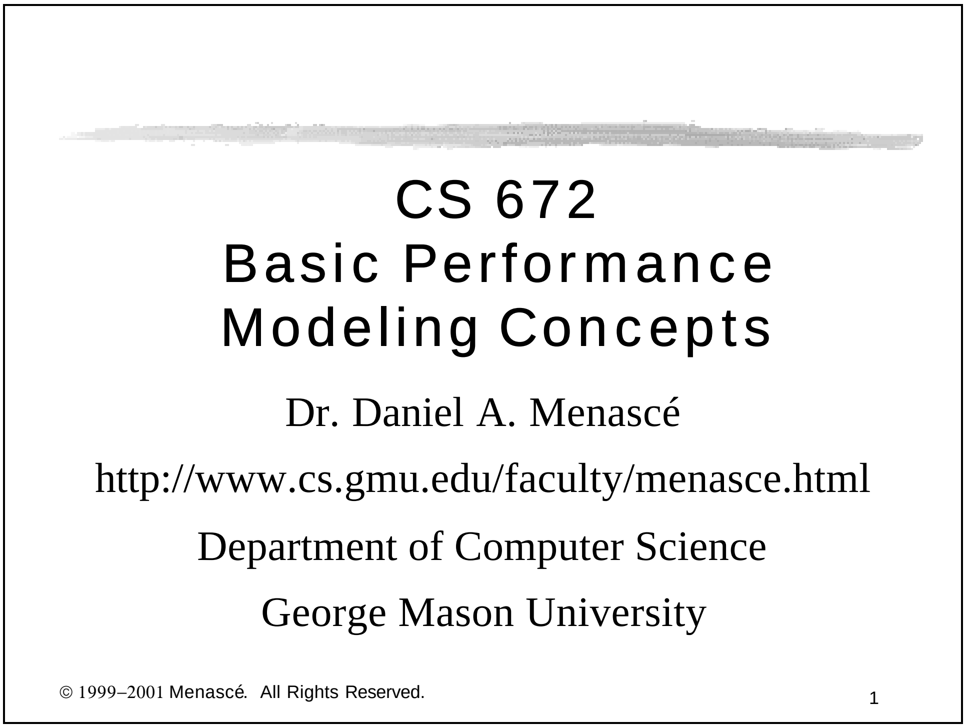


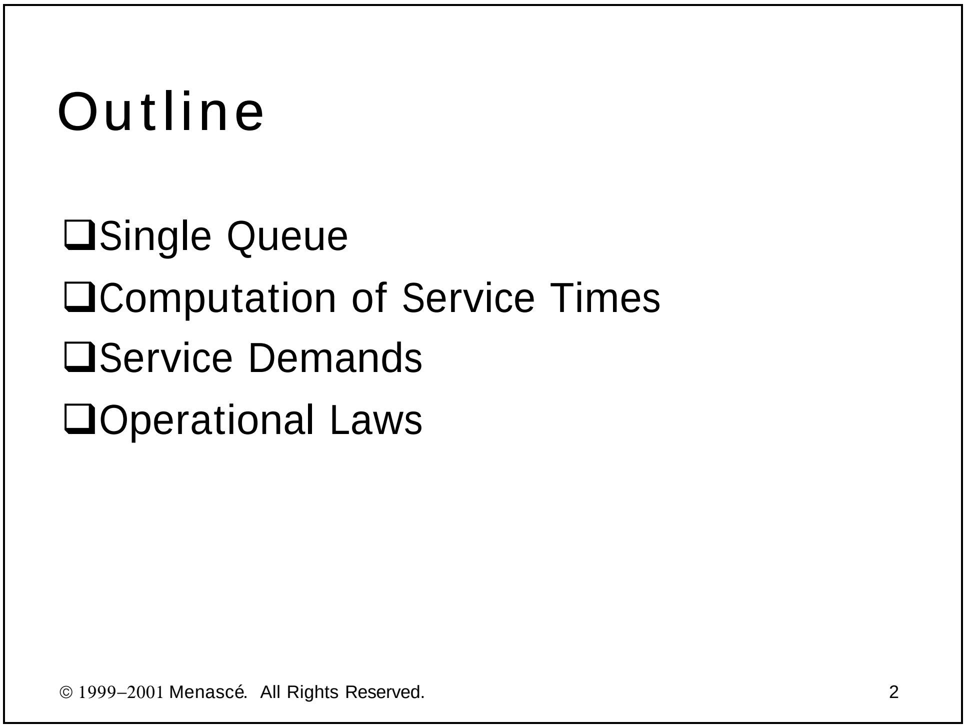
CS 672 Basic Performance Modeling Concepts

Dr. Daniel A. Menascé

http://www.cs.gmu.edu/faculty/menasce.html

Department of Computer Science

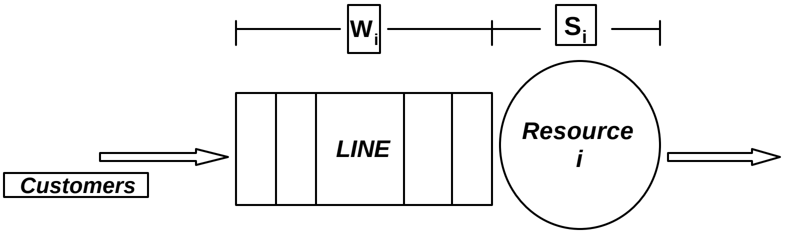
George Mason University



Outline

- ❑Single Queue
- ❑Computation of Service Times
- ❑Service Demands
- ❑Operational Laws

A Resource and its Queue

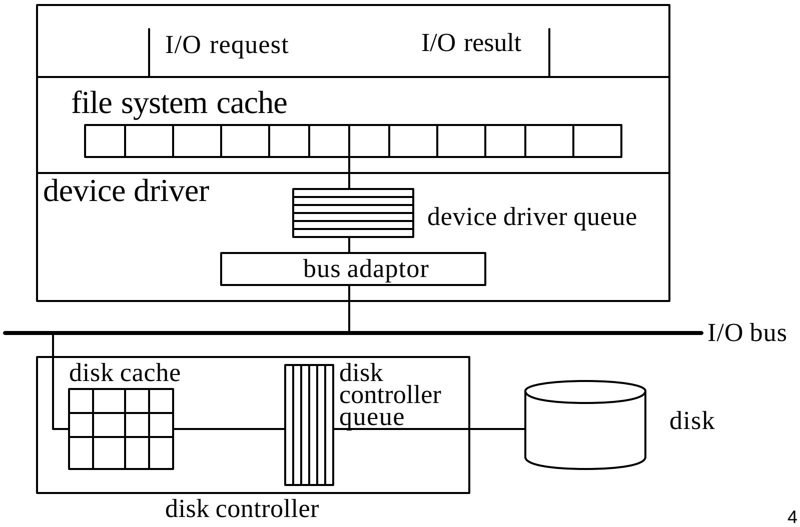


S_i : service time

W_i : waiting time

resource: CPU, disk, network, etc.

Computing Disk Service Times



Computing Disk Service Times

$$s_d = ContrTime + P_{miss}(Seek + Latency + TransferT)$$

$$TransferT = \frac{BlockSize}{TransferRate}$$

Computing Disk Service Times
Types of Workloads

Random Workload:
10, 201, 15, 1023, 45, 39, 782

Sequential Workload:

$$4, \underbrace{102, 103, 104, 105, 106}_{\text{run length= 5}}, 25, 88, \underbrace{32, 33, 34, 35, 36, 37, 38}_{\text{run length= 7}}, 29, 15$$

