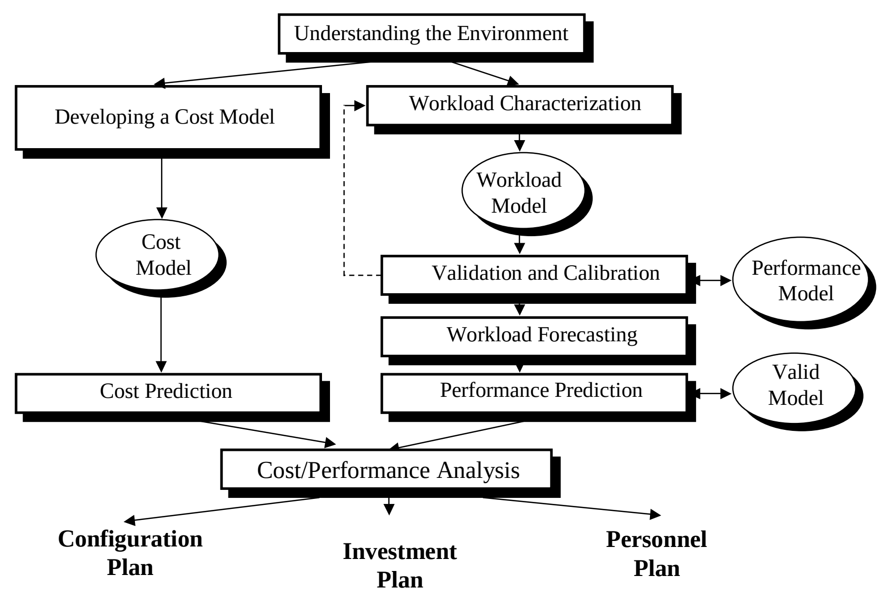
CS 672
System Level Performance Models of Computer Systems

Dr. Daniel A. Menascé
http://www.cs.gmu.edu/faculty/menasce.html
Department of Computer Science
George Mason University



Part V: Learning Objectives

Characterize system-level models

Present State Transition Diagram (STD) technique

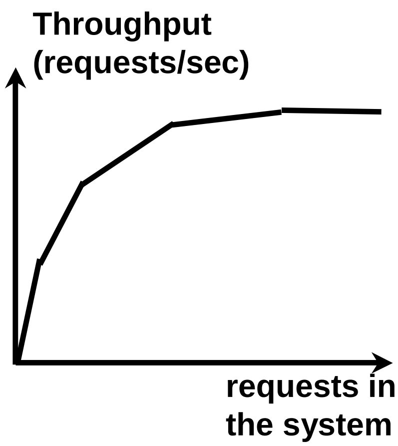
Show general solution to STDs

Show how to obtain performance metrics from the solution of STDs

3

System-level Models

- System is seen as a black box.
- Only its input-output characteristics are considered.
- Inputs: arrivals of requests
- Output: throughput.



4

System-level Example

- A Web server receives 10 requests/sec.
- The maximum number of requests in the server is 3.
- Requests that arrive and find three requests being processed are rejected.

5

System-level Example

- The measured throughput as a function of the number of requests is:

Number of requests	Throughput (req/sec)
0	0
1	12
2	15
3	16

6

System-level Example: a few questions

- Q1: What is the probability that an incoming request is rejected?
- Q2: What is the average number of requests in execution?
- Q3: What is the average throughput of the Web server?
- Q4: What is the average time spent by an HTTP request in the Web server?

