

Capa Aplicación: P2P

ELO322: Redes de Computadores Agustín J. González

Este material está basado en:

- Material de apoyo al texto *Computer Networking: A Top Down Approach Featuring the Internet*. Jim Kurose, Keith Ross.

Capítulo 2: Capa Aplicación

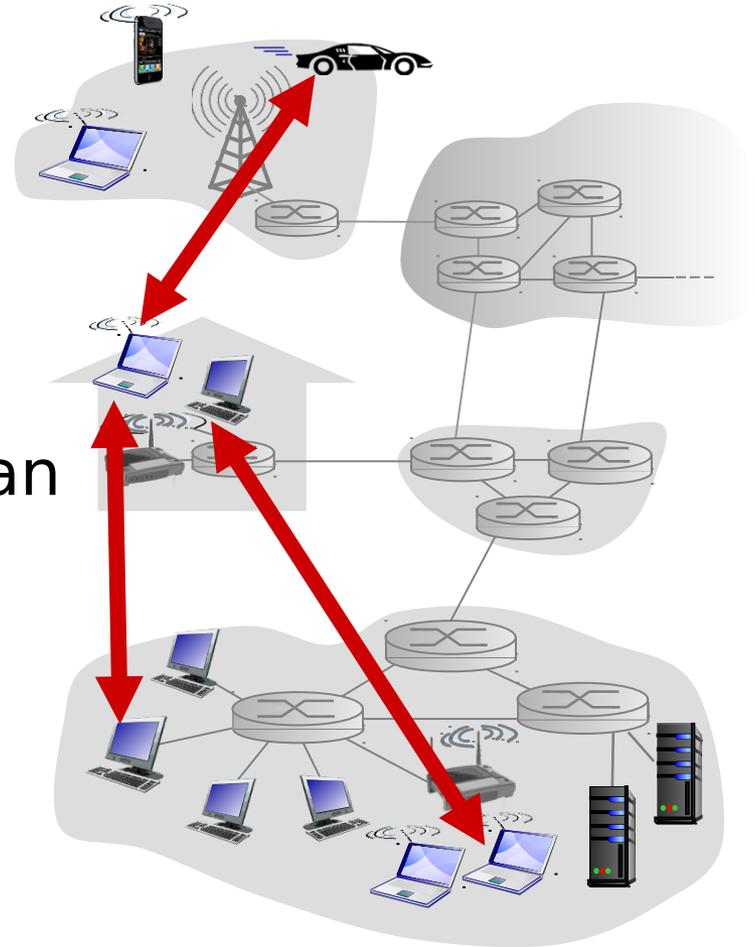
- ❑ 2.1 Principios de la aplicaciones de red
- ❑ 2.2 Web y HTTP
- ❑ 2.3 Correo Electrónico
 - SMTP, POP3, IMAP
- ❑ 2.4 DNS
- ❑ 2.5 P2P para archivos compartidos
- ❑ 2.6 Video streaming y redes de distribución de contenidos (Redes II)
- ❑ 2.7 Programación de sockets con UDP y TCP

Arquitectura P2P pura

- ❑ No hay servidor operando siempre
- ❑ Sistemas terminales arbitrarios se comunican directamente
- ❑ Pares se conectan intermitentemente y cambian dirección IP

Ejemplos (no necesariamente puras):

