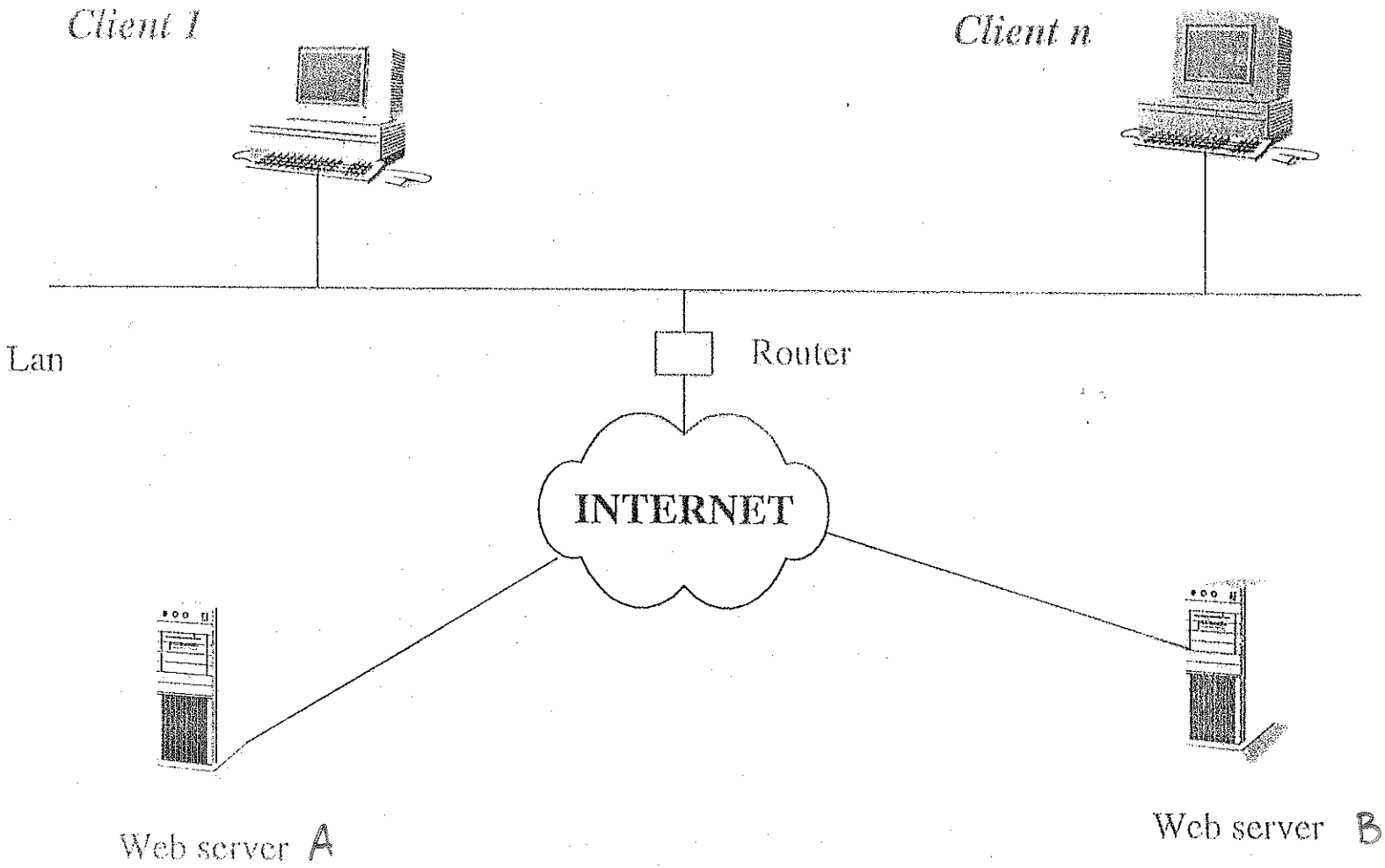
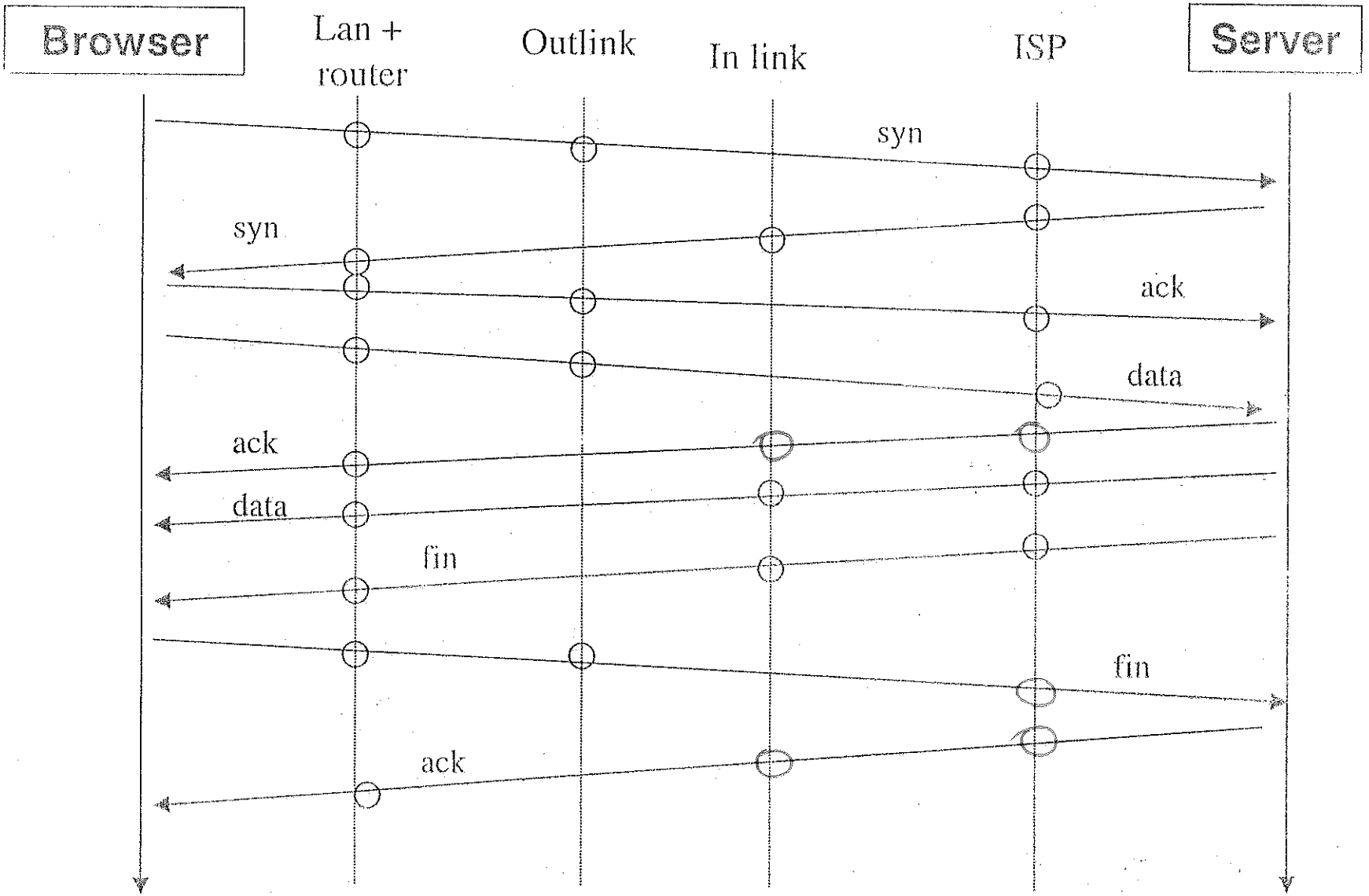
Client side models
no cache proxy server case

