

## Tema 5: Análisis operacional en servidores

¿Cómo modelar el rendimiento y predecir la capacidad de mi servidor?

Analistas, administradores y diseñadores



## Objetivos del tema

- Proporcionar un modelo analítico de comportamiento de un sistema informático como punto de partida para obtener índices de rendimiento.
- Entender la importancia de los cuellos de botella como limitadores del rendimiento de los sistemas informáticos.
- Saber aplicar las leyes operacionales en ejemplos sencillos para obtener índices de rendimiento.
- Saber interpretar los límites optimistas del rendimiento que establece el análisis operacional.
- Saber evaluar de forma cuantitativa el efecto de diferentes terapias de mejora o estrategias de diseño sobre el rendimiento de un servidor.

2

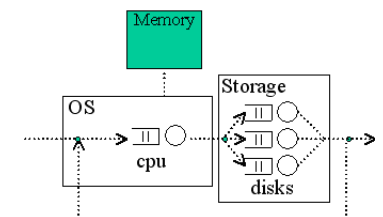
## Bibliografía

- *Evaluación y modelado del rendimiento de los sistemas informáticos*. X. Molero, C. Juiz, M. Rodeño. Pearson Educación, 2004. Capítulos 4 y 5.
- *The art of computer system performance analysis*. R. Jain. John Wiley & Sons, 1991. Capítulos 30, 32, 33 y 34.
- *Measuring computer performance: a practitioner's guide*. David J. Lilja, Cambridge University Press, 2000. Capítulo 11.

3

## Contenido

- Introducción: Redes de colas de espera.
- Variables y leyes operacionales.
- Límites optimistas del rendimiento.
- Técnicas de mejora del rendimiento.
- Algoritmos de resolución de redes de colas.



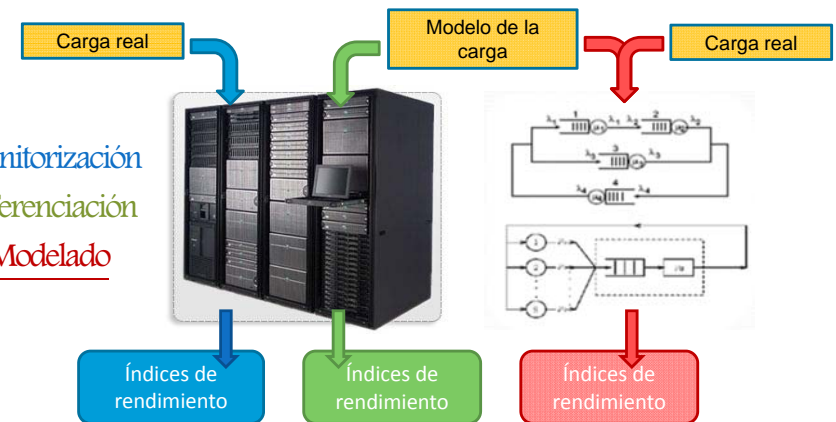
4

## 5.1. Introducción: Redes de Colas de Espera

### Como decíamos en el tema 1...

- ¿Cómo podemos evaluar/analizar/estudiar el rendimiento de un servidor?

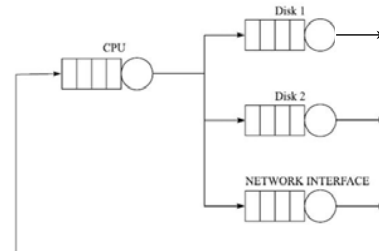
- Monitorización
- Referenciación
- Modelado



6

## El modelo de un sistema informático

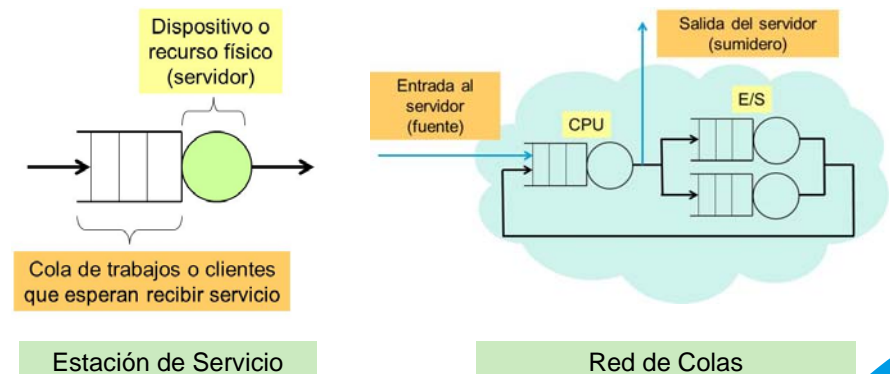
- Abstracción del sistema informático real.
  - Conjunto de dispositivos interrelacionados y trabajos que los usan (carga).
    - Dispositivos (*resources*): procesador, discos, memoria, red, etc.
    - Trabajos (*jobs*): procesos, peticiones, etc.
  - Normalmente un dispositivo o recurso solo puede ser usado por un trabajo a la vez. El resto de trabajos habrá de esperar.
- Modelos basados en redes de colas (*queueing networks*).
  - Introducidos en la década de 1950.
  - Objetivo básico: cálculo del tiempo de respuesta que experimenta un trabajo procesado por el sistema.
  - Aproximación estadística.
- Otros modelos: redes de Petri, cadenas de Markov, ...



7

## Red de colas: concepto

- Conjunto de estaciones de servicio conectadas entre sí.
- Estación de servicio (*service station*): Objeto abstracto compuesto por un dispositivo (recurso físico) que presta un servicio y una cola de espera para los trabajos (clientes) que demandan un servicio de él.



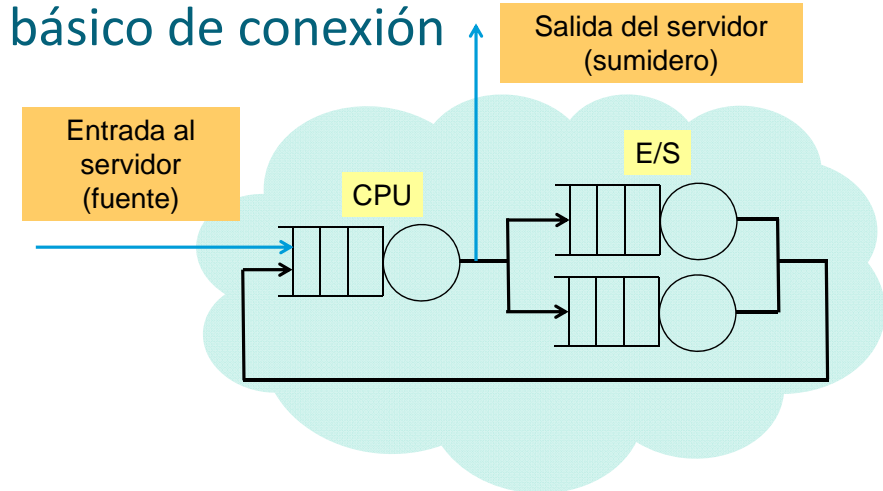
8

## El modelo de servidor central

- Es el modelo de redes de colas que más se ha utilizado para representar el comportamiento de los programas en un servidor de cara a extraer información sobre su rendimiento.
- ¿Cuál es este comportamiento?
  - Un trabajo que “llega” al servidor comienza utilizando el procesador.
  - Después de “abandonar” el procesador, el trabajo puede:
    - terminar (sale del servidor), o bien
    - realizar un acceso a una unidad de entrada/salida (discos, red,...).
  - Después de una operación con una unidad de entrada/salida, el trabajo vuelve a “visitar” al procesador.
- Recursos considerados
  - Procesador o procesadores.
  - Entrada/salida: unidades de disco, red, etc.

9

## Modelo de servidor central: Diagrama básico de conexión



- Propiedad de este modelo: Cada trabajo que abandona el servidor (=proceso ejecutado) ha “visitado” la CPU tantas veces como la suma de sus “visitas” a cada uno los dispositivo de E/S más 1.

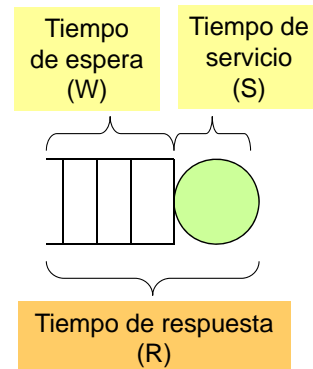
10

## Variables temporales de un trabajo en una estación de servicio

- Tiempo de espera en cola ( $W$ , *waiting time*)
  - Tiempo transcurrido desde que un trabajo solicita hacer uso del recurso físico hasta que realmente empieza a utilizarlo.
- Tiempo de servicio ( $S$ , *service time*)
  - Tiempo transcurrido desde que un trabajo hace uso del recurso físico hasta que lo libera.
- Tiempo de respuesta de la estación de servicio ( $R$ , *response time*)
  - Suma de los dos tiempos anteriores.

$$R = W + S$$

- En general, se trata de magnitudes estadísticas (distribuciones de probabilidad).



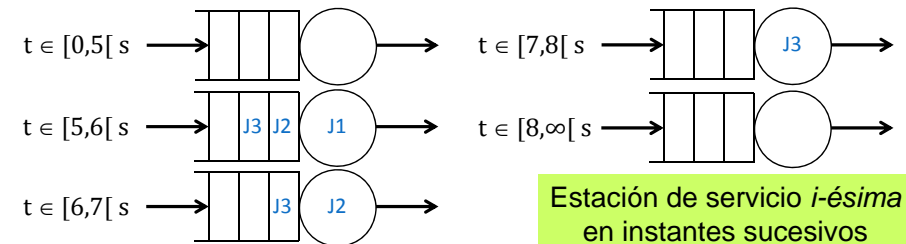
11

## Ejercicio

Suponga que la estación de servicio  $i$ -ésima de una red de colas tiene un tiempo de servicio constante  $S_i = 1s$ . Suponga que los trabajos (*jobs*) llegan con la siguiente distribución temporal:

- Durante los primeros 5 segundos no llega ningún trabajo.
- En  $t=5s$  llegan 3 trabajos:  $J_1$ ,  $J_2$  y  $J_3$  (por ese orden).