Computing Disk Service Times

Random Workload:

$$P_{miss} = 1$$

$$RunLength = 1$$

$$SeekTime = S_{rand}$$

$$Latency = 1/2 \times \mathrm{Re}\,volutionTime$$

Computing Disk Service Times

Sequential Workload:

$$P_{miss} = 1/\,RunLength$$

$$SeekTime = S_{rand}\ /\ RunLength$$

$$Latency = \frac{1/2 + (RunLength - 1)[(1 + U_d)/2]}{RunLength} \times$$

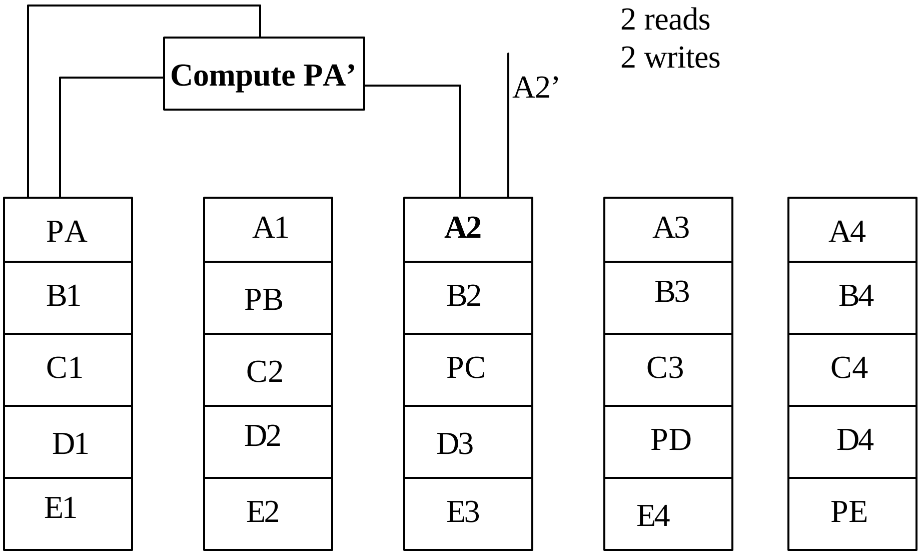
$$\mathrm{Re}\,volutionTime$$

$$U_d = l_d \times S_D$$

Disk Arrays

PA	A1	A2	A3	A4
B1	PB	B2	B3	B4
C1	C2	PC	C3	C4
D1	D2	D3	PD	D4
E1	E2	E3	E4	PE

Disk Arrays - Write One Stripe Unit

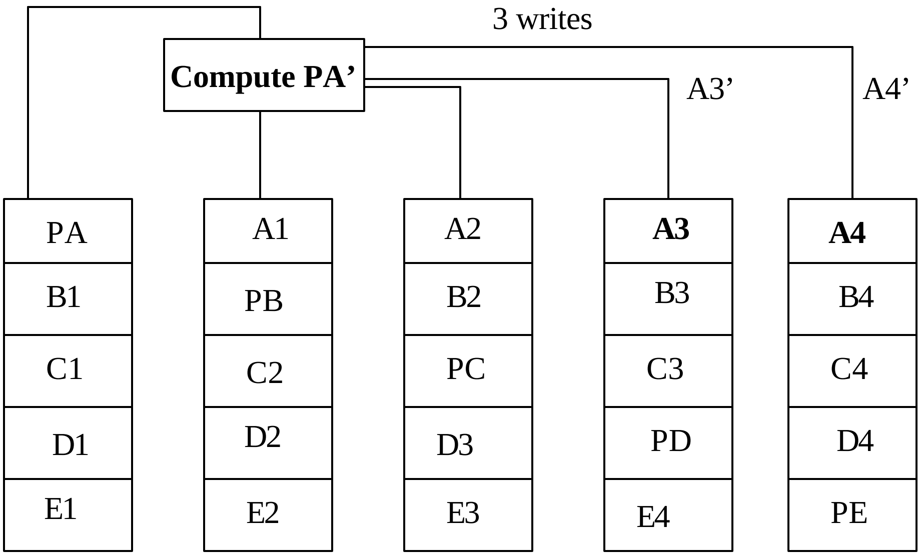


2 reads
2 writes

PA	A1	**A2**	A3	A4
B1	PB	B2	B3	B4
C1	C2	PC	C3	C4
D1	D2	D3	PD	D4
E1	E2	E3	E4	PE

Disk Arrays - Write Two Stripe Units

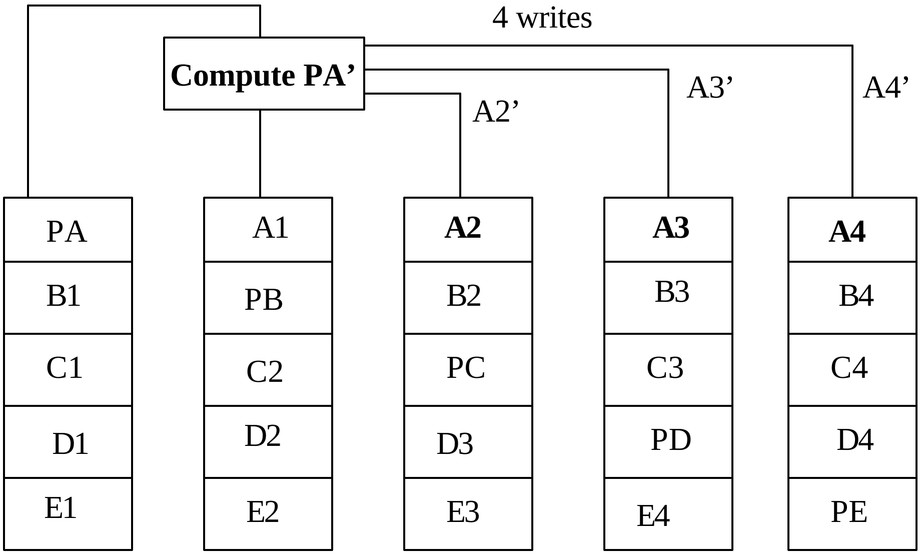
2 reads
3 writes



PA	A1	A2	**A3**	**A4**
B1	PB	B2	B3	B4
C1	C2	PC	C3	C4
D1	D2	D3	PD	D4
E1	E2	E3	E4	PE

Disk Arrays - Write Three Stripe Units

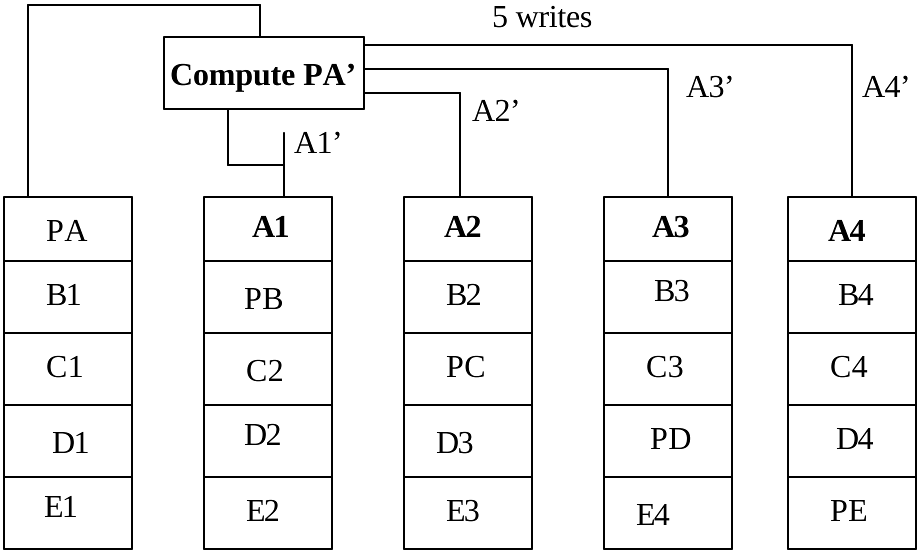
1 read
4 writes



PA	A1	**A2**	**A3**	**A4**
B1	PB	B2	B3	B4
C1	C2	PC	C3	C4
D1	D2	D3	PD	D4
E1	E2	E3	E4	PE