Prof. Bruno Ciciani
Facoltà di Ingegneria
Università di Roma La Sapienza

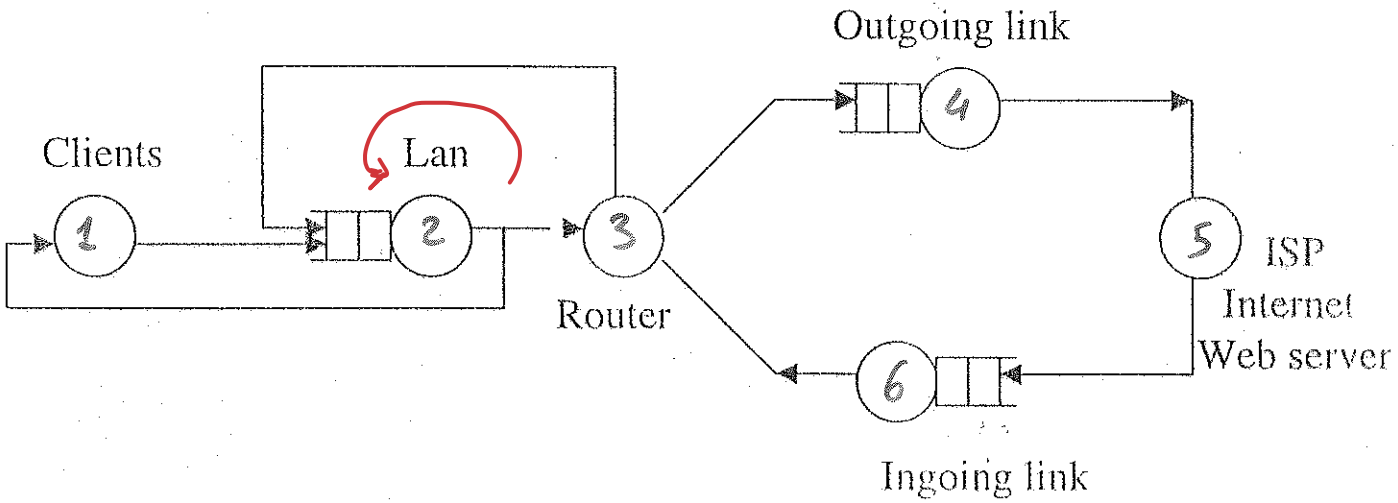
Accesso di un client al server senza proxy



Connessione TCP



QN model


Outgoing link
Clients
Lan
1
2
3
4
5
6
Router
ISP
Internet
Web server
Ingoing link

Input parameters

- LAN BANDWIDTH : in Megabits/sec
- Max PDU : maximum Protocol Data Unit size for the LAN's network layer protocol (in bytes)
  (MTU : Maximum Transmission Unit at Network Layer)
- Frame Ovhd : frame overhead of the LAN's link layer protocol (in bytes)
- Router Latency : router latency in microseconds per packet
- Link Bandwidth : bandwidth of the connection to the ISP (in megabits/sec.)
- Internet Delay RTT : Internet Round Trip Time (in milliseconds)
- Internet Data Rate : Internet data transfer rate (in Kbps)
- Browser Rate : rate, in HTTP operations/sec, at which a browser requests a new document when the user is in the think mode : the inverse of the user's think time

- NumberClients: number of client workstations
- PercentActive: percent of client workstations using the Web
- Avg Size HTTP Request: average size of the HTTP request sent by the browser to the server, in bytes
- Document requested
  - R: number of categories
  - DocumentSize$_r$: average document size (in KB), of category r ($r: 1 \ldots R$)
  - PercentSize$_r$: percent of documents requested that are in category r ($r: 1 \ldots R$)

$$\text{Document Size} = \left(\sum_{r=1}^{R} \text{DocumentSize}_r \cdot \text{PercentSize}_r \right) / \text{In Kibbytes}$$

CALCOLO DELLE SERVICE DEMAND

7 new

service demand at queue 2: LAN

- 2 TCP segment (without "body" message)
  x synchronization del protocollo HTTP
  (il terzo è "piggybaked" nella richiesta HTTP)
  ∴ diventano 2 DATAGRAMS

- 1 TCP segment
  x la HTTP request
  diventano N1 datagrams, funzione
  delle dimensioni della richiesta
  (Avg Size HTTP Request)

- Trasmissione del documento richiesto
  diventano N2 datagrams, funzione
  delle dimensioni del documento
  (1.024 x Document size)

