



**REDES DE ORDENADORES**  
*Área de Ingeniería Telemática*

# Otros servicios

## Tema 5.- Nivel de aplicación en Internet

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

*Redes de Ordenadores*

*Ingeniero Técnico de Telecomunicación Especialidad en Sonido e Imagen, 3º curso*



# Temario

- 1.- Introducción
- 2.- Nivel de enlace en LANs
- 3.- Interconexión de redes IP
- 4.- Nivel de transporte en Internet
- 5.- Nivel de aplicación en Internet



# Temario

- 1.- Introducción
- 2.- Nivel de enlace en LANs
- 3.- Interconexión de redes IP
- 4.- Nivel de transporte en Internet
- 5.- Nivel de aplicación en Internet**
  - Nivel de aplicación: WWW
  - FTP. Telnet. Ejemplos
  - E-mail
  - DNS. P2P. Mensajería...



# Contenido

- DNS
- P2P
- Mensajería
- Proxy



# Contenido

- **DNS**
- P2P
- Mensajería
- Proxy



# El problema de los nombres

- Las direcciones IP, que identifican a los interfaces de los hosts, son números de 32 bits
- Sencillas de manejar para las máquinas, complicado para los humanos
- Más sencillo memorizar nombres textuales
- Hace falta “traducir” el nombre textual en la dirección numérica para que se pueda realizar la comunicación. Esto se llama “resolver el nombre”
- La traducción se realiza mediante el Sistema de Nombres de Dominio o DNS (Domain Name System)



# Domain Name System

- Es una **base de datos distribuida**
- Servidores de nombres organizados **jerárquicamente**
- Es un **protocolo de aplicación**
- Permite a los hosts traducir entre nombres y direcciones
  - Funcionalidad vital
  - Implementada como protocolo a nivel de aplicación
  - Complejidad en los extremos de la red

## ¿Por qué no centralizado?

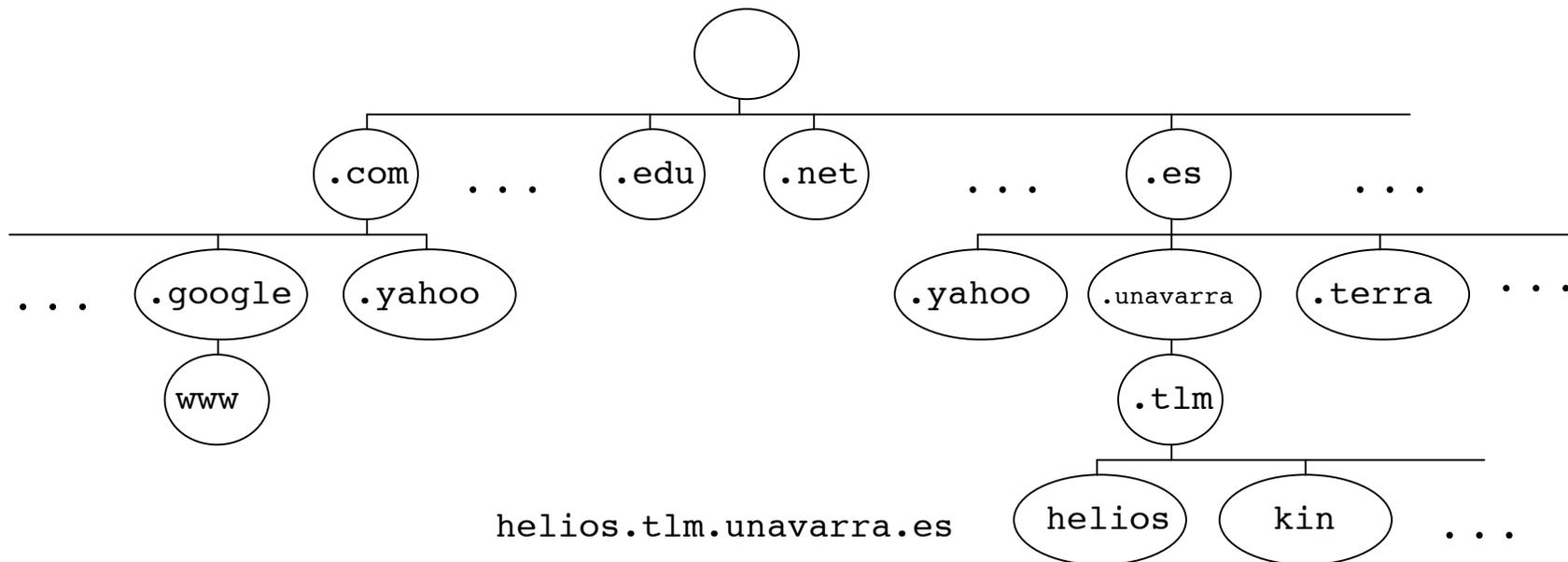
- Punto de fallo
- Volumen de tráfico
- Base de datos centralizada lejana
- Mantenimiento