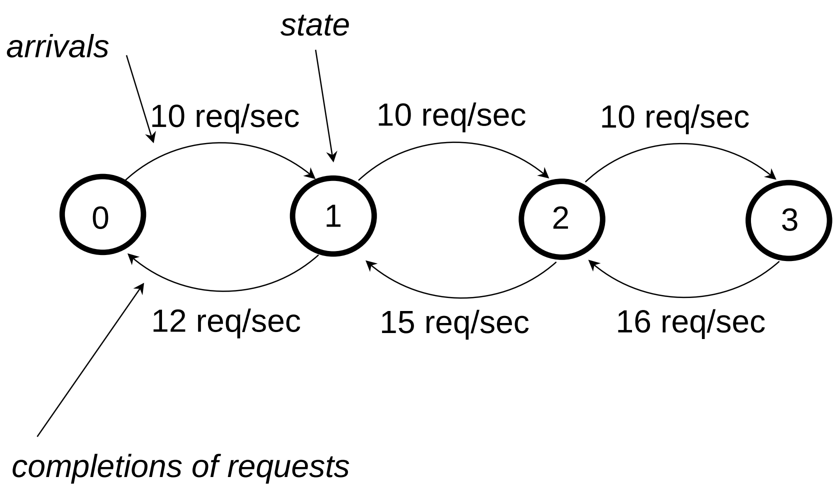
7

System-level Example

- Characterize the Web server by its state, i.e., the number *k* of requests in the Web server.
- Assumptions made:
  - homogeneous workload: all requests are equivalent
  - memoryless: how the system arrived at system *k* does not matter.
  - operational equilibrium: no. requests at beginning of interval = no. request at the end.

8

System-level Example



9

System-level Example

- Assume we are able to find the values of:
  - P_k = probability that there are *k* requests in the Web server.
- Question: can we answer all the questions posed before as a function of the P_k's?

10

System-level Example: a few questions

- Q1: What is the probability that an incoming request is rejected?
- A:

11

System-level Example: a few questions

- Q1: What is the probability that an incoming request is rejected?
- A: It is the probability that an arriving HTTP request finds 3 requests already being processed. The answer is then P_3.

12

System-level Example: a few questions

- Q2: What is the average number of requests in execution?
- A: using the definition of average:

13

System-level Example: a few questions

- Q2: What is the average number of requests in execution?
- A: using the definition of average:

$n_{\mathbf{req}} = 0 \text{ x } P_{\mathbf{0}} + 1 \text{ x } P_{\mathbf{1}} + 2 \text{ x } P_{\mathbf{2}} + 3 \text{ x } P_{\mathbf{3}}$

14

System-level Example: a few questions

- Q3: What is the average throughput of the Web server?
- A: again, using the definition of average:

15

System-level Example: a few questions

- Q3: What is the average throughput of the Web server?
- A: again, using the definition of average:

$$X = 0 \times P_0 + 12 \times P_1 + 15 \times P_2 + 16 \times P_3$$

throughput value at each state

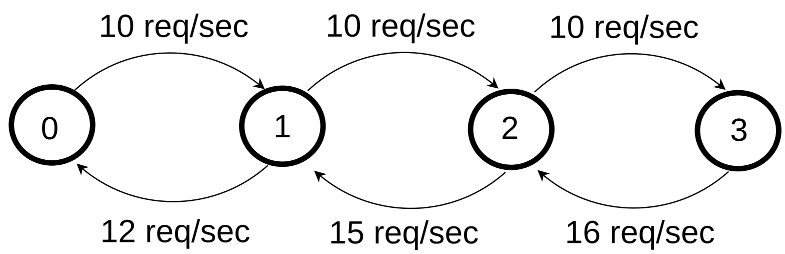
16

System-level Example: a few questions

- Q4: What is the average time spent by an HTTP request in the Web server?
- A: It will be a function of the average number of requests, $n_{\mathbf{req}}$, and the average throughput X. More on this later...

17

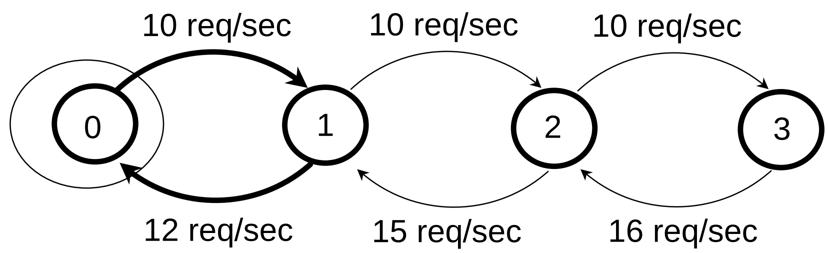
System-level Example: computing the $P_{\mathbf{k}}$'s



- use the flow in = flow out principle: the flow into a set of states is equal to the flow out of this set of states in equilibrium.