Calcule los tiempos de espera en la cola y los tiempos de respuesta que experimentan **cada uno** de los tres trabajos. Calcule finalmente los valores medios de  $W$  y  $R$ .



$$W_i(J_1) = 0s$$

$$R_i(J_1) = 1s$$

$$W_i(J_2) = 1s$$

$$R_i(J_2) = 2s$$

$$W_i(J_3) = 2s$$

$$R_i(J_3) = 3s$$

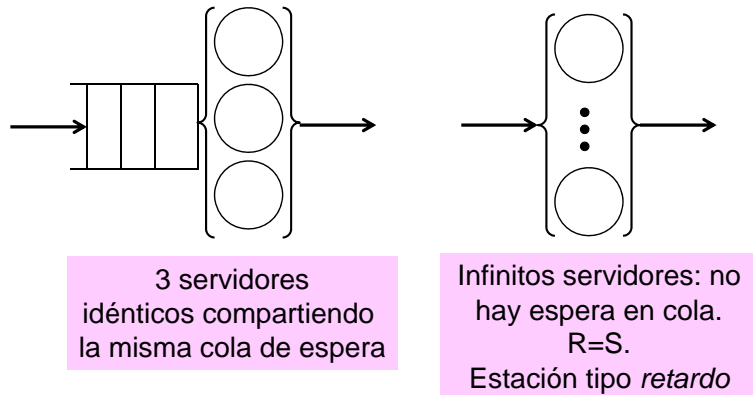
$$\text{Valor medio } W_i = 1s$$

$$\text{Valor medio } R_i = 2s$$

12

## Estaciones con más de un servidor

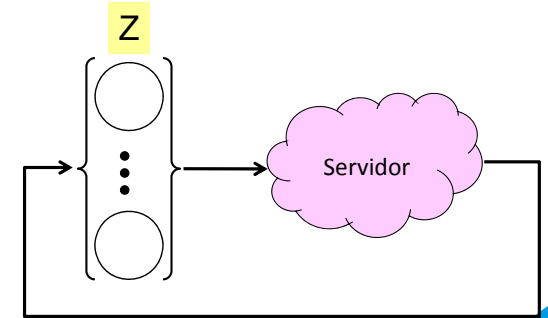
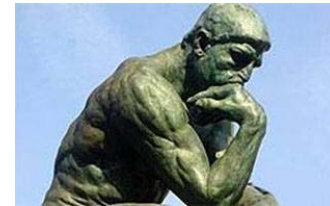
- Son capaces de atender a más de un trabajo en paralelo:



13

## El tiempo de reflexión ( $Z$ , *think time*)

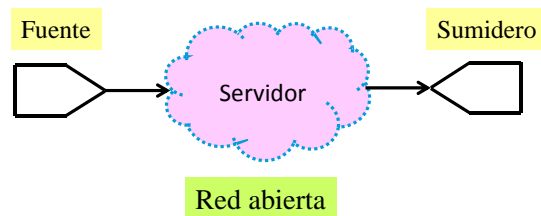
- Es un parámetro ( $Z$ ) que representa el tiempo que requiere el usuario antes de volver a lanzar una petición al servidor.
- Se suele modelar mediante una estación de servicio tipo retardo con un tiempo de servicio =  $Z$ .



14

## Redes de colas abiertas

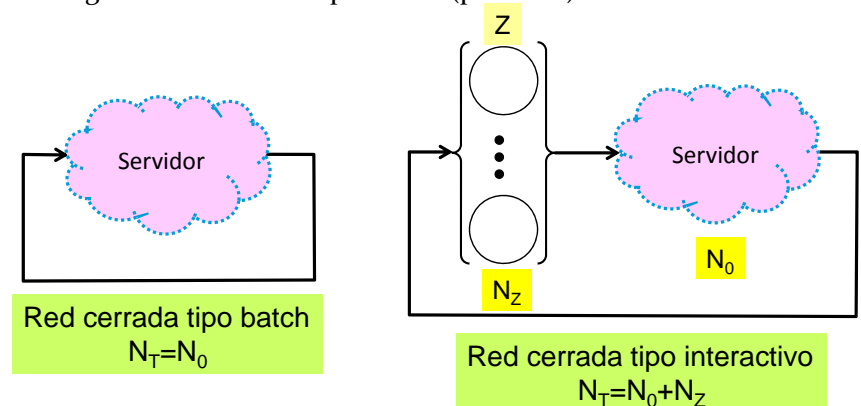
- Los trabajos llegan a la red a través de una fuente externa que no controlamos. Tras ser procesados, salen de ella a través de uno o más sumideros.
  - El número de trabajos en el servidor ( $N_0$ ) puede variar con el tiempo.
  - Normalmente, se conoce la tasa de llegada de trabajos al servidor ( $\lambda_0$ ).



15

## Redes de colas cerradas

- Presentan un número constante de trabajos que van recirculando por la red ( $N_T$ ). Dependiendo de si hay o no interacción con usuarios se distingue entre redes de tipo **batch** (por lotes) o redes **interactivas**.



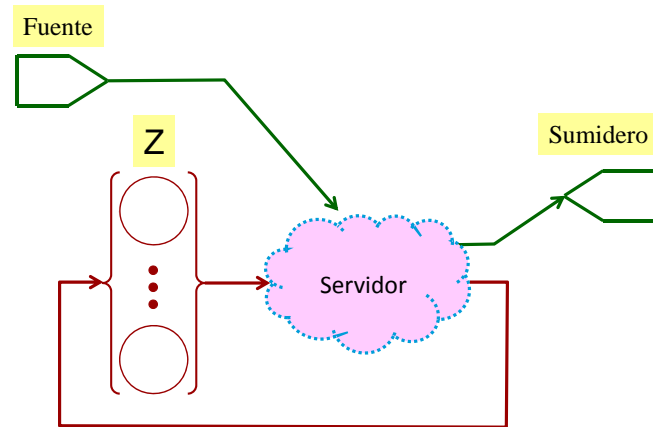
$N_0$  = Número de trabajos en el servidor.

$N_Z$  = Número de trabajos en reflexión (esperando a que los usuarios vuelvan a introducirlos en el servidor). **Siempre supondremos 1 usuario = 1 trabajo.**

16

## Redes mixtas

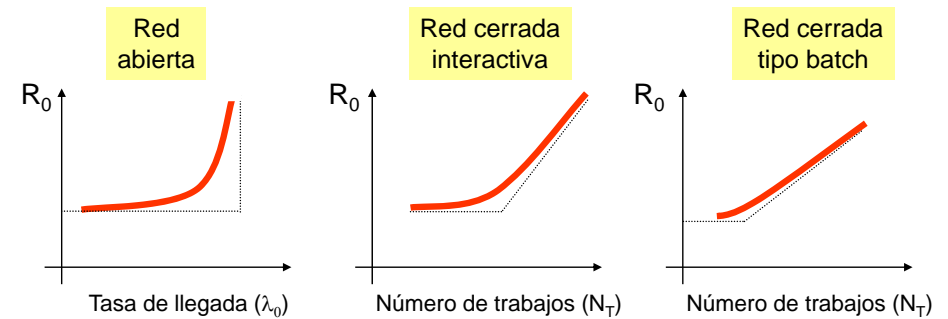
- Cuando el modelo no corresponde a ninguno de los dos anteriores.



17

## Evolución del tiempo medio de respuesta del servidor ( $R_0$ ) según el tipo de red

- Se mide desde que el trabajo entra al servidor hasta que lo abandona.



18

## 5.2. Variables y leyes operacionales

### El análisis operacional

- Técnica de análisis de redes de colas presentada por Denning y Buzen en 1978.
- Basada en valores medios de *variables operacionales* (=magnitudes medibles) del sistema informático.

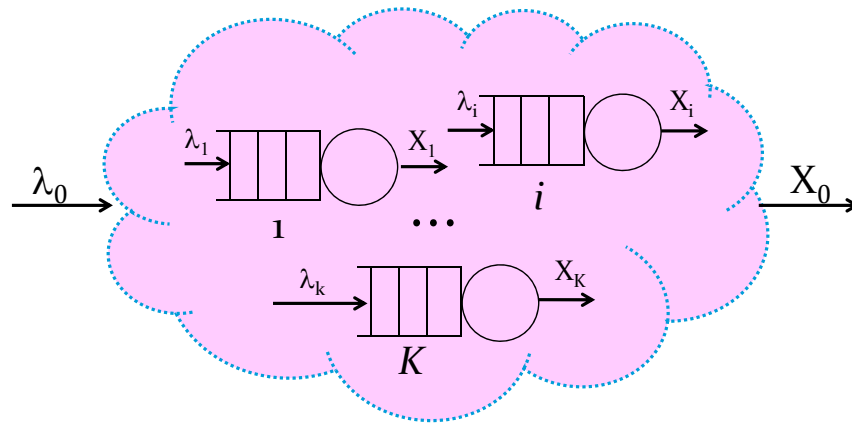


- Leyes operacionales: relaciones entre las variables operacionales.
- Límites optimistas de las prestaciones del servidor por medio de cálculos muy sencillos.

20

## Variables: servidor vs. estación de servicio

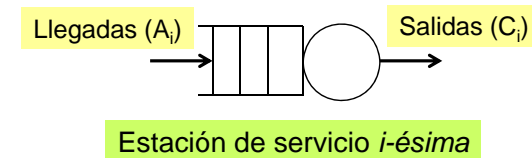
- El servidor contiene  $K$  estaciones de servicio (recursos o dispositivos).
- A todo el servidor en su globalidad lo denotamos como dispositivo “cero”.



21

## Variables operacionales básicas

- Variable global temporal:
  - $T$  Duración del periodo de medida para el que se extrae el modelo.
- Variables operacionales básicas de la estación de servicio  $i$ -ésima medidas durante el tiempo  $T$ :
  - $A_i$  Número de trabajos solicitados a la estación (llegadas, *arrivals*).
  - $B_i$  Tiempo que el dispositivo está ocupado (*busy time*).
  - $C_i$  Número de trabajos completados por la estación (salidas, *completions*).