- 3 TCP segments to close connection

$$D_{LAN} = 2 \cdot NetworkTime(0, LANBandwidth) +$$

$$NetworkTime(AvgSizeHTTPRequest, LANBandwidth) +$$

$$NetworkTime(1024 \times DocumentSize, LANBandwidth) +$$

$$3\ NetworkTime(0, LANBandwidth)$$

dove

$$NetworkTime(m, B) = \frac{8 \times [m + Overhead(m)]}{10^6 \cdot B}$$

Caso senza frammentaz. IP & $m \leq 65515$

$$Overhead(m) = (NDatagrams(m) - 1) \cdot (IPOvhd + FrameOvh$$

$$+ TCPOvhd + IPOvhd + FrameOvhd =$$

$$= TCPOvhd + NDatagrams(m) \cdot (IPOvhd + FrameOv$$

$$NDatagrams(m) = \left\lceil \frac{m + TCPOvhd}{\min_n \{MTU_n - IPOvhd\}} \right\rceil$$

Caso senza framment. IP

$l_m \leq 65515$ p.e. 40'000 byte

segmento TCP

20	40'000

hp: rete "Ethernet"

Overhead: 18

Max data area: 1500

} Max PDU Size = 1518

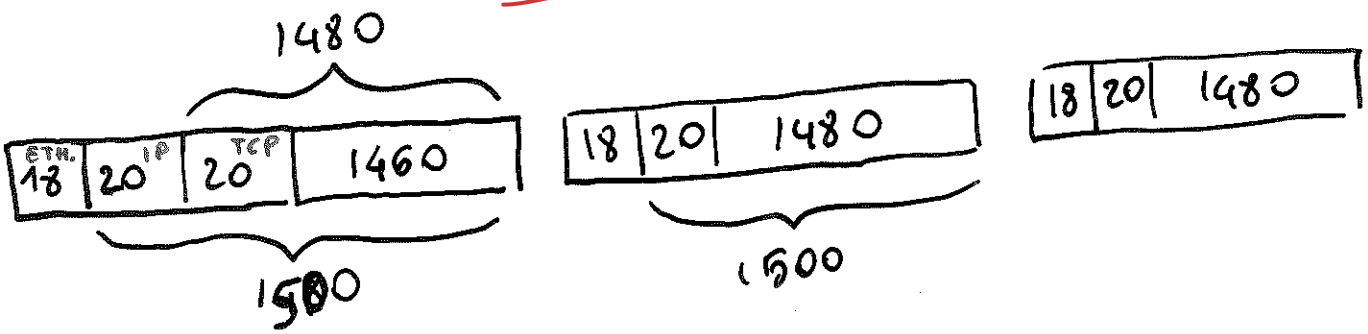
↓

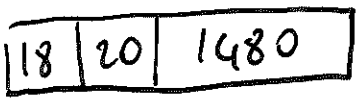
senza framm. IP: i datagram vengono generati con

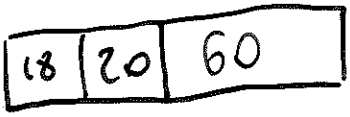
Overhead: 20

Max data area: 1480

} Max PDU Size = 1500



... 



$$N_{Datagram}(m) = \left\lceil \frac{40\,000 + 20}{1480} \right\rceil = 28$$

Caso senza framm. IP

& $m > 65515$ p.e. 100'000

Segment. TCP

20	65515

20	34485

hp: rete Ethernet

senza framm. IP

1° SEGMENTO

18	20	20	1460

18	20	1480

18	20	1480

...

18	20	1480

18	20	415

2° SEGMENTO

18	20	20	1460

18	20	1480

18	20	1480

...

18	20	1480

18	20	465

$$N_{segments} = \left\lceil \frac{Message\ Size}{65515} \right\rceil \quad \text{i.e.} \quad \left\lceil \frac{100'000}{65515} \right\rceil = 2$$

$$N_{Datagrams}(I\ segm) = \left\lceil \frac{m + TCPOvh}{\min_m \{MTU_m - IPOverh\}} \right\rceil =$$

$$\text{i.e.} \quad \left\lceil \frac{65515 + 20}{1480} \right\rceil = 45$$

$$N_{Datagrams}(II\ segm) = \left\lceil \quad \right\rceil =$$

$$\text{i.e.} \quad \left\lceil \frac{3485 + 20}{1480} \right\rceil = 24$$

$$N_{Datagrams} = N_{datagrams}(I\ segm) + N_{datagrams}(II\ segm)$$

stima approssimata

$$N_{datagrams} = \left\lceil \frac{Message\ Size + N_{Segments} \times TCPOvh}{\min_m \{MTU_m - IPOvh\}} \right\rceil$$