**¡ No escala !**



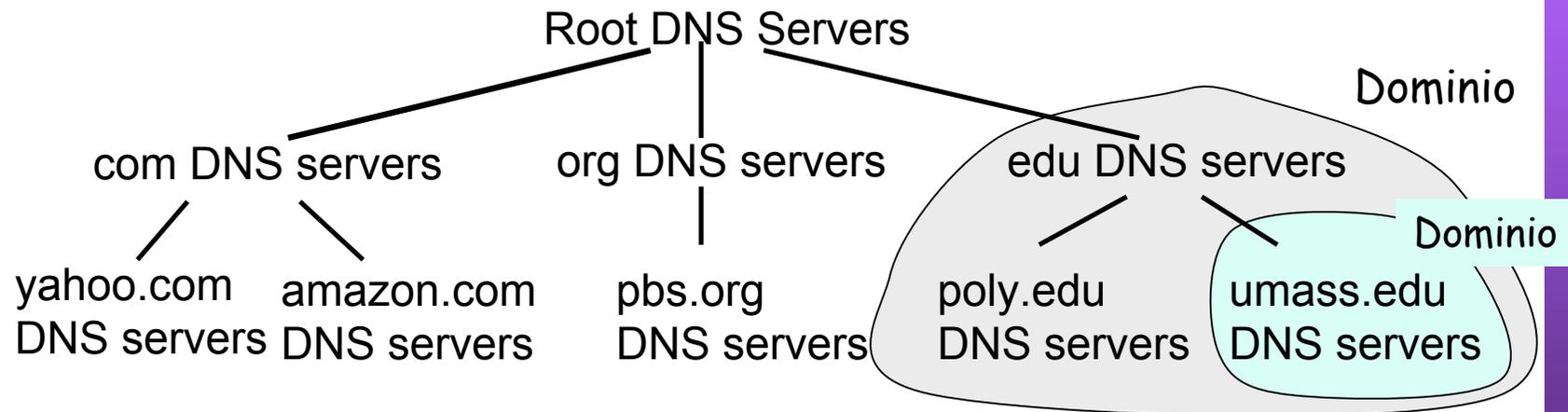
# Jerarquía de nombres

- Los nombres están formados por segmentos alfanuméricos separados por puntos (no distingue mayúsculas)  
helios.tlm.unavarra.es  
www.google.com
- Estructura jerárquica (. . .)





