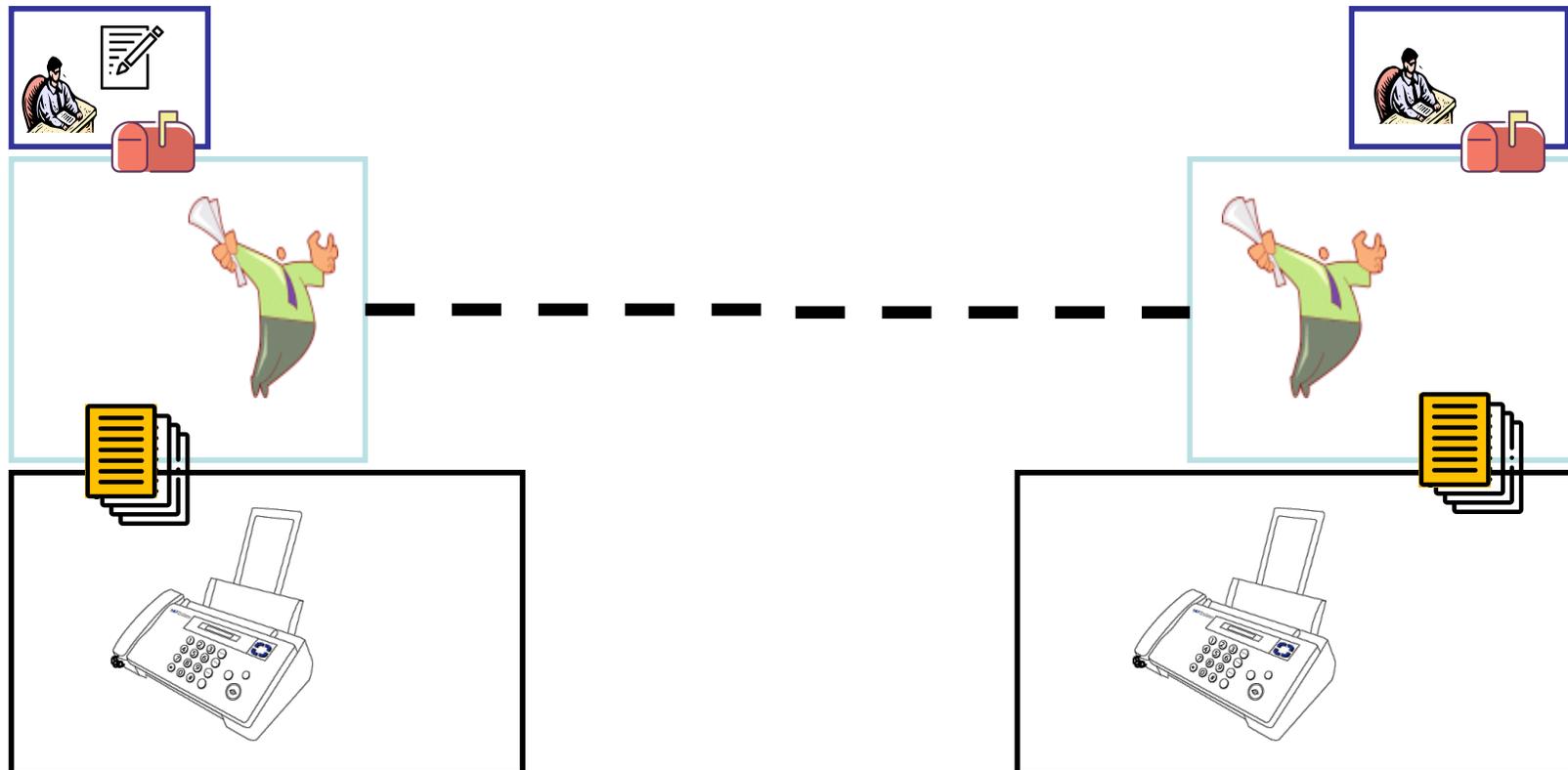
# Comunicación horizontal

- Los homólogos tienen una forma de comunicarse
  - Los usuarios se escriben memorándums en cierto idioma
  - Lo que uno envía (deja en el buzón) lo recibe el otro



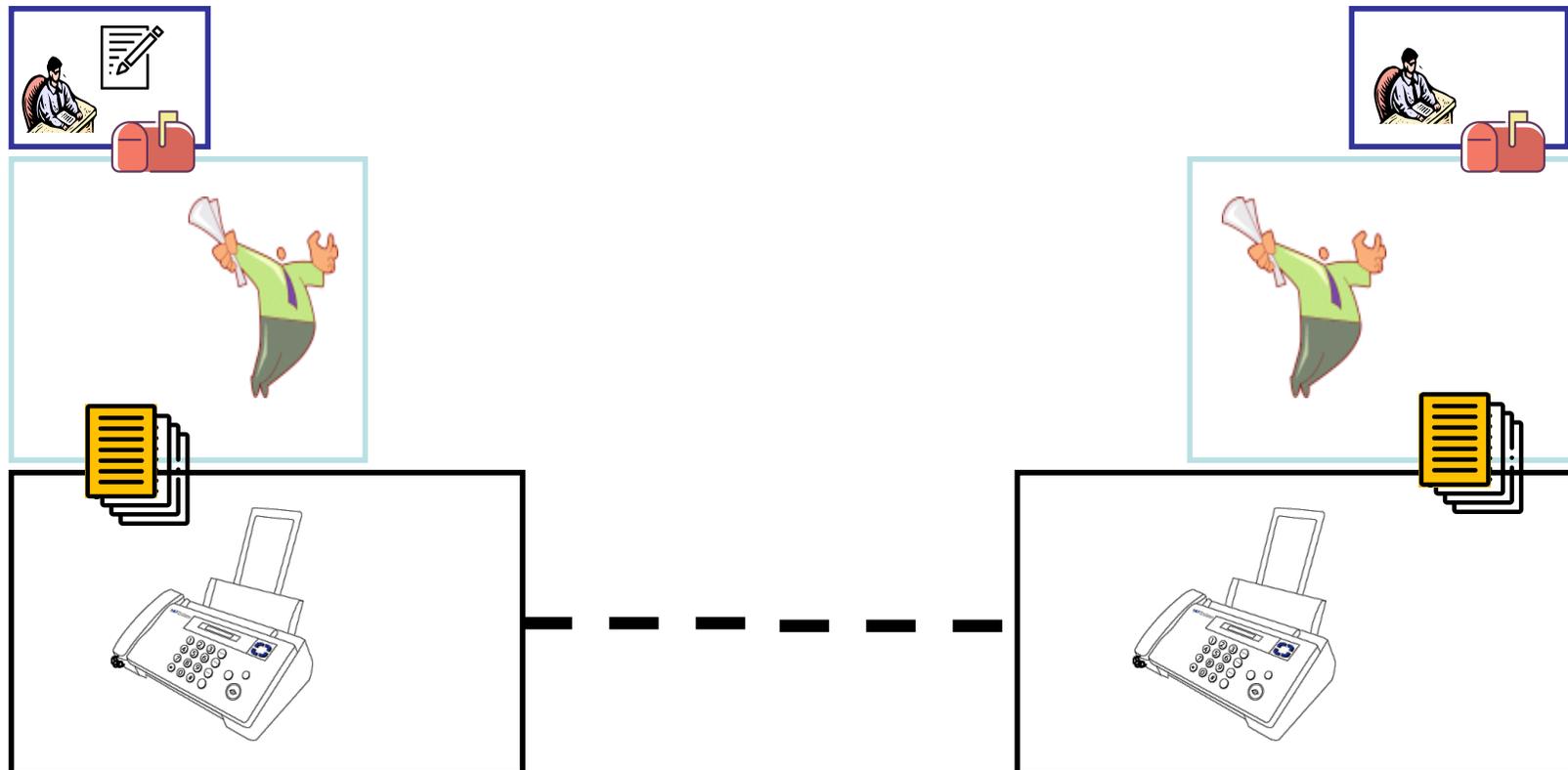
# Comunicación horizontal

- Los homólogos tienen una forma de comunicarse
  - Los asistentes se intercambian conjuntos de papeles
  - Solo entienden lo que pone en la portada
  - No saben nada de memorándums; como si quieren ser listas de la compra escritas en Klingon