18

System-level Example: computing the P_k's

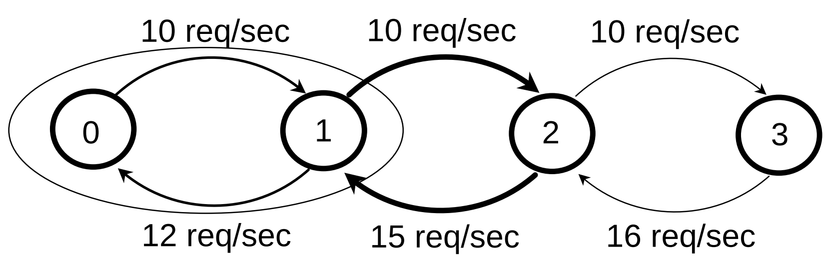


flow in = flow out

$$12 \times P_1 = 10 \times P_0$$

19

System-level Example: computing the P_k's

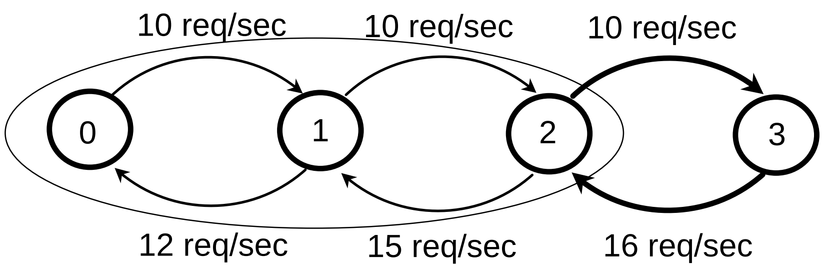


flow in = flow out

$$15 \times P_2 = 10 \times P_1$$

20

System-level Example: computing the P_k's



flow in = flow out

$$16 \times P_3 = 10 \times P_2$$

System-level Example: computing the P_k's

- Putting it all together:

$$12 \times P_1 = 10 \times P_0 \Rightarrow P_1 = 10/12\ P_0$$

$$15 \times P_2 = 10 \times P_1 \Rightarrow P_2 = 10/15\ P_1 = \frac{10 \times 10}{15 \times 12} P_0$$

$$16 \times P_3 = 10 \times P_2 \Rightarrow P_3 = 10/16\ P_2 = \frac{10 \times 10 \times 10}{16 \times 15 \times 12} P_0$$

System-level Example: computing the P_k's

- Putting it all together:

$P_1 = 10/12\, P_0$; $P_2 = \frac{10 \times 10}{15 \times 12} P_0$; and

$P_3 = \frac{10 \times 10 \times 10}{16 \times 15 \times 12} P_0$

- But, the Web server has to be in one of the four states at any time. So, $P_0 + P_1 + P_2 + P_3 = 1$.

23

System-level Example: computing the P_k's

- Solving for P_0 and then for the other P_k's we get:

k	Pk
0	0.365
1	0.305
2	0.203
3	0.127

24

System-level Example: answering the questions

- Q1: What is the probability that an incoming request is rejected?
- A: It is the probability that an arriving HTTP request finds 3 requests already being processed. The answer is then

  $P_3 = 0.127 = 12.7\%$.