# B.D. jerárquica distribuida



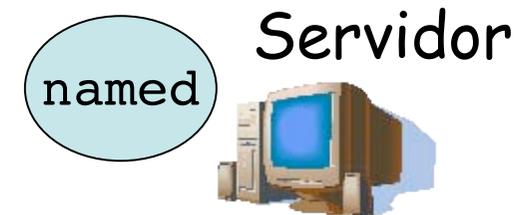
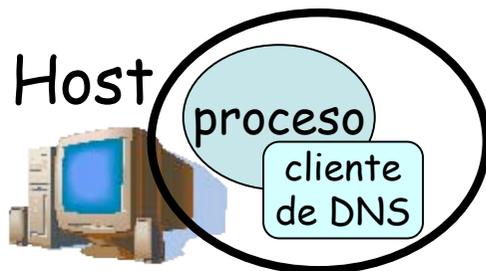
## El cliente busca la IP de **www.amazon.com**:

- El cliente pregunta a un **servidor Root** para encontrar el servidor de DNS del dominio **com**
- El cliente pregunta al **servidor del dominio com** para obtener el servidor del dominio **amazon.com**
- El cliente pregunta al servidor DNS del dominio **amazon.com** para obtener la IP de **www.amazon.com**.



# Implementación

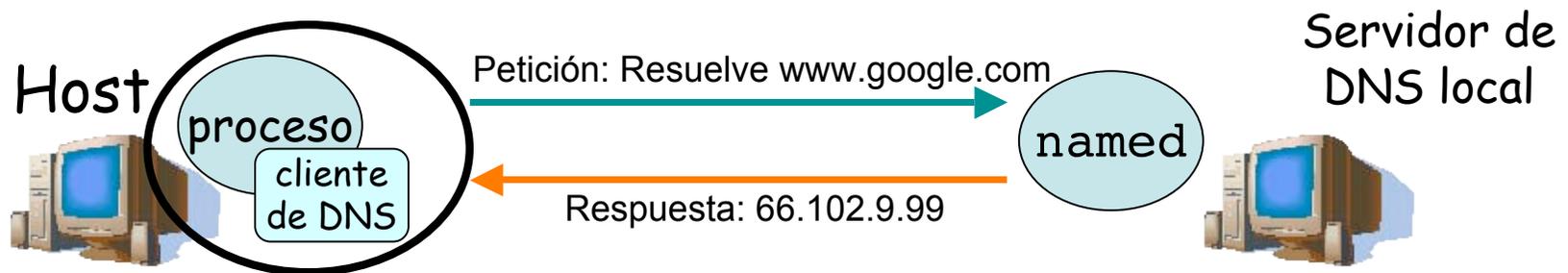
- El servidor es un programa específico pero el cliente es generalmente solo unas funciones en una librería (*resolver*) (...)
- La aplicación cliente de DNS es la propia aplicación del usuario (...)
- El software típico que lo implementa es BIND (Berkeley Internet Name Domain) (el programa servidor se llama *named*) (...)
- Emplea UDP (puerto servidor 53) o TCP si el mensaje de respuesta es de más de 512 Bytes.





# Funcionamiento

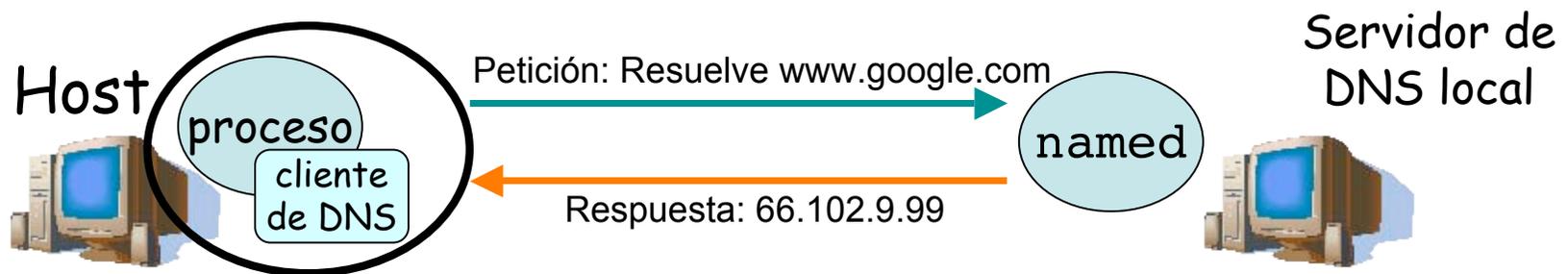
- Cada ISP posee un servidor de nombres local (...)
- Los hosts tienen configurado a su servidor local
- Cuando un host desea resolver un nombre hace la petición a su servidor local el cual le devuelve la respuesta (... ..)