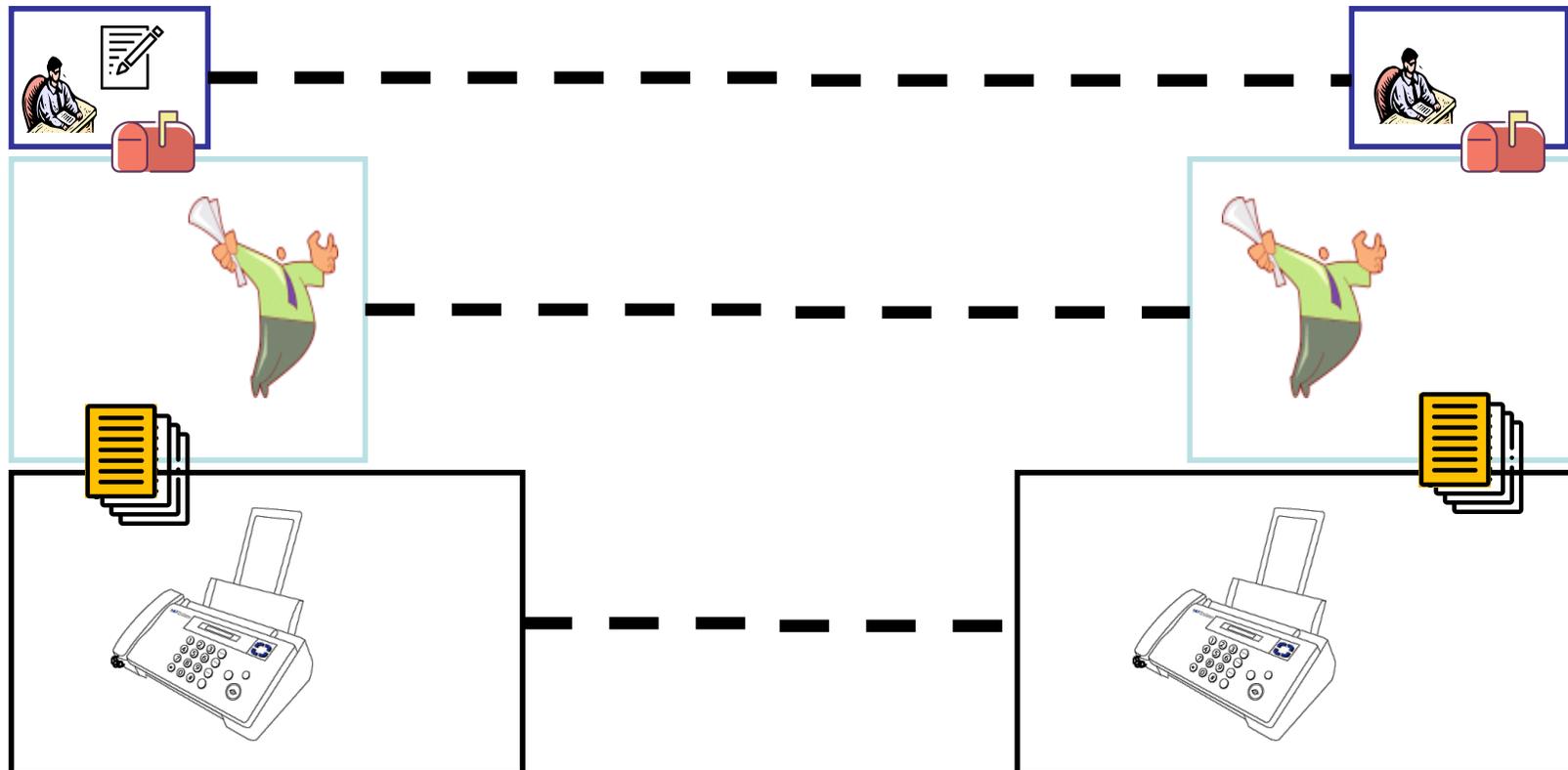
# Comunicación horizontal

- Los homólogos tienen una forma de comunicarse
  - Las máquinas de fax intercambian 0s y 1s
  - Saben que representan imágenes a plasmar en un papel
  - No entienden nada de lo que aparece en el papel



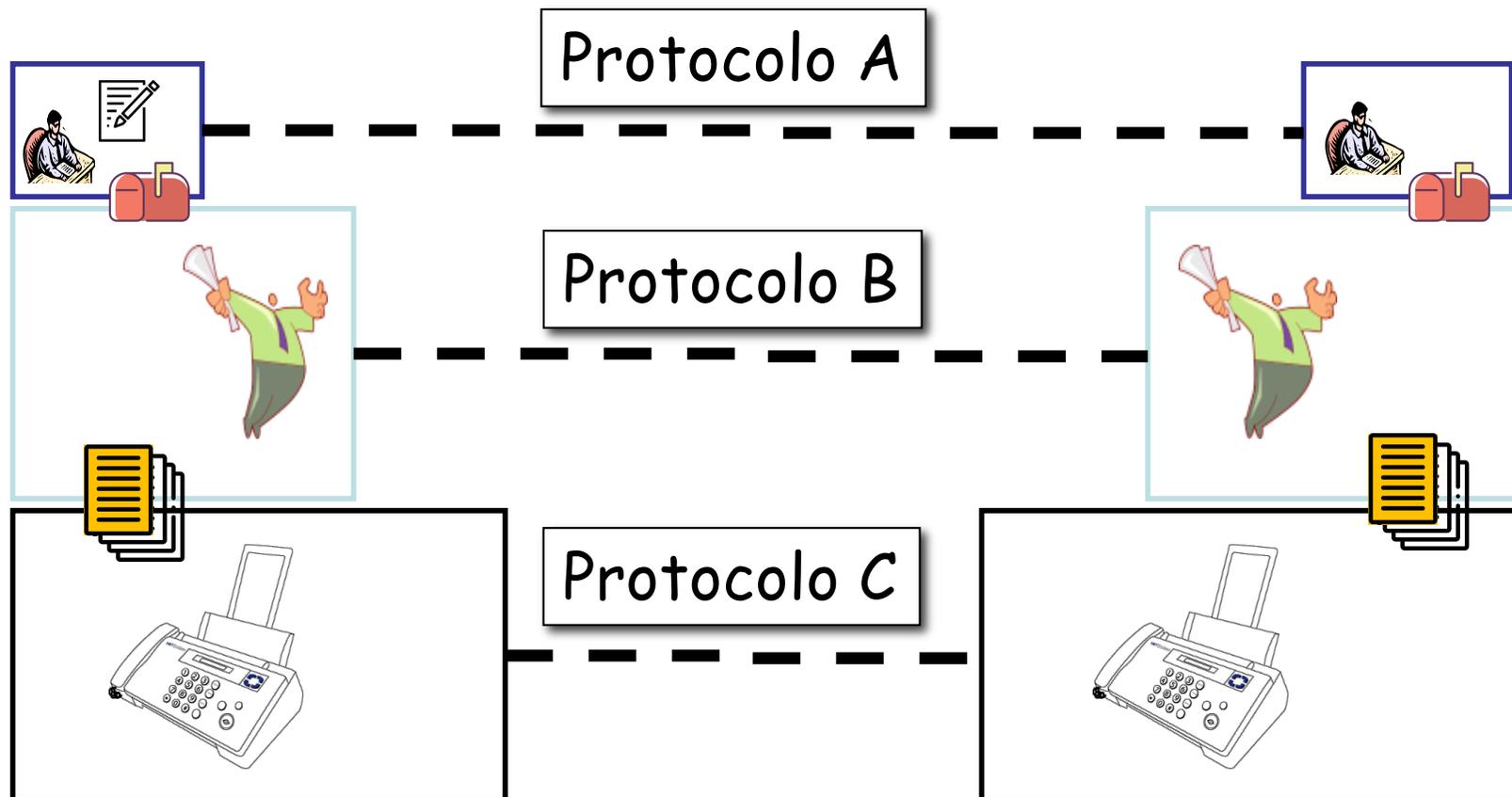
# Comunicación horizontal

- Cada uno se comunica con su homólogo
- No le importa quién está “encima” ni qué datos le da, los trata de forma “transparente”
- No le importa cómo resuelve el problema el que está “debajo”, solo que hace llegar su mensaje al otro extremo



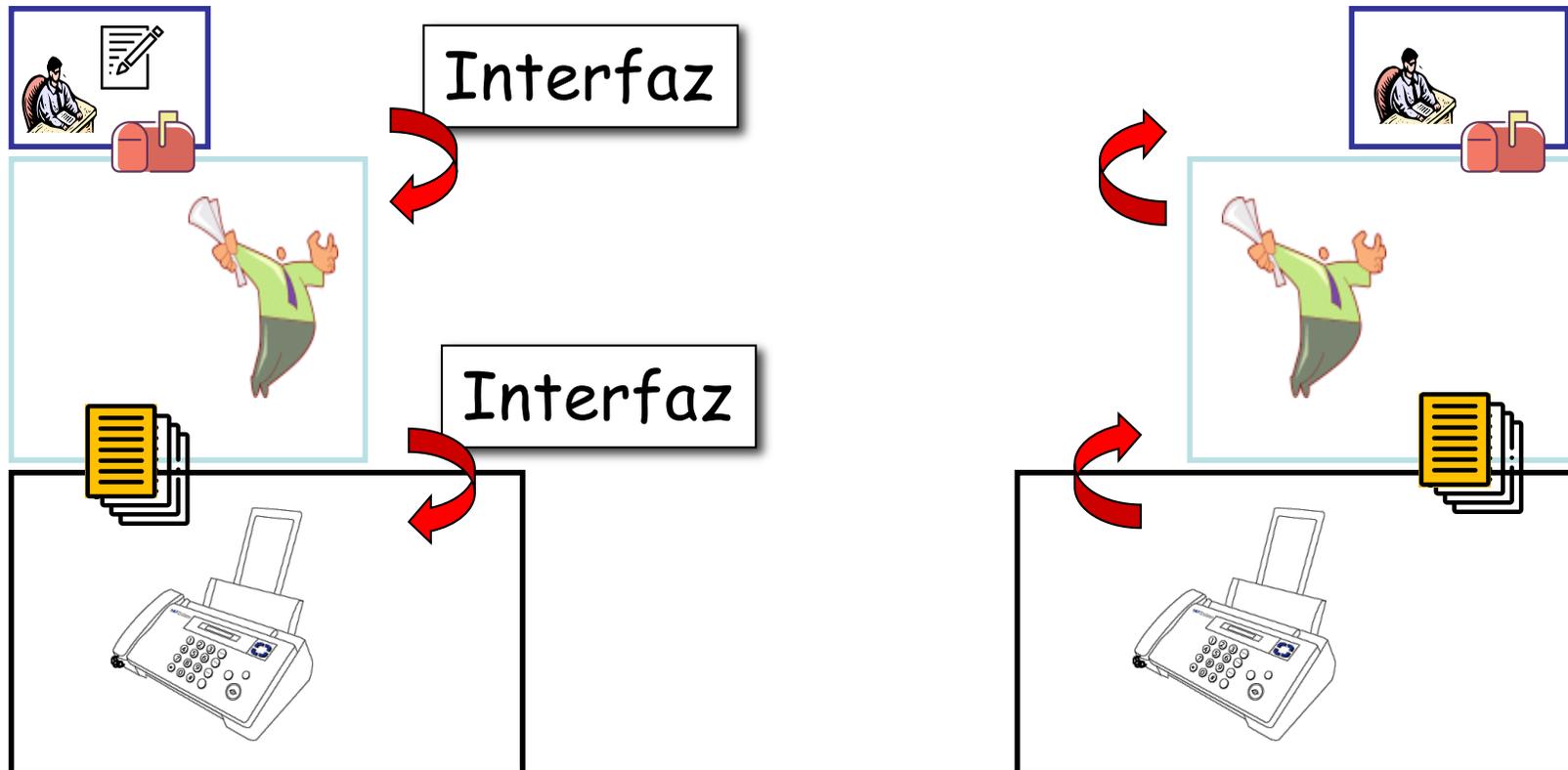
# Protocolo

- La forma de comunicarse uno con su homólogo viene determinado por un “protocolo”
- Por ejemplo el añadir una portada y qué contenido debe tener
- Solo lo entienden los homólogos en cada extremo