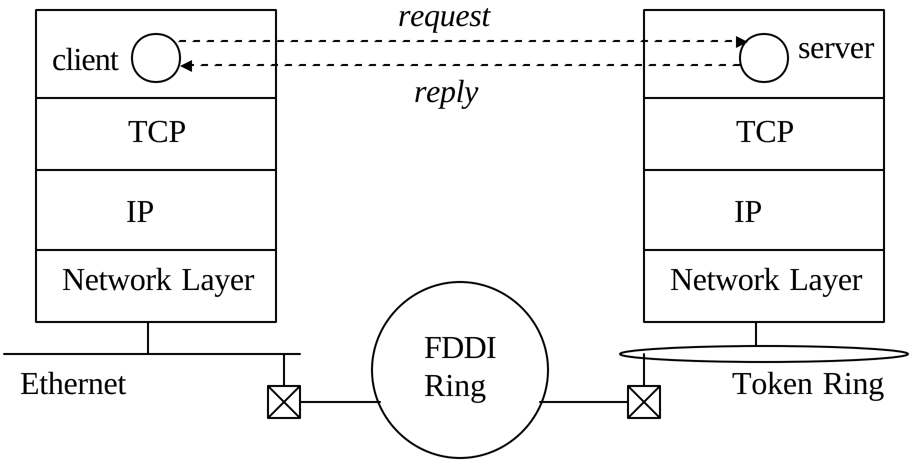
Disk Arrays - Write Four Stripe Units

0 reads
5 writes

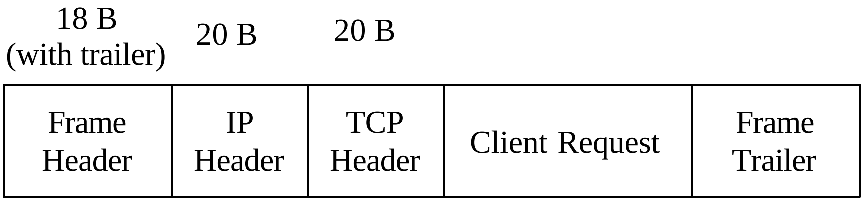


PA	**A1**	**A2**	**A3**	**A4**
B1	PB	B2	B3	B4
C1	C2	PC	C3	C4
D1	D2	D3	PD	D4
E1	E2	E3	E4	PE

Network Service Times



Network Service Times

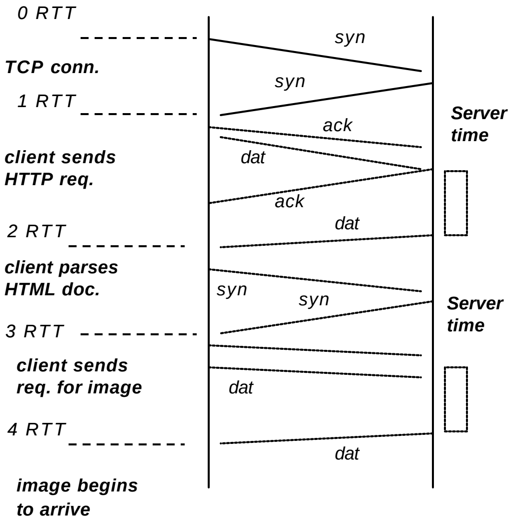


MTU=1500 bytes
Client Message Size = 2500 bytes
No Datagrams = ⌈2500 / (1500–20–20)⌉= 2
Total Overhead = 2 * (18+20+20)=116 bytes
Message Service Time = [2500+116]*8/10,000,000=0.02098 sec

Web Page Download Times

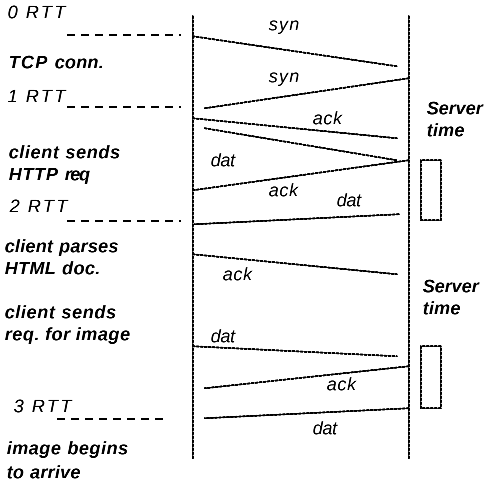
❑ Depend on
- ➢ type of HTTP protocol used
- ➢ page parameters
- ➢ network parameters
- ➢ TCP parameters

HTTP 1.0 interaction



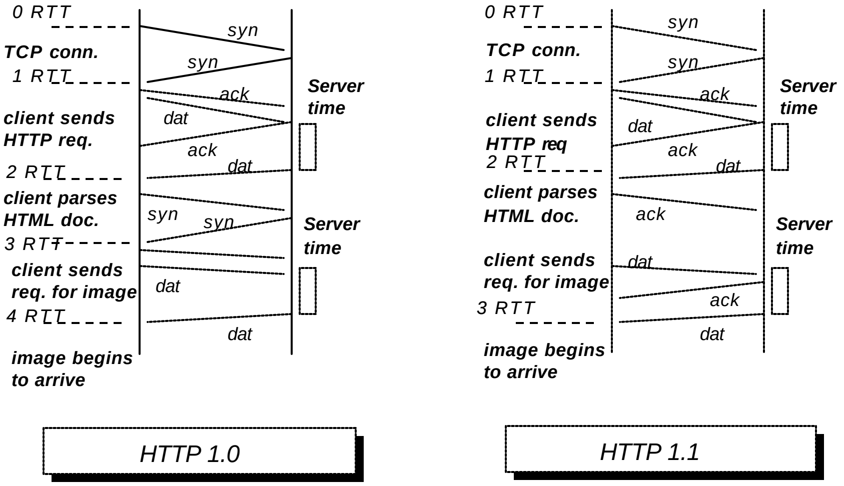
17

HTTP 1.1 interaction



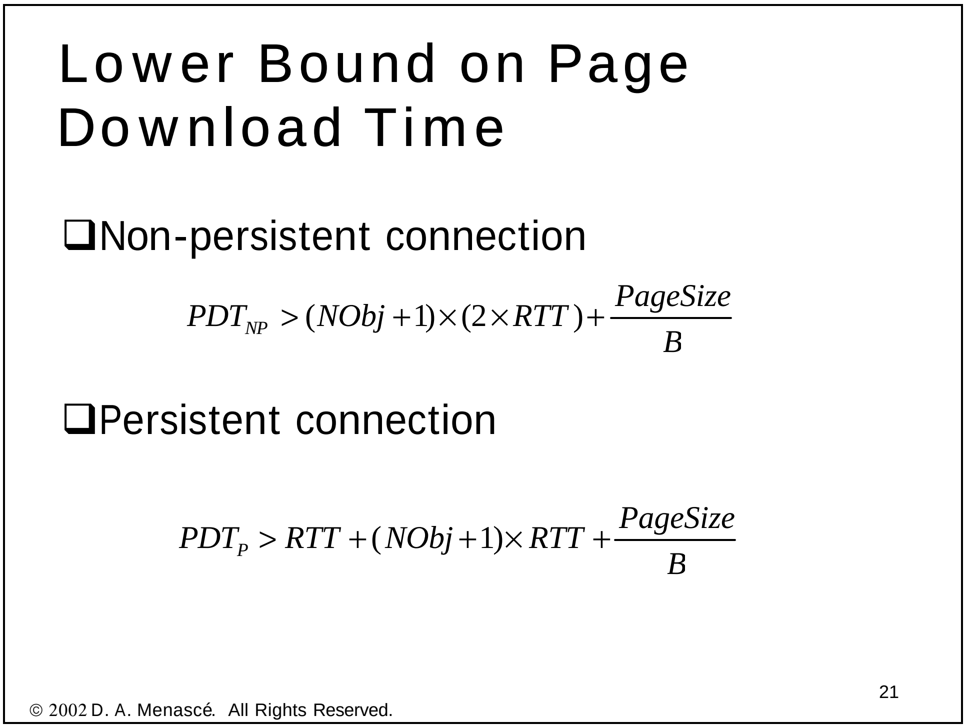
18

HTTP 1.0 and 1.1



Lower Bound on Page Download Time

- ❑PageSize: size, in bytes, of all objects of a page, including the HTTP header (290 bytes).
- ❑B: effective network bandwidth (in bps)
- ❑RTT: network round trip time (in sec)
- ❑NObj: Number of embedded objects in a page.