# Funcionamiento

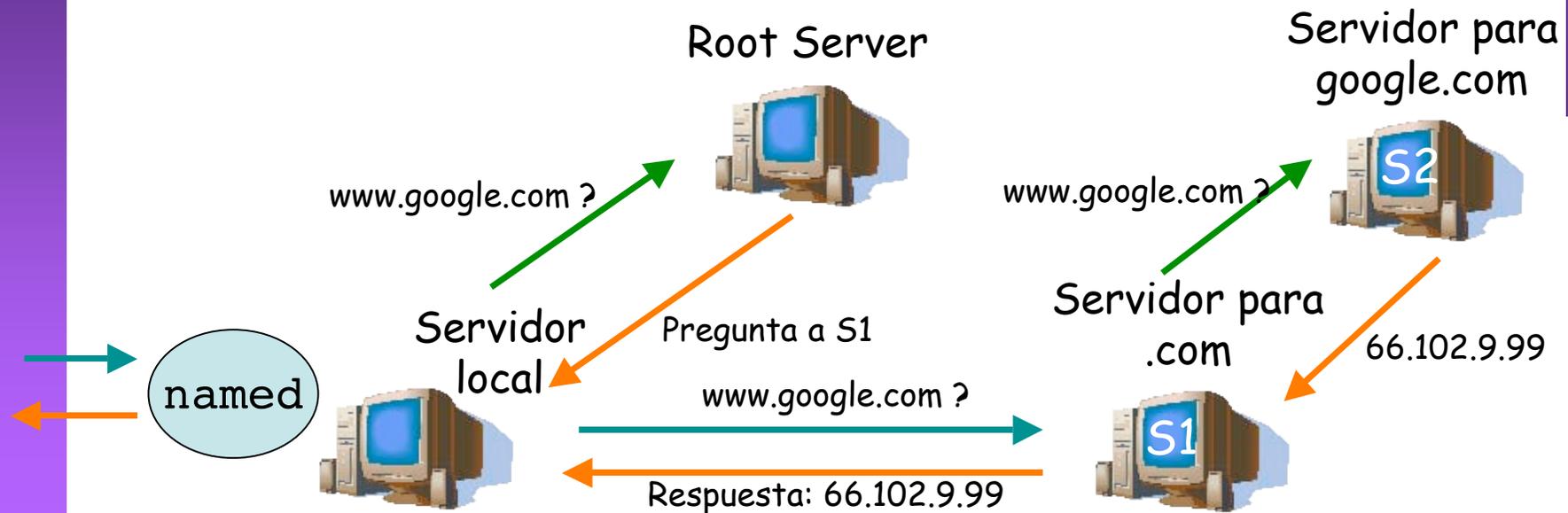
- ¿Cómo conoce la respuesta el servidor local?
  - Si es el servidor autoritario (**authoritative server**) para el dominio en el que está esa máquina él tiene la porción de la base de datos distribuida en la que está el mapeo (**zone file**)
  - Si no lo es preguntará a un **Root Server**





# Funcionamiento

- El servidor local pregunta a un **Root Server** (...)
- Éste le devuelve la dirección de un servidor intermedio (petición **iterativa**) (...)
- El Servidor local hace una petición recursiva a ese servidor (...)
- Continuará haciendo la petición (**recursiva**) hasta que llegue un servidor autoritario (. . .)
- Todas las peticiones son recursivas menos la petición al Root Server para reducir la carga sobre los Root