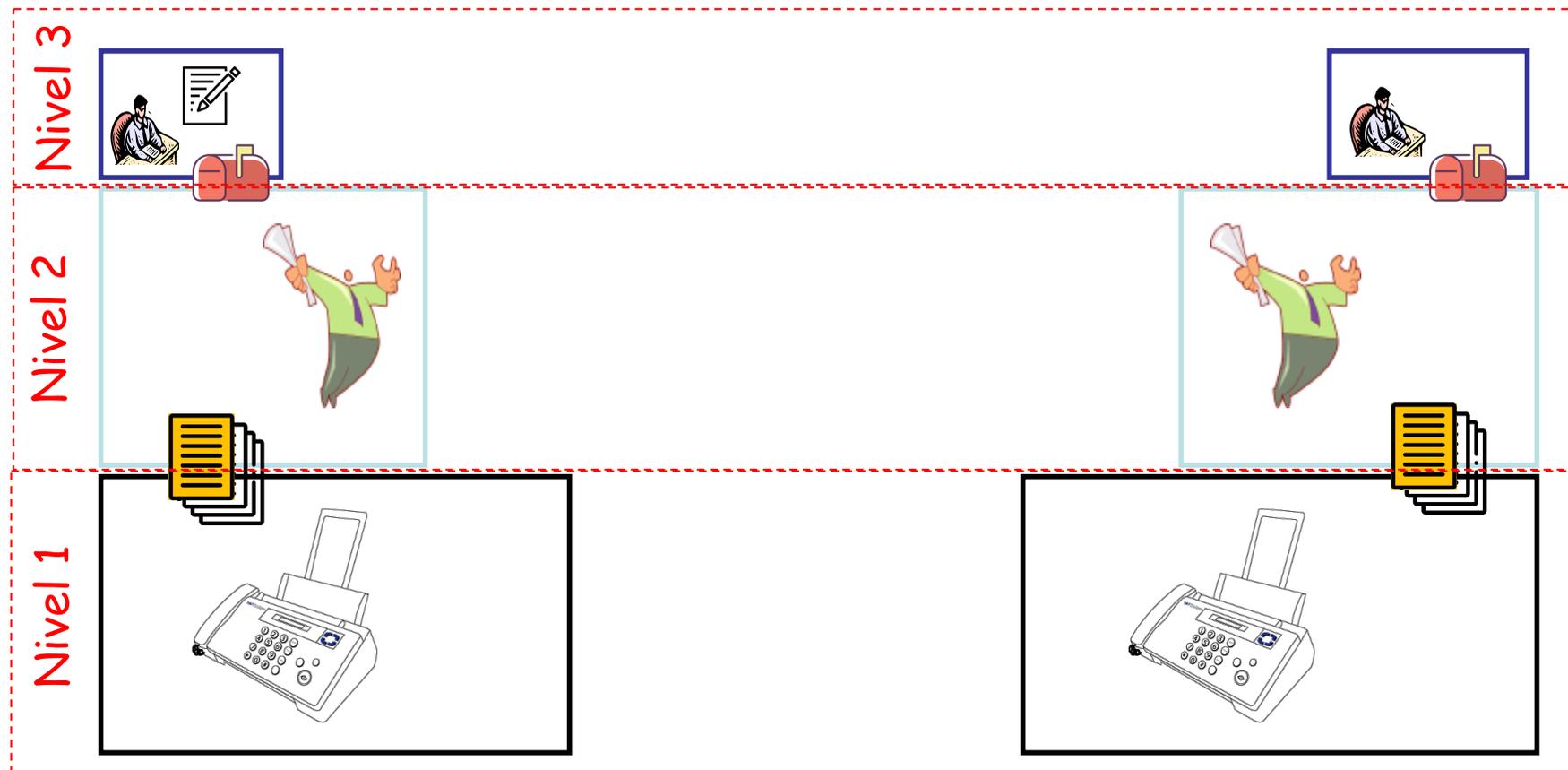
# Comunicación vertical

- Mediante un “interfaz”
- Ej: el buzón y cómo indica el usuario el destinatario (un postit?)
- El: la bandeja del fax y los controles para marcar el número destino
- El interfaz resuelve la comunicación vertical en ambos sentidos (suponiendo comunicación duplex)



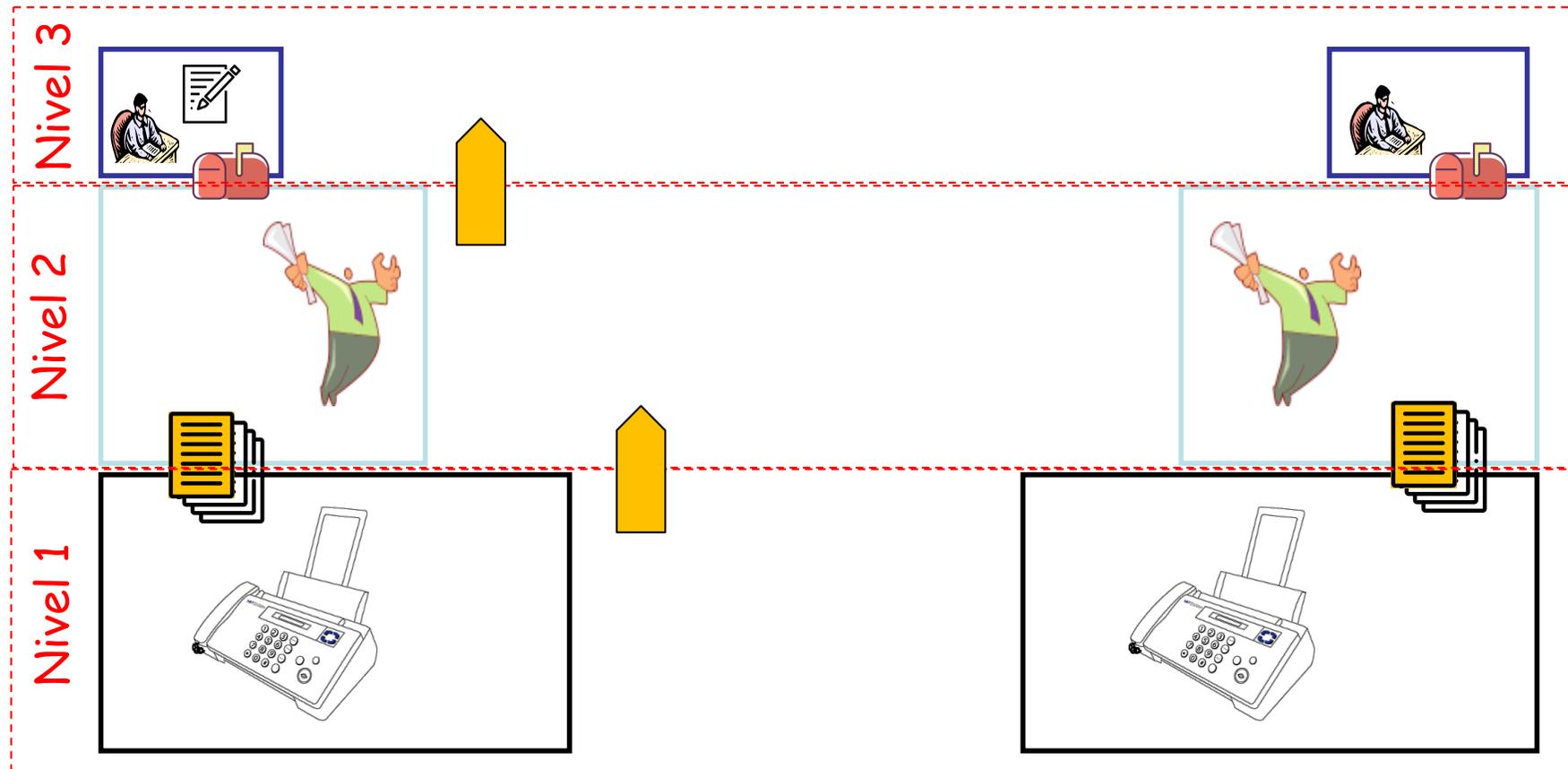
# Arquitectura

- Hablamos de “capas” o “niveles” en esta arquitectura
- Numerados de “abajo” a “arriba”
- “Abajo” suele ser cerca del suelo, lo “físico”, por eso ahí en general estará la comunicación real física entre elementos separados