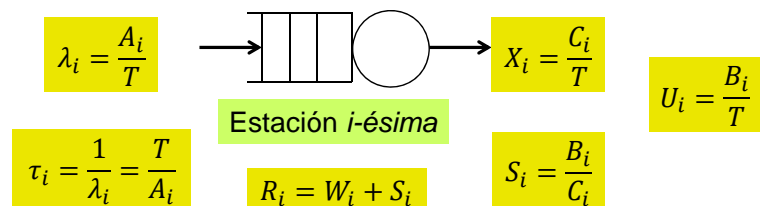


22

## Variables operacionales deducidas

- Se deben poder estimar a partir de las variables básicas:
 

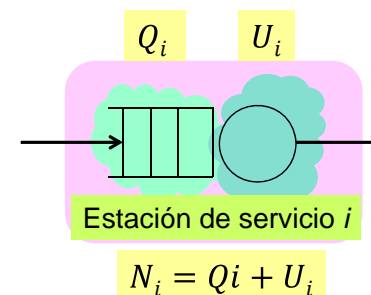
• $\lambda_i$ Tasa media de llegada ( <i>arrival rate</i> )	trabajos/segundo
• $\tau_i$ Tiempo medio entre llegadas ( <i>interarrival time</i> )	segundos [/trabajo]
• $X_i$ Productividad media ( <i>throughput</i> )	trabajos/segundo
• $S_i$ Tiempo medio de servicio ( <i>service time</i> )	segundos [/trabajo]
• $W_i$ Tiempo medio de espera en cola ( <i>waiting time</i> )	segundos [/trabajo]
• $R_i$ Tiempo medio de respuesta ( <i>response time</i> )	segundos [/trabajo]
• $U_i$ Utilización ( <i>utilization</i> )	sin unidades



23

## Variables operacionales deducidas (II)

- $N_i$ : Número medio de trabajos en la estación de servicio (cola más recurso).
- $Q_i$ : Número medio de trabajos en cola de espera (*jobs in queue*).
- $U_i$ : Número medio de trabajos siendo servidos por el dispositivo,  $U_i = N_i - Q_i$ . Coincide numéricamente con la **utilización** = proporción de tiempo que el dispositivo ha estado en uso (*busy*) con respecto al intervalo total de medida ( $T$ ) (como máximo 1 si  $B_i = T$ ).



$$U_i = \frac{B_i}{T}$$



24

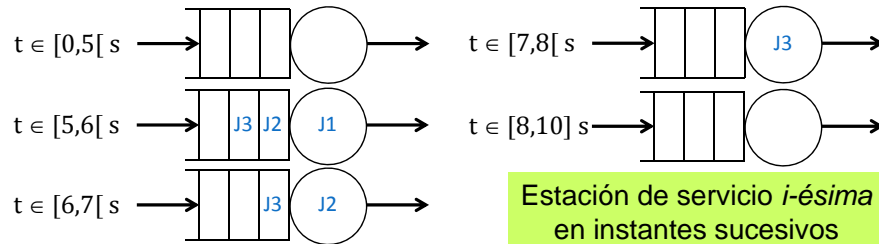


## Ejercicio

Suponga que la estación de servicio  $i$ -ésima de una red de colas tiene un tiempo de servicio constante  $S_i = 1s$ . Suponga que los trabajos llegan con la siguiente distribución temporal:

- Durante los primeros 5 segundos no llega ningún trabajo.
- En  $t=5s$  llegan 3 trabajos: J1, J2 y J3 (por ese orden).

Para el intervalo de medida  $[0, 10]s$ , calcule  $A_i$ ,  $B_i$ ,  $C_i$ ,  $\lambda_i$ ,  $\tau_i$ ,  $X_i$ ,  $Q_i$ ,  $N_i$ ,  $U_i$ .

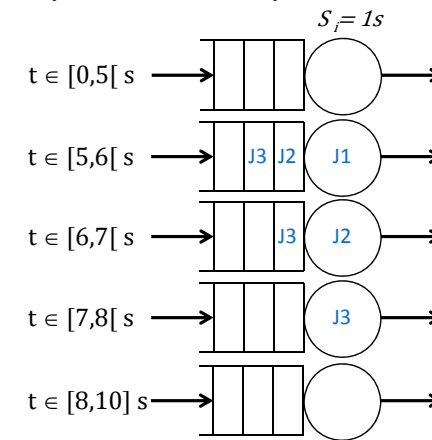


$A_i = 3$  trabajos,  $B_i = 3s$ ,  $C_i = 3$  trabajos,  $\lambda_i = X_i = 3/10 = 0.3$  trabajos/s,  $\tau_i = 3.3$  s.

25

## Ejercicio (cont.)

Cálculo de la utilización media ( $U_i$ ) y del número medio de trabajos en la cola ( $Q_i$ ) y en la estación ( $N_i$ ).



$$Q_i = \frac{0 \times 5s + 2 \times 1s + 1 \times 1s + 0 \times 3s}{10s} = 0.3 \text{ tr.}$$

$$N_i = \frac{0 \times 5s + 3 \times 1s + 2 \times 1s + 1 \times 1s}{10s} = 0.6 \text{ tr.}$$

Formas alternativas de calcular  $U_i$ :

$$U_i = N_i - Q_i = 0.3 \text{ tr.}$$

$$U_i = \frac{0 \times 5s + 1 \times 3s + 0 \times 2s}{10s} = 0.3 \text{ tr.}$$

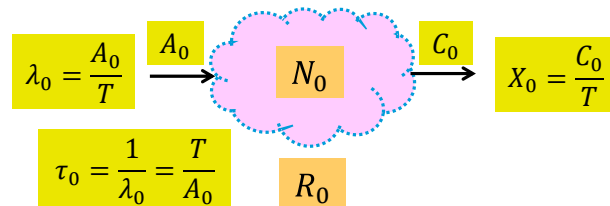
$$U_i = \frac{B_i}{T} = \frac{3}{10} = 0.3$$

Estación de servicio  $i$ -ésima en instantes sucesivos

26

## Variables globales del servidor

- Variables básicas
  - $A_0$  Número de trabajos solicitados al servidor (*arrivals*).
  - $C_0$  Número de trabajos completados por el servidor (*completions*).
- Variables deducidas
  - $\lambda_0$  Tasa media de llegada al servidor (*arrival rate*).
  - $\tau_0$  Tiempo medio entre llegadas al servidor (*interarrival time*).
  - $X_0$  Productividad media del servidor (*throughput*).
  - $R_0$  Tiempo medio de respuesta del servidor (*response time*).
  - $N_0$  Número medio de trabajos en el servidor (*#jobs*) =  $N_1 + N_2 + \dots + N_K$ .



27

## Razón de visita y demanda de servicio

- Razón de visita  $V_i$  (*visit ratio*). Representa la proporción entre el número de trabajos completados por el servidor y el número de trabajos completados por la estación de servicio  $i$ -ésima (dispositivo). Es como si el trabajo tuviera que “visitar” el dispositivo  $i$ -ésimo una media de  $V_i$  veces antes de poder abandonar el servidor.

$$V_i = \frac{C_i}{C_0}$$

- Demanda de servicio  $D_i$  (*service demand*). Cantidad de tiempo que, por término medio, el dispositivo  $i$ -ésimo dedica a cada trabajo que abandona el servidor.

$$D_i = \frac{B_i}{C_0} = V_i \times S_i$$

Nótese que la demanda de servicio no tiene en cuenta la posible espera de un trabajo en su cola.

28

## Ejercicio

Después de monitorizar el **disco duro** de un servidor web durante un periodo de **30 segundos**, se sabe que ha estado en funcionamiento un total de **27 segundos**. Asimismo, se han contabilizado durante ese periodo un total de **74** peticiones de lectura/escritura al disco duro y un total de **72** peticiones completadas. Se ha estimado que cada consulta atendida por el servidor web ha requerido una media de 4 accesos de E/S al disco duro. Calcule:

- ¿Cuál es la tasa media de llegada al disco duro?
- ¿Cuál es la productividad media del disco duro?
- Determine la utilización del disco duro, su tiempo de servicio y su demanda de servicio.
- ¿Cuál es la productividad media del servidor web?

29

## Leyes operacionales

- Casi todas las variables utilizadas en el análisis operacional son **valores medios** para el intervalo de monitorización T. Por tanto, el valor de dichas variables dependerá del intervalo de observación T.
- Existen, sin embargo, una serie de relaciones entre algunas variables operacionales que se mantienen válidas para cualquier intervalo de observación y que no dependen de suposiciones sobre la distribución de los tiempos de servicio o de la forma en la que llegan los trabajos. Estas relaciones se denominan **leyes operacionales**.
- Estas leyes son tanto más útiles cuando se cumple la denominada **hipótesis del equilibrio de flujo** que establece que en un servidor **no saturado**, si se escoge un intervalo de observación suficientemente largo, se cumple que:
  - El número de trabajos que llegan al servidor coincide aproximadamente con el que sale ( $A_o \approx C_o$ ). Dicho de otra manera, la tasa media de llegada coincide aproximadamente con la productividad media ( $\lambda_o \approx X_o$ ).
  - El número de trabajos que llegan a cada estación de servicio coincide aproximadamente con el que sale: ( $A_i \approx C_i \Rightarrow \lambda_i \approx X_i, \forall i=1...K$ ).

30

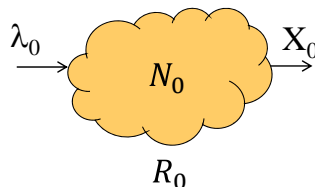
## Ley de Little

*"The long-term average number of customers in a stable system is equal to the long-term average arrival rate multiplied by the average time a customer spends in the system"*

- Aplicada a un servidor, relaciona el número medio de trabajos en el servidor ( $N_o$ ) con el tiempo medio de respuesta ( $R_o$ ) y su tasa de llegada ( $\lambda_o$ ).

$$N_o = \lambda_o \times R_o$$

- Bajo la hipótesis del equilibrio de flujo (servidor no saturado), esta ley relaciona las dos variables más importantes que reflejan el rendimiento de un servidor: su productividad ( $X_o$ ) y su tiempo de respuesta ( $R_o$ ).



$$N_o = \lambda_o \times R_o = X_o \times R_o$$

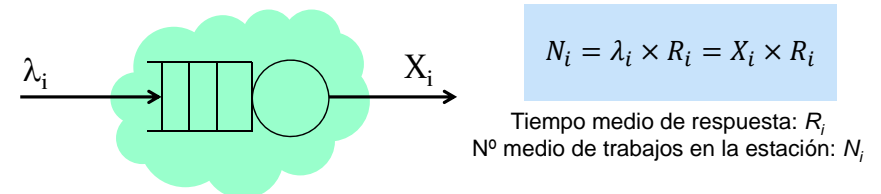
**Ley de Little aplicada a un servidor no saturado**

31

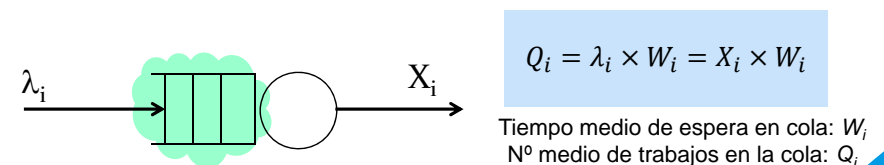
## Ley de Little (cont.)