$$Overhead_n = N_{segments} \times TCP\ Ovh + N_{Datagrams} \times (IP\ Ovh + Frame\ Ovh_n)$$

$$Service\ Time(m) = Network\ Time(m, B) =$$

$$= \frac{8 \times [msg\ size^{(m)} + Overhead(m)]}{10^6 \cdot B}$$

Service demand at queue 3: router

$$D_{router} = \left[N_{DATAGRAMS} \left(1024 \times Document\ Size \right) + 6 \right] \times Router\ Latency \times 10^{-6}$$

↑ microsecond

6: 2 + 1 for synchronization segments

1 is piggy-backed in the HTTP request

1 for HTTP request

3 TCP segments to close connection

Service Demand at queue 4: OUTGOING LINK

$$D_{OUT_L} = \text{NetworkTime}[\text{Avg Size HTTP Request}, \text{Link Bandwidth}] + 3 \cdot \text{NetworkTime}[0, \text{Link Bandwidth}]$$

3 ← # of TCP synchronization segments to open and close the HTTP protocol

Service Demand at queue 5: ISP, Internet and WS

$$D_{int} = \frac{\text{Internet Delay RTT}}{10^3} + \frac{1}{2}\frac{\text{Internet Delay RTT}}{10^3} +$$

- $\frac{\text{Internet Delay RTT}}{10^3}$ → Time to establish a TCP connection
- $\frac{1}{2}\frac{\text{Internet Delay RTT}}{10^3}$ → Time to send the HTTP request
- $\frac{\text{Document Size}}{\text{Internet Data Rate}}$ → Time to transfer the document from the WS to the client

Service Demand at queue 6: INCOMING LINK

$$D_{INL} = NetworkTime[DocumentSize, LinkBandwidth] + 2\ NetworkTime[0, Linkbandwidth]$$

2 TCP synch. segments to open and close the connection

Esempio 10.4

	Residence Time (sec)	Utilization (%)
LAN	0.019	0.56
Router	0.001	0.03
Outgoing Link	0.076	2.18
Incoming Link	46.62	99.33