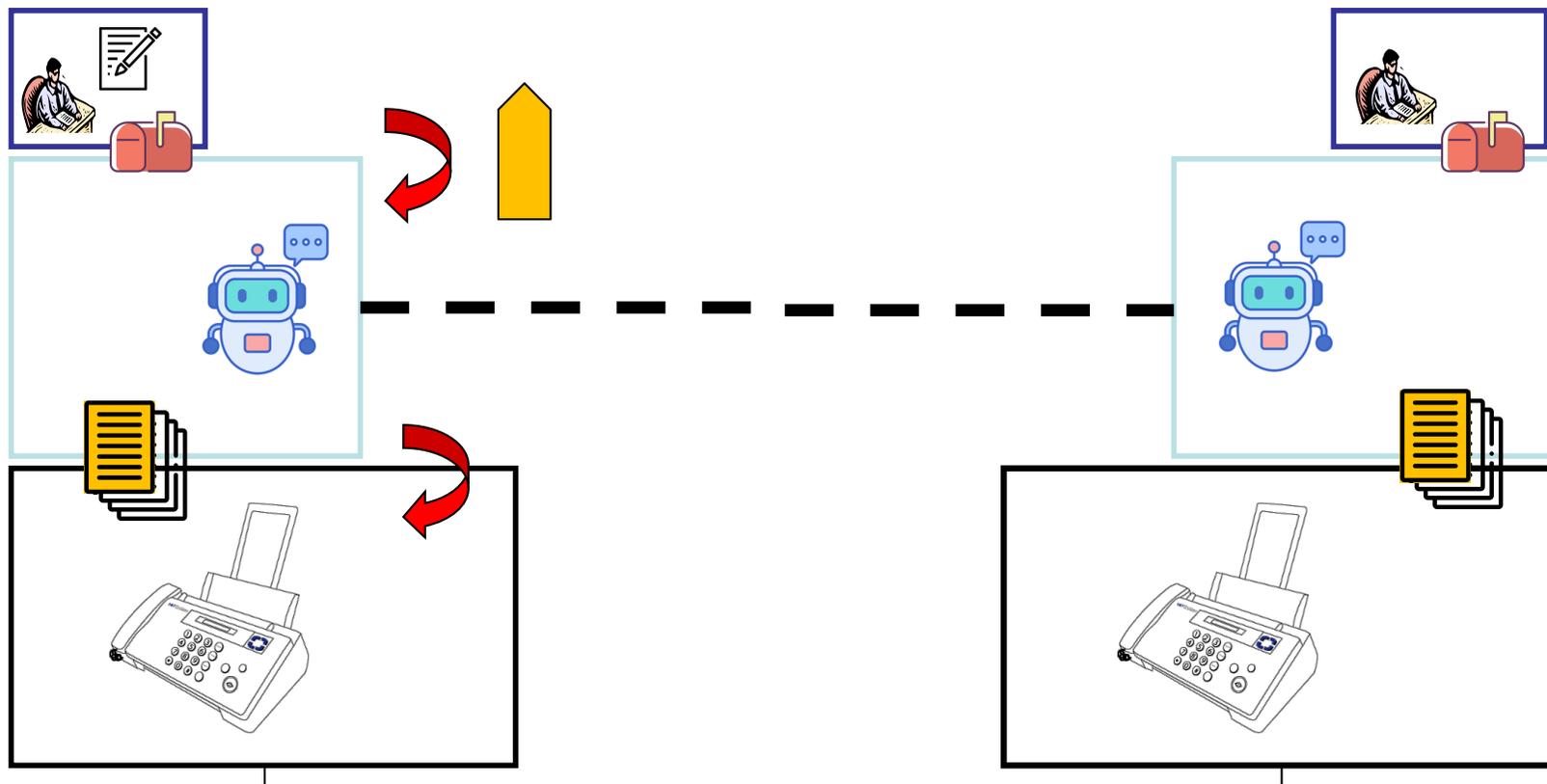
# Servicio

- Cada nivel ofrece un “servicio” al nivel superior (su usuario)
- El servicio puede tener (o no) muchas características: entrega fiable, ordenada, confidencialidad, tamaño variable, prioridades, etc
- El servicio ofrecido suele depender de las características del protocolo empleado



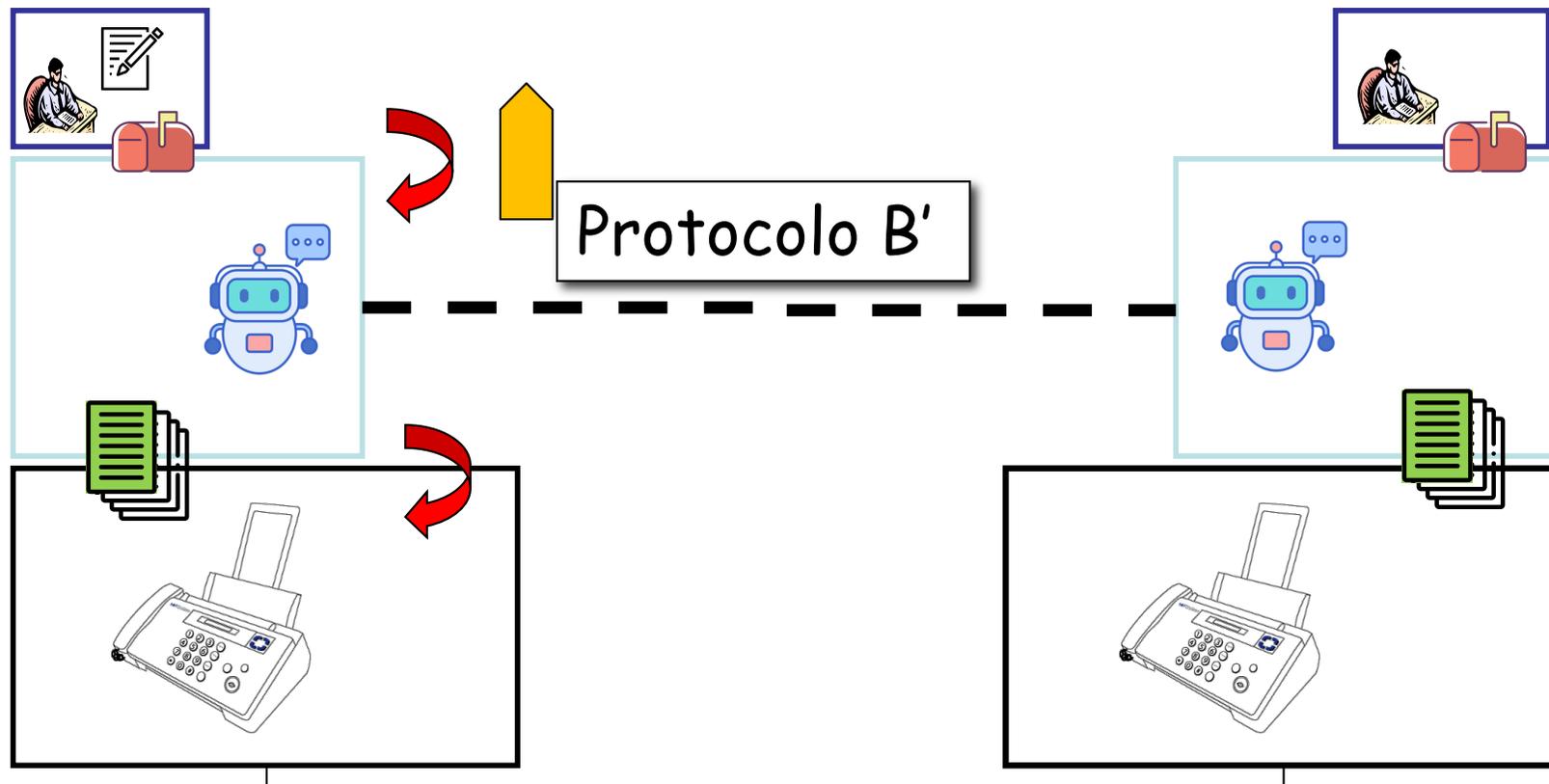
# Independencia

- Podemos sustituir nuestra implementación de uno de esos niveles
- Por ejemplo, sustituimos el nivel de los asistentes por otros
- El nuevo asistente debe ser capaz de usar la máquina de fax
- Y lo más sencillo sería que se comunicara con el usuario con el mismo interfaz, ofreciendo el mismo servicio



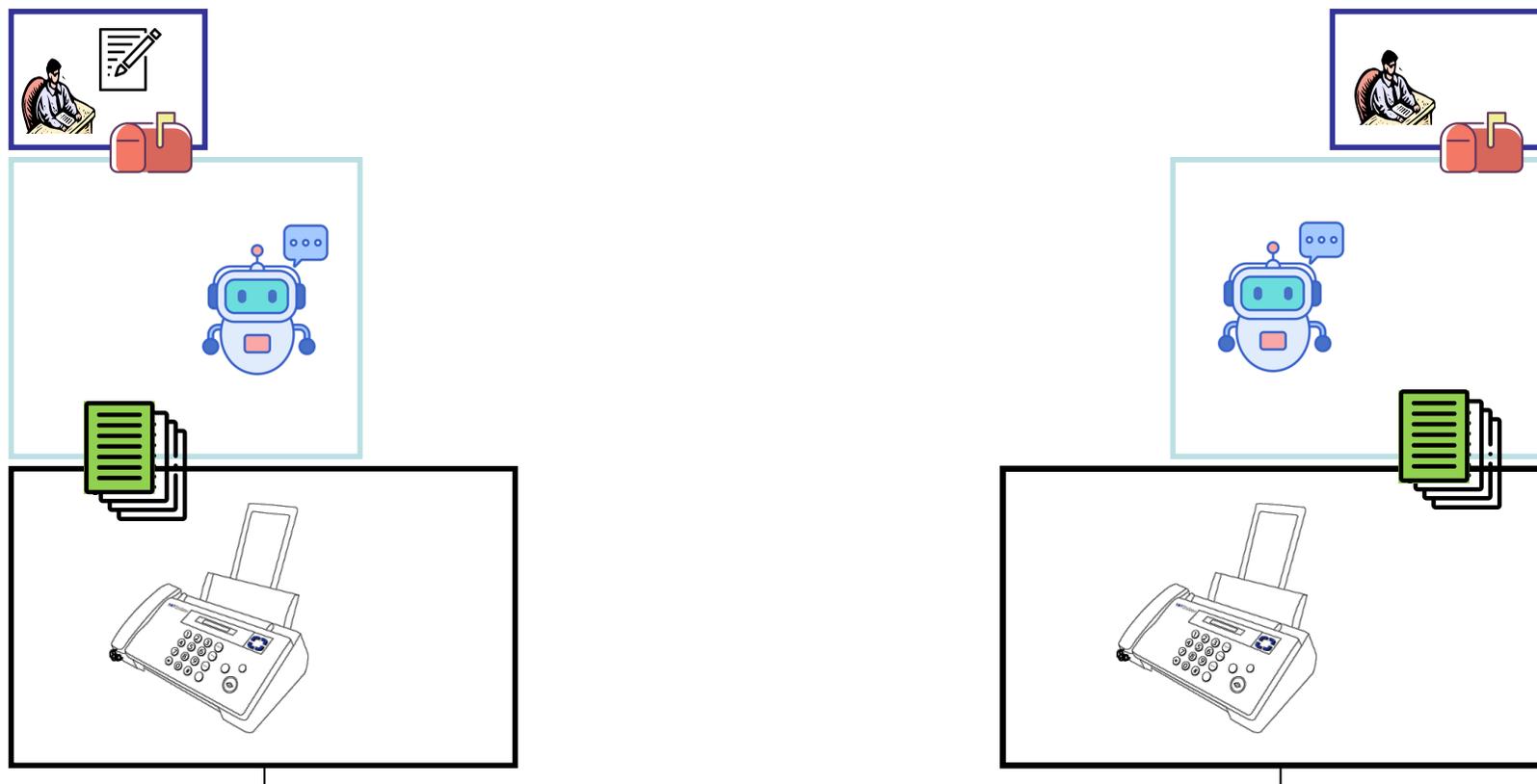
# Independencia

- La nueva implementación puede emplear un protocolo diferente
- Por ejemplo, la portada es más grande, con más o menos información, escrita en otra idioma, o además pone otra hoja al final, o intercala una página entre cada dos, o envía cada página en una llamada diferente
- Nada de todo eso es visible para los otros dos niveles
- No hay que cambiar nada en ellos