La ley de Little puede ser aplicada no solo al servidor en su totalidad, sino a cada estación de servicio y a cada uno de los diferentes sub-niveles de una estación de servicio.

- Aplicación a toda una estación de servicio:



- Aplicación a la cola de una estación de servicio:



32

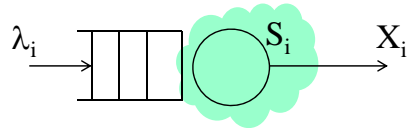


## Ley de la Utilización

- Relaciona la utilización de un dispositivo con el número de trabajos que es capaz de realizar por unidad de tiempo (=su productividad) y el tiempo medio que le dedica a cada uno de ellos (=su tiempo de servicio):

$$U_i = \frac{B_i}{T} = \frac{C_i}{T} \times \frac{B_i}{C_i} = X_i \times S_i \quad \text{Ley de la Utilización}$$

- Para un sistema en equilibrio de flujo (servidor no saturado), se podía haber deducido también la ley de la utilización sin más que aplicar la ley de Little al recurso físico (dispositivo) de una estación:



$$U_i = \lambda_i \times S_i = X_i \times S_i$$

**Ley de la Utilización**  
(servidor no saturado)

33

## Ejemplo de aplicación

- Como consecuencia de unas medidas sobre el disco duro principal de un servidor de base de datos **no saturado**, sabemos que el tiempo medio de respuesta del disco es 48ms, su tiempo medio de servicio es 30 ms y su productividad 25 peticiones de E/S completadas por segundo. Calcule el tiempo medio de espera en la cola del disco duro así como el número medio de trabajos en su cola de espera.

- Solución:

- Calcule el tiempo medio de espera en cola del disco duro:

$$W_{disco} = R_{disco} - S_{disco} = 48 - 30 = 18 \text{ ms}$$

- Calcule el número medio de trabajos en su cola de espera:

$$Q_{disco} = N_{disco} - U_{disco} = 1.2 - 0.75 = 0.45 \text{ trabajos}$$

$$\text{(ley de Little): } N_{disco} = X_{disco} \times R_{disco} = 25 \text{ tr/s} \times 0.048 \text{ s} = 1.2 \text{ trabajos}$$

$$\text{(ley de la Utilización): } U_{disco} = X_{disco} \times S_{disco} = 25 \text{ tr/s} \times 0.03 \text{ s} = 0.75 \text{ trabajos (utilización 75\%)}$$

- Pregunta: ¿Por qué el tiempo de respuesta (48 ms) es mucho mayor que el tiempo de servicio (30 ms) si la utilización del disco no llega al 100%?

34

## Ley del flujo forzado

- Las productividades (flujos) a diferentes niveles del servidor tienen que ser proporcionales a la productividad global del servidor.
  - La ley del flujo forzado relaciona la productividad del servidor con la de cada uno de los dispositivos que integran el mismo:

$$V_i = \frac{C_i}{C_0} = \frac{X_i}{X_0}$$



$$X_i = X_0 \times V_i$$

**Ley del Flujo Forzado**

- Como consecuencia de la ley del flujo forzado, las utilizaciones de cada dispositivo son proporcionales a las demandas de servicio del mismo, siendo la constante de proporcionalidad precisamente la productividad global del servidor:

$$U_i = X_i \times S_i = X_0 \times V_i \times S_i = X_0 \times D_i$$

**Relación Utilización-Demanda de Servicio**

35

## Ejemplo de aplicación II

- Un ingeniero informático ha monitorizado un servidor de ayuda a la toma de decisiones durante 24 horas, contabilizando un total de 14400 consultas al servidor. Durante ese tiempo, el monitor *sar* le ha indicado que el procesador ha ejecutado una media de 40 procesos por minuto y que su utilización ha sido del 40%. Suponiendo que el servidor no está saturado, calcule la razón de visita, la demanda de servicio y el tiempo medio de servicio del procesador.

- Solución:

Necesitamos calcular primero  $X_0 = \lambda_0$  (ya que no saturado) =  $14400 \text{ tr} / (24 \text{ hr} \times 60 \text{ min/hr}) = 10 \text{ tr/min}$

$$V_{CPU} = X_{CPU} / X_0 = \frac{40 \text{ tr/min}}{10 \text{ tr/min}} = 4$$

$$D_{CPU} = U_{CPU} / X_0 = \frac{0.4}{10 \text{ tr/min}} = 0.04 \text{ min/[tr]} = 2.4 \text{ s/[tr]}$$

$$S_{CPU} = D_{CPU} / V_{CPU} = 2.4 \text{ s} / 4 = 0.6 \text{ s}$$

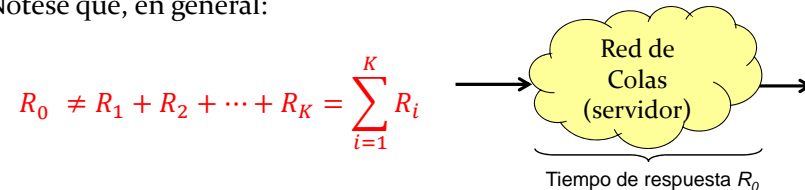
36

## Ley general del tiempo de respuesta

- Si el servidor no está saturado y tenemos en cuenta que el número medio de trabajos dentro de un servidor no es más que la suma del número medio de trabajos que hay en cada estación de servicio:  $N_0 = N_1 + N_2 + \dots + N_K$ , y que  $N_i = X_i \times R_i = X_0 \times V_i \times R_i$ , es fácil demostrar que:

$$R_0 = V_1 \times R_1 + V_2 \times R_2 + \dots + V_K \times R_K = \sum_{i=1}^K V_i \times R_i$$

expresión conocida como “Ley general del tiempo de respuesta”.  
Nótese que, en general:

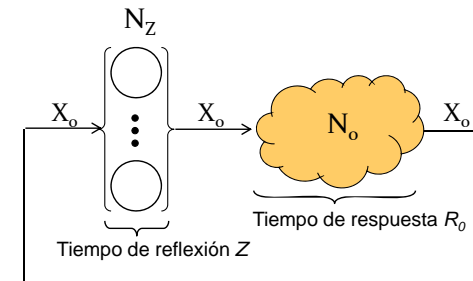


37

## Ley del tiempo de respuesta interactivo

- Se obtiene mediante la aplicación de la ley de Little a una red cerrada de tipo interactivo.
- Válida para servidores no saturados.

Recordemos:  $N_Z$  = Número medio de trabajos (=clientes) en reflexión.



$$N_Z = X_0 \times Z; \quad N_0 = X_0 \times R_0$$

$$N_T = N_Z + N_0 = X_0 \times Z + X_0 \times R_0 = X_0 \times (Z + R_0)$$

$$R_0 = \frac{N_T}{X_0} - Z$$

que se conoce como la *Ley del tiempo de respuesta interactivo*.

Nótese que al ser una red cerrada, el número total de trabajos (=clientes) en la red cerrada ( $N_T = N_Z + N_0$ ), es constante.

38

## Ejemplo de aplicación III

- Durante un tiempo T, se encuentran haciendo uso de un servidor de ficheros un total de 3000 usuarios, cada uno asociado a un único fichero (1 usuario = 1 fichero = 1 trabajo de nuestro modelo). Suponiendo que el tiempo medio de reflexión de cada usuario es de 20 segundos y que el tiempo medio de respuesta del servidor es de 10 segundos por cada fichero, ¿cuál es la productividad del servidor y cuántos trabajos se encuentran, de media, en reflexión?

$$X_0 = \frac{N_T}{R_0 + Z} = \frac{3000}{10 + 20} = 100 \text{ tr/s} \quad N_Z = X_0 \times Z = 2000 \text{ tr}$$

- Si se quiere conseguir una productividad de 125 trabajos por segundo, ¿qué tiempo de respuesta debería tener el servidor?

$$R_0 = \frac{N_T}{X_0} - Z = \frac{3000}{125} - 20 = 4 \text{ s}$$

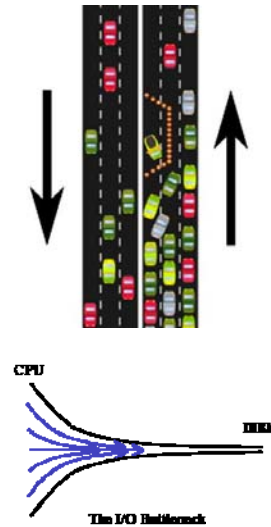
- ¿Qué habría que hacer para conseguir una productividad de 200 trabajos/s?

39

## 5.3. Límites optimistas del rendimiento

## Limitaciones en el rendimiento: cuello de botella

- Todo servidor presenta alguna limitación en su rendimiento.
- En esta sección veremos que la localización del elemento limitador no solo depende del servidor sino también de la carga.
  - Al elemento limitador del rendimiento del servidor se le denomina cuello de botella (*bottleneck*).
  - Además, puede haber más de uno de estos elementos limitadores.
- Veremos que la única manera de mejorar las prestaciones de un servidor de manera significativa es actuando sobre el cuello de botella.