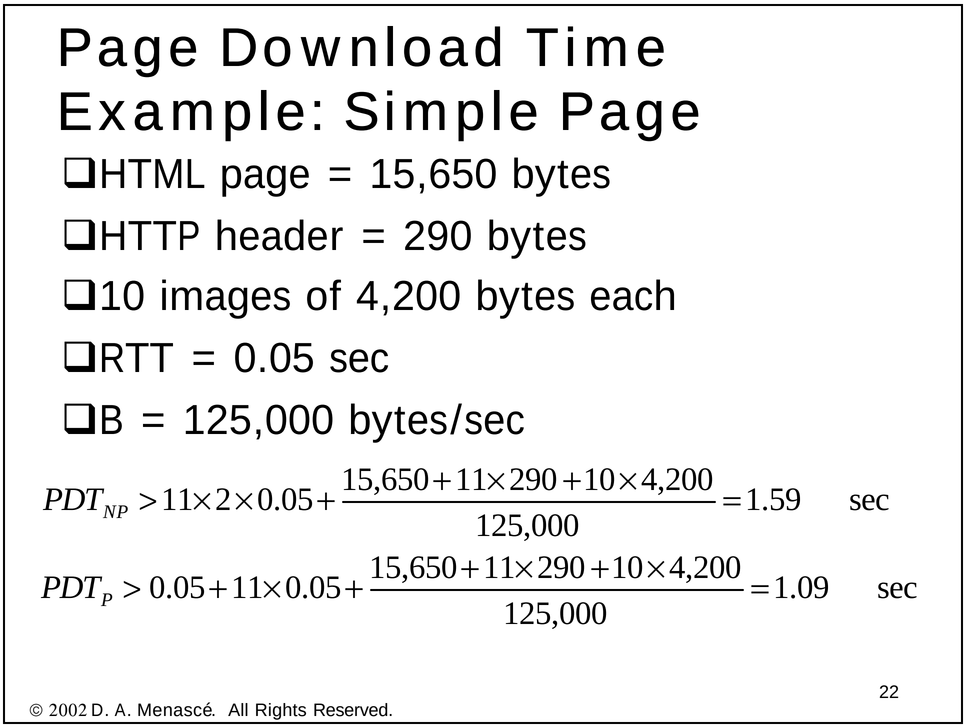
Lower Bound on Page Download Time

- ❑Non-persistent connection

$$PDT_{NP} > (NObj + 1) \times (2 \times RTT) + \frac{PageSize}{B}$$

- ❑Persistent connection

$$PDT_P > RTT + (NObj + 1) \times RTT + \frac{PageSize}{B}$$



Page Download Time Example: Simple Page

- ❑HTML page = 15,650 bytes
- ❑HTTP header = 290 bytes
- ❑10 images of 4,200 bytes each
- ❑RTT = 0.05 sec
- ❑B = 125,000 bytes/sec

$$PDT_{NP} > 11 \times 2 \times 0.05 + \frac{15{,}650 + 11 \times 290 + 10 \times 4{,}200}{125{,}000} = 1.59 \quad \text{sec}$$

$$PDT_P > 0.05 + 11 \times 0.05 + \frac{15{,}650 + 11 \times 290 + 10 \times 4{,}200}{125{,}000} = 1.09 \quad \text{sec}$$

Page Download Time Example: Elaborate Page

❑HTML page = 15,650 bytes

❑HTTP header = 290 bytes

❑20 images of 20,000 bytes each

❑RTT = 0.05 sec

❑B = 125,000 bytes/sec

$$PDT_{NP} > 21 \times 2 \times 0.05 + \frac{15{,}650 + 21 \times 290 + 20 \times 20{,}000}{125{,}000} = 5.47 \quad \text{sec}$$

$$PDT_{P} > 0.05 + 21 \times 0.05 + \frac{15{,}650 + 21 \times 290 + 20 \times 20{,}000}{125{,}000} = 4.47 \quad \text{sec}$$

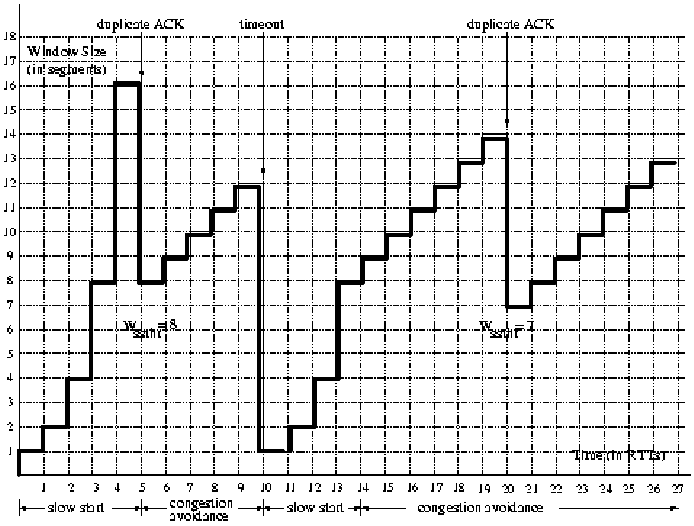
TCP Throughput

❑Depends on:

- ➢Packet Loss Ratio
- ➢Round Trip Time
- ➢Wm: Maximum Receiver Window Size (advertised by the receiver at connection establishment time)
- ➢TCP timeout
- ➢Network Bandwith
- ➢Maximum Segment Size

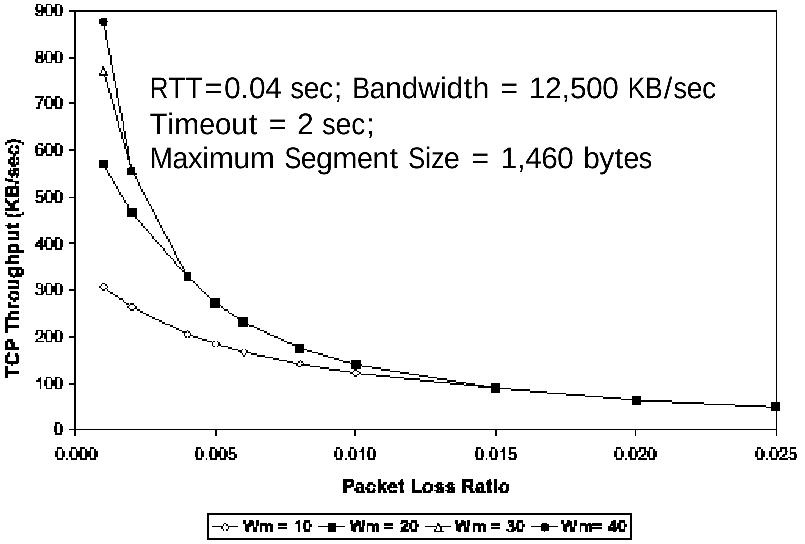
TCP: Window size vs time (in RTTs)

Window Size (segments)



Time (in RTTs)

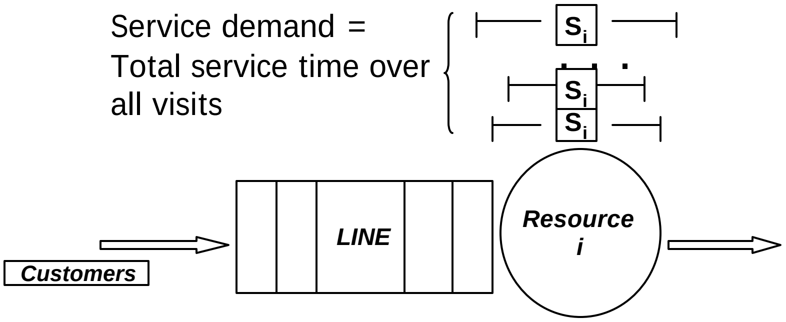
TCP Throughput



RTT=0.04 sec; Bandwidth = 12,500 KB/sec
Timeout = 2 sec;
Maximum Segment Size = 1,460 bytes

Service Demand (D_i)

Service demand = Total service time over all visits



S_i : service time
D_i : service demand

Service Demand Example

- ❑Requests to a Web site use two disks. The service times at each of the disks for each I/O carried out by a single request are

	Service Time (msec)	
I/O	**Disk 1**	**Disk 2**
1	12	12
2	20	15
3	15	14
4	18	-
	65	41

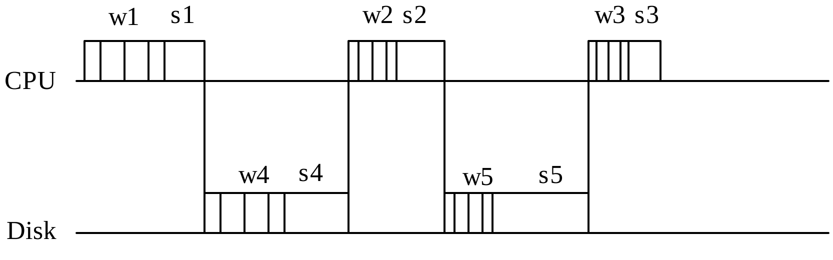
Service demand at disk 1

Service demand at disk 2

Important take home!