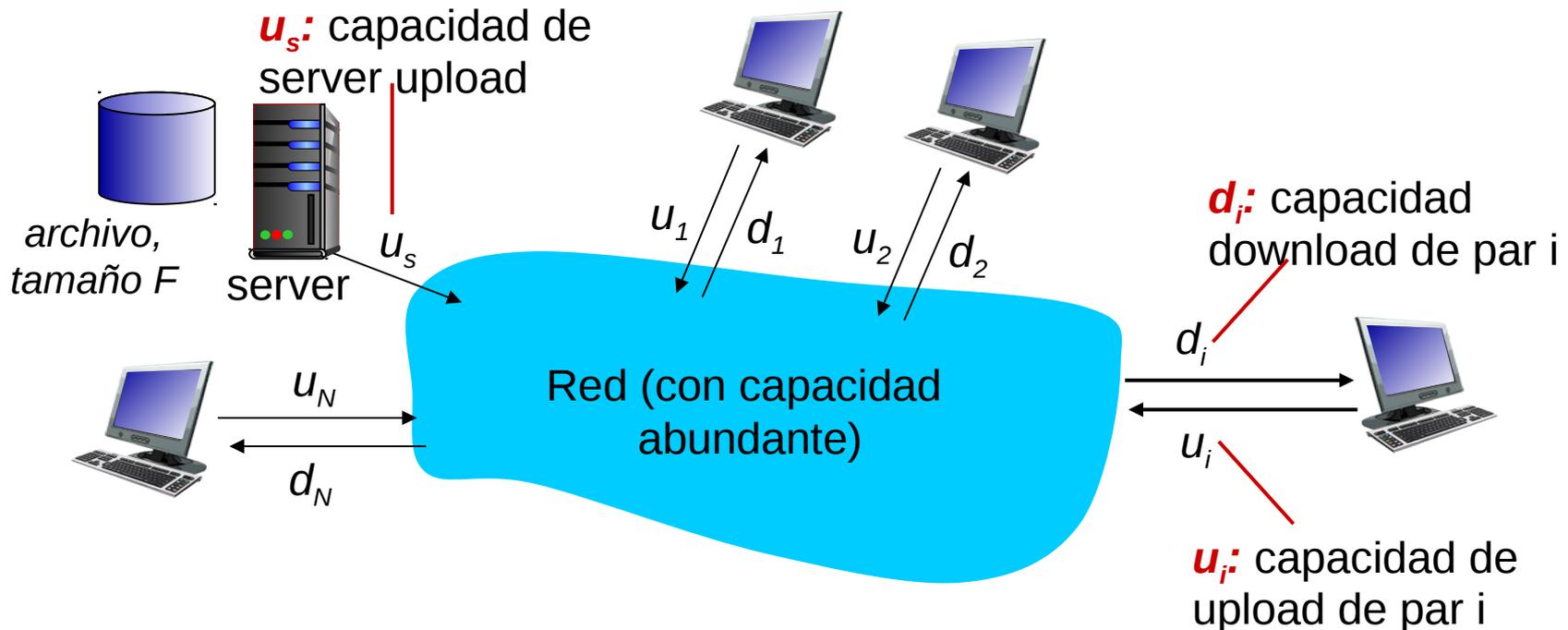
- Distribución de archivos (BitTorrent)
- Streaming (KanKan)
- VoIP (Skype)



Distribución de archivos: client-server vs P2P

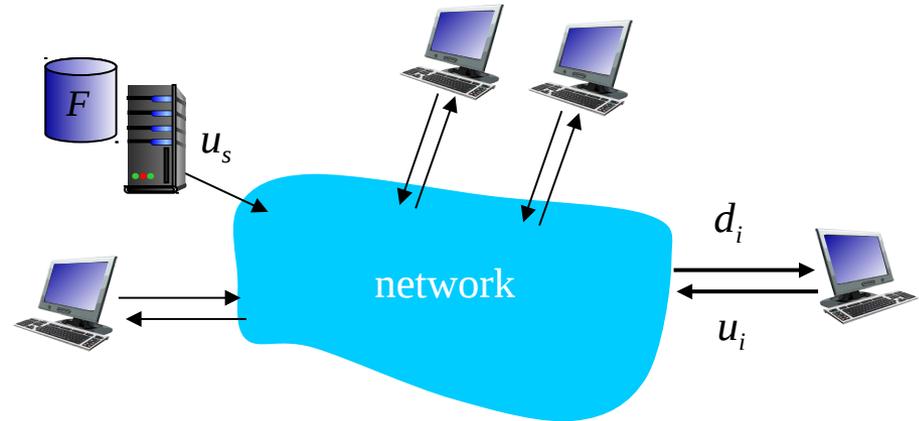
Pregunta: Cuánto demora distribuir un archivo (tamaño F) desde un servidor a N pares?

- La capacidad de upload/download de pares es un recurso limitado



Distribución de archivos: client-server

- **Transmisión del servidor:**
debe enviar secuencialmente (upload) N copias del archivo:
 - Tiempo envío 1 copia: F/u_s
 - Tiempo envío N copias: NF/u_s
- **Cliente:** cada cliente debe bajar una copia
 - d_{\min} = tasa mínima de bajada de cliente
 - Tiempo máximo de bajada de cliente: F/d_{\min}