41

## Identificación de un cuello de botella

- El cuello de botella es el dispositivo con **mayor utilización** ya que será el primero que llegue a saturarse ( $U_i=1$ ) cuando aumente la carga ( $\lambda_0$  mayor).

$$U_i = X_i \times S_i = X_0 \times V_i \times S_i = X_0 \times D_i = \lambda_0 \times D_i$$

- Podemos identificar fácilmente el cuello de botella de un servidor simplemente buscando el dispositivo con **mayor demanda de servicio** ( $U_i \propto D_i$ ).
- No hace falta llevar el servidor al límite para identificar el cuello de botella.
- Denotaremos por “b” al índice del dispositivo cuello de botella. Su demanda de servicio y su utilización vendrán dadas por.

$$D_b = \max_{i=1 \dots K} \{D_i\} = V_b \times S_b$$

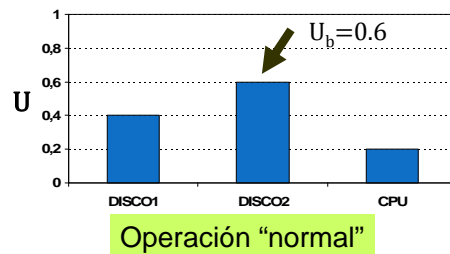
$$U_b = \max_{i=1 \dots K} \{U_i\} = X_0 \times D_b$$

42

## Saturación del servidor

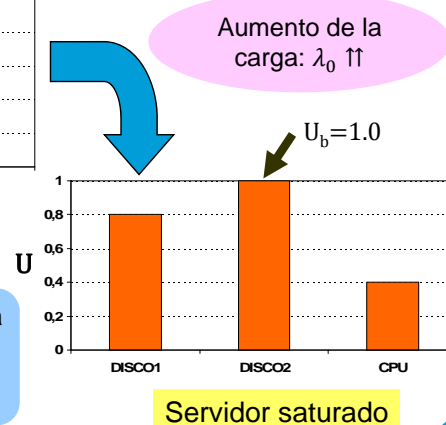
- El servidor se satura cuando lo hace el cuello de botella ya que, por definición, éste será el primero en alcanzar una utilización = 1 cuando aumente la carga.

$$U_b = X_0 \times D_b = \lambda_0 \times D_b$$



En saturación, el servidor alcanza su productividad máxima:

$$X_0^{max} = 1/D_b$$



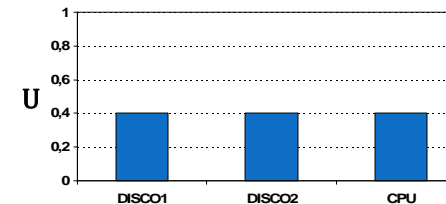
Servidor saturado

43

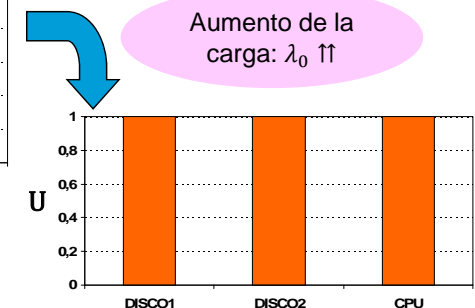
## Servidor equilibrado (*balanced system*)

- Servidor en que todos los dispositivos tienen la misma demanda de servicio o utilización (la carga se absorbe equitativamente):

$$U_i \approx U_j \quad \forall i, j = 1 \dots K \quad \rightarrow \quad D_i \approx D_j \quad \forall i, j = 1 \dots K$$



Todos los dispositivos son cuellos de botella



44

## Límites Optimistas del Rendimiento

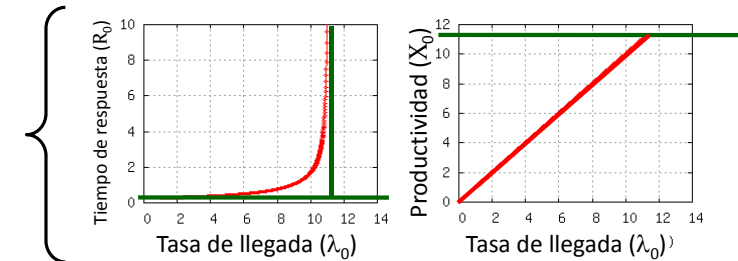
- Se trata de encontrar una cota superior de la productividad e inferior para el tiempo de respuesta del servidor. Son, por tanto, **límites optimistas** del rendimiento. En particular, sabremos:
  - ¿Cuál es la productividad máxima ( $X_0^{max}$ ) del servidor?
  - ¿Cuál es el tiempo de respuesta mínimo ( $R_0^{min}$ ) del servidor?
- Campos de aplicación:
  - Planificación de la capacidad (*capacity planning*).
  - Estimación de la mejora potencial de prestaciones que pueden reportar ciertas acciones sobre el servidor.



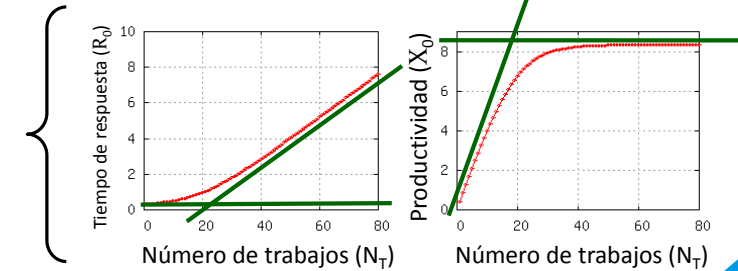
45

## Localización de los límites (asíntotas)

Redes abiertas



Redes cerradas



46

## Límites optimistas: redes abiertas

- El valor máximo de la tasa de llegada del servidor, será aquél que sature completamente el dispositivo cuello de botella ( $U_b=1$ ):

$$U_b = X_0 \times D_b = \lambda_0 \times D_b \quad \text{Si } U_b = 1 \Rightarrow X_0^{max} = \frac{1}{D_b}$$

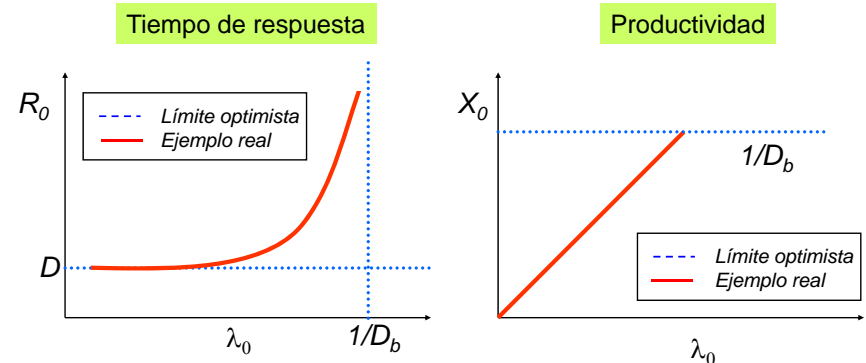
A partir de ese momento, una tasa de llegada mayor provocaría la saturación del servidor y, por tanto, dejaría de cumplirse la hipótesis del equilibrio de flujo ( $X_0$  ya no podría seguir a  $\lambda_0$ ).

- El valor más optimista (= el valor mínimo) del tiempo medio de respuesta del servidor ( $R_0^{min}$ ) será el que experimenta un trabajo cuando llega al servidor sin que haya otros trabajos previamente ( $W_i = 0 \rightarrow R_i^{min} = S_i \forall i=1..K$ ):

$$R_0^{min} = \sum_{i=1}^K V_i \times R_i^{min} = \sum_{i=1}^K V_i \times S_i = \sum_{i=1}^K D_i \equiv D$$

47

## Resumen: Límites optimistas en redes abiertas



$$R_0^{min} = D \equiv \sum_{i=1}^K D_i$$

$$X_0^{max} = \frac{1}{D_b}$$

48

## Ejemplo de red abierta

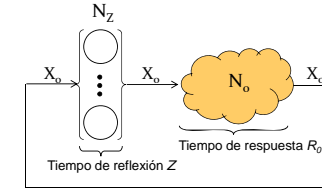
Dispositivo	$V_i$	$S_i$ (ms)	$D_i$ (s)
CPU	16	10	0.16
DISCO A	7	20	0.14
DISCO B	8	30	0.24



- Tiempo de respuesta mínimo:
  - $R_0^{min} = 0.16 + 0.14 + 0.24 = 0.54s$
- Productividad máxima:
  - $X_0^{max} = \frac{1}{D_b} = \frac{1}{0.24} = 4.2 \text{ trbj./s}$
- Utilización máxima de la CPU:
  - $U_{CPU}^{max} = X_0^{max} \times D_{CPU} = 0.67.$
- Productividad máxima de la CPU:
  - $X_{CPU}^{max} = X_0^{max} \times V_{CPU} = 67.2 \text{ tr/s}.$
- $U_i$  con  $\lambda_o = 2$  trabajos/s ( $=X_o$ )
  - $U_{CPU} = X_o \times D_{CPU} = 0.32$
  - $U_A = X_o \times D_A = 0.28$
  - $U_B = X_o \times D_B = 0.48$

49

## Límites optimistas: redes cerradas



Ley de Little a la red completa ( $N_T = N_Z + N_o$ ):

$$N_T = X_0 \times (R_0 + Z)$$

- a) Para valores de carga bajos ( $N_T$  pequeño):
  - Valor optimista del tiempo de respuesta: cuando los trabajos siempre encuentran los dispositivos sin ocupar ( $W_i = 0$ , por lo que  $R_i = S_i \forall i=1..K$ ):

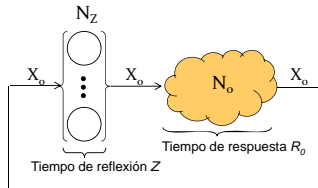
$$R_o \rightarrow R_o^{min} = \sum_{i=1}^K V_i \times R_i^{min} = \sum_{i=1}^K V_i \times S_i = \sum_{i=1}^K D_i \equiv D$$

- Valor optimista de la productividad a partir del valor optimista del tiempo de respuesta (sin más que reemplazar ese valor de  $R_o$  en la ley de Little a la red completa):

$$X_0 \rightarrow \frac{N_T}{R_o^{min} + Z} = \frac{N_T}{D + Z}$$

50

## Límites optimistas: redes cerradas (II)



Ley de Little a la red completa ( $N_T = N_Z + N_o$ ):

$$N_T = X_0 \times (R_0 + Z)$$

- b) Para valores de carga altos ( $N_T$  grande):
  - Valor optimista de la productividad. Suponemos que el dispositivo cuello de botella está cerca de la saturación

$$U_b = X_0 \times D_b$$

$$\text{Si } U_b \rightarrow 1 \Rightarrow X_o \rightarrow X_o^{max} = \frac{1}{D_b}$$

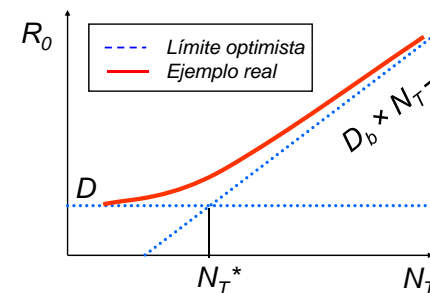
- Valor optimista del tiempo de respuesta, a partir del valor optimista de la productividad (sin más que reemplazar ese valor de  $X_o$  en la ley de Little a la red completa):

$$R_0 \rightarrow \left( \frac{N_T}{X_o^{max}} \right) - Z = D_b \times N_T - Z$$

51

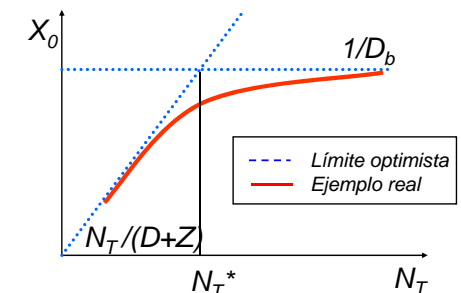
## Resumen: límites optimistas en redes cerradas

Tiempo de respuesta



$$R_0 \geq \max\{D, D_b \times N_T - Z\}$$

Productividad



$$X_0 \leq \min\left\{\frac{N_T}{D + Z}, \frac{1}{D_b}\right\}$$

52

## Punto teórico de “saturación” (knee point)

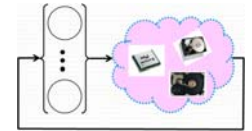
- Es el valor de  $N_T$  en donde las asíntotas coinciden:

$$D = D_b \times N_T^* - Z \Rightarrow N_T^* = \frac{D + Z}{D_b}$$

- Propiedades del punto teórico de “saturación”  $N_T^*$ :
  - Para un número total de trabajos  $N_T > N_T^*$ , los límites asíntóticos vienen impuestos únicamente por el cuello de botella del servidor.
  - A partir de  $N_T^*$  trabajos ya no se puede conseguir el tiempo de respuesta mínimo ya que se empiezan a formar colas de espera en, al menos, el dispositivo cuello de botella (en la práctica, esto sucede bastante antes).
  - En principio, podría parecer el número ideal de trabajos en la red ya que, al menos teóricamente, para  $N_T = N_T^*$  se podría conseguir la productividad máxima y el tiempo de respuesta mínimo absolutos del servidor (en la práctica esto nunca se puede conseguir de forma simultánea).