- ❑ Service demands are <u>important</u> parameters for performance models
- ❑ Service demands are easy to measure. Service times are much harder to obtain!
- ❑ Service demands are associated with a type of request and a resource.
- ❑ Service demands are measured in time units (e.g., sec, msec)
- ❑ Service demands are load independent!
- ❑ More on this to come ...

Service Demand

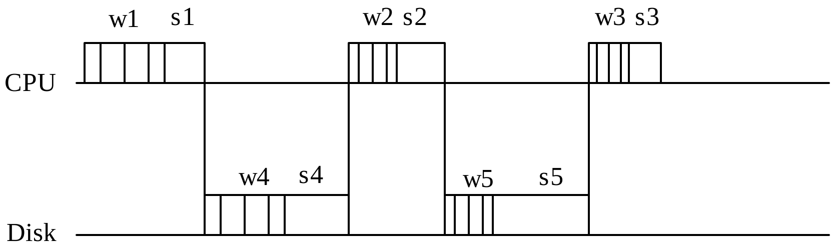


Service demand at the CPU = s1 + s2 + s3
Service demand at the disk = s4 + s5

▥ Waiting time

□ Service time

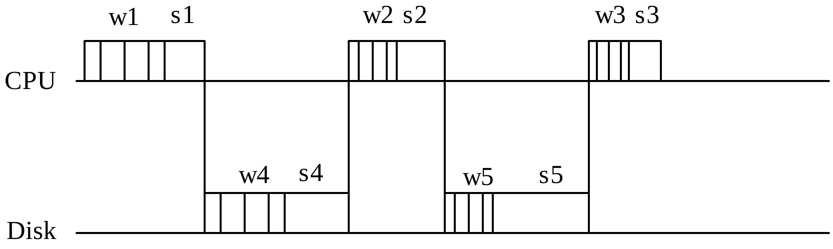
Queuing Time



Queuing time at the CPU = w1 + w2 + w3
Queuing time at the disk = w4 + w5

▥ Waiting time　　□ Service time

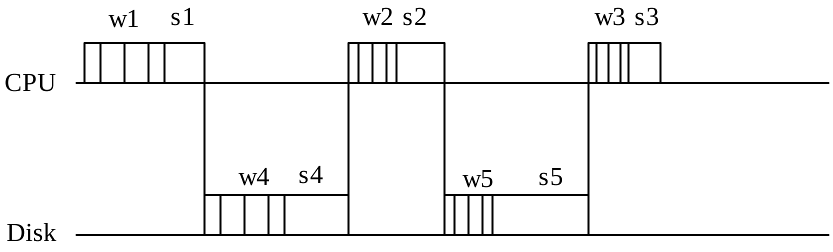
Residence Time



Residence time at the CPU = w1 + s1 + w2 + s2 + w3 + s3
Residence time at the disk = w4 + s4 + w5 + s5

▥ Waiting time　　□ Service time

Response Time



Response time = Residence time at the CPU + Residence time at the disk

▥ Waiting time

□ Service time

Queuing Basic Concepts

- ❑ Total time spent by a request during the j^{th} visit to a resource *i*:
  - ➢ Service time (S_i^j): period of time a request is receiving service from resource *i*, such as CPU or disk.
  - ➢ Waiting time (W_i^j): the time spent by a request waiting access to resource *i*

Basic Queuing Concepts

- ❑ <u>Service Demand</u> (D_i) is the sum of all service times for a request at resource *i*

$$D_{scpu} = S^1_{scpu} + S^2_{scpu}$$

- ❑ <u>Queuing Time</u> (Q_i) is the sum of all waiting times for a request at resource *i*

$$Q_{scpu} = W^1_{scpu} + W^2_{scpu}$$

Basic Queuing Concepts

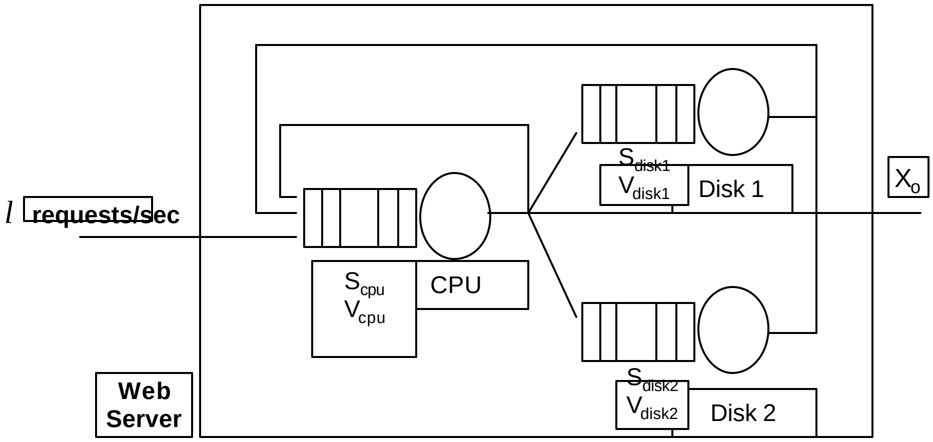
- ❑ <u>Residence Time</u> (R'_i) at resource *i* is the sum of service demand plus queuing time.

$$R'_i = Q_i + D_i$$

- ❑ <u>Response time</u> (R_r) of a request *r* is the sum of that request's residence time at all resources.

$$R_{server} = R'_{cpu} + R'_{disk}$$

Computer Systems Have Many Resources!



l requests/sec

Web Server

S_{cpu} V_{cpu}

CPU

S_{disk1} V_{disk1}

Disk 1

S_{disk2} V_{disk2}

Disk 2

X_o

Notation (1)

❑V_i: average number of visits to queue *i* by a request;

❑$S_{i:}$ average service time of a request at queue *i* per visit to the resource;

❑λ_i average arrival rate of requests to queue *i*

❑D_i service demand of a request at queue *i*,

❑$D_i = V_i \times S_i$

Notation (2)

❑ $N_{i:}$ average number of requests at queue *i*, waiting or receiving service from the resource

❑ $X_{i:}$ average throughput of queue *i*, i.e. average number of requests that complete from queue *i* per unit of time

❑ $X_{o:}$ average system throughput, defined as the number of requests that complete per unit of time.

Basic Performance Results

Utilization Law

❑ The utilization (U_i) of resource i is the fraction of time that the resource is busy.

$$U_i = X_i * S_i = \lambda_i * S_i$$

Example of Utilization Law: iostat in Unix

r/s	w/s	Kr/s	Kw/s	svc_t_(msec)
0.8	7.4	6.2	131.2	136.7
0.2	4.4	1.6	113.6	61
1	14.8	8	438.4	61.3
13	1.2	128	134.4	16.8
0.2	0	1.6	0	12.4
0	0.2	0	25.6	40.9
0	0	0	0	0
0	4	0	28.6	116
0	0	0	0	0
0	0	0	0	0
3	0	24	0	11.4
0	0.6	0	35.2	35.2
0	0	0	0	0
0	0.2	0	1.6	17.3
1.30	2.34	12.10	64.90	36.36

$$X_{disk} = 1.3 + 2.34 = 3.64 \text{ IOs/sec}$$

$$U_{disk} = X_{disk} \times S_{disk} = 3.64 \times 0.03636 = 13.24\%$$

Utilization Law: example

- ❑ A network segment transmits 1,000 packets/sec. Each packet has an average transmission time equal to 0.15 msec.
- ❑ What is the utilization of the LAN segment?

Utilization Law: example

- ❑ A network segment transmits 1,000 packets/sec. Each packet has an average transmission time equal to 0.15 msec.
- ❑ What is the utilization of the LAN segment?

$$U_{LAN} = X_{LAN} * S_{LAN} = 1{,}000 * 0.00015 = 0.15 = 15\%$$

Basic Performance Results

Forced Flow Law

- ❑ By definition of the average number of visits V_i, each completing request has to pass V_i times, on the average, by queue *i*. So, if X_o requests complete per unit of time, V_i*X_o requests will visit queue *i.*

$$X_i = V_i * X_o$$

Forced Flow Law: example

- ❑ Database transactions perform an average of 4.5 I/O operations on the database server. During a one-hour monitoring period, 7,200 transactions were executed.
- ❑ What is the average throughput of the disk?
- ❑ If each I/O takes 20 msec on the average, what is the disk utilization?

