



Asignatura..... **SISTEMAS INFORMÁTICOS II** Grupo.....
 Apellidos Nombre.....
 Ejercicio del día..... **19 de mayo de 2017.**.....

T1 (1)	T2 (1)	T3 (1)	T4 (1)	T5 (1)	Total Teoría (5)

PARTE I

(33,33% de la nota del examen)

1.- TEORÍA (5 puntos). Contesta de modo claro y conciso a las siguientes cuestiones.

1. **(1 punto)** Define en qué consiste la transparencia de datos y enumera las tres alternativas posibles en cuanto al tipo de acuerdo a tomar entre cliente y servidor.

Diapositiva 6 (definición) – Tema 1

Diapositiva 8 (3 alternativas) – Tema 1

2. **(1 punto)** Explica cuál es la funcionalidad del *Server Stub* en RPCs.

Diapositiva 18 – Tema 1

3. **(1 punto)** Compara los servicios web basados en SOAP frente a los servicios web basados en REST en términos de la complejidad de implementación del lado del cliente y el lado del servidor.

Diapositiva 37 – Tema 1

4. **(1 punto)** ¿Qué es *.Net Remoting*?

Diapositiva 58 – Tema 1

5. **(1 punto)** Enumera y explica brevemente los cuatro objetivos del Network time Protocol, NTP.

Diapositiva 90 – Tema 1



SISTEMAS INFORMÁTICOS II

Asignatura..... Grupo.....
 Apellidos..... Nombre.....
 Ejercicio del día **19 de mayo de 2017.**

2 (3)	3 (2)	Total Problemas (5)

2. PROBLEMA. Considerar los siguientes casos particulares de sistemas distribuidos:

- Servicio de solicitud de asistencia en carretera.** Se quiere desarrollar un sistema cliente-servidor que permita a los usuarios de una compañía aseguradora solicitar asistencia en carretera a través de una aplicación móvil. La aplicación enviará las coordenadas GPS con la ubicación del vehículo, el identificador del cliente y la hora de la solicitud. La aplicación recibirá una respuesta indicando si la petición ha sido o no aceptada y el la hora esperada de llegada del servicio técnico. Los clientes serán de arquitecturas heterogéneas. Además, se requiere que el sistema trabaje en tiempo real, por lo que el coste del procesamiento de los mensajes es crítico.
- API para empresa de venta online.** Una empresa de venta online desea desarrollar una API con el fin de facilitar el desarrollo de aplicaciones que mejoren su sistema (por ejemplo recomendaciones de productos asociados o productos similares). Los usuarios de la API pueden utilizar el servicio para consultar el carrito de la compra de diferentes usuarios, el perfil público de los usuarios del sistema (edad, género, profesión, etc.), publicar las recomendaciones de productos similares o productos relacionados, etc. En general, la complejidad de las operaciones y estructuras de datos es alta. La empresa cuenta además 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.

(3 puntos – 1.5 puntos por escenario) Para cada uno de ellos se pide elegir razonadamente el mecanismo de comunicación más adecuado entre los vistos en la parte de teoría de la asignatura (UDP, TCP, RPC, WS-SOAP, REST, CORBA, Java-RMI o Colas de Mensajes). Indicar así mismo si será necesario implementar algún **mecanismo adicional de traducción de datos**.

No se tendrán en cuenta respuestas sin justificación (cuantas más justificaciones, mayor la puntuación).

1. Servicio de solicitud de asistencia en carretera.

- El mecanismo más adecuado sería comunicación no orientada a conexión (UDP) debido a las siguientes razones:
 - Las interacciones serían acopladas en el tiempo (servidor y cliente conectados a la vez).
 - La funcionalidad es muy elemental, no merece la pena usar CORBA o servicios web. La información a enviar es muy pequeña, sólo las coordenadas GPS, el identificador de cliente y la marca temporal. La respuesta del servidor también es sencilla, indicando la aceptación o no del servicio y la hora estimada de llegada.
 - No merece la pena establecer una conexión para el envío de un único mensaje de petición y respuesta, por lo que se descarta TCP. Además, en caso de pérdida de un



SISTEMAS INFORMÁTICOS II

Asignatura..... Grupo.....
 Apellidos..... Nombre.....
 Ejercicio del día **19 de mayo de 2017.**

mensaje, éste se podría reenviar sin problema puesto que las interacciones son del tipo petición/respuesta e idempotentes.