25

System-level Example: answering the questions

- Q2: What is the average number of requests in execution?
- A: using the definition of average:

$$n_{req} = 0 \text{ x } 0.365 + 1 \text{ x } 0.305 + 2 \text{ x } 0.203 + 3 \text{ x } 0.127 = 1.091 \text{ requests}$$

26

System-level Example: answering the questions

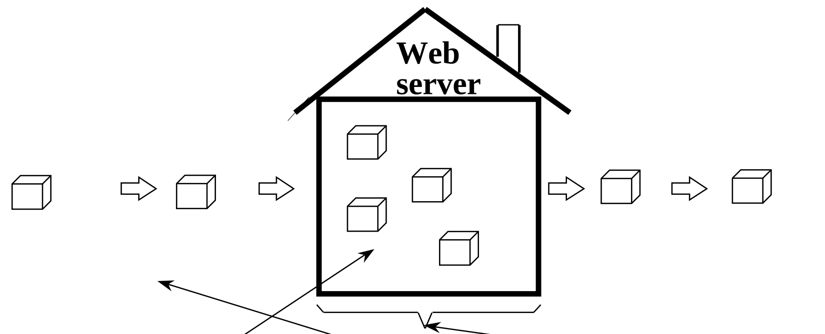
- Q3: What is the average throughput of the Web server?
- A: again, using the definition of average:

$$X = 0 \times 0.365 + 12 \times 0.305 + 15 \times 0.203 + 16 \times 0.127 = 8.731 \text{ requests/sec.}$$

System-level Example: answering the questions

- Q4: What is the average time spent by an HTTP request in the Web server?
- A: It is a function of the average number of requests, $n_{\mathbf{req}}$, and the average throughput X. We need Little's Law to answer this question.

Little’s Law



avg. number requests in the server = **avg. departure rate from the server** X **avg. time spent at the server**

1.091 req = 8.731 req/sec X ?

System-level Example: answering the questions

- Q4: What is the average time spent by an HTTP request in the Web server?
- A: From Little’s Law,

$$R = n_{\mathbf{req}} / X = 1.091 / 8.731 = 0.125 \text{ sec.}$$

Practice Drill

Using Models for Decision Making

- What happens if the maximum number of allowed TCP connections changes from 3 to 10?
- What if the load on the server doubles?
- What is the impact of a threefold increase in the server's capacity?

Types of System-level Models

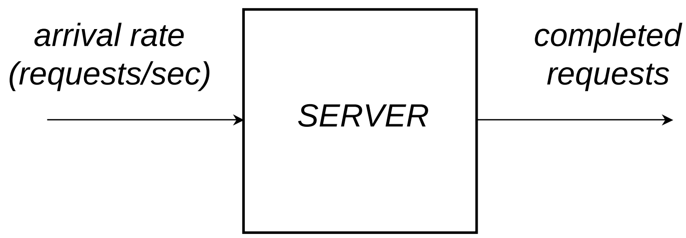
- Population Size:
  - infinite
  - finite
- Service Rate:
  - fixed
  - variable
- Maximum Queue Size:
  - unlimited
  - limited

Types of System-level Models (population size)

- Infinite Population: the number of clients is very large. The rate at which requests arrive to the system does not depend on the number of requests in the system.
  - e.g., requests arriving from the Internet to a public Web server.

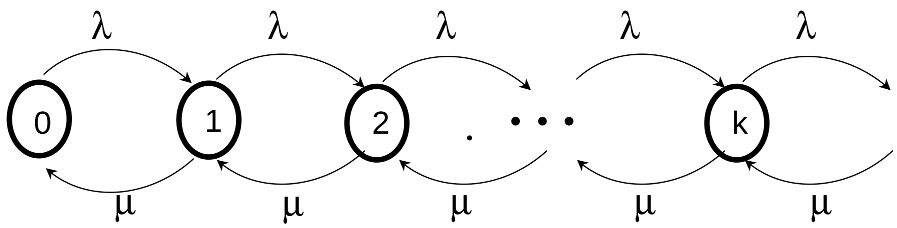
33

Types of System-level Models (infinite population)