53

## Ejemplo de red cerrada



Tiempo de reflexión (Z) 18 s			
Dispositivo	$V_i$	$S_i$ (s)	$D_i$ (s)
CPU	5	1	5
DISCO A	2	2	4
DISCO B	2	1.5	3

- Asíntotas:

$$R_0 \geq \max\{D, D_b \times N_T - Z\}$$

$$= \max\{12, 5 \times N_T - 18\}$$

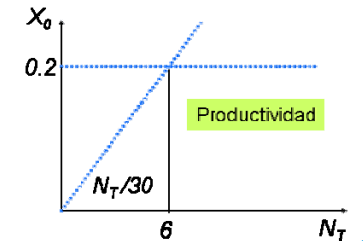
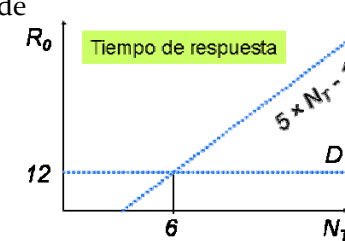
$$X_0 \leq \min\left\{\frac{N_T}{D + Z}, \frac{1}{D_b}\right\} = \min\left\{\frac{N_T}{30}, 0.2\right\}$$

- Punto teórico de saturación:

$$N_T^* = \frac{D + Z}{D_b}$$

$$= \frac{12 + 18}{5}$$

$$= 6 \text{ trabajos}$$

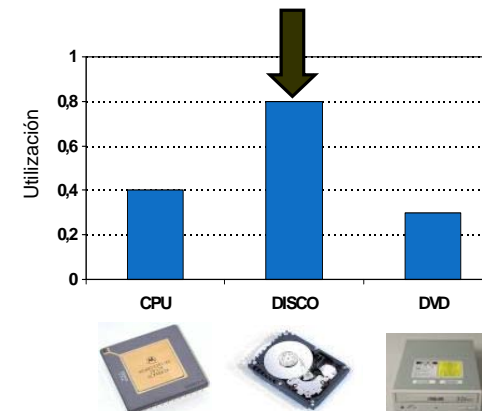


54

## 5.4. Técnicas de mejora

## Técnicas para mejorar las prestaciones

- Para mejorar las prestaciones de manera significativa hay que actuar sobre el cuello de botella del servidor.
  - Sintonización o ajuste (tuning).
  - Actualización o ampliación (upgrading).

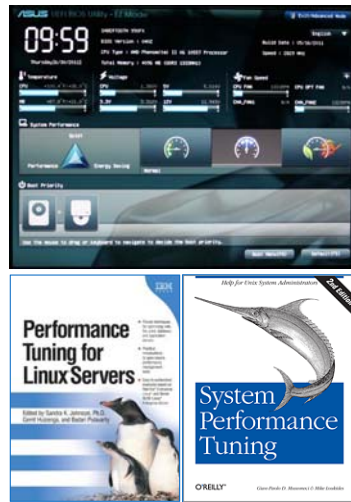


56



## Sintonización o ajuste (*tuning*)

- Optimización del funcionamiento de componentes existentes:
  - Componentes hardware.
  - Sistema operativo.
  - Aplicaciones.
- Muchos ajustes se hacen en el sistema operativo:
  - Políticas de gestión de procesos y memoria virtual.
  - Distribución de la información entre discos.
- Algunos problemas:
  - Hay que conocer muy bien el sistema operativo y el funcionamiento de los componentes hardware.
  - Posible alteración de la fiabilidad.



57

## Actualización o ampliación (*upgrading*)

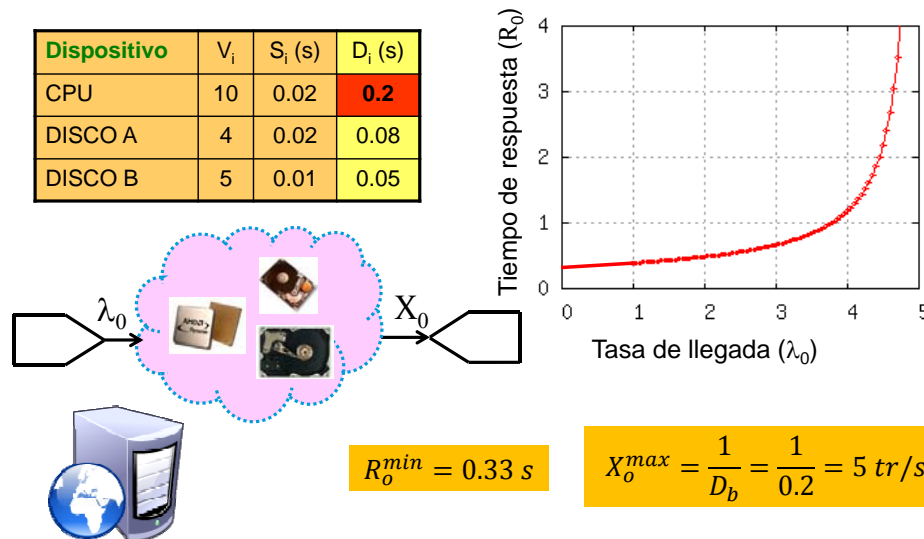
- Reemplazar dispositivos por otros más rápidos → Disminuimos el tiempo medio de servicio.
  - Procesador, memoria, placa base, disco...
- Añadir dispositivos para poder realizar más tareas en paralelo → Disminuimos la razón de visita.
  - Ejemplo: multiprocesadores, matrices de discos (RAID),...
- Algunos problemas:
  - Compatibilidad de los nuevos elementos con los existentes.
  - Facilidad del servidor para dejarse actualizar (extensibilidad/escalabilidad).



58

## Ejemplo: red abierta (servidor web)

Dispositivo	$V_i$	$S_i$ (s)	$D_i$ (s)
CPU	10	0.02	0.2
DISCO A	4	0.02	0.08
DISCO B	5	0.01	0.05

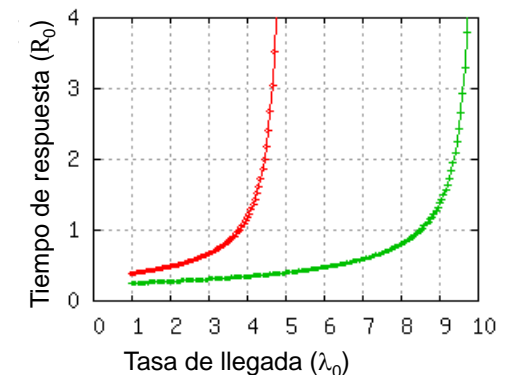


59

## Actualización: CPU doble de rápida

Dispositivo	$V_i$	$S_i$ (s)	$D_i$ (s)
CPU	10	0.01	0.1
DISCO A	4	0.02	0.08
DISCO B	5	0.01	0.05

La CPU se mantiene como cuello de botella pero con menor demanda



60

## Actualización: discos doble de rápidos

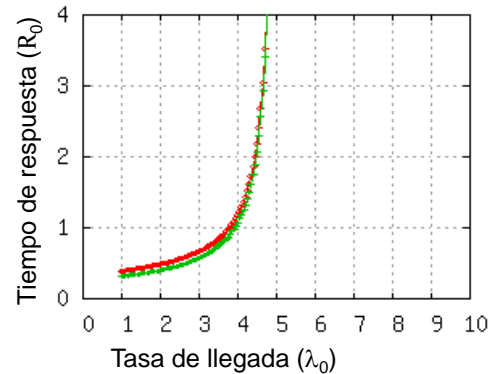
Dispositivo	$V_i$	$S_i$ (s)	$D_i$ (s)
CPU	10	0.02	0.2
DISCO A	4	0.01	0.04
DISCO B	5	0.005	0.025

La CPU se mantiene como cuello de botella



$$R_o^{\min} = 0.265 \text{ s}$$

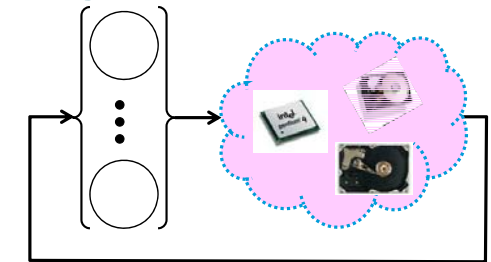
$$X_o^{\max} = \frac{1}{D_b} = \frac{1}{0.2} = 5 \text{ tr/s}$$



61

## Ejemplo: red cerrada interactiva (servidor de streaming)

Tiempo de reflexión (Z)		4 s	
Dispositivo	$V_i$	$S_i$ (s)	$D_i$ (s)
CPU	10	0.02	0.2
DISCO A	4	0.02	0.08
DISCO B	5	0.01	0.05