- b. Habría que implementar algún tipo de funcionalidad adicional para garantizar la transparencia de datos al tratarse de sistemas heterogéneos, pero la información intercambiada es pequeña y sencilla de codificar.

2. API para empresa de venta online.

- a. El mecanismo más adecuado sería servicios web basados en REST debido a las siguientes razones:
- Descartamos UDP y TCP debido a que los clientes serán heterogéneos y se tiene una alta complejidad en los datos intercambiados (estos mecanismos no proporcionan transparencia de datos), así como en las operaciones a realizar.
 - 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, ya que es menos pesado que SOAP y se requiere evitar un tamaño de mensajes excesivamente grande.
- b. Dado que se tiene clientes heterogéneos, la representación JSON requeriría marshalling/unmarshalling de los datos únicamente a nivel de texto.

3. PROBLEMA (2 puntos). Considera el siguiente fichero WSDL asociado a un servicio web SOAP para la solicitud de información sobre un producto identificado por un id (string) y actualización del stock:

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

<definitions name="MyService"
targetNamespace=http://www.examples.com/wsd1/MyService.wsd1
xmlns="http://schemas.xmlsoap.org/wsd1/"
xmlns:soap=http://schemas.xmlsoap.org/wsd1/soap/
xmlns:tns="http://www.examples.com/wsd1/MyService.wsd1"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">

    <message name="productInfoRequest">
        <part name="idproduct" type="xsd:string"/>
    </message>

    <message name="productInfoResponse">
        <part name="description" type="xsd:string"/>
    </message>

    <message name="stockUpdateRequest">
```



SISTEMAS INFORMÁTICOS II

Asignatura..... Grupo.....

Apellidos Nombre.....

Ejercicio del día **19 de mayo de 2017.**

```

    <part name="idproduct" type="xsd:string"/>
    <part name="newstock" type="xsd:integer"/>
</message>
<message name="stockUpdateResponse">
    <part name="status" type="xsd:integer"/>
</message>

<portType name="Service_PortType"><operation name="getInfo">
    <input message="tns:productInfoRequest"/><output
message="tns:productInfoResponse"/>
</operation><operation name="updateStock">
    <input message="tns:stockUpdateRequest"/><output
message="tns:stockUpdateResponse"/>
</operation></portType>

<binding name="Service_Binding" type="tns:Service_PortType">
<soap:binding style="rpc" transport="http://schemas.xmlsoap.org/soap/http"/>
    <operation name="getInfo"><soap:operation soapAction="getInfo"/>
    <input><soap:body encoding="literal"/></input><output>
    <soap:body encoding="literal"/></output></operation>
    <operation name="updateStock"><soap:operation soapAction="updateStock"/>
    <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.examplessi2.com/Mitienda/" />
</port> </service>
</definitions>

```

3.1. (1.5 puntos). Indicar los cambios que habría que realizar para incluir una nueva función a la interfaz que a partir del identificador del producto (`idproducto`) y un número entero (`r`) devuelva un valor entero indicando el número de tiendas que disponen del producto en un radio de `r` kilómetros. Justificar los cambios realizados.

Habría que definir una nueva operación con nombre "stockRatio" (por ejemplo). Esta petición involucraría un mensaje de envío y otro de respuesta. El mensaje de petición "stockRatioRequest" y el mensaje de respuesta "stockRatioResponse" se definen como <message> como sigue:

```

<message name="stockRatioRequest">

```



SISTEMAS INFORMÁTICOS II
 Asignatura..... Grupo.....
 Apellidos Nombre.....
 Ejercicio del día **19 de mayo de 2017.**

```
<part name="idproduct" type="xsd:string"/>
<part name="ratio" type="xsd:integer"/>
</message>
<message name="stockRatioResponse">
  <part name="status" type="xsd:integer"/>
</message>
```

Hay que incluir la operación “stockRatio” con la secuencia del intercambio de mensajes en el campo PortType:

```
<PortType ...
<operation name="stockRatio">
  <input message="tns:stockRatioRequest"/>
  <output message="tns:stockRatioResponse"/>
</operation>
...
</portType>
```

Y finalmente hay que definir el modo en que los mensajes se transmiten sobre un protocolo de RPC, con extensiones específicas para SOAP.