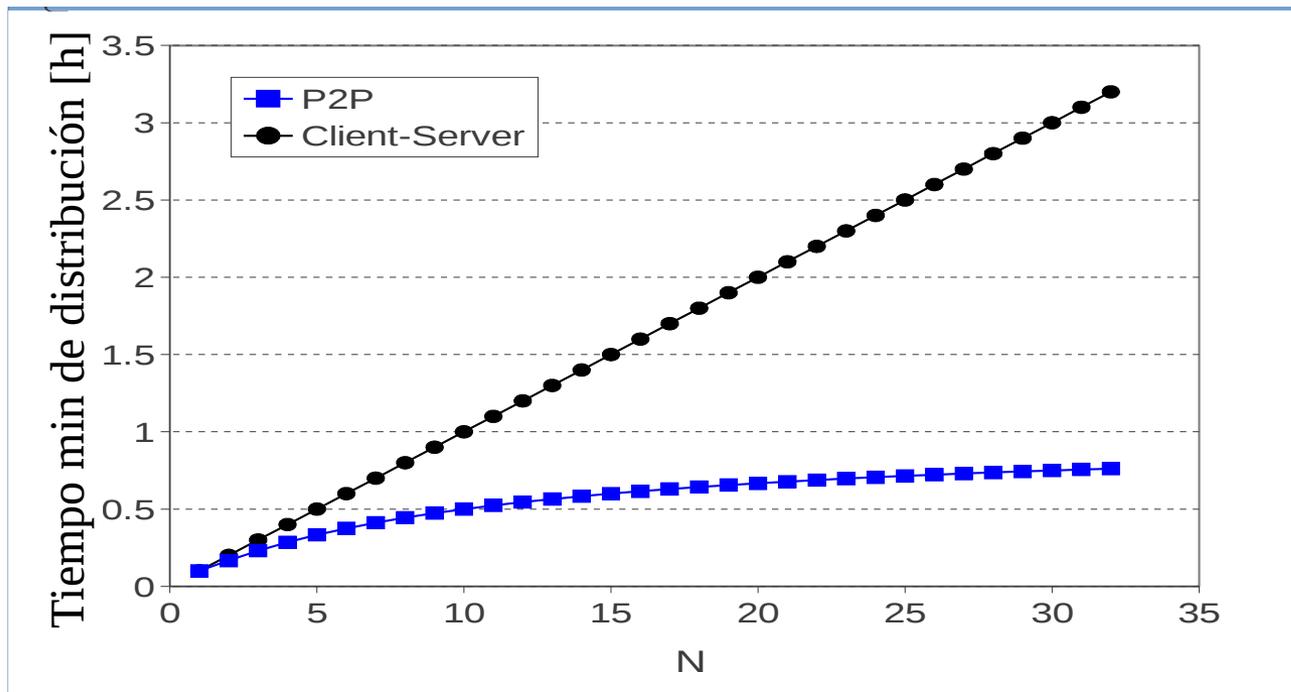
*Tiempo para distribuir F
a N clientes usando
enfoque client-server*