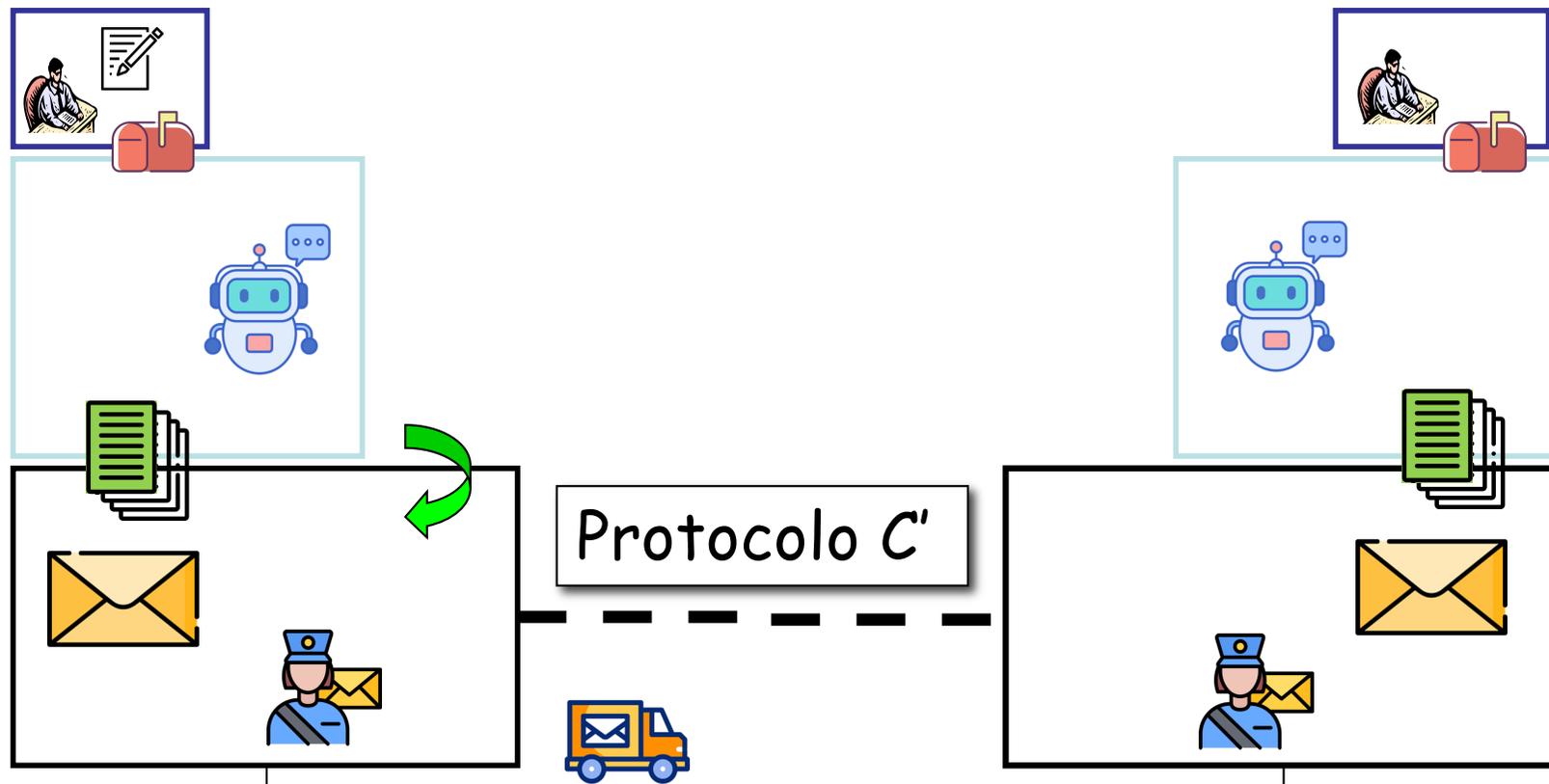
# Independencia

- También podríamos sustituir el nivel inferior
- Por ejemplo, cambiamos el fax por un servicio de mensajería



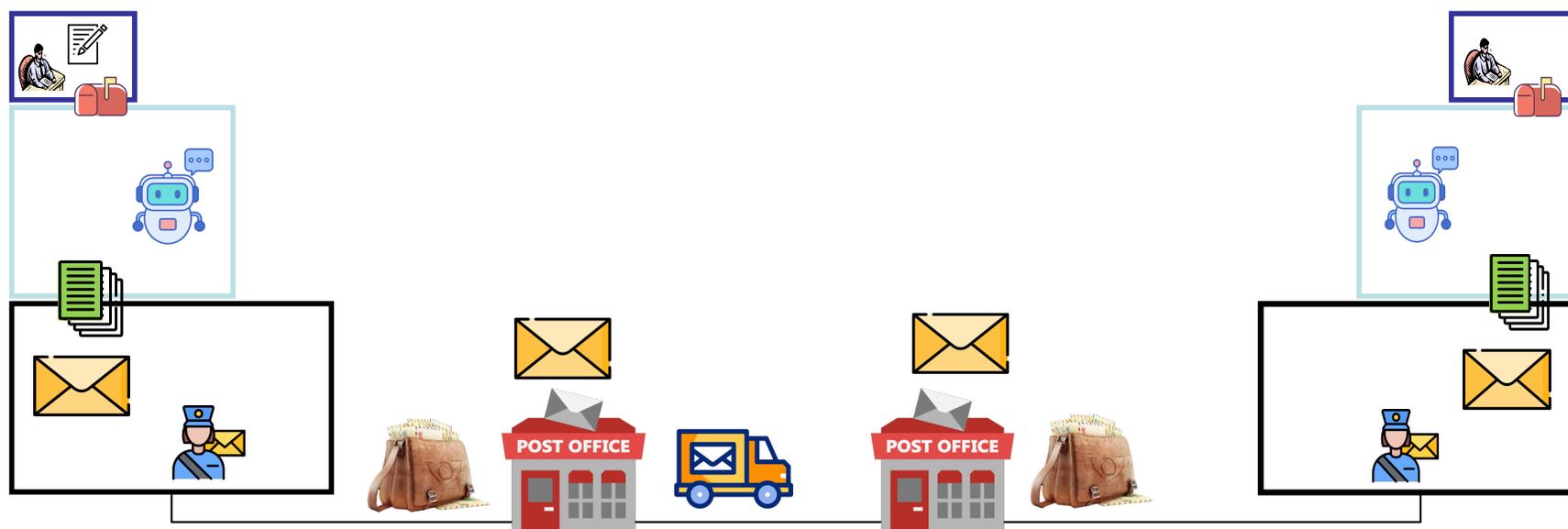
# Independencia

- En este caso seguramente cambie el interfaz porque no hay que marcar un número de teléfono sino indicar una dirección postal
- Supongamos que el propio cartero mete los papeles en el sobre (nueva encapsulación)
- El servicio será diferente: tarda más pero no depende tanto de que funcione la corriente eléctrica y no escanea el mensaje (mayor calidad)