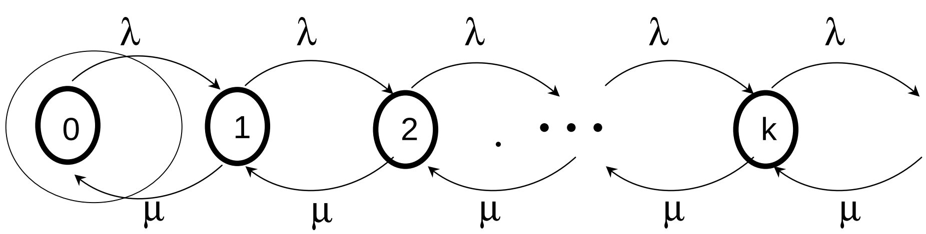
34

Infinite Population/Infinite Queue



flow in = flow out

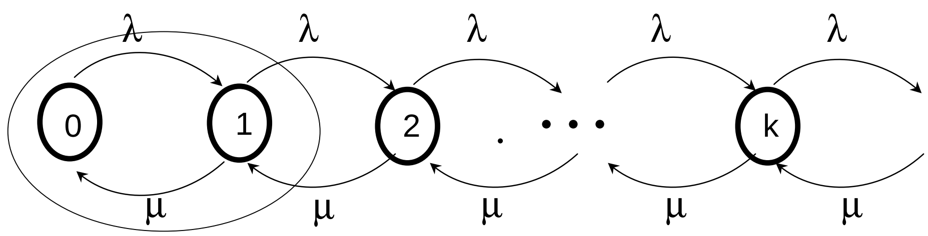
Infinite Population/Infinite Queue



flow out = flow in

$\lambda P_0 = \mu P_1$

Infinite Population/Infinite Queue

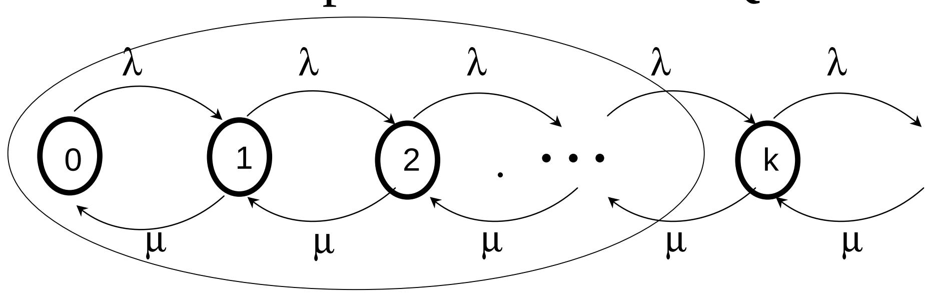


flow out = flow in

$\lambda P_0 = \mu P_1$

$\lambda P_1 = \mu P_2$

Infinite Population/Infinite Queue



flow out = flow in

$\lambda P_o = \mu P_1$

$\lambda P_1 = \mu P_2$

. . .

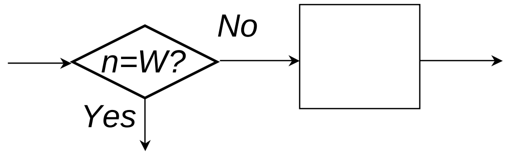
$\lambda\ _{Pk-1} = \mu P_k$

Infinite Population/Infinite Queue Example

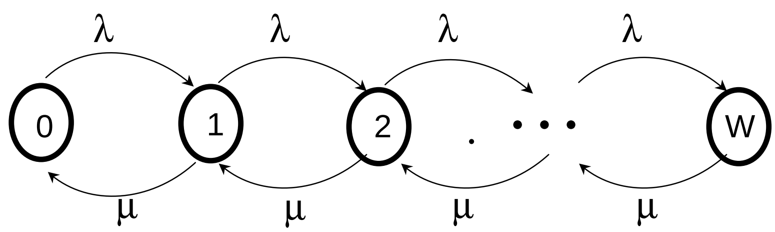
- A DB server receives 30 req/sec. Each request takes 0.02 sec on the average. Find:
  - Fraction of requests in the DB server?
  - Average response time.
  - Average response time for a server twice as fast.
  - Average response time for a server twice as fast for twice the arrival rate.

