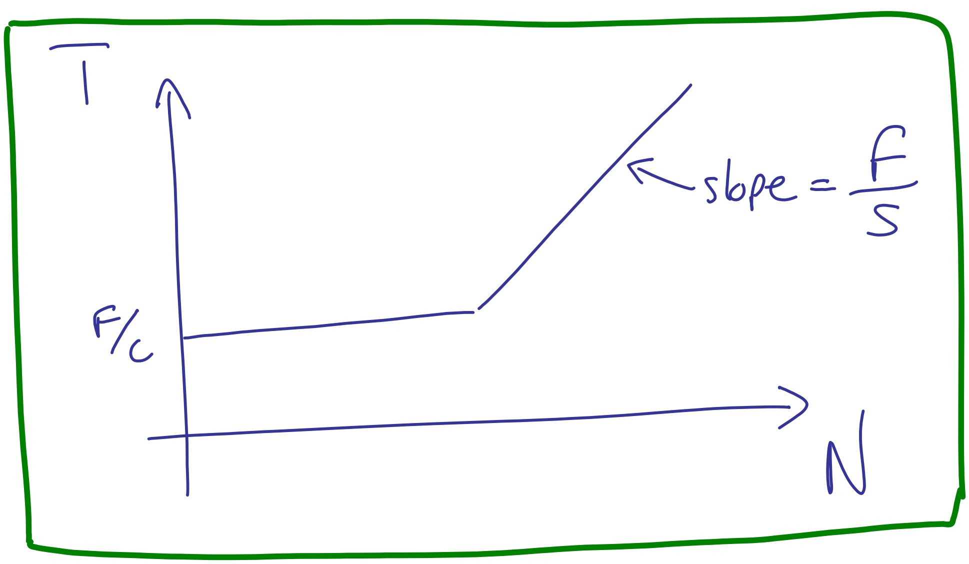
Client-server architecture

- server bandwidth S
- client bandwidth C
- number of clients N
- server has file of size F to be given to all clients
- time to download T
- if server is bottleneck, $T = \frac{NF}{S}$
- if clients are bottleneck, $T = \frac{F}{C}$
- so $$T = \max\left(\frac{NF}{S}, \frac{F}{C}\right)$$



P2P architecture

- no server
- client bandwidth C
- number of clients N
- each client has one "song" of size F/N that must be given to all other clients.
  (So total size of data is $\frac{F}{N} \times N = F$, same as for the client-server architecture).
- time to distribute all songs is T.

$$T = \frac{F}{N} \times \frac{1}{C} \times N = \frac{F}{C}$$

each client distributes its song at the same time

$\frac{F}{N}$: song size; $\frac{1}{C}$: rate of transfer; N: number of clients

(about N:) technically should be $N-1$, because each client distributes to $N-1$ other machines. But difference is negligible for large N.

The book gives a more sophisticated analysis with different upload + download speeds for each client, and assuming a server starts with all the songs.

But the basic message is:

sufficiently many peers (or 'clients') can always outperform a single server by working simultaneously.