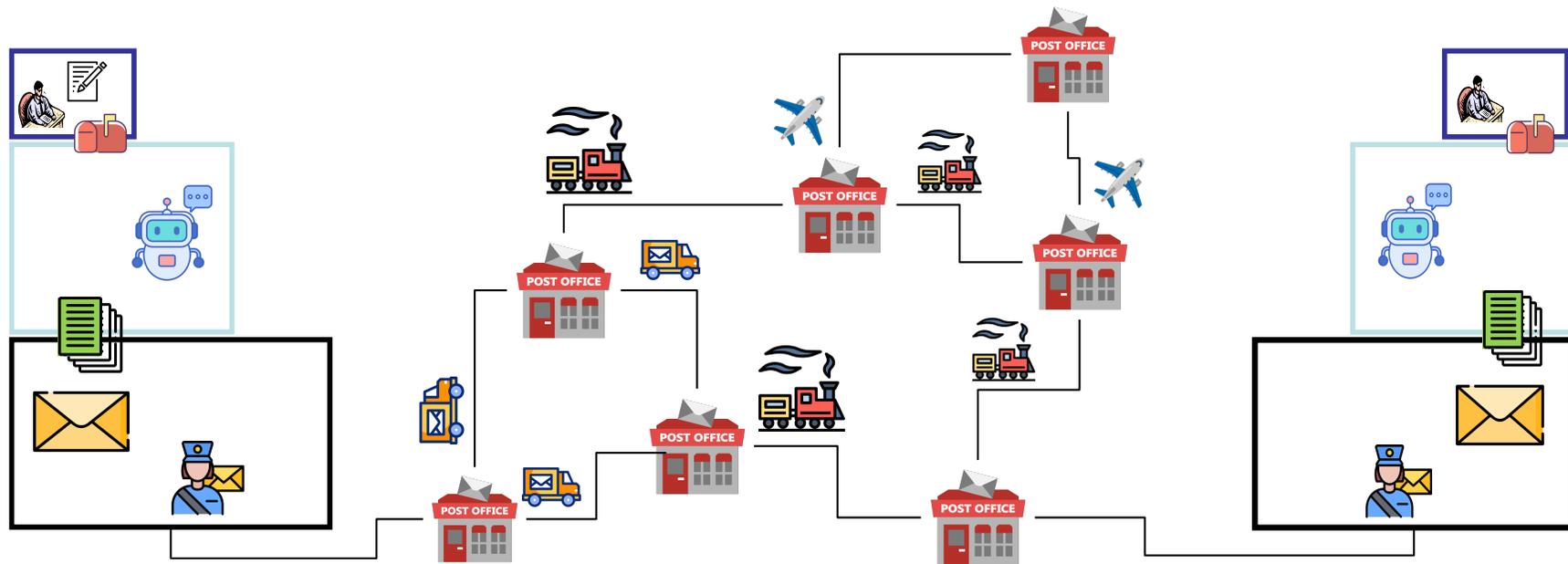
# Elementos intermedios

- Podemos forzar un poco más esta analogía para incluir el camino entre los dos extremos
- El mensajero introduce el sobre en su bolso y lo lleva a la oficina
- Sale del bolso y alguien mira el destinatario para introducir el sobre en un camión que viaja a otra oficina
- En esa oficina alguien saca el sobre para mirar el destinatario y se lo coloca en el bolso al cartero que hará el reparto
- Los trabajadores de oficinas deben entender el protocolo de correos
- No miran el contenido del sobre ni necesitan entender nada de lo que hay dentro



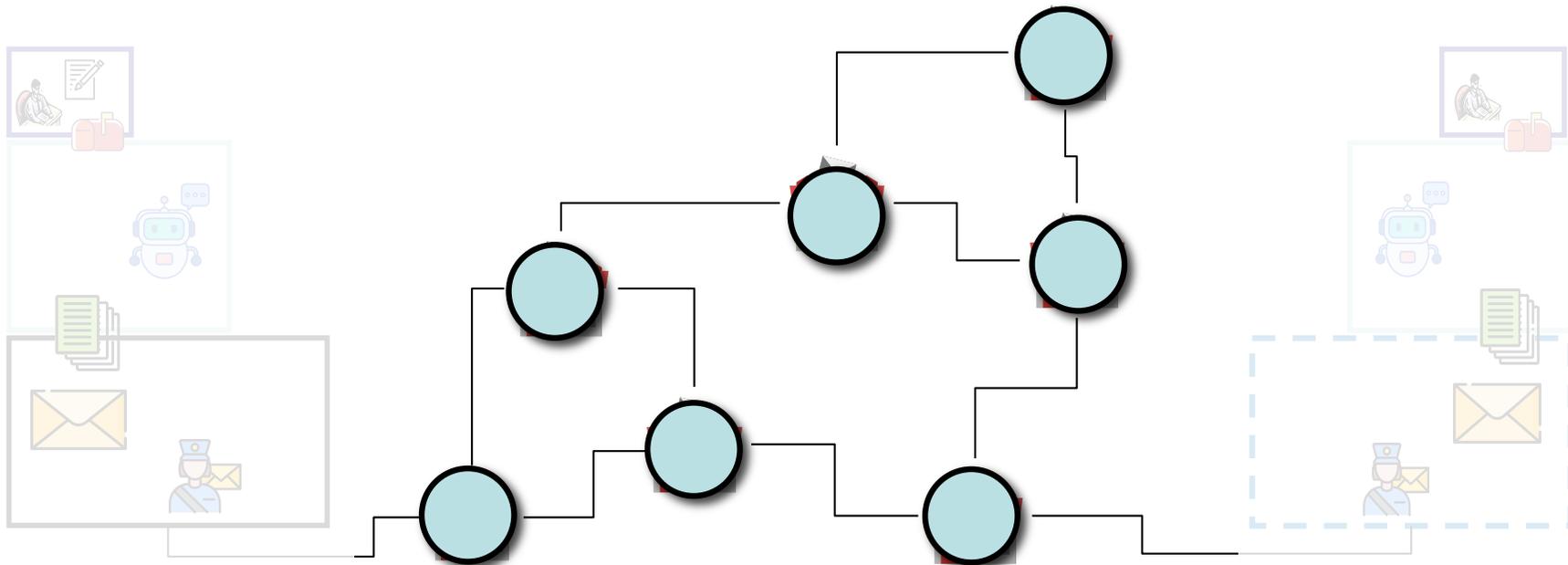
# Elementos intermedios

- Puede haber más oficinas intermedias para organizar el reparto
- Transporte en camión, avión, tren, etc (diferentes medios)
- ¿A qué se parece esto?



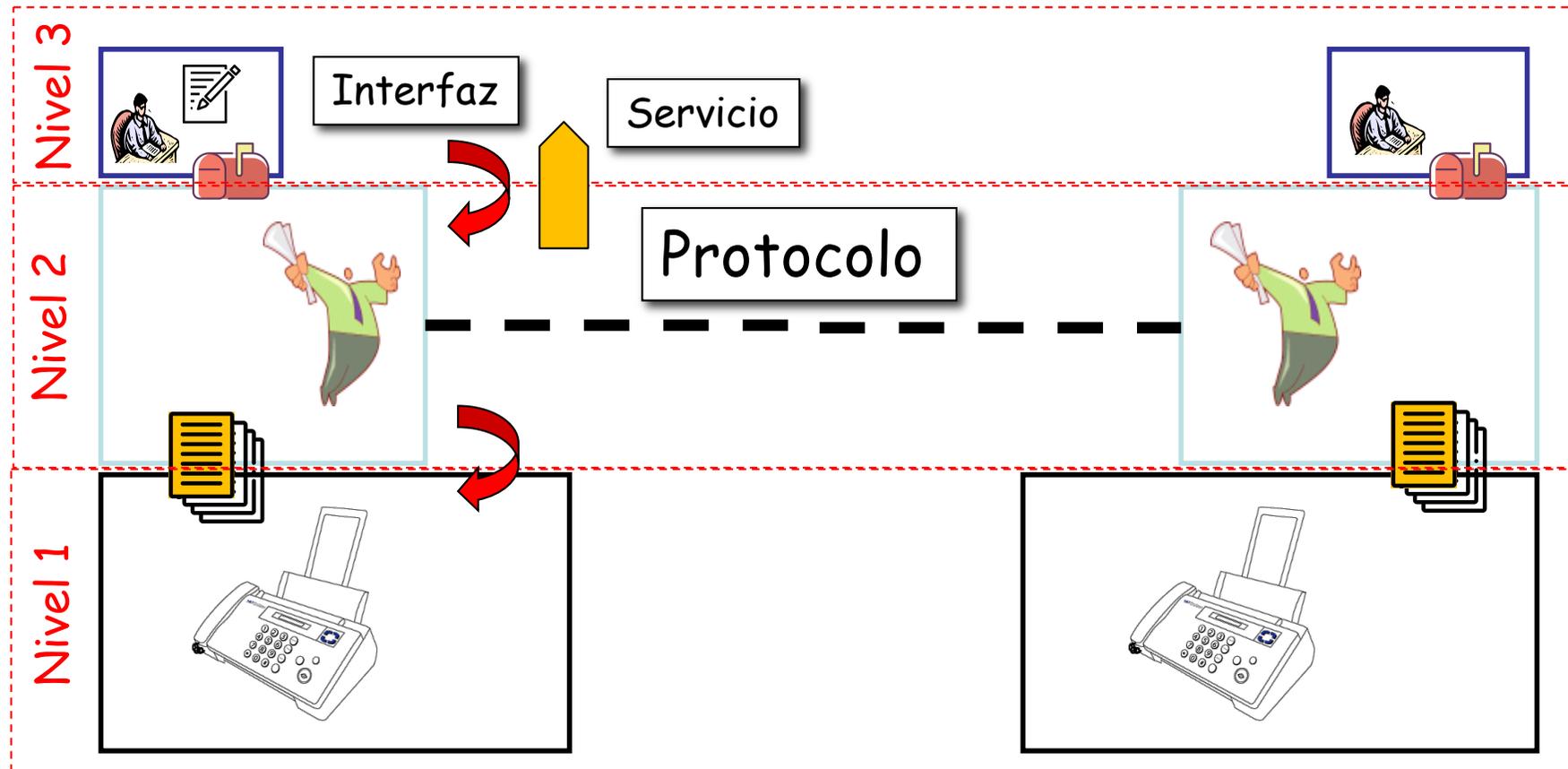
# Elementos intermedios

- Puede haber más oficinas intermedias para organizar el reparto
- Transporte en camión, avión, tren, etc (diferentes medios)
- ¿A qué se parece esto?
- Los nodos que hacen reenvío son las oficinas postales
- Los enlaces son rutas de reparto
- Puede haber varios caminos posibles y alguien debe elegir uno
- A ver si va a resultar que trabajamos en un servicio de logística... entregando 0s y 1s