# DNS: Root name servers

- 13 en el mundo
- En el fichero de configuración de cada servidor de DNS





# TLDs, Authoritative Servers, cache

## Servidores de Top-level domains (TLD):

- Responsables de *com, org, net, edu*, (etc) y de los dominios raíz de países (*es, uk, fr, ca, jp, etc*)
- ESNIC para el TLD *.es* (<http://www.nic.es>)

## Authoritative DNS servers:

- Servidores DNS de organizaciones
- Mantienen el mapeo autorizado para los nombres dentro del dominio de la organización

## Fully Qualified Domain Name (FQDN)

- En realidad la raíz del árbol tiene también “nombre” pero es nulo
- Un FQDN incluye el nombre hasta la raíz, o sea, termina en un “.”  
[www.tlm.unavarra.es.](http://www.tlm.unavarra.es)

## Una vez que un servidor de DNS aprende un mapeo lo cachea

- Las entradas en la cache caducan tras un tiempo
- Normalmente los servidores de los TLD van a estar cacheados en los servidores locales
  - Así que los Root no se suelen visitar



# Contenido

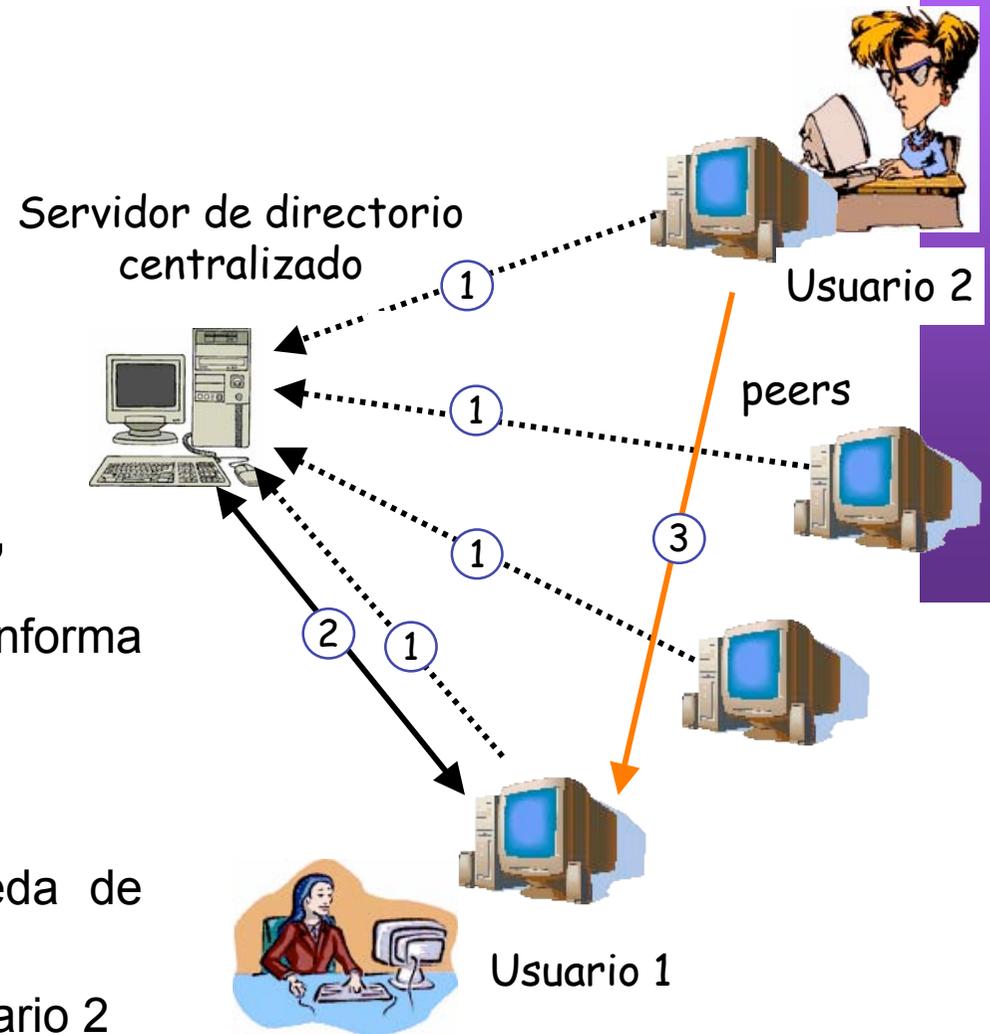
- DNS
- **P2P**
- Mensajería
- Proxy



# P2P: directorio centralizado

## Diseño original de “Napster”

- 1) Cuando un peer se conecta, informa al servidor central:
  - Dirección IP
  - contenido
- 2) Usuario 1 hace una búsqueda de “Requiem”
- 3) Usuario 1 pide el fichero a Usuario 2





# Ventajas e inconvenientes

## Ventajas

- Todos los peers son servidores
- **Altamente escalable**

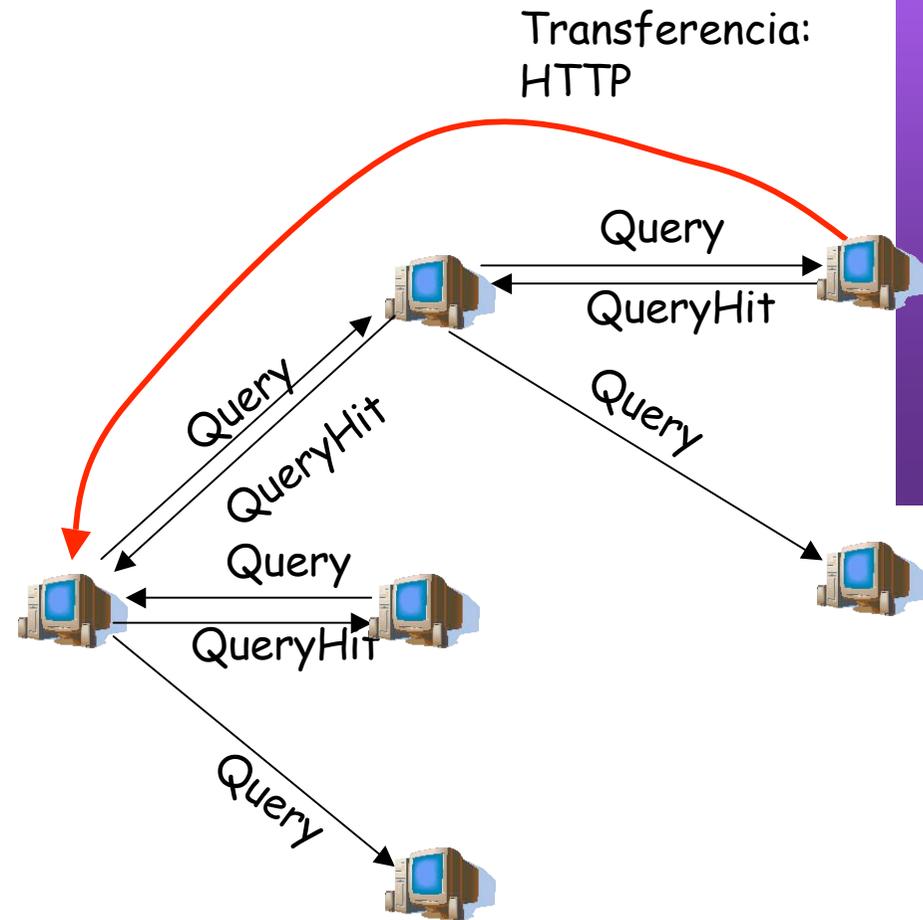
## Inconvenientes

- **Un punto de fallo central**
- Impone un límite de prestaciones
- Infracción de copyrights!