Forced Flow Law: example

- ❑ Database transactions perform an average of 4.5 I/O operations on the database server. During a one-hour monitoring period, 7,200 transactions were executed.
- ❑ What is the average throughput of the disk?
- ❑ If each I/O takes 20 msec on the average, what is the disk utilization?

$$X_{server} = 7{,}200 / 3{,}600 = 2 \text{ tps}$$

$$X_{disk} = V_{disk} * X_{server} = 4.5 * 2 = 9 \text{ tps}$$

$$U_{disk} = X_{disk} * S_{disk} = 9 * 0.02 = 0.18 = 18\%$$

Basic Performance Results

Service Demand Law

❑ The service demand D_i is related to the system throughput and utilization by the following:

$$D_i = V_i * S_i = (X_i/X_o)(U_i/X_i) = U_i / X_o$$

Example of Service Demand Law: vmstat

in	sy	cs	us	sy	idle
119	65	24	1	0	99
296	2491	289	13	6	81
260	5586	213	44	7	49
326	2822	474	21	7	72
352	1913	271	13	4	83
304	2058	280	17	5	78
275	3072	506	21	7	72
322	3340	417	18	8	74
301	2000	201	9	3	87
261	1952	282	10	4	86
251	1870	220	9	4	87
412	4646	763	33	12	54
					76.83

Interval:
12*5sec= 60 sec

Number of Requests:
20

$$U_{cpu} = 1 - 0.7683 = 0.232 = 23.2\%$$

$$X_0 = 20 / 60 = 0.333 \text{ requests/sec}$$

$$D_{cpu} = \frac{U_{cpu}}{X_0} = 0.232 / 0.333 = 0.695 \text{sec}$$

Service Demand Law: example

- ❑ A Web server running on top of a Unix system was monitored for 10 minutes. It was observed that the CPU was 90% busy during the monitoring period. The number of HTTP requests counted in the log was 30,000.
- ❑ What is the CPU service demand of an HTTP request?

Service Demand Law: example

- ❑ A Web server running on top of a Unix system was monitored for 10 minutes. It was observed that the CPU was 90% busy during the monitoring period. The number of HTTP requests counted in the log was 30,000.
- ❑ What is the CPU service demand of an HTTP request?

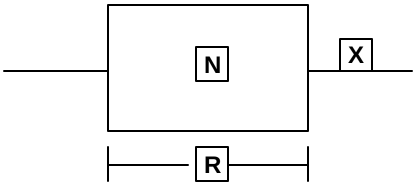
$$U_{cpu} = 90\%$$

$$X_{server} = 30,000 / (10*60) = 50 \text{ requests/sec}$$

$$D_{cpu} = V_{cpu} * S_{cpu} = U_{cpu} / X_{server} = 0.90 / 50 = 0.018 \text{ sec}$$

Basic Performance Results

Little’s Law

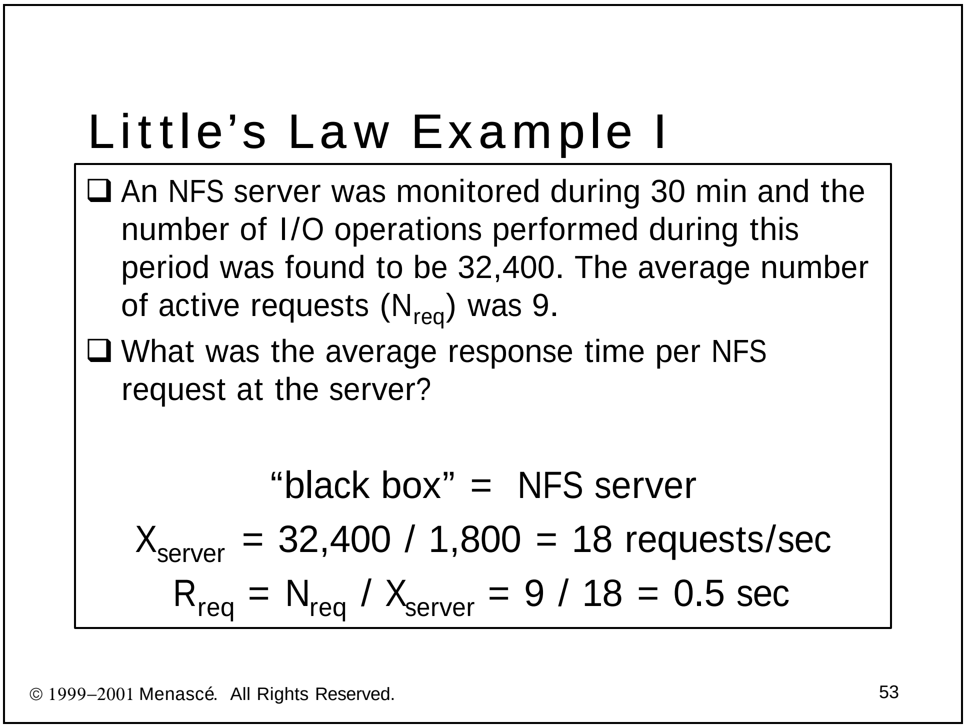


- The average number of customers in a “black box” is equal to average time each customer spends in the “box” times the throughput of the “box”.

$$N = R * X$$

Little’s Law Example I

- An NFS server was monitored during 30 min and the number of I/O operations performed during this period was found to be 32,400. The average number of active requests (N_{req}) was 9.
- What was the average response time per NFS request at the server?



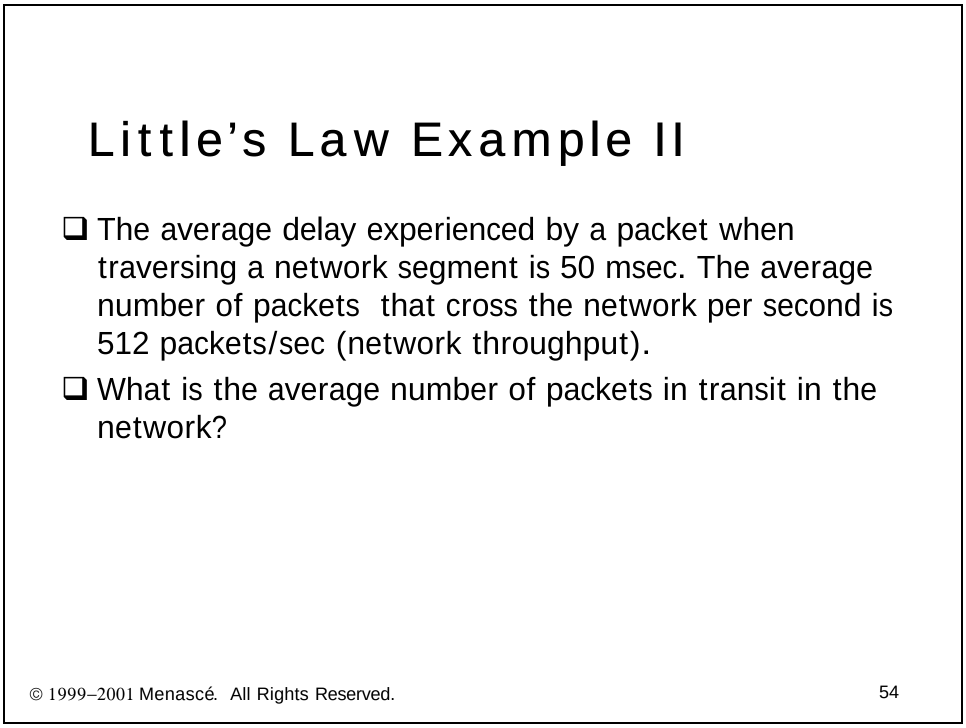
Little’s Law Example I

- ❑ An NFS server was monitored during 30 min and the number of I/O operations performed during this period was found to be 32,400. The average number of active requests (N_{req}) was 9.
- ❑ What was the average response time per NFS request at the server?