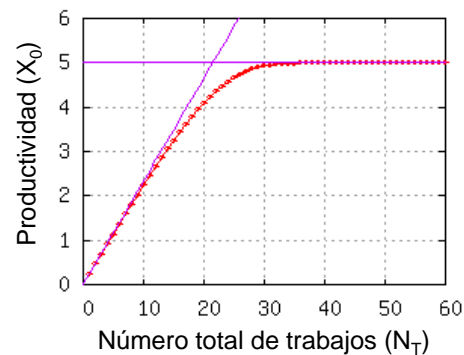
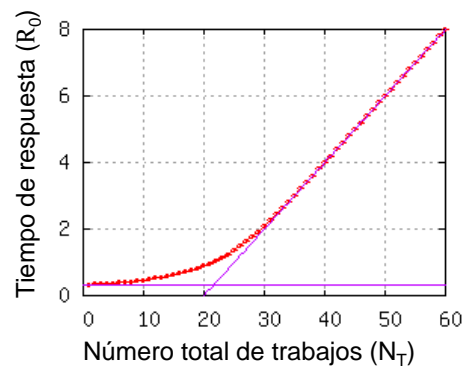
Asíntotas:

$$R_o \geq \max\{D, D_b \times N_T - Z\} = \max\{0.33, 0.2 \times N_T - 4\}$$

$$X_o \leq \min\left\{\frac{N_T}{D+Z}, \frac{1}{D_b}\right\} = \min\left\{\frac{N_T}{4.33}, 5\right\}$$

62

## Rendimiento del servidor original



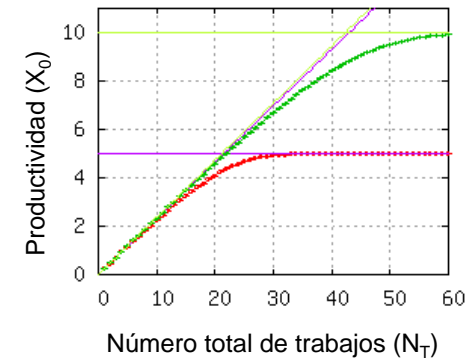
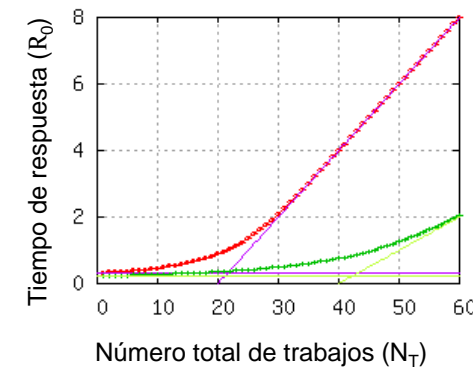
Punto teórico de saturación:

$$N_T^* = \frac{D+Z}{D_b} = 21.7 \text{ trabajos}$$

Los valores óptimos de ambos índices para  $N_T$  altos están determinados por el cuello de botella (CPU)

63

## Actualización: CPU doble de rápida



Punto teórico de saturación:

$$N_T^* = \frac{D+Z}{D_b} = 42.3 \text{ trabajos}$$

Asíntotas:

$$R_o \geq \max\{D, D_b \times N_T - Z\} = \max\{0.23, 0.1 \times N_T - 4\}$$

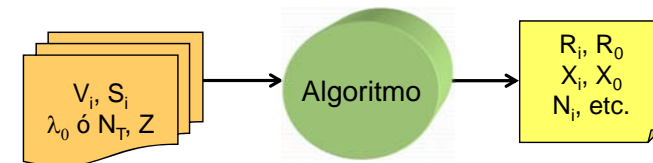
$$X_o \leq \min\left\{\frac{N_T}{D+Z}, \frac{1}{D_b}\right\} = \min\left\{\frac{N_T}{4.23}, 10\right\}$$

64

## 5.5. Algoritmos de resolución de modelos de redes de colas

## Algoritmos de resolución de redes de colas

- En este apartado vamos a proporcionar una metodología (algoritmo) para resolver modelos de redes de colas. Supondremos conocido:
  - El número de estaciones de servicio ( $K$ ).
  - Por cada estación:
    - Razón de visita medio de cada estación ( $V_i$ ).
    - Tiempo de servicio medio de cada estación ( $S_i$ ).
  - Si la red es abierta: Tasa de llegada al servidor ( $\lambda_0$ ).
  - Si la red es cerrada:
    - Número total de trabajos en la red ( $N_T$ ).
    - Si la red es interactiva: Tiempo medio de reflexión de los usuarios ( $Z$ ).

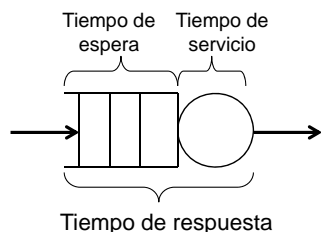


66

## Hipótesis del “peor escenario posible”

- Peor escenario posible: cuando un trabajo llega a la estación de servicio  $i$ -ésima de una **red de colas abierta** supondremos que:
  - Tiene que esperar a que se procesen los  $N_i$  trabajos que, de media, hay en ese momento en la estación, uno comenzando a ser servido y el resto esperando:
 
$$W_i = N_i \times S_i$$
  - Luego, para ser procesado, tendrá que esperar su propio tiempo de servicio.
- Por lo tanto, el tiempo de respuesta medio vendrá dado, en este peor escenario posible, por:

$$R_i = W_i + S_i = N_i \times S_i + S_i$$



De donde se deduce que (aplicando que  $N_i = \lambda_i \times R_i$ ):

$$R_i = \lambda_i \times R_i \times S_i + S_i \Rightarrow$$

$$R_i = \frac{S_i}{1 - \lambda_i \times S_i} = \frac{S_i}{1 - U_i} = \frac{S_i}{1 - \lambda_0 \times D_i}$$

↑  
si no saturado

67

## Resolución de redes de colas abiertas

- Modelos abiertos (suponemos conocidos:  $\lambda_0, V_i$  y  $S_i \forall i=1..K$ ).



Paso 1.- Calculamos la demanda media de servicio de cada estación:

$$D_i = V_i \times S_i$$

Paso 2.- Calculamos el tiempo medio de respuesta de cada estación:

$$R_i = \frac{S_i}{1 - \lambda_i \times S_i} = \frac{S_i}{1 - \lambda_0 \times D_i}$$

Paso 3.- Calculamos el tiempo medio de respuesta del servidor:

$$R_0 = \sum_{i=1}^K V_i \times R_i = \sum_{i=1}^K \frac{V_i \times S_i}{1 - \lambda_0 \times D_i} = \sum_{i=1}^K \frac{D_i}{1 - \lambda_0 \times D_i}$$

- El resto de variables operacionales ( $X_i, U_i, N_i, N_0, W_i, Q_i, \dots$ ) se pueden calcular usando sus expresiones habituales.

68

## Ejemplo: resolución de redes abiertas

Recurso	$V_i$	$S_i$ (s)
CPU	9	0.010
DISCO	3	0.020
RED	5	0.016



Suponiendo que la tasa de llegada al servidor es de 5 trabajos/s:

- Calcule las demandas de servicio de cada recurso.
- ¿Qué recurso es el cuello de botella? ¿Cuál es la productividad máxima del servidor? ¿Está el servidor saturado?
- Calcule el tiempo de respuesta, para el peor caso posible, de cada recurso y del servidor.
- Calcule el nº medio de trabajos en el servidor.
- Calcule la utilización, el tiempo medio de espera en la cola y el número de medio de trabajos en la cola de cada recurso.

69

## Solución del ejemplo

Recurso	$V_i$	$S_i$ (s)	$D_i$ (s)
CPU	9	0.010	0.09
DISCO	3	0.020	0.06
RED	5	0.016	0.08

b) La CPU es el cuello de botella (el recurso de mayor demanda de servicio).  
- Productividad máxima del servidor:

$$X_0^{max} = 1/D_b = 11.1 \text{ tr/s}$$

Como  $\lambda_0 = 5 \text{ tr/s} < X_0^{max}$  el servidor no está saturado.

c) Peor caso posible:  $R_i = N_i \times S_i + S_i \Rightarrow R_i = \frac{S_i}{1 - \lambda_0 \times D_i}$

$$R_{CPU} = \frac{S_{CPU}}{1 - \lambda_0 \times D_{CPU}} = \frac{0.01 \text{ s}}{1 - 5 \text{ tr/s} \times 0.09 \text{ s}} = 0.018 \text{ s}$$

Igualmente,  $R_{DISCO} = 0.029 \text{ s}$ ,  $R_{RED} = 0.027 \text{ s}$ .

Finalmente,  $R_0 = V_{CPU} \times R_{CPU} + V_{DISCO} \times R_{DISCO} + V_{RED} \times R_{RED} = 0.38 \text{ s}$

d)  $N_0 = X_0 \times R_0 = 5 \frac{\text{tr}}{\text{s}} \times 0.38 \text{ s} = 1.9 \text{ tr}$

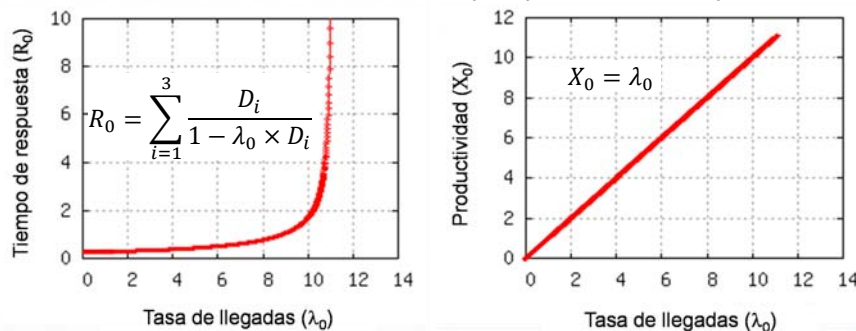
70

## Solución del ejemplo (cont.)

e)  $U_i = X_0 \times D_i = 5 \text{ tr/s} \times D_i$ ;  $W_i = R_i - S_i$   
 $Q_i = \lambda_i \times W_i = X_i \times W_i = X_0 \times V_i \times W_i = 5 \text{ tr/s} \times V_i \times W_i$

Recurso	$V_i$	$S_i$ (s)	$D_i$ (s)	$R_i$ (s)	$U_i$	$W_i$ (s)	$Q_i$ (tr.)
CPU	9	0.010	0.09	0.018	0.45	0.008	0.37
DISCO	3	0.020	0.06	0.029	0.30	0.009	0.13
RED	5	0.016	0.08	0.027	0.40	0.011	0.27

Adicionalmente, podríamos representar  $R_0$  y  $X_0$  en función de  $\lambda_0$ :



71

## Resolución con solvenet

- Programa muy sencillo que resuelve redes de colas utilizando los algoritmos de esta sección.
  - Disponible el código fuente en lenguaje C (SWAD).
  - Los parámetros del modelo se indican en la línea de comandos.

```
Usage: solvenet [0|1] [lambda0] NT Z] K S1 V1...SK VK
With no parameters, shows this message
network: 0 (open) and 1 (closed)
lambda0: arrival rate = throughput (only open networks)
NT:      total number of jobs in the net (only closed nets)
Z:       think time (only interactive closed networks)
K:       number of service stations
S1:      service time of device 1
Vi:      ratio visit of device i
```