# Gnutella

- Completamente distribuido
- Dominio público
- Overlay network
  - Grafo
  - Cada conexión un enlace
- Petición de búsqueda enviada sobre las conexiones TCP
- peers reenvían la petición
- Respuesta enviada por el camino inverso
- Escalabilidad: limitar el alcance de la inundación





# Contenido

- DNS
- P2P
- **Mensajería**
- Proxy



# Servicios de conversación

- Conversación a líneas en máquinas UNIX: **write**

```
[daniel@t1m13 daniel]$ write lir
Hola tu
[daniel@t1m13 daniel]$
```

```
[lir@t1m13 lir]$
Message from daniel@t1m13.net.t1m.unavarra.es
on pts/0 at 18:39 ...
Hola tu
EOF
```

- Conversación en terminal UNIX completo: **talk**

```
[daniel@t1m13 daniel]$ talk lir@t1m21
```

```
[Connection established]
Hola
```

```
[-----]
Pues hola
```

```
Message from Talk_Daemon@t1m13.net.t1m.unavarra.es
at 18:30 ...
talk: connection requested by daniel@t1m13.
talk: respond with: talk daniel@t1m13.
[lir@t1m13 lir]$ talk daniel@t1m13
```

```
[Connection established]
Pues hola
```

```
[-----]
Hola
```



# Evolución de los servicios clásicos de conversación

- Internet Relay Chat (IRC):
  - Los clientes se conectan a un servidor central
  - Existen “habitaciones”. Todos los usuarios que ejecuten el comando para “unirse” a una habitación podrán leer lo que cualquier otro en esa habitación escriba (... ..)

```
otilio[9]% irc dmorato irc.arrakis.es

*** Connecting to port 6667 of server
-atreides.irc-hispano.org- *** Looking
-atreides.irc-hispano.org- *** Found y
*** As2gcu.DMOyqd.virtual is now your
+atreides.irc-hispano.org)
*** Welcome to the IRC-Hispano Interne
+atreides.irc-hispano.org)

% /join #pamplona
*** dmorato (daniel@As2gcu.DMOyqd.virt
*** Topic for #pamplona: Pamplona.cjb.
*** #pamplona CHaN 1065118105
*** #pamplona 1006939646
-CHaN- Bienvenido a #Pamplona,ciudad s
```

```
X-Chat Windows User Modes Settings Scripts & Plugins Help
#pamplona
Pamplona.cjb.net T N S I P M L K
3 ops, 43 total
MacServer
Marylinm
|Krom|
behennah
destroyz
LadyMarga
LaytoZZzz
Luke
Navarrico
usys
ZyArk
|Piro|
|[THC]|
^FIoS^
Op DeOp
Ban Kick
Send Dialog
Lookup Whois
dmorato
```