Types of System-level Models (maximum queue size)

- Unlimited Queue Size: all arriving requests are queued for service. No requests are rejected!
- Limited Queue Size: requests that find more than W requests waiting for service are rejected.



Infinite Population/Finite Queue



• arriving requests that find the server in state W are lost.

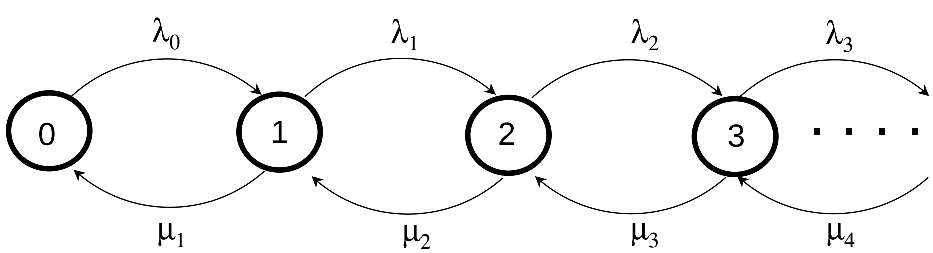
Infinite Population/Finite Queue Example

- A DB server receives 30 req/sec. Each request takes 0.02 sec on the average. At most 4 request can be queued. Find:
  – Fraction of requests in the DB server?
  – Average response time.
  – Average response time for a server twice as fast.
  – Average response time for a server twice as fast for twice the arrival rate.

Infinite Population/Finite Queue Example (cont'd)

- What is the maximum value for the maxium number of request queued so that less then 1% of the requests are rejected?

Generalized System-level Models



Generalized System-level Models can be solved using the ***flow in = flow out*** *principle!*

Generalized System-level Models

$$p_k = \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}}$$

$$p_0 = \left[\sum_{k=0}^{\infty} \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}} \right]^{-1}$$

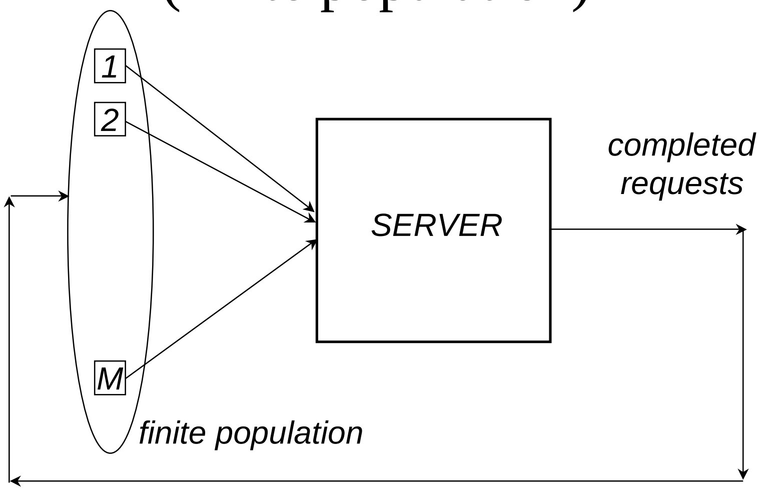
45

Types of System-level Models (population size)

- Finite Population: the number of clients is limited. The rate at which requests arrive to the system depends on how many have already arrived.
  - e.g., requests arriving to an intranet Web server from a known number of clients within the organization.