Throughput : 0.2910 pkt/sec

Response Time : 48.0 sec

$$\lceil 3.2 \rceil + \lceil 1.7 \rceil \neq \lceil 4.9 \rceil$$

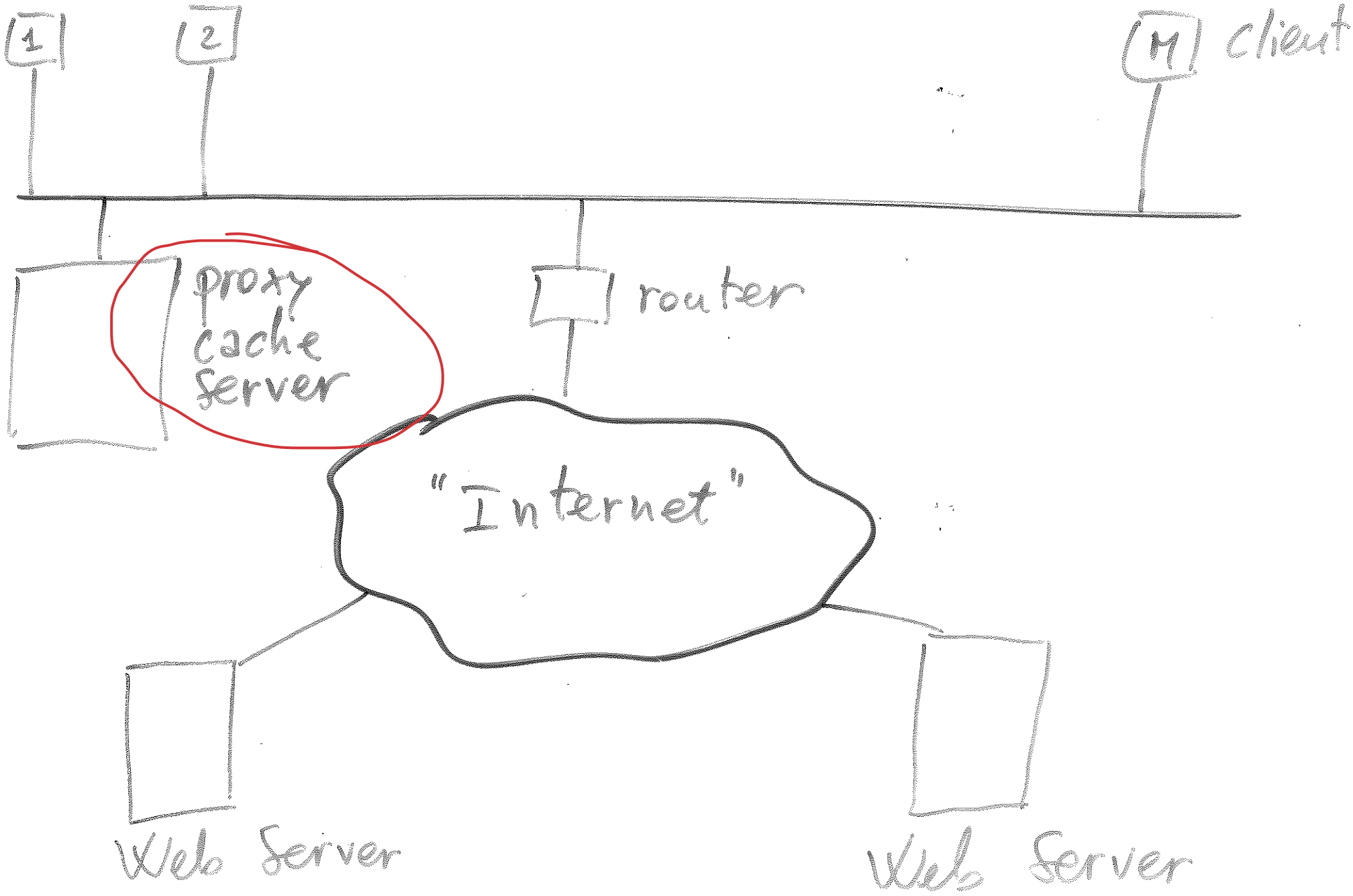
$$4 \qquad 2$$

=

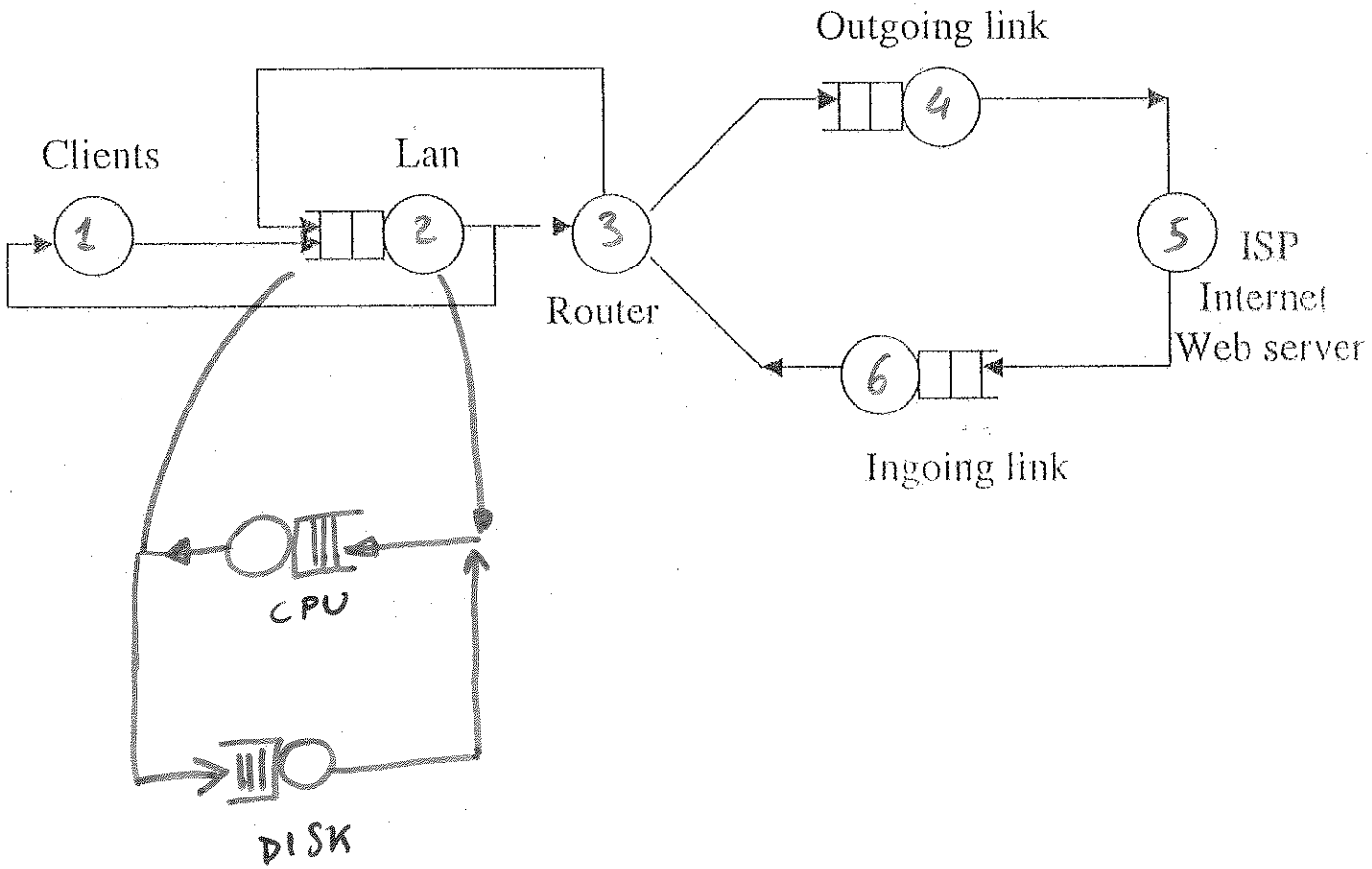
CLIENT SIDE MODELS

CACHE PROXY SERVER

CASE


1
2
M
client
proxy
cache
server
router
"Internet"
Web Server
Web Server

QN model


Outgoing link
Clients
Lan
1
2
3
4
5
6
Router
ISP
Internet
Web server
Ingoing link
CPU
DISK

Phit: fraction of requests that can be served from the proxy server's cache

Hit CPU time: CPU time, in seconds, needed to process the request at the cache proxy server in case of hit

Miss CPU time: CPU time, in seconds, needed to process a request at the cache proxy server in case of miss:

- process the request
- ask the request to the WS
- store the document
- replace a document
- send the document to the client