# Evolución de los servicios clásicos de conversación

## Messengers (...)





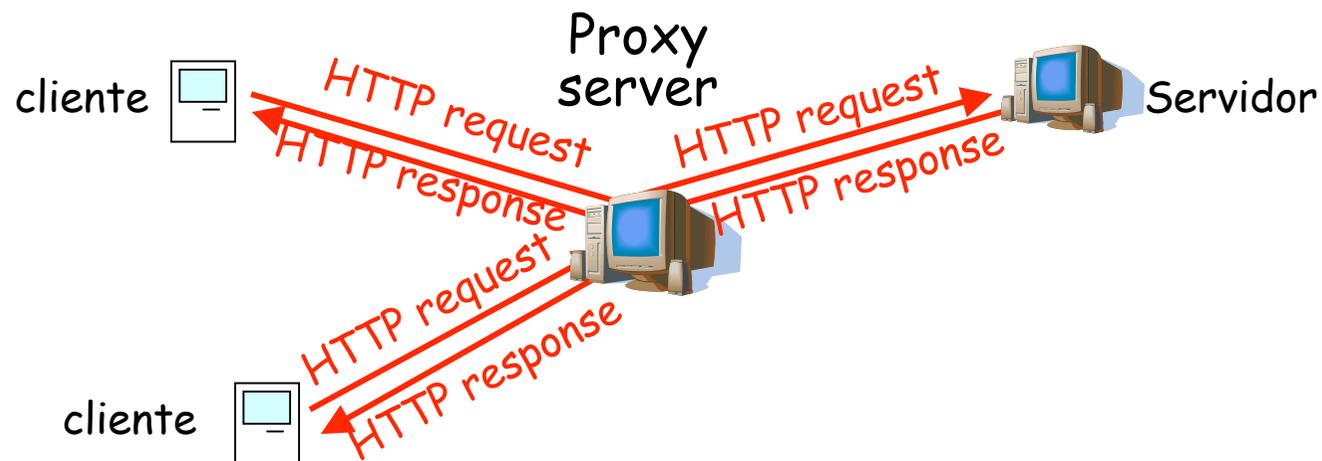
# Contenido

- DNS
- P2P
- Mensajería
- **Proxy**



# Proxy

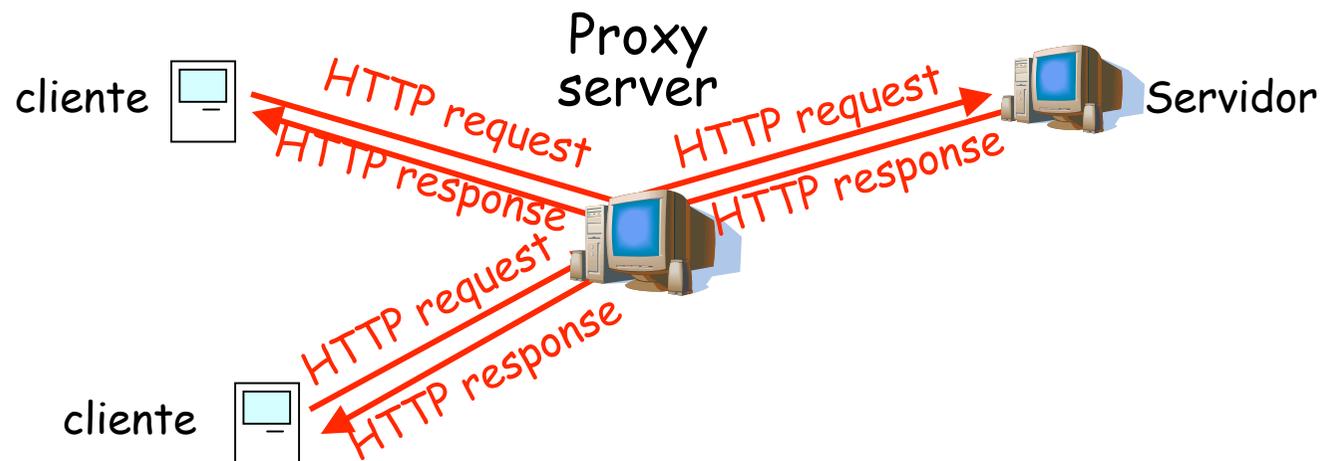
- Satisface las peticiones del cliente
- El cliente no requiere contactar con el servidor
- Normalmente es una máquina con un software de Proxy
- Proxy para numerosos servicios
- El más común es la Web
- El cliente, en vez de solicitar el documento al servidor lo solicita al Proxy
- El proxy lo solicita al servidor





# Proxy-cache

- El proxy puede guardar en cache los documentos que ha obtenido por solicitudes anteriores
- Cuando le llega una nueva petición busca el documento en la cache
- Si no lo encuentra lo pediría al servidor pero si lo encuentra lo entrega directamente de la cache





# Temario

- 1.- Introducción
- 2.- Nivel de enlace en LANs
- 3.- Interconexión de redes IP
- 4.- Nivel de transporte en Internet
- 5.- Nivel de aplicación en Internet**
  - Nivel de aplicación: WWW
  - FTP. Ejemplos
  - E-mail
  - DNS. P2P. Mensajería...



# Próxima clase

## *Problemas*