```
<binding ...
  <operation name="stockRatio"><soap:operation soapAction="stockRatio"/>
  <input><soap:body encoding="literal"/></input><output>
  <soap:body encoding="literal"/></output></operation>
  ...
</binding>
```

3.2 (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”:

```
<soap:address location="http://www.examplessi2.com/Mitienda/" />
```



Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....
 Apellidos Nombre.....
 Ejercicio del día **19 de mayo de 2017**.....

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

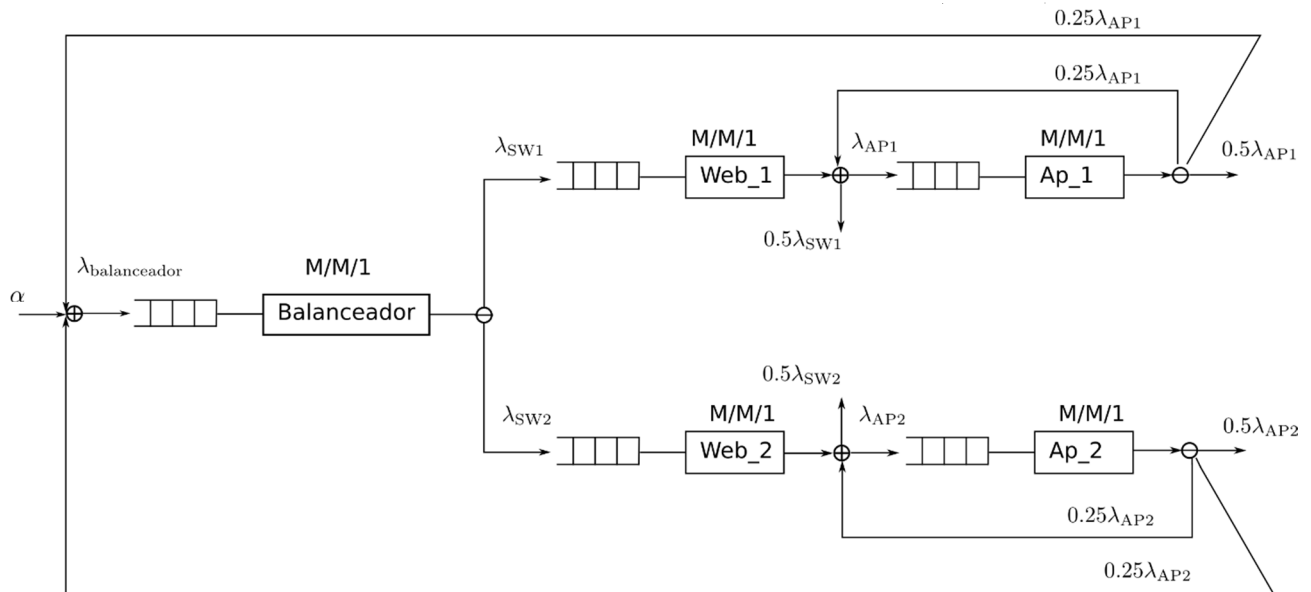
PARTE II

(33,33% de la nota del examen)

1. PROBLEMA (7 puntos). Una empresa presta un servicio a sus usuarios. Las peticiones de los usuarios son recibidas inicialmente por un balanceador de carga que reparte las peticiones **por igual** y de forma aleatoria entre **dos** servidores web de **iguales características**. Se estima que una petición recibida por uno de los servidores web accederá únicamente a contenido estático con una probabilidad del **50%**. Una vez procesadas estas peticiones se dan por terminadas. El **50%** restante necesitará acceder además a contenido dinámico. En este caso, tras ser procesadas en el servidor web, las peticiones han de invocar un servlet en un servidor de aplicaciones. La empresa cuenta con **dos** de estos servidores, de iguales características. Cada servidor de aplicaciones es accedido **exclusivamente** por uno de los servidores web. Se estima que con una probabilidad del **25%**, un servlet requeriría la invocación de otro servlet en el mismo servidor de aplicaciones, y con un **25%** de probabilidad requerirá la realización de otra petición al sistema. Estas peticiones son recibidas por el balanceador de carga. El resto de peticiones se dan por terminadas. Se ha estimado que, en promedio, la CPU del balanceador de carga tarda **100ms** en procesar una petición, mientras que la CPU de un servidor web tarda **200ms**. Finalmente, la CPU de un servidor de aplicaciones tarda en promedio **250ms** en procesar una petición. El servicio prestado recibe peticiones de los usuarios siguiendo un proceso Poisson con una media de **2** peticiones por segundo.

Suponer que todos los tiempos de servicio están distribuidos de forma exponencial y que los servidores tienen cola de espera de