“black box” = NFS server

$$X_{server} = 32{,}400 \; / \; 1{,}800 = 18 \text{ requests/sec}$$

$$R_{req} = N_{req} \; / \; X_{server} = 9 \; / \; 18 = 0.5 \text{ sec}$$



Little’s Law Example II

- ❑ The average delay experienced by a packet when traversing a network segment is 50 msec. The average number of packets that cross the network per second is 512 packets/sec (network throughput).
- ❑ What is the average number of packets in transit in the network?

Little’s Law Example II

- ❑ The average delay experienced by a packet when traversing a network segment is 50 msec. The average number of packets that cross the network per second is 512 packets/sec (network throughput).
- ❑ What is the average number of packets in transit in the network?

“black box” = network segment

$$N_{packets} = R_{packet} * X_{network}$$

$$N_{packets} = 0.05 * 512 = 25.6 \text{ packets}$$

Little’s Law Example III

- ❑ The disk of a Web server receives requests at a rate of 20 requests/sec. The average disk service time, considering both random and sequential requests, is 8.02 msec.
- ❑ What is the average disk utilization?

Little’s Law Example III

- ❑ The disk of a Web server receives requests at a rate of 20 requests/sec. The average disk service time, considering both random and sequential requests, is 8.02 msec.
- ❑ What is the average disk utilization?

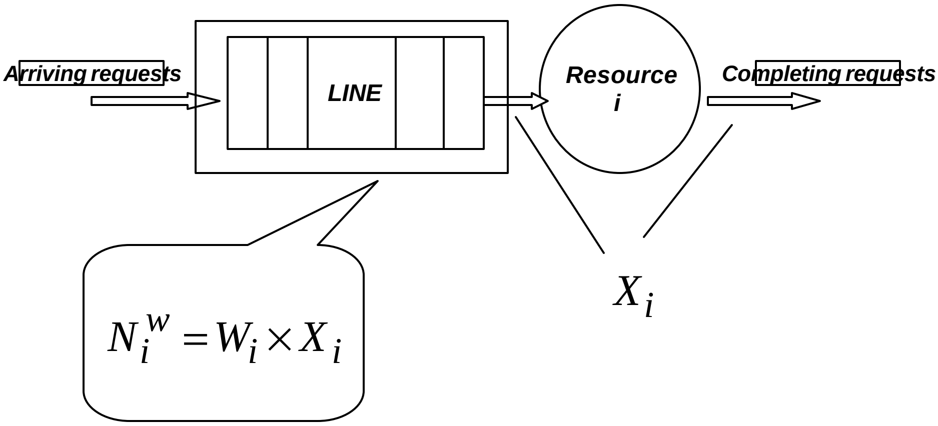
“black box” = disk

$\lambda_{disk} = X_{disk} = 20$ requests/sec

$S_{request} = 0.00802$ sec

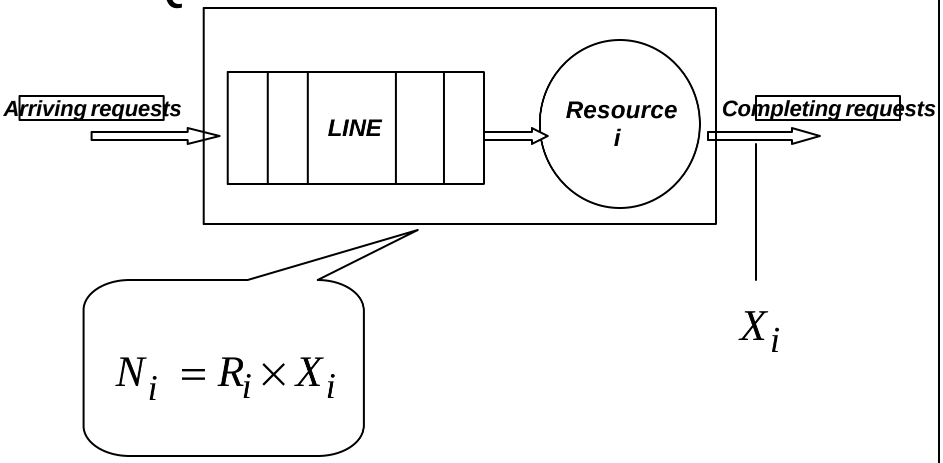
$U_{disk} = S_{request} * X_{disk} = 0.00802 * 20 = 16.04\%$