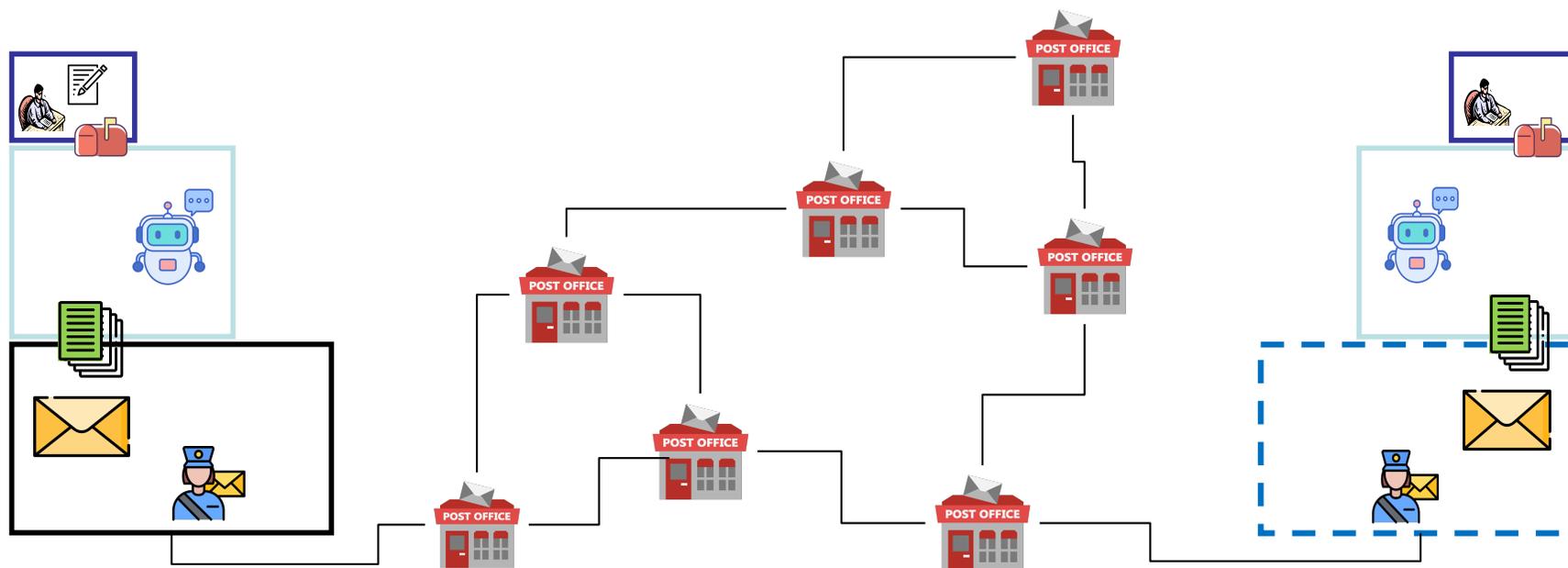
# Recapitulando

- Niveles (capas)
- Protocolos
- Interfaces
- Servicios
- Encapsulación



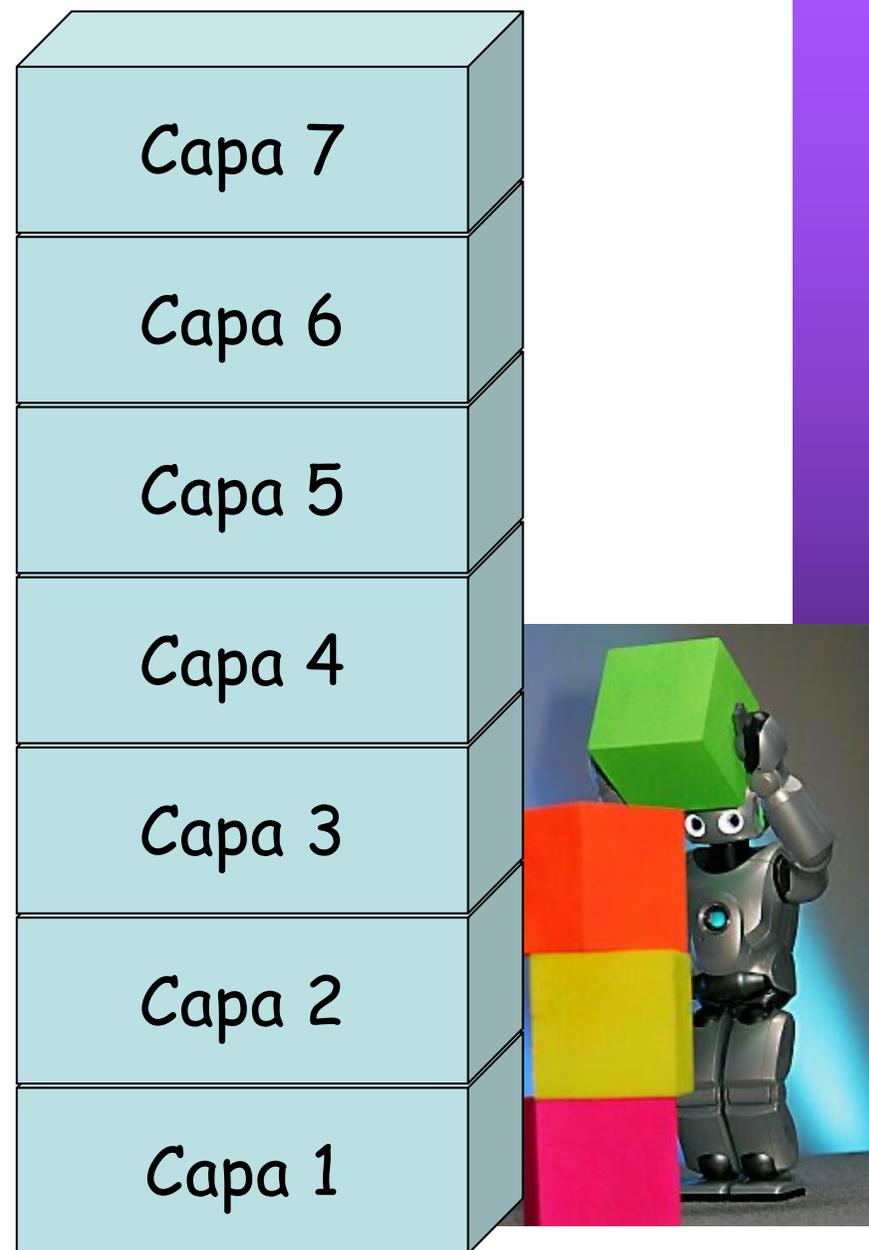
# Recapitulando

- No solo en los extremos, también en elementos de la red
- Pero los elementos de la red pueden no entender todas las capas
- Al menos necesitarán las inferiores
- Las capas que llamamos inferiores son las más cercanas a la comunicación física
- Veremos que la capa 1 es de hecho donde está el medio físico, el transmisor y el receptor
- Las capas superiores se implementan mediante electrónica o software
- Las más altas son software



# ¿Por qué capas?

- Sistemas complejos
- Una estructura ayuda en la identificación de funciones y relaciones
- La modularización facilita el mantenimiento y actualización del sistema
- Cambios en una capa pueden ser transparentes a las demás



upna

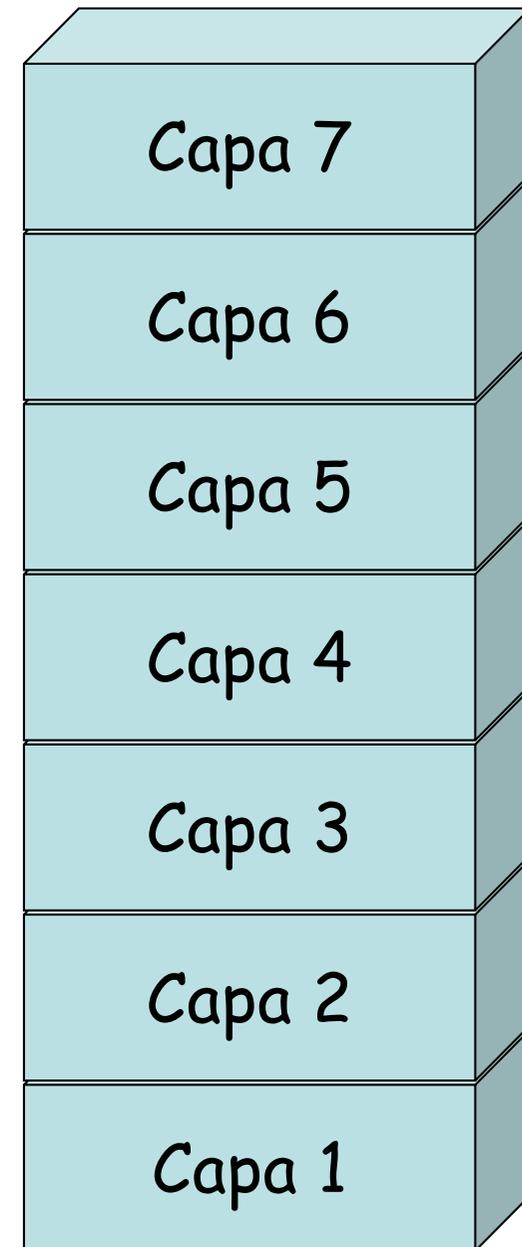
Universidad Pública de Navarra  
Nafarroako Unibertsitate Publikoa