Disk-time: disk time for kilobyte at the cache proxy server (in millisecond)

$D'_{LAN} = p_{hit} D_{LAN} + (1 - p_{hit}) 2 D_{LAN}$

$D'_{router} = (1 - p_{hit}) D_{router}$

$D'_{OutL} = (1 - p_{hit}) D_{OutL}$

$D'_{Internet} = (1 - p_{hit}) D_{Internet}$

$D'_{InL} = (1 - p_{hit}) D_{InL}$

$D'_{CPU} = p_{hit}$ Hit CPU time +
$(1 - p_{hit})$ Miss CPU time

$D'_{Disk} = $ Disk time $\times \frac{\text{Document size}}{1000}$

Esempio 10.5

stessi parametri di 10.4

+

CPU Time x hit : 0.25 msec

CPU Time x miss : 0.50 msec

Disk service time : 6 msec x Kilobyte

Phit	Throughput (req/sec)	Response Time (sec)
0.20	0.364	37.82
0.30	0.416	32.72
0.40	0.483	27.61
0.50	0.581	22.50
0.0	0.292	48.0

Esempio 10.6

Stessi parametri di esempi 10.5 e 10.4

Link sostituito con un T1 link (1.544 Mbps)

Cache hit ratio : 40%

↓

new bottleneck : disk 45% utilz.

Response time 1.149 sec.

Throughput 3.347 rep/sec.