72

## Resolución con solvenet: redes abiertas

Recurso	$V_i$	$S_i$ (s)	$D_i$ (s)
CPU	9	0.010	0.09
DISCO	3	0.020	0.06
RED	5	0.016	0.08

$\lambda_o$  5.0 trabajos/s

solvenet 0 5.0 3 0.01 9 0.02 3 0.016 5



```
*****
*   NAME   *   Ui   *   Ni   *   Ri   *   Xi   *   Wi   *
*****
*   *   *   *   *   *   *   *   *
* DEV 1 *   0.4500*   0.8182*   0.0182*   45.0000*   0.0082*
*   *   *   *   *   *   *   *   *
* DEV 2 *   0.3000*   0.4286*   0.0286*   15.0000*   0.0086*
*   *   *   *   *   *   *   *   *
* DEV 3 *   0.4000*   0.6667*   0.0267*   25.0000*   0.0107*
*   *   *   *   *   *   *   *   *
*****
```

73

## Resolución con solvenet: redes abiertas (II)

Recurso	$V_i$	$S_i$ (s)	$D_i$ (s)
CPU	9	0.010	0.09
DISCO	3	0.020	0.06
RED	5	0.016	0.08

$\lambda_o$  5.0 trabajos/s

solvenet 0 5.0 3 0.01 9 0.02 3 0.016 5



```
*****
*   NAME   *   Vi   *   Si   *   Di   *   Qi   *
*****
*   *   *   *   *   *   *   *   *
* DEV 1 *   9.0000*   0.0100*   0.0900*   0.3682*
*   *   *   *   *   *   *   *   *
* DEV 2 *   3.0000*   0.0200*   0.0600*   0.1286*
*   *   *   *   *   *   *   *   *
* DEV 3 *   5.0000*   0.0160*   0.0800*   0.2667*
*   *   *   *   *   *   *   *   *
*****
```

74

## Resolución con solvenet: redes abiertas (III)

Recurso	$V_i$	$S_i$ (s)	$D_i$ (s)
CPU	9	0.010	0.09
DISCO	3	0.020	0.06
RED	5	0.016	0.08

$\lambda_o$  5.0 trabajos/s

solvenet 0 5.0 3 0.01 9 0.02 3 0.016 5

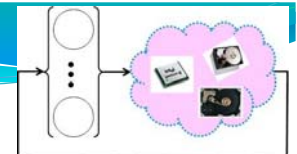


```
*****
*   SYSTEM VARIABLES   *
*****
*   #JOBS IN SYSTEM (N0) *   1.9134*
*   *   *   *   *   *
*   RESPONSE TIME (R0) *   0.3827*
*   MINIMUM RESPONSE TIME *   0.2300*
*   *   *   *   *   *
*   THROUGHPUT (X0) *   5.0000*
*   MAXIMUM THROUGHPUT *   11.111*
*   *   *   *   *   *
*****
```

```
*****
*   ASIMPTOTIC BOUNDS   *
*****
*   R0_min =   0.2300   *
*   X0_max =   11.1111   *
*   *   *   *   *   *
*****
```

75

## Resolución de redes cerradas



- Modelos cerrados (suponemos conocidos:  $V_i$ ,  $S_i$ ,  $N_T$  y  $Z$ )
  - Método: Debemos ir resolviendo la red para valores incrementales del número de trabajos en la red hasta alcanzar  $N_T$ :  $n_T=1, \dots, N_T$ .
  - Notación:  $N_i(n_T)$ : Número de trabajos en la estación de servicio  $i$ -ésima si en la red hubiese  $n_T$  trabajos. Ídem para los tiempos de respuesta  $R_i(n_T)$  y las productividades  $X_i(n_T)$ .

For  $i = 1$  to  $K$  do  $N_i(0) = 0$  ← Inicialización del nº de trabajos en cada estación

For  $n_T = 1$  to  $N_T$  do

For  $i = 1$  to  $K$  do  $R_i(n_T) = (N_i(n_T - 1) + 1) \times S_i$  ← Hipótesis del peor caso para redes cerradas

$R_0(n_T) = \sum_{i=1}^K V_i \times R_i(n_T), X_0(n_T) = \frac{n_T}{Z + R_0(n_T)}$  ← Tiempo de respuesta y productividad del servidor.

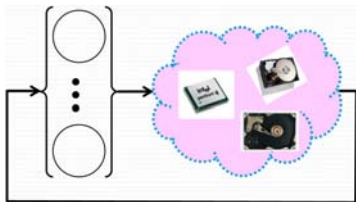
For  $i = 1$  to  $K$  do  $N_i(n_T) = X_0(n_T) \times V_i \times R_i(n_T)$  ← Actualización del número de trabajos en cada estación.

76



## Ejemplo: resolución de redes cerradas

T. reflexión (Z)		2 s
Recurso	$V_i$	$S_i$ (s)
CPU	10	0.01
DISCO1	5	0.02
DISCO2	4	0.03



$$N_{CPU}(0) = N_{DISCO1}(0) = N_{DISCO2}(0) = 0$$

$$n_T = 1$$

$$R_{CPU}(1) = S_{CPU} = 0.01s$$

$$R_{DISCO1}(1) = S_{DISCO1} = 0.02s$$

$$R_{DISCO2}(1) = S_{DISCO2} = 0.03s$$

$$R_0(1) = 10 \times 0.01 + 5 \times 0.02 + 4 \times 0.03 = 0.32s$$

$$X_0(1) = \frac{1}{2 + 0.32} = 0.43 \text{ trabajos/s}$$

$$N_{CPU}(1) = 0.43 \times 10 \times 0.01 = 0.043$$

$$N_{DISCO1}(1) = 0.43 \times 5 \times 0.02 = 0.043$$

$$N_{DISCO2}(1) = 0.43 \times 4 \times 0.03 = 0.052$$

77

## Ejemplo: resolución de redes cerradas (II)

$$N_{CPU}(1) = 0.043; N_{DISCO1}(1) = 0.043$$

$$N_{DISCO2}(1) = 0.052$$

$$n_T = 2$$

$$R_{CPU}(2) = (0.043 + 1) \times S_{CPU} = 0.01043s$$

$$R_{DISCO1}(2) = (0.043 + 1) \times S_{DISCO1} = 0.0209s$$

$$R_{DISCO2}(2) = (0.052 + 1) \times S_{DISCO2} = 0.0316s$$

$$R_0(2) = \sum_{i=1}^3 V_i \times R_i(2) = 0.3348s$$

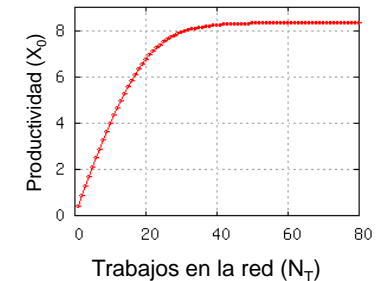
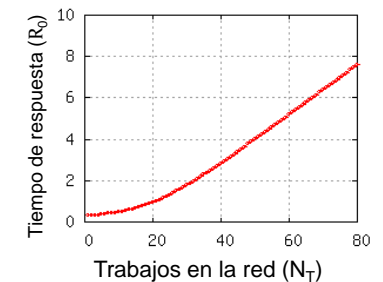
$$X_0(2) = \frac{2}{2 + 0.3348} = 0.857 \text{ trabajos/s}$$

$$N_{CPU}(2) = 0.857 \times 10 \times 0.01043 = 0.0894$$

$$N_{DISCO1}(2) = 0.857 \times 5 \times 0.0209 = 0.0894$$

$$N_{DISCO2}(2) = 0.857 \times 4 \times 0.0316 = 0.1081$$

etc. hasta llegar al valor  $n_i = n_T$  que nos pidan



78

## Resolución con solvenet: redes cerradas

Recurso	$V_i$	$S_i$ (s)
CPU	10	0.01
DISCO1	5	0.02
DISCO2	4	0.03

T. reflexión (Z)		2 s
Nº Trabajos en la red cerrada ( $N_T$ )		2
solvenet 1 2 2 3 0.01 10 0.02 5 0.03 4		



NAME	$U_i$	$N_i$	$R_i$	$X_i$	$W_i$
DEV 1	0.0857*	0.0894*	0.0104*	8.5659*	0.0004*
DEV 2	0.0857*	0.0894*	0.0209*	4.2830*	0.0009*
DEV 3	0.1028*	0.1081*	0.0316*	3.4264*	0.0016*

79

## Resolución con solvenet: redes cerradas (II)

Recurso	$V_i$	$S_i$ (s)
CPU	10	0.01
DISCO1	5	0.02
DISCO2	4	0.03

T. reflexión (Z)		2 s
Nº Trabajos en la red cerrada ( $N_T$ )		2
solvenet 1 2 2 3 0.01 10 0.02 5 0.03 4		



NAME	$V_i$	$S_i$	$D_i$	$Q_i$
DEV 1	10.0000*	0.0100*	0.1000*	0.0037*
DEV 2	5.0000*	0.0200*	0.1000*	0.0037*
DEV 3	4.0000*	0.0300*	0.1200*	0.0053*

80



## Resolución con solvenet: redes cerradas (III)

Recurso	$V_i$	$S_i$ (s)
CPU	10	0.01
DISCO1	5	0.02
DISCO2	4	0.03

T. reflexión (Z)	2 s
Nº Trabajos en la red cerrada ( $N_T$ )	2

solvenet 1 2 2 3 0.01 10 0.02 5 0.03 4



```
*****
*          SYSTEM VARIABLES          *
*****
*          *          *
* #JOBS IN SYSTEM (N0) *      0.2868*
* #INTERACTIVE USERS(NZ)*    1.7132*
* #JOBS IN THE NET (NT) *      2*
* SATURATION POINT (N*) *     20*
*          *          *
* RESPONSE TIME (R0)    *    0.3348*
* MINIMUM RESPONSE TIME*    0.3200*
*          *          *
* THROUGHPUT (X0)       *    0.8566*
* MAXIMUM THROUGHPUT    *    8.3333*
*          *          *
*****
```

```
*****
*          ASIMPTOTIC BOUNDS          *
*****
*          *          *
* R0 >= max{0.32,  0.12*NT-2.00} *
*          *          *
* X0 <= min {NT/2.32,  8.33} *
*          *          *
*****
```