**ARQUITECTURA DE REDES, SISTEMAS Y SERVICIOS**  
*Área de Ingeniería Telemática*

# TCP/IP y OSI

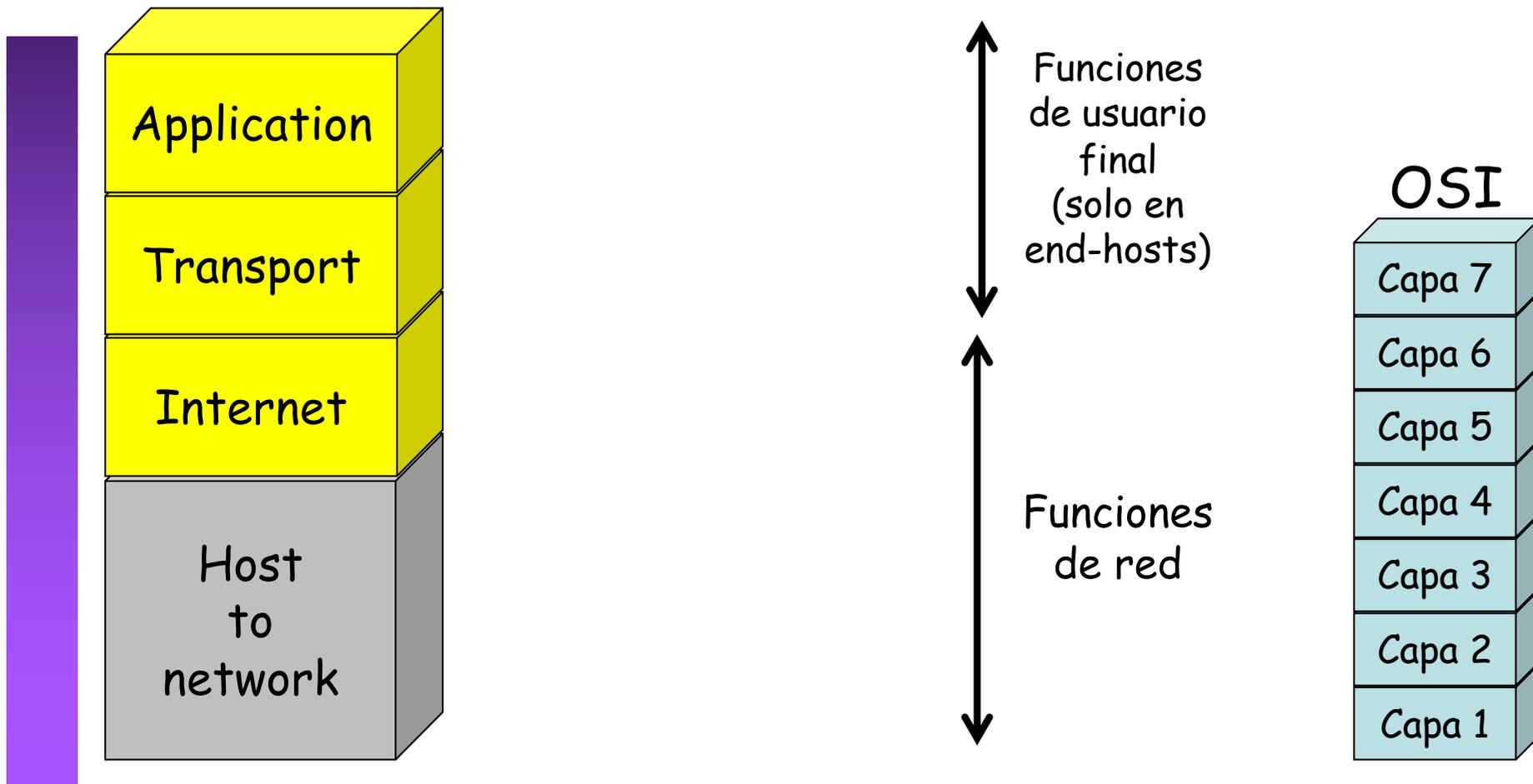
# Modelo de referencia OSI de ISO

- OSI = Open Systems Interconnection
- ISO = International Organization for Standardization
- ISO 7498 (1984)
- Modelo para arquitecturas de protocolos de computadores
- CCITT X.200 (compatible)
- Capa:
  1. Capa Física
  2. Capa de Enlace de Datos
  3. Capa de Red
  4. Capa de Transporte
  5. Capa de Sesión
  6. Capa de Presentación
  7. Capa de Aplicación



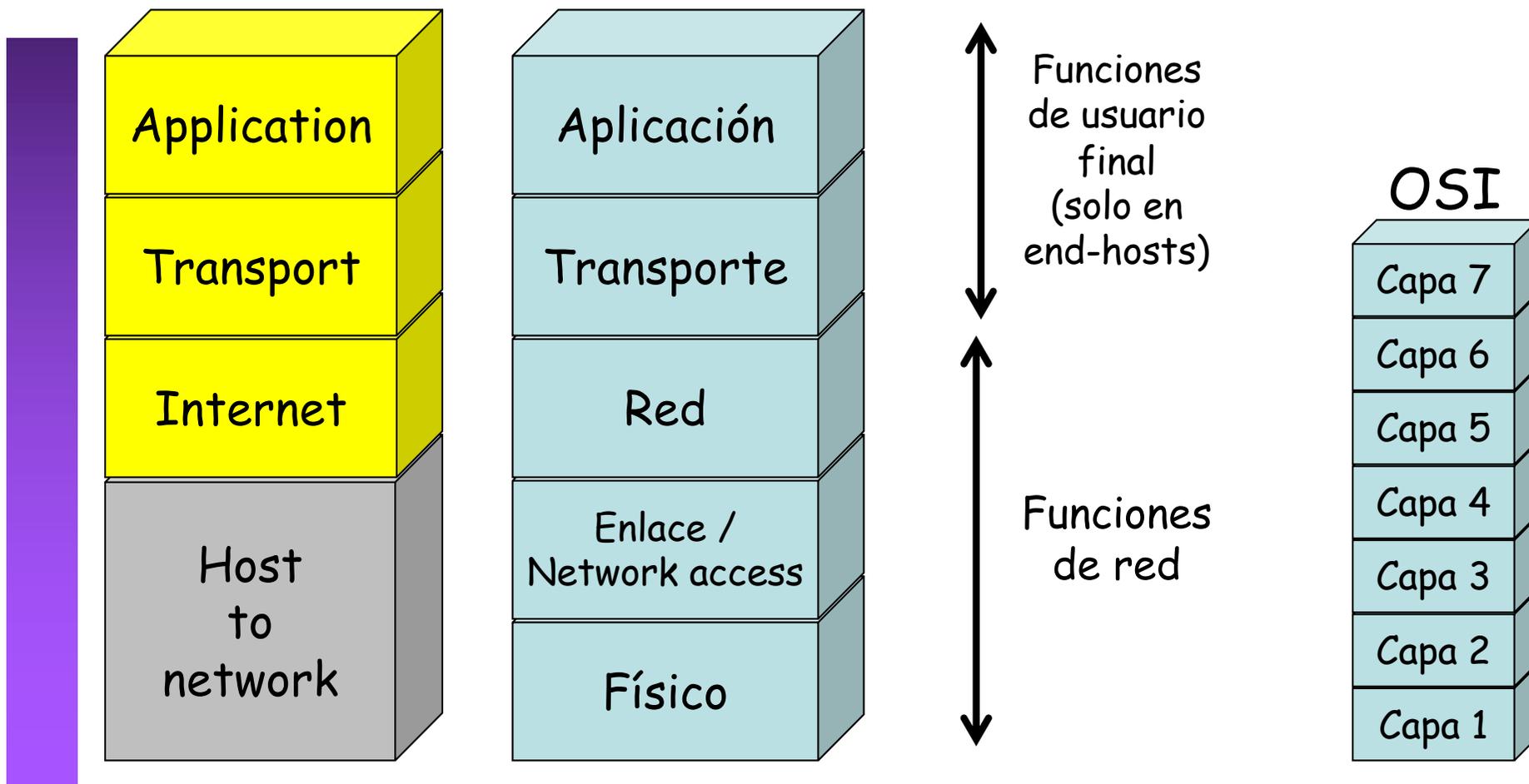
# Arquitectura de protocolos TCP/IP

- Arquitectura dominante
- Múltiples desarrollos paralelos en conmutación de paquetes
- Importante financiación de un proyecto del DARPA (*Defense Advanced Research Projects Agency*)



# Arquitectura de protocolos TCP/IP

- Al hablar de ella añadiremos los dos niveles inferiores OSI
- Capas 5 y 6 se consideran dentro de la aplicación en la arquitectura TCP/IP
- (...)



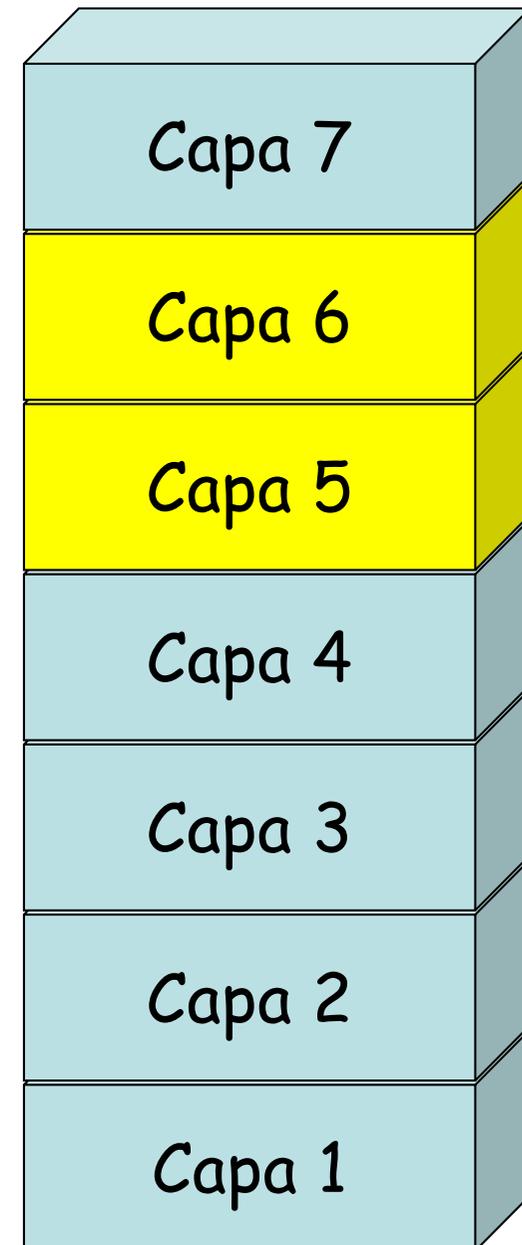
# Sesión y presentación

## Capa de Sesión

- Control y disciplina de diálogo entre aplicaciones (ej. half/full duplex)
- Agrupamiento (grupos de datos para la aplicación)
- Recuperación (puntos de comprobación)
- Gestiona *sesiones* entre aplicaciones

## Capa de Presentación

- Ofrece a la aplicación independencia en la representación de datos (sintaxis)
- Codificación y formato de los datos
- Compresión de los datos
- Cifrado de datos



# Arquitectura de protocolos TCP/IP

- Empezaremos con escenarios sencillos
- Lo mínimo que necesitaremos normalmente será las capas 1 (físico) y 2 (enlace)
- Por ejemplo, vamos a ver el caso de Ethernet