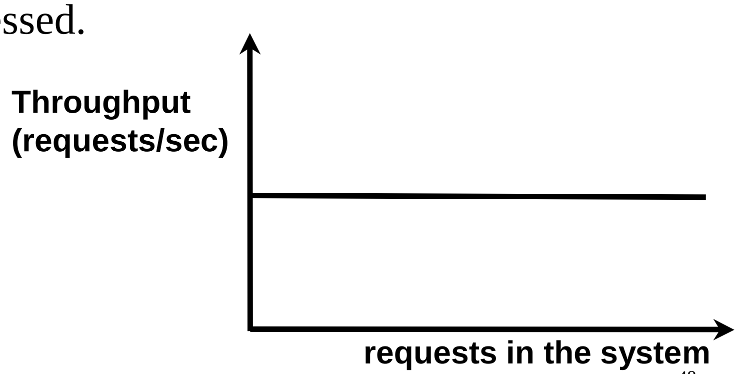
46

Types of System-level Models (finite population)



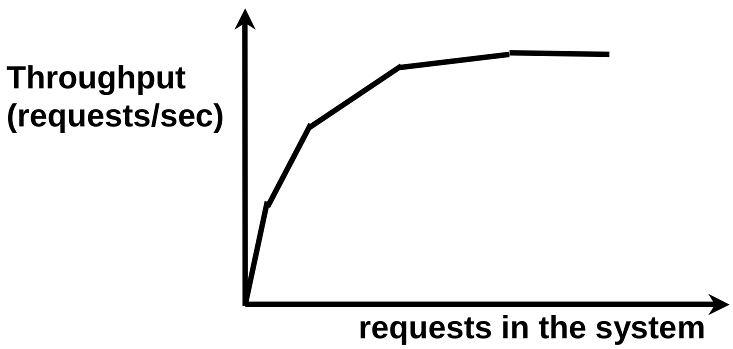
Types of System-level Models (service rate)

- Fixed Service Rate: the throughput does not vary with the number of requests being processed.



Types of System-level Models (service rate)

- Variable Service Rate: the throughput depends on the number of requests being processed.



Types of System-level Models

Population	Service Rate	Queue Size
infinite	fixed	unlimited
infinite	fixed	limited
infinite	variable	unlimited
infinite	variable	finite
finite	fixed	
finite	variable	

System-level Models Example

A Web server receives 30 requests/sec. Its throughput function is given below. The server queue is limited to five requests. What is the server utilization, avg. throughput, avg. no. requests, avg. response time, and fraction of lost requests?

No. of requests	Throughput (req/sec)
1	18
2	35
3 or more	50

System-level Models Example (cont'd)

Using the Generalized System-level model equations we get that

Average Number of Requests	1.850
Server Utilization	82.700%
Server Average Throughput	28.4 req/sec
Fraction of Lost Requests	0.05343

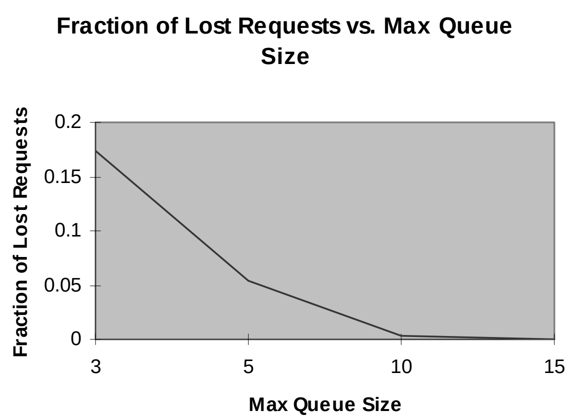
From Little's Law,
avg. response time = avg. no requests/
avg. throughput =
1.85 / 28.4 = 0.065 sec.

System-level Models Example (cont'd)

Max. Queue Size	3	5	10	15
Average Number of Requests	1.433	1.850	2.212	2.264
Server Utilization	79.81%	82.7%	83.9%	83.96%
Avg. Server Throughput (req/sec)	24.8	28.4	29.9	30
Average Response Time (sec)	0.058	0.065	0.074	0.075
Fraction of Lost Requests	0.173077	0.05343	0.003869	0.000299

53

System-level Models Example (cont'd)



54

Summary

System-level models view a server as a black box. Only its arrival process and throughput functions are relevant.

State Transition Diagrams (STDs) can be used to find the probability that *k* requests are in the server. Use the *flow in = flow out* principle.

Little's Law can be used to compute the response time from the average number of requests and from the throughput.

55