$$D_{c-s} \geq \max\{NF/u_s, F/d_{\min}\}$$

Aumento lineal con N

Ejemplo: Client-server vs. P2P

Tasa subida clientes = u , $F/u = 1$ [h], $u_s = 10u$, $d_{min} \geq u_s$

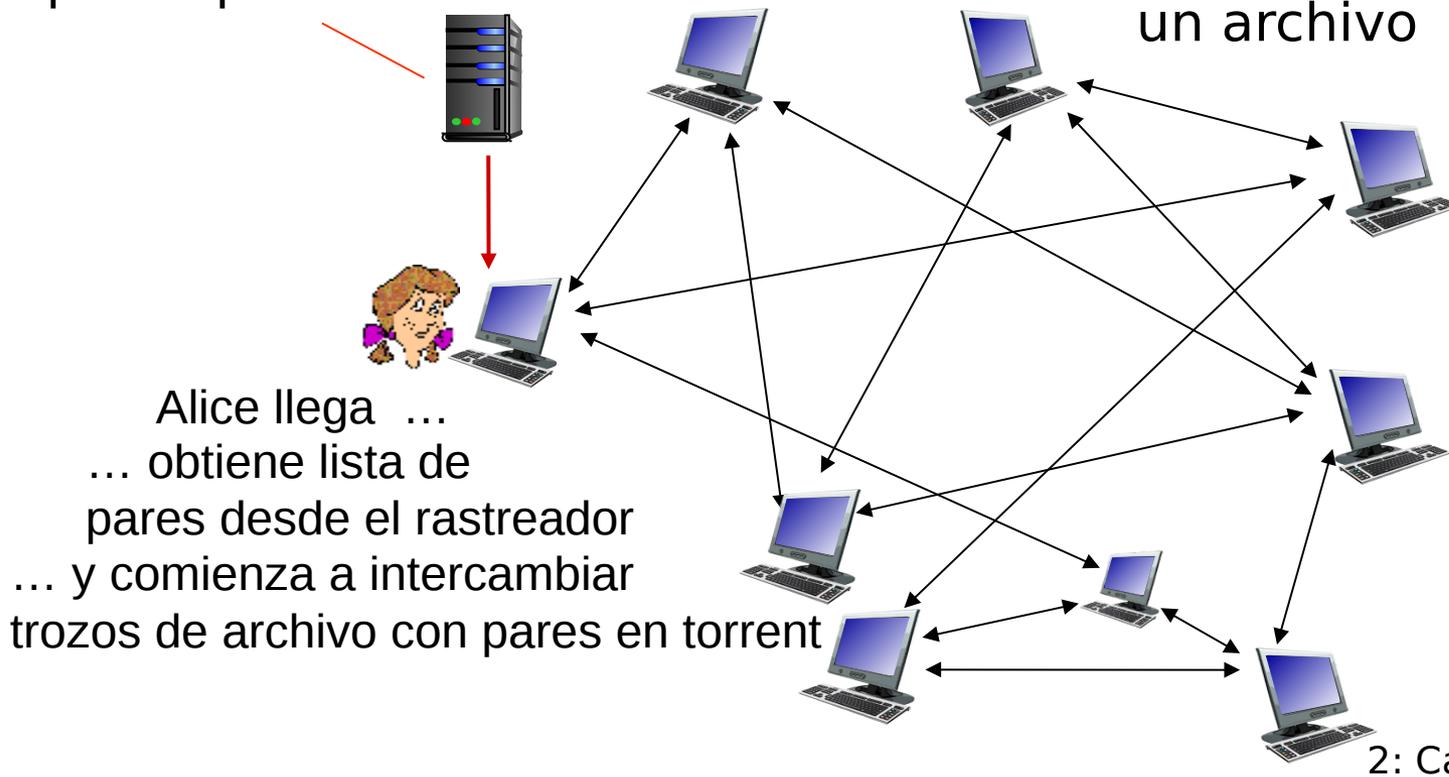


Distribución de archivos P2P: BitTorrent

- Archivo es dividido en 256Kb trozos
- Pares en torrent envían/reciben trozos de archivo

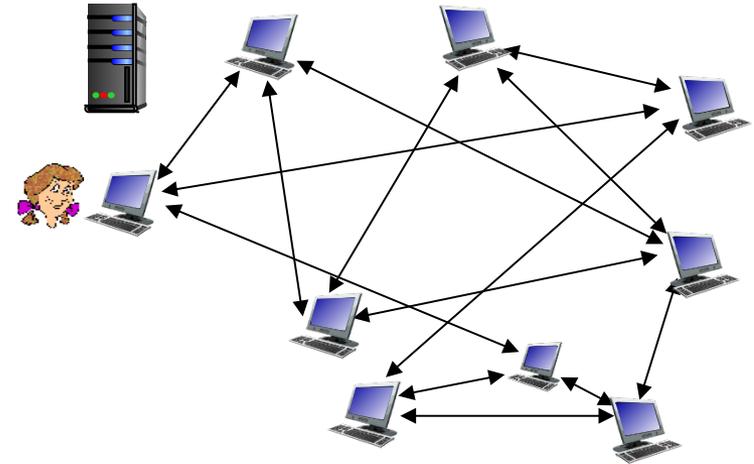
tracker: rastrea pares participados en torrent

torrent: grupo de pares intercambiando trozos de un archivo



Distribución de archivos P2P: BitTorrent

- Par se incorpora a torrent:
 - No tiene trozos, pero los acumulará en el tiempo desde otros pares
 - Se registra con rastreador para obtener lista de pares, se conecta a subconjunto de pares (“vecinos”)
- Mientras un usuario baja datos, éste sube trozos a otros pares
- Pares pueden ir y venir
- Cuando se tiene el archivo completo, un par puede irse (egoísta) o permanecer en torrent (altruista)



BitTorrent: petición y envío de trozos de archivo

Petición de trozos:

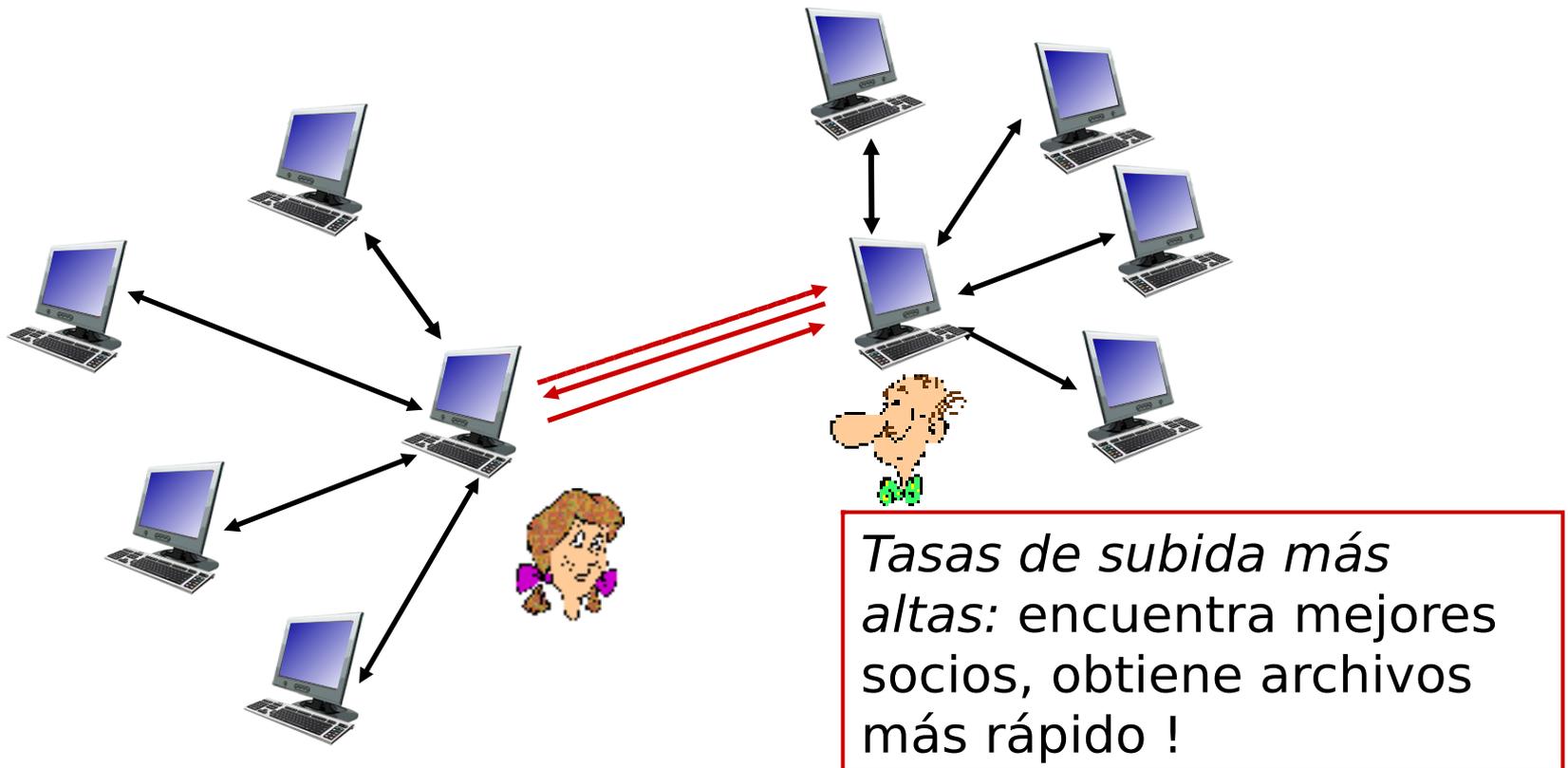
- En cualquier momento, pares diferentes tienen distinto subconjunto de trozos del archivo
- periódicamente, Alice pide a cada par la lista de trozos que ellos tienen
- Alice pide trozos ausentes desde pares, partiendo por el más escaso.

Envío de trozos

- Alice envía trozos a los 4 pares que le están enviando trozos a las mayores tasas.
 - Los otros pares son restringidos por Alicia (no reciben trozos de ella)
 - Cada 10 segundos se re-evalúa los 4 mejores.
- cada 30 [s]: aleatoriamente selecciona otro par, y comienza a enviar trozos (responde a peticiones)
 - “optimistamente coopera” con este par
 - El nuevo par escogido puede llegar a formar parte de los top 4

BitTorrent: Mano de vuelta

- (1) Alice “optimistamente coopera” con Bob
- (2) Alice se convierte en un proveedor top-4 de Bob; Bob devuelve mano
- (3) Bob se convierte en un proveedor top-4 de Alice



Capítulo 2: Capa Aplicación

- ❑ 2.1 Principios de la aplicaciones de red
- ❑ 2.2 Web y HTTP
- ❑ 2.3 Correo Electrónico
 - SMTP, POP3, IMAP
- ❑ 2.4 DNS
- ❑ 2.5 P2P para archivos compartidos
- ❑ 2.6 Video streaming y redes de distribución de contenidos (Redes II)
- ❑ 2.7 Programación de sockets con UDP y TCP