Applying Little’s Law to the Waiting Line



$$N_i^w = W_i \times X_i$$

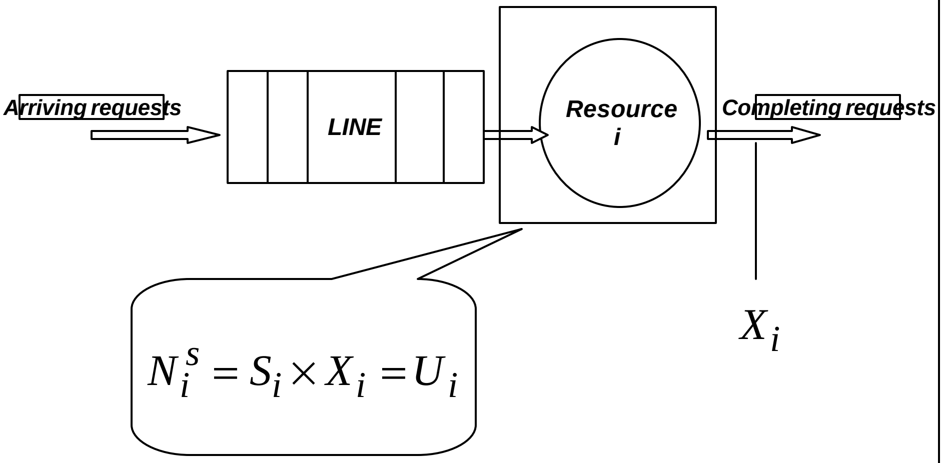
Applying Little’s Law to the Queue



$$N_i = R_i \times X_i$$

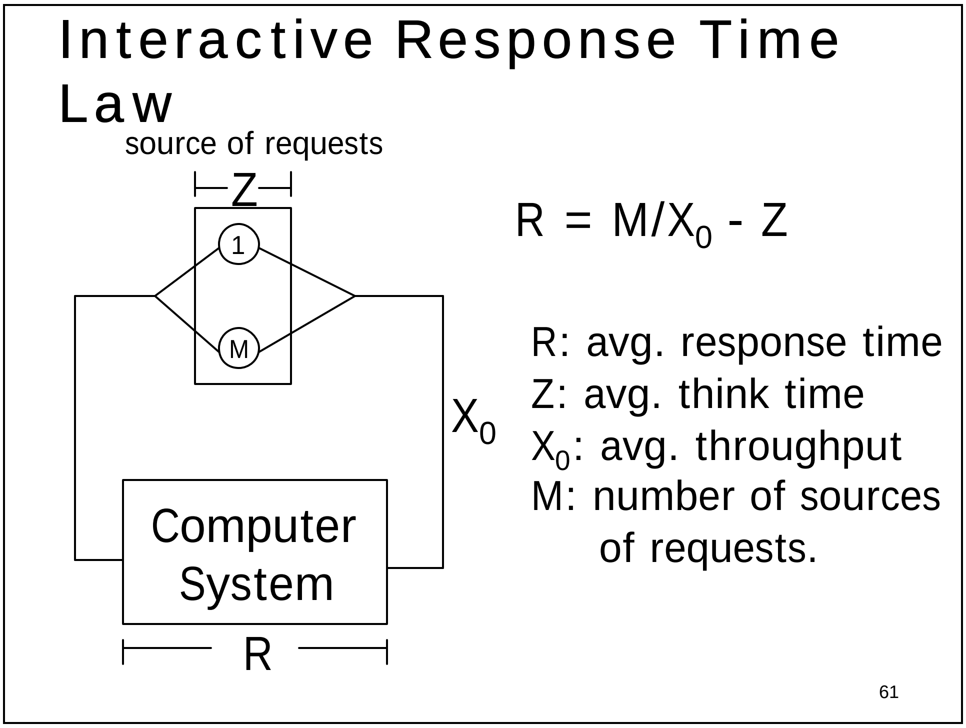
X_i

Applying Little’s Law to the Server



$$N_i^s = S_i \times X_i = U_i$$

X_i

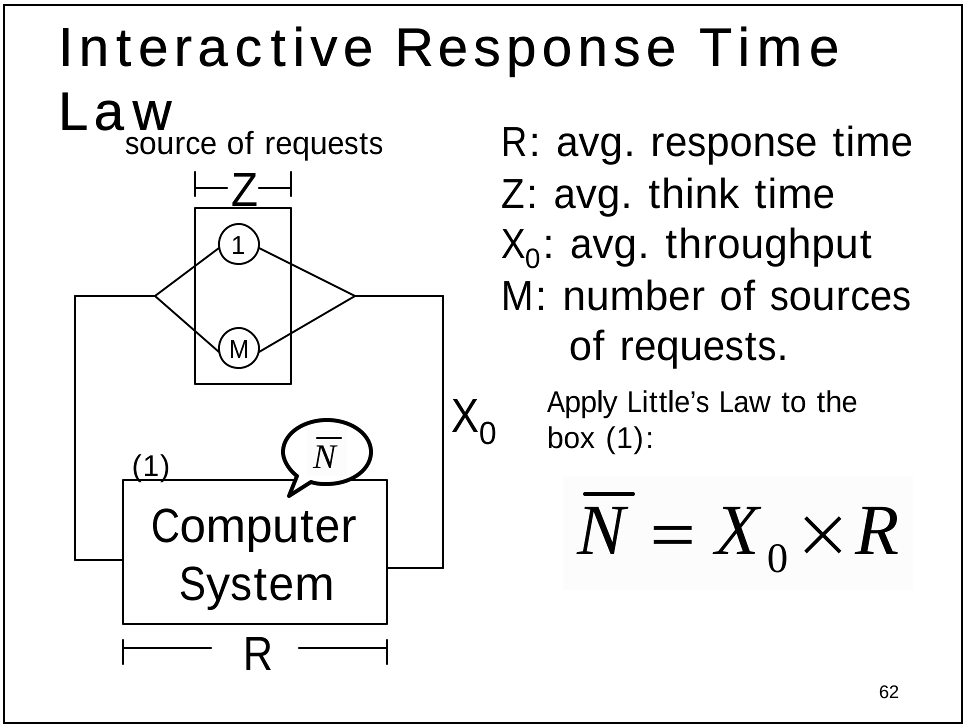


Interactive Response Time Law

$$R = M/X_0 - Z$$

R: avg. response time
Z: avg. think time
X_0: avg. throughput
M: number of sources of requests.

61



Interactive Response Time Law

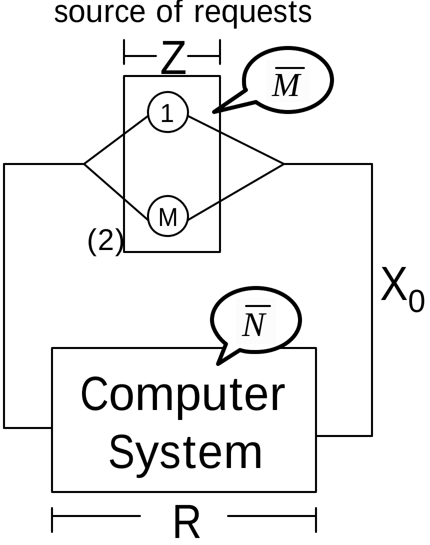
R: avg. response time
Z: avg. think time
X_0: avg. throughput
M: number of sources of requests.

Apply Little's Law to the box (1):

$$\overline{N} = X_0 \times R$$

62

Interactive Response Time Law



R: avg. response time

Z: avg. think time

X_0: avg. throughput

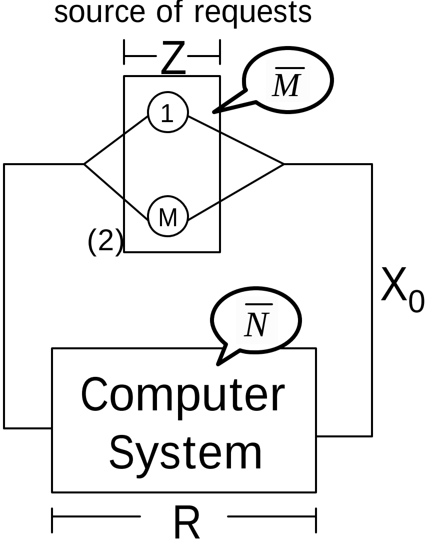
M: number of sources of requests.

Apply Little’s Law to box (2):

$$\overline{M} = X_0 \times Z$$

63

Interactive Response Time Law



R: avg. response time

Z: avg. think time

X_0: avg. throughput

M: number of sources of requests.

Combining the results:

$$\overline{N} = X_0 \times R$$

$$\overline{M} = X_0 \times Z$$

$$\overline{N} + \overline{M} = M = X_0(R+Z)$$

$$\Rightarrow \boxed{R = \frac{M}{X_0} - Z}$$

64

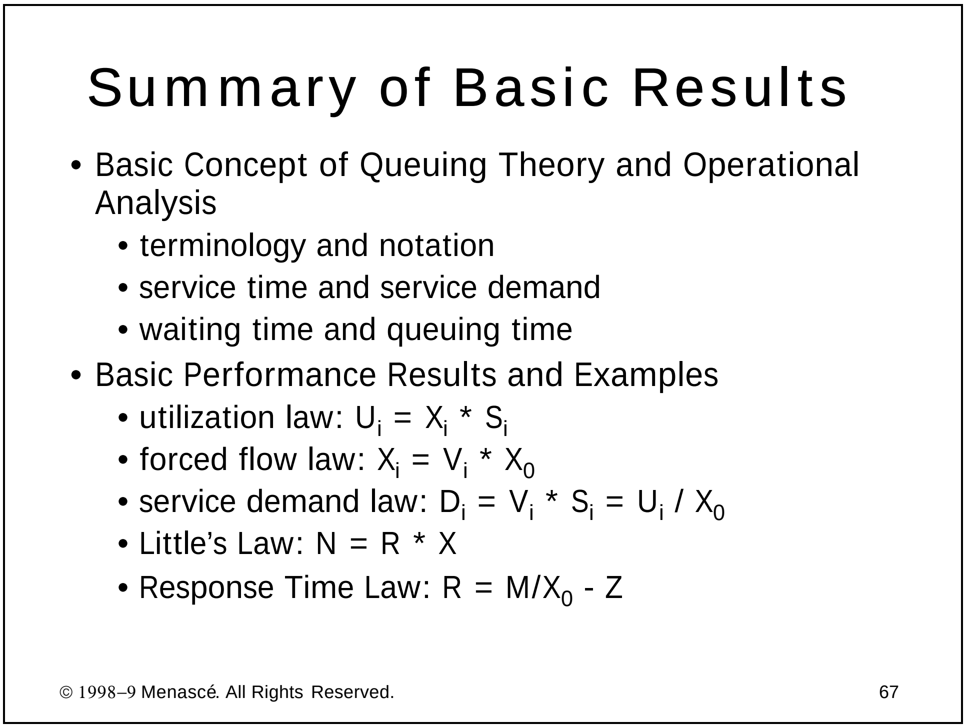
Response Time Law Example

- ❑A database server is capable of processing 20 requests/sec. The average think time is 15 sec. What is the maximum number of client machines that can be supported so that the average response time does not exceed 2 seconds?

Response Time Law Example

❑A database server is capable of processing 20 requests/sec. The average think time is 15 sec. What is the maximum number of client machines that can be supported so that the average response time does not exceed 2 seconds?

- ❑Z = 15 sec, X_0 = 20 req/sec. So,
- ❑M = (R + 15) * 20 ○ (2 + 15) * 20 = 340



Summary of Basic Results

- Basic Concept of Queuing Theory and Operational Analysis
  - terminology and notation
  - service time and service demand
  - waiting time and queuing time
- Basic Performance Results and Examples
  - utilization law: $U_i = X_i * S_i$
  - forced flow law: $X_i = V_i * X_0$
  - service demand law: $D_i = V_i * S_i = U_i / X_0$
  - Little’s Law: $N = R * X$
  - Response Time Law: $R = M/X_0 - Z$