



Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....  
 Apellidos ..... Nombre.....  
 Ejercicio del día **13 de mayo de 2016**.....

1.1 (1.5)	1.2 (1.5)	1.3 (1.5)	1.4 (1.5)	2 (1.5)	3 (2.5)	Total Parte I (10)

## PARTE I

(33,33% de la nota del examen)

**1.- (6 puntos).** Resuelve de modo claro y conciso las siguientes cuestiones:

**1.1. (1.5 puntos).** Describe en qué consiste la transparencia de datos, y nombra tres mecanismos que se podrían utilizar para conseguirla.

**1.2. (1.5 puntos).** Indica las componentes que aparecen en el estándar de directorios X.500 y los protocolos que se utilizan para el intercambio de mensajes. Indicar qué componente se ejecuta en el cliente y qué componente se ejecuta en el servidor.



Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....  
 Apellidos ..... Nombre.....  
 Ejercicio del día **13 de mayo de 2016**.....

**1.3. (1.5 puntos).** Explica el ciclo de desarrollo de una aplicación distribuida basada en CORBA.

**1.4. (1.5 puntos).** Describe el modelo publicador / suscriptor para el diseño de sistemas distribuidos (componentes que lo forman, interacciones entre ellas, y tareas a llevar a cabo por cada una). Indica qué mecanismo de comunicación es más adecuado para su implementación.



Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....  
 Apellidos ..... Nombre.....  
 Ejercicio del día **13 de mayo de 2016**.....

**2.- (1.5 puntos).** Considera el siguiente fichero WSDL asociado a un servicio web SOAP:

```
<?xml version="1.0"?>

<definitions name="MyService" targetNamespace=http://www.examples.com/wSDL/MyService.wsdl
xmlns="http://schemas.xmlsoap.org/wSDL/" xmlns:soap=http://schemas.xmlsoap.org/wSDL/soap/
xmlns:tns="http://www.examples.com/wSDL/MyService.wsdl"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">

    <message name="loginRequest"><part name="user" type="xsd:string"/></message>
    <message name="loginResponse"><part name="greeting" type="xsd:string"/></message>
    <message name="addressRequest"><part name="user" type="xsd:string"/></message>
    <message name="addressResponse"><part name="address" type="xsd:string"/></message>

    <portType name="Service_PortType"><operation name="login">
        <input message="tns:loginRequest"/><output message="tns:loginResponse"/>
    </operation><operation name="address">
        <input message="tns:addressRequest"/><output message="tns:addressResponse"/>
    </operation></portType>

    <binding name="Service_Binding" type="tns:Service_PortType">
    <soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
        <operation name="login"><soap:operation soapAction="login"/>
            <input><soap:body encoding="literal"/></input><output>
            <soap:body encoding="literal"/></output></operation>
            <operation name="address"><soap:operation soapAction="address"/>
            <input><soap:body encoding="literal"/></input><output>
            <soap:body encoding="literal"/></output></operation>
        </binding>

    <service name="My_Service"><port binding="tns:Service_Binding" name="Service_Port">
        <soap:address location="http://www.examples.com/MyService/" />
    </port> </service>
</definitions>
```

**2.1. (1 punto).** Indicar las operaciones soportadas por el servicio web y el intercambio de mensajes asociado a cada operación.



Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....  
 Apellidos ..... Nombre.....  
 Ejercicio del día **13 de mayo de 2016**.....

El servicio web soporta dos operaciones "login" y "address". Cada operación involucra enviar un mensaje de petición y uno de respuesta. Para la operación "login" el mensaje de petición es "loginRequest" y el mensaje de respuesta es "loginResponse". Para la operación "address" el mensaje de petición es "addressRequest" y el mensaje de respuesta es "addressResponse".

**2.2 (0.5 puntos).** Indicar la URL que se utilizará para solicitar la invocación del servicio (dirección del servicio).

La URL es la que aparece en el apartado "service": <http://www.examples.com/MyService/>

**3.- (2.5 puntos).** Considerar los siguientes casos particulares de sistemas distribuidos basados en el modelo cliente / servidor.

- a) **Red social de micro-posts:** los usuarios pueden utilizar el servicio para consultar los posts de otros usuarios, publicar un nuevo post, modificar un post ya publicado, seguir a otro usuario, consultar la lista de usuarios seguidos, etc. La empresa cuenta con un corta-fuegos a través del cual deberán pasar los clientes. El ancho de banda de la empresa es reducido y por lo tanto se desearía evitar que el tamaño de los mensajes sea excesivamente grande. Los clientes que usarán el servicio serán heterogéneos.
- b) **Servicio de tiempo:** clientes heterogéneos utilizan el servicio para solicitar información sobre la hora actual. Los clientes envían al servidor un mensaje. Tras esto el servidor contesta con otro mensaje que incluye ciertas marcas de tiempo. Los clientes utilizan esta información para actualizar la hora de sus relojes.

Para cada uno de ellos se pide elegir justificadamente (**cuantas más razones se den para la elección mayor la puntuación**) el mecanismo de comunicación más adecuado entre UDP, Servicios web basados en REST, Servicios web basados en SOAP o Colas de Mensajes. Indicar además si será necesario usar algún mecanismo adicional para garantizar la **transparencia de datos**.

- a) Descartamos UDP debido a la limitación de tamaño de mensaje, y al hecho de que los clientes serán heterogéneos y éste mecanismo no proporciona transparencia de datos. Descartamos colas de mensajes ya que la interacción será síncrona, y colas de mensajes tampoco proporciona transparencia de datos. Otras razones para descartar estos mecanismos es la existencia del corta-fuegos, que los filtraría. Los servicios web pasarían a través del corta-fuegos. Entre ellos elegimos REST ya que es más eficiente que SOAP. Podríamos usar JSON para la representación de los datos, lo que nos proporcionaría transparencia de datos con lo que no tendríamos que traducirlos aunque los clientes sean heterogéneos.
- b) En este servicio parece crítico el envío rápido de mensajes. Esto descartaría colas de mensajes (comunicación asíncrona) y también los servicios web basados en SOAP, ya que procesar sus mensajes es muy costoso, y son muy grandes. REST puede ser más eficiente que SOAP, sin embargo la solución más rápida sería utilizar UDP para realizar el envío de los mensajes. Como los mensajes son pequeños (marcas de tiempo), no tenemos problemas por espacio. Las operaciones son idempotentes, así que tampoco habría



Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....  
 Apellidos ..... Nombre.....  
 Ejercicio del día **13 de mayo de 2016**.....

problema porque se perdiesen los mensajes. Simplemente se podrían volver a enviar el mensaje sin problema. El único problema es que UDP no proporciona transparencia de datos. Sin embargo, bastaría con introducir algún mecanismo de traducción para garantizar esto último. Un ejemplo podría ser traducir todo a Little-endian.



Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....  
 Apellidos ..... Nombre.....  
 Ejercicio del día **13 de mayo de 2016**.....

1 (7)	2 (3)	Total Parte II (10)

## PARTE II

(33,33% de la nota del examen)

**1. PROBLEMA (7 puntos).** Un servidor web de aplicaciones recibe peticiones de los clientes según un proceso de Poisson con una media de **20 peticiones por segundo**. Las peticiones son recibidas inicialmente por un servidor caché que analiza el contenido de las mismas para determinar si se trata de peticiones de páginas estáticas que tenga en caché. En caso de ser así, el servidor caché devuelve la respuesta al cliente y la petición finaliza. Este servidor cuenta con una única CPU que tarda en promedio **20 ms** en analizar cada petición. Se estima que en promedio, el **30%** de las peticiones recibidas se encuentran en caché y el **70%** de las peticiones no están cacheadas.

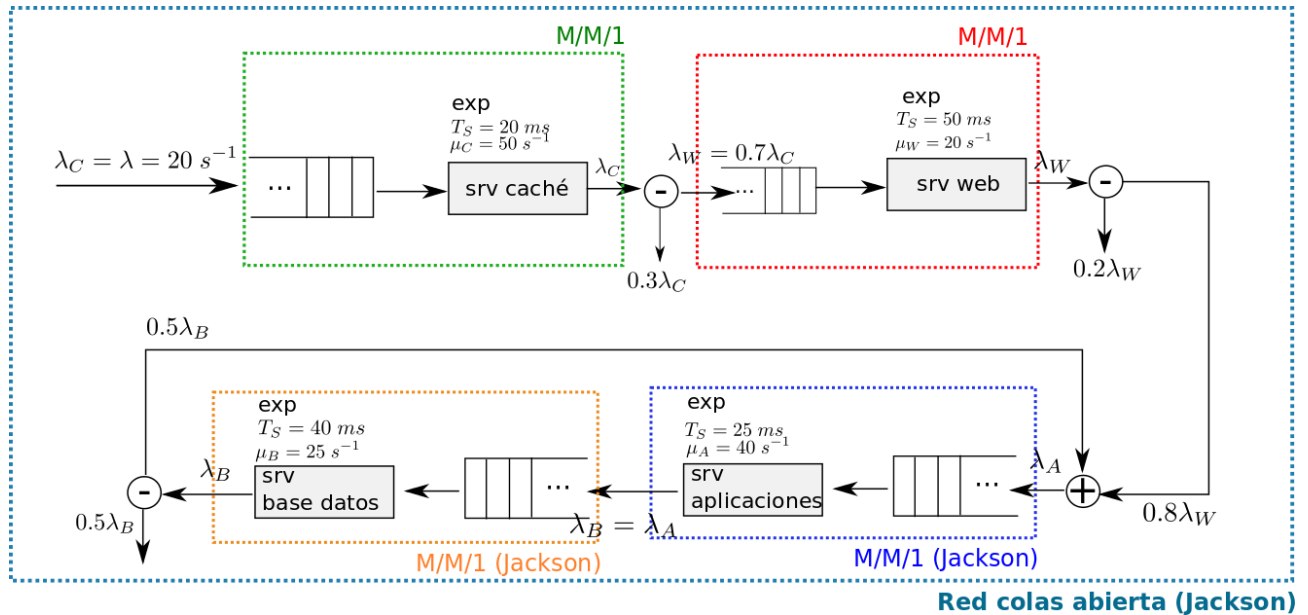
Aquellas peticiones que no puedan ser resueltas por el servidor caché, pasan a un servidor web que cuenta con una CPU. El tiempo que tarda la CPU en procesar el trabajo tiene un valor medio de **50 ms**. Un **80%** de las peticiones procesadas por el servidor web requieren que, adicionalmente al proceso realizado por dicho servidor, se ejecute un programa externo. Esta ejecución se realiza en un servidor de aplicaciones. El 20% restante de peticiones salen del sistema.

El servidor de aplicaciones cuenta con una única CPU que tarda en promedio **25 ms** en procesar una petición. Cada vez que el servidor de aplicaciones procesa una petición, necesita acceder a un servidor de base de datos. Este servidor tiene un tiempo de servicio con media **40 ms**. Tras pasar por el servidor de aplicaciones y el servidor de bases de datos, la **mitad** de las peticiones necesita realizar otra petición al servidor de aplicaciones, y el resto de peticiones salen del sistema.

Suponer que todos los tiempos de servicio están distribuidos de forma exponencial, que los servidores tienen una cola de espera de tamaño infinito y que existe un número muy grande de clientes, de modo que el número de peticiones pendientes de servicio no afecta al ritmo de llegada de nuevas peticiones.

**Redondear todos los resultados a tres posiciones decimales.**

**1.1 (1 punto)** Dibujar el diagrama de proceso del sistema completo, y expresar (no calcular) las tasas de llegada a la entrada de cada servidor, indicando las suposiciones realizadas. Dar una explicación razonada de qué modelo, según la notación de Kendall, será aplicable a cada una de sus componentes.



Las tasas de llegadas a la entrada de cada servidor se pueden obtener al suponer que los sistemas se encuentran en estado estacionario. Entonces:

- La tasa de entrada al servidor caché será la tasa de llegadas exterior del sistema. Por tanto, la tasa de llegadas  $\lambda_C$  al servidor caché sigue una distribución de Poisson. Al encontrarse en estado estacionario y no haber retroalimentación, el servidor caché sigue un modelo **M/M/1** y por el Teorema de Burke, su tasa de salida es  $\lambda_C$  con distribución de Poisson.
- La tasa de entrada a los servidores web será el 70% de la tasa de salidas del servidor caché. Como se trata de una división aleatoria de un proceso de Poisson, la entrada servidor web también seguirá una distribución de Poisson con tasa de llegadas  $\lambda_W = 0.7\lambda_C$ . Al encontrarse en estado estacionario y no haber retroalimentación, el servidor web sigue un modelo **M/M/1** y por el Teorema de Burke, su tasa de salida es  $\lambda_W$  con distribución de Poisson.
- La tasa de entrada al servidor de aplicaciones viene dada por la suma de la tasa de llegadas procedentes del 80% de las peticiones salientes del servidor web y el 50% de la tasa de salidas del servidor de bases de datos (retroalimentación). Por tanto, al haber retroalimentación, la tasa de llegadas  $\lambda_A$  al servidor de aplicaciones no sigue una distribución de Poisson. Sin embargo, al encontrarse en estado estacionario, la tasa de salida del distribuidor será  $\lambda_A$ . A pesar de no tener una tasa neta de llegadas Poissoniana, el **Teorema de Jackson** nos permite modelar el número de clientes en el servidor de aplicaciones como un modelo **M/M/1**.
- El mismo razonamiento anterior se aplica para el servidor de bases de datos, que se modelará de acuerdo a un modelo **M/M/1**. En este caso se denotará su tasa de entrada como  $\lambda_B$  y  $\lambda_B = \lambda_A$ .



Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....  
 Apellidos..... Nombre.....  
 Ejercicio del día **13 de mayo de 2016**.....

El sistema en su conjunto se puede modelar como una red de colas abiertas, ya que se asume que cada uno de los sistemas está en estado estacionario, las llegadas desde el exterior siguen una distribución de Poisson, y existe una probabilidad no nula de que las peticiones salgan del sistema.

**1.2 (1 punto) Calcular la tasa de llegadas efectiva a la entrada de cada servidor (servidor caché, servidor web, servidor de aplicaciones y servidor de bases de datos).**

- Tasa de llegadas efectiva a la entrada del servidor caché:  $\lambda_C = \lambda = 20 \text{ s}^{-1}$
- Tasa de llegadas efectiva a la entrada del servidor web:  $\lambda_W = 0.7 \lambda_C = 0.7 \cdot 20 = 14 \text{ s}^{-1}$ .
- Tasa de llegadas efectiva a la entrada del servidor de aplicaciones:  $\lambda_A = 0.5 \lambda_A + 0.8 \cdot \lambda_W = 0.8 \cdot 0.7 \cdot \lambda \Rightarrow \lambda_A = \frac{0.56}{0.5} \lambda = 22.4 \text{ s}^{-1}$ .
- Tasa de llegadas efectiva a la entrada del servidor de bases de datos:

Observar que en todos los casos la tasa de llegadas a cada servidor es menor que la tasa de servicio, y, por tanto, los sistemas se encuentran efectivamente en estado estacionario.

**1.3 (1 punto) Calcular justificadamente el número medio de peticiones en cola en cada uno de los componentes (servidor caché, servidor web, servidor de aplicaciones y servidor de bases de datos).**

Para calcular el número de peticiones en cola en los modelos M/M/1 usaremos que  $L^q = L - \rho^1$ .

Para calcular el número medio de clientes  $L_C$  en el servidor caché aplicamos las ecuaciones del modelo M/M/1:

$$L_C = \frac{\rho_C}{1 - \rho_C} = \frac{\lambda_C}{\mu_C - \lambda_C} = \frac{20}{50 - 20} = 0.667 \text{ clientes}$$

Y por tanto, el número medio de clientes en cola para el servidor caché será:  $L_C^q = L_C - \rho_C = L_C - \frac{\lambda_C}{\mu_C} = 0.667 - \frac{20}{50} = 0.267 \text{ clientes}$ .

Para calcular el número medio de clientes  $L_W$  en el servidor web aplicamos las ecuaciones del modelo M/M/1:

$$L_W = \frac{\rho_W}{1 - \rho_W} = \frac{\lambda_W}{\mu_W - \lambda_W} = \frac{14}{20 - 14} = 2.333 \text{ clientes}$$

Y por tanto, el número medio de clientes en cola para el servidor web será:  $L_W^q = L_W - \rho_W = L_W - \frac{\lambda_W}{\mu_W} = 2.333 - \frac{14}{20} = 1.633 \text{ clientes}$ .

<sup>1</sup> Se puede resolver de otras formas. Por ejemplo, calculando el tiempo medio de espera en cola y aplicando el Teorema de Little.





Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....  
 Apellidos..... Nombre.....  
 Ejercicio del día **13 de mayo de 2016**.....

Para calcular el número medio de clientes  $L_A$  en el servidor de aplicaciones aplicamos las ecuaciones del modelo M/M/1:

$$L_A = \frac{\rho_A}{1 - \rho_A} = \frac{\lambda_A}{\mu_A - \lambda_A} = \frac{22.4}{40 - 22.4} = 1.273 \text{ clientes}$$

Y por tanto, el número medio de clientes en cola para el servidor de aplicaciones será:  $L_A^q = L_A - \rho_A = L_A - \frac{\lambda_A}{\mu_A} = 1.273 - \frac{22.4}{40} = 0.713 \text{ clientes}$ .

Para calcular el número medio de clientes  $L_B$  en el servidor de bases de datos aplicamos las ecuaciones del modelo M/M/1:

$$L_B = \frac{\rho_B}{1 - \rho_B} = \frac{\lambda_B}{\mu_B - \lambda_B} = \frac{22.4}{25 - 22.4} = 8.615 \text{ clientes}$$

Y por tanto, el número medio de clientes en cola para el servidor de bases de datos será:  $L_B^q = L_B - \rho_B = L_B - \frac{\lambda_B}{\mu_B} = 8.615 - \frac{22.4}{25} = 7.719 \text{ clientes}$ .

#### 1.4 (1.5 puntos) Calcular justificadamente el tiempo medio de respuesta de todo el sistema.

Aplicamos el Teorema de Little sobre el sistema global, teniendo en cuenta que el número medio de clientes en el sistema total es la suma del número medio de clientes en cada uno de los subsistemas y la tasa de llegadas externas al sistema es  $\lambda$ :

$$W = \frac{L_C + L_W + L_A + L_B}{\lambda} = \frac{0.667 + 2.333 + 1.273 + 8.615}{20} = 0.644 \text{ s}$$

#### 1.5 (1.5 puntos) Calcular justificadamente el tiempo medio de respuesta para las peticiones que requieren pasar por el servidor de aplicaciones.

$$W = W_C + W_W + \frac{L_A + L_B}{0.8 \lambda_W} = \frac{L_C}{\lambda_C} + \frac{L_W}{\lambda_W} + \frac{L_A + L_B}{0.8 \lambda_W} = \frac{0.667}{20} + \frac{2.333}{14} + \frac{1.273 + 8.615}{0.8 \cdot 14} = 1.083 \text{ s}$$

#### 1.6 (1 punto) Determinar justificadamente si el servidor caché actual es suficiente para que al menos el 60% de las peticiones que solo pasan por este servidor tengan una latencia inferior a 25 milisegundos. No se tendrán en cuenta respuestas no razonadas.

Como las peticiones solo pasan por el servidor caché, el tiempo de estancia en el sistema (latencia) de esas peticiones será únicamente el del servidor caché. Se calcula la probabilidad de que el



Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....  
 Apellidos ..... Nombre.....  
 Ejercicio del día **13 de mayo de 2016**.....

tiempo de estancia en el servidor caché sea menor o igual a 25 milisegundos de acuerdo a la ecuación de la función de distribución del tiempo de estancia en el sistema para el modelo M/M/1:

$$F_W(t) = 1 - e^{-(\mu_c - \lambda_c)t} = 1 - e^{-(50-20)0.025} = 0.528 < 0.60$$

Por tanto, no es posible satisfacer el requisito con el servidor caché actual.

**2. PROBLEMA (3 puntos)** Una empresa cuenta con un servidor A que recibe las peticiones de los clientes sobre un conjunto de servicios web y que tiene un tiempo medio de servicio igual a **40ms** distribuido de forma exponencial. Por construcción, dicho servidor solo admite **4 peticiones** esperando en cola. En caso de que se reciba una petición y la cola esté llena, la petición se rechaza. El servidor A recibe peticiones de los clientes siguiendo un proceso Poisson con una media de **20 peticiones por segundo**.

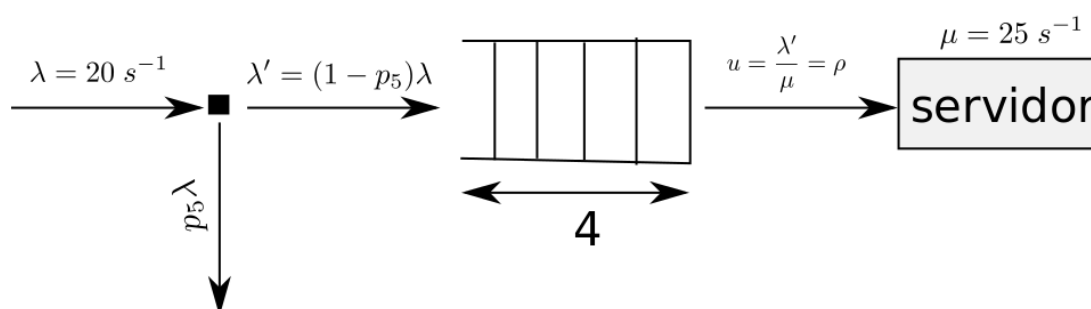
Suponer que existe un número muy grande de clientes, de modo que el número de peticiones pendientes de servicio no afecta al ritmo de llegada de nuevas peticiones.

**Redondear todos los resultados a tres posiciones decimales.**

**2.1 (0.5 punto) Dibujar el diagrama del sistema y justificar un modelo válido para describirlo.**

- Tiempo de servicio exponencial con valor esperado 40 ms  $\rightarrow T_s = 0.04 \text{ s}$  y  $\mu = 25$  peticiones/s.
- Cola de espera limitada, de modo que rechaza cualquier nueva solicitud que reciba cuando ya se encuentra procesando una y hay otras dos en espera  $\rightarrow$  Cola de tamaño 4 y 1 servidor  $\rightarrow$  Capacidad máxima de clientes que puede contener el sistema  $K=5$ .
- Las peticiones que se realizan siguen un ritmo de Poisson con tasa de 4 peticiones por segundo  $\rightarrow$  Tiempo entre llegadas exponencial con tasa de llegadas  $\lambda = 20$  peticiones/s.

Por tanto, se tarda de un modelo M/M/1/5 como se muestra en la siguiente figura:



**2.2 (1 punto) Calcular la probabilidad de que se rechace una solicitud.**

La probabilidad de rechazar una solicitud viene dada por la probabilidad de que haya actualmente 5 clientes en el sistema (1 siendo servido y 4 en cola). Por tanto, la probabilidad de rechazo viene dada por  $p_5$ .

Podemos calcularla de acuerdo a la expresión de la distribución estacionaria de Markov para el modelo M/M/1/K:

$$p_5 = p_0 \left( \frac{\lambda}{\mu} \right)^5$$



Asignatura SISTEMAS INFORMÁTICOS II Grupo.....  
 Apellidos..... Nombre.....  
 Ejercicio del día 13 de mayo de 2016

Conocemos  $\lambda$  y  $\mu$ , pero necesitamos calcular  $p_0$ . Usamos la expresión para  $p_0$  derivada del segundo axioma de probabilidad:

$$p_0 = \frac{1 - \frac{\lambda}{\mu}}{1 - \left(\frac{\lambda}{\mu}\right)^{K+1}} = \frac{1 - \frac{20}{25}}{1 - \left(\frac{20}{25}\right)^6} = 0.271$$

Con este valor de  $p_0$  podemos calcular  $p_5$  como sigue:

$$p_5 = p_0 \left(\frac{\lambda}{\mu}\right)^5 = 0.271 \cdot \left(\frac{20}{25}\right)^5 = 0.089$$

Por tanto, la probabilidad de rechazo es del 8.9%.

### 2.3 (1 punto) Calcular el tiempo medio de respuesta del sistema para las peticiones que no son rechazadas.

El tiempo medio de estancia en el sistema de las peticiones que son procesadas viene dado por  $W$ . Para calcular  $W$  podemos seguir los siguientes pasos:

1. Calcular el número de clientes en el sistema  $L$ .
2. Obtener  $W$  a partir de  $L$  usando el Teorema de Little.

$$L = \frac{\frac{\lambda}{\mu}}{1 - \frac{\lambda}{\mu}} \left[ \frac{1 - (K+1) \left(\frac{\lambda}{\mu}\right)^K + K \left(\frac{\lambda}{\mu}\right)^{K+1}}{1 - \left(\frac{\lambda}{\mu}\right)^{K+1}} \right] = \frac{\frac{20}{25}}{1 - \frac{20}{25}} \left[ \frac{1 - (K+1) \left(\frac{20}{25}\right)^5 + K \left(\frac{20}{25}\right)^6}{1 - \left(\frac{20}{25}\right)^6} \right] = 1.868 \text{ clientes}$$

Ahora se calcula el tiempo medio de estancia en el sistema (latencia) en función de  $L$  y la tasa efectiva de llegadas. Tener en cuenta que el Teorema de Little se calcula sobre la tasa efectiva de llegadas.

$$W = \frac{L}{\lambda'} = \frac{1.868}{(1 - 0.089)20} = 0.103 \text{ s}$$

**2.4 (0.5 puntos)** Dada la popularidad del servicio, el tráfico de entrada se incrementa, siendo ahora de 25 peticiones por segundo, con el tiempo entre llegadas distribuido de forma exponencial. Esta creciente popularidad hace que se quiera mejorar la disponibilidad del servicio, para lo cual se desea que a lo sumo el 1% de las peticiones puedan ser rechazadas. Asumiendo que no es posible cambiar el servidor ni incorporar nuevos servidores, calcular el tamaño mínimo de la cola para satisfacer este requisito.



Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....  
Apellidos ..... Nombre.....  
Ejercicio del día **13 de mayo de 2016**.....

En este nuevo escenario,  $\lambda = 25 \text{ p/s}$  y, dado que el servidor sigue siendo el mismo,  $\mu = 25 \text{ p/s}$ . Aplicando las ecuaciones de modelo M/M/1/K con  $\lambda = \mu$ , se tiene que la probabilidad de rechazo para un sistema con capacidad para K clientes será:

$$p_K = p_0 = \frac{1}{K+1} = 0,01$$

Despejando K:  $K = 99$  clientes, y, por tanto, el tamaño de la cola debería ser 98.

**Formulario:****Modelo M/M/1**

$$p_n = (1 - \rho)(\rho)^n$$

$$\rho = \lambda / \mu$$

$$L = \frac{\rho}{1 - \rho}$$

$$F_W(t) = 1 - e^{-(\mu - \lambda)t}$$

**Modelo M/M/c:**

$$p_n = \begin{cases} p_0 \frac{(\lambda/\mu)^n}{n!} & (n < c) \\ p_0 \frac{c^c}{c!} \left( \frac{\lambda}{c\mu} \right)^n & (n \geq c) \end{cases}$$

$$\rho = \frac{\lambda}{c\mu}$$

$$p_0 = \left[ \left( \sum_{n=0}^{c-1} \frac{(\lambda/\mu)^n}{n!} \right) + \frac{(\lambda/\mu)^c}{c!(1-\rho)} \right]^{-1}$$

$$P_q = \frac{p_c}{1-\rho} = E_c(c, \rho)$$

$$L = \frac{P_q \rho}{1-\rho} + c\rho$$

**Modelo M/M/c/c:**

$$p_n = p_0 \left( \frac{\lambda}{\mu} \right)^n \frac{1}{n!} \quad (0 \leq n \leq c)$$

$$p_0 = \left[ \sum_{n=0}^c \left( \frac{\lambda}{\mu} \right)^n \frac{1}{n!} \right]^{-1}$$

**Modelo M/G/1:**

$$L = \frac{\lambda^2 E[S^2]}{2(1-\rho)} + \rho$$

$$\rho = \lambda / \mu$$

**Modelo M/M/1/K:**

$$p_n = p_0 \left( \frac{\lambda}{\mu} \right)^n \quad (0 \leq n \leq K)$$

$$p_0 = \begin{cases} \left[ \frac{1 - \lambda/\mu}{1 - (\lambda/\mu)^{K+1}} \right] & (\lambda \neq \mu) \\ \frac{1}{K+1} & (\lambda = \mu) \end{cases}$$

$$\rho = \begin{cases} \frac{\lambda}{\mu} \left[ \frac{1 - (\lambda/\mu)^K}{1 - (\lambda/\mu)^{K+1}} \right] & (\lambda \neq \mu) \\ \frac{K}{K+1} & (\lambda = \mu) \end{cases}$$

$$L = \begin{cases} \frac{\lambda/\mu}{1 - \lambda/\mu} \left[ \frac{1 - (K+1)(\lambda/\mu)^K + K(\lambda/\mu)^{K+1}}{1 - (\lambda/\mu)^{K+1}} \right] & (\lambda \neq \mu) \\ \frac{K}{2} & (\lambda = \mu) \end{cases}$$

**Modelo M/M/1/M**

$$p_n = p_0 \binom{M}{n} n! \left( \frac{\lambda}{\mu} \right)^n = p_0 \frac{M!}{(M-n)!} \left( \frac{\lambda}{\mu} \right)^n$$

$$p_0 = \left[ \sum_{n=0}^M \frac{M!}{(M-n)!} \left( \frac{\lambda}{\mu} \right)^n \right]^{-1}$$

$$\rho = 1 - p_0$$

$$L = M - \frac{\lambda'}{\lambda} = M - \frac{\mu}{\lambda} \rho$$

**Modelo M/M/c/M**

$$p_n = \begin{cases} p_0 \binom{M}{n} \left( \frac{\lambda}{\mu} \right)^n & (0 \leq n < c) \\ p_0 \binom{M}{n} \frac{n!}{c^{n-c} c!} \left( \frac{\lambda}{\mu} \right)^n & (c \leq n < M) \end{cases}$$

$$p_0 = \left[ \sum_{n=0}^{c-1} \binom{M}{n} \left( \frac{\lambda}{\mu} \right)^n + \sum_{n=c}^M \binom{M}{n} \frac{n!}{c^{n-c} c!} \left( \frac{\lambda}{\mu} \right)^n \right]^{-1}$$

$$\rho = 1 - \sum_{n=0}^{c-1} p_n \frac{c-n}{c}$$

$$L = M - \frac{\lambda'}{\lambda} = M - \frac{c\mu}{\lambda} \rho$$



Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....  
 Apellidos ..... Nombre.....  
 Ejercicio del día **13 de mayo de 2016**.....

1.1 (1.25)	1.2 (1.25)	1.3 (1.25)	1.4 (1.25)	Total Parte III Teoría (5)

### PARTE III

(33,33% de la nota del examen)

### TEORÍA

**1.- (5 puntos).** Resuelve de modo claro y conciso las siguientes cuestiones:

**1.1. (1.25 puntos).** Definición de clúster de alta disponibilidad (*High Availability Cluster*) y clúster de alto rendimiento (*High Performance Cluster*). Indicar las diferencias entre ambos.

**1.2. (1.25 puntos).** ¿Para qué se utiliza el protocolo *Open Shortest Path First* (OSPF)? ¿Para qué se utilizan los mensajes OSPF Hello y LSA (*Link State Advertisements*)?



Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....  
 Apellidos ..... Nombre.....  
 Ejercicio del día **13 de mayo de 2016**.....

**1.3. (1.25 puntos).** Enumerar y explicar los diferentes elementos que deben definirse en la configuración de un balanceador de carga.

**1.4. (1.25 puntos).** Explicar en qué consisten las copias *on-line* en los sistemas de creación de copias de la información. ¿Qué implicaciones tienen en términos del RTO (*Recovery Time Objective*) frente a las copias *off-line*? Enumerar los diferentes mecanismos más habituales de menor a mayor fiabilidad.





Asignatura **SISTEMAS INFORMÁTICOS II**

Grupo.....

Apellidos..... Nombre.....

Ejercicio del día **13** de mayo de 2016.....

2.1 (1)	2.2 (1.5)	2.3(0.5)	2.4(1)	3.1 (1)	Total Parte III Problemas (5)

## PARTE III

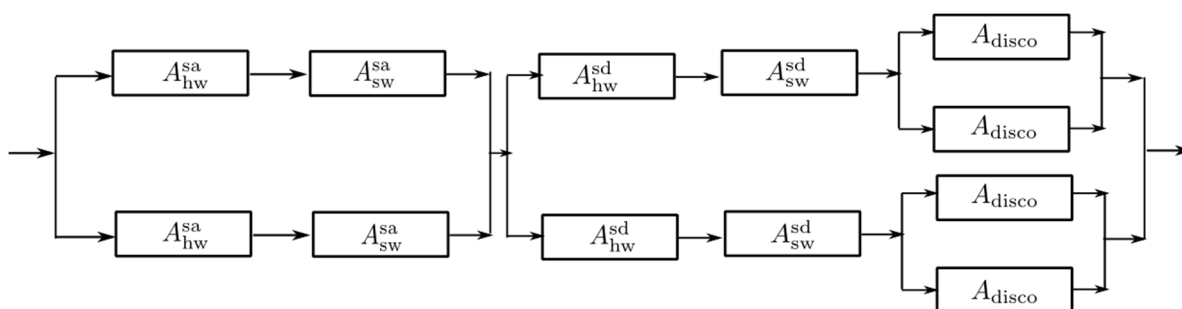
(33,33% de la nota del examen)

## PROBLEMAS

**2.- (4 puntos).** Una empresa proporciona un servicio que implica ejecutar una aplicación en un servidor de aplicaciones y acceder a cierto espacio de almacenamiento ubicado en un servidor de disco. El servidor de disco tiene **dos** discos duros configurados en RAID-1 (mirroring). Sin embargo, el software que gestiona el servidor de disco es experimental y se estima que falla **una vez cada 5 días** en promedio. **Lo mismo ocurre con el software del servidor de aplicaciones.** Cuando falla el software de alguno de los servidores, se tarda en promedio **1 hora** en detectar dicho fallo. Para corregirlo, es necesario reiniciar el servidor, lo que lleva en promedio **20 minutos**.

Para aumentar la disponibilidad del sistema, la empresa decide **duplicar servidores** y se adquiere un servidor de aplicaciones y un servidor de disco adicional, de **iguales características a los anteriores**. La empresa cuenta además con un contrato de mantenimiento que garantiza la reparación del hardware de cualquier servidor (servidor de aplicaciones o servidor de disco) o disco duro en **24 horas**. Finalmente, se sabe que el tiempo medio hasta el fallo del hardware de cualquier servidor (servidor de aplicaciones o servidor de disco) es de **5,000 horas**, mientras que el tiempo medio hasta el fallo de cada disco duro del servidor de disco es de **2,000 horas**.

**2.1 (1 punto)** Dibujar el diagrama de disponibilidad del sistema teniendo en cuenta que es necesario que al menos un servidor de cada tipo esté operativo para poder prestar el servicio, y para que un servidor esté operativo debe funcionar correctamente su componente software y hardware, así como el espacio del almacenamiento, en el caso del servidor de disco.



**2.2 (1.5 puntos)** Calcular la disponibilidad total del sistema e indicar si existe un punto simple de fallo (SPOF) en el mismo. Suponer que todos los fallos son independientes. Hacer los cálculos utilizando 4 decimales.

$$A_{hw}^{sa} = A_{hw}^{sd} = \frac{5000}{5000 + 24} = 0.9952$$



Asignatura **SISTEMAS INFORMÁTICOS II**

Grupo.....

Apellidos..... Nombre.....

Ejercicio del día **13 de mayo de 2016**.....

$$A_{sw}^{sa} = A_{sw}^{sd} = \frac{5 \cdot 24}{5 \cdot 24 + 1.3} = 0.9893$$

$$A_{disco} = \frac{2000}{2000 + 24} = 0.9881$$

$$A^{sa} = A_{hw}^{sa} \cdot A_{sw}^{sa} = 0.9952 \cdot 0.9893 = 0.9846$$

$$A^{sd} = A_{hw}^{sd} \cdot A_{sw}^{sd} (1 - (1 - A_{disco})^2) = 0.9952 \cdot 0.9893 \cdot (1 - 0.0119^2) = 0.9844$$

$$A = (1 - (1 - A^{sa})^2) \cdot (1 - (1 - A^{sd})^2) = 0.9998 \cdot 0.9998 = 0.9996$$

Todas las componentes están replicadas luego no existe SPOF.

**2.3 (0.5 puntos).** Suponiendo que todos los tiempos de vida son independientes y están distribuidos de forma exponencial, calcular el tiempo medio hasta el fallo (bien por fallo de software o de hardware) de un servidor de aplicaciones.

Como los tiempos de vida están distribuidos de forma exponencial su función tasa de fallos es constante e igual a  $1 / \text{MTTF}$ . Al tener las componentes colocadas en serie, las funciones tasa de fallo se suman. Esto nos permite calcular el MTTF del servidor de aplicaciones que es igual a:

$$\text{MTTF} = 1 / (1 / \text{MTTF}_{hw} + 1 / \text{MTTF}_{sw}) = 1 / (1 / 5000 + 1 / (5 \cdot 24)) = 117.19 \text{ horas.}$$

**2.4 (1 punto)** Actualmente cada pareja compuesta por un servidor de aplicaciones más servidor de disco se encuentra localizada en un edificio diferente. La distancia entre los dos edificios es de 1 km. Estos edificios se encuentran conectados mediante enlaces de alta velocidad privados (Ethernet + Fibra óptica). Indicar si dicha configuración será resistente a desastres, y si es así, a **qué tipo de desastres será tolerante**. Indicar también el **tipo de desastres a los que no será tolerante**. Suponiendo que los dos servidores de aplicaciones están configurados según un clúster Activo-Pasivo asimétrico de dos nodos, indicar **dos** mecanismos que se podrían emplear para transportar la señal Heartbeat.

El sistema será tolerante a desastres locales que ocurran en un único edificio (inundaciones incendios etc.). No será tolerante a desastres de áreas extensas que afecten a ambos edificios.

Para transportar la señal Heartbeat se podría utilizar un enlace Ethernet dedicado, o escrituras periódicas en áreas compartidas de almacenamiento localizadas en los servidores de disco.

**3.1- (1 puntos).** Determinar el conmutador raíz y el estado de cada uno de los puertos de los conmutadores de la figura como puerto raíz (RP), puerto designado (DP) o puerto alternativo (AP) tras aplicar el RSTP para eliminar los bucles de la red. El identificador asignado a cada conmutador aparece marcado con un número a su lado (menor identificador es mayor prioridad).

El conmutador raíz sería el que tiene mayor prioridad que es igual a menor ID. Luego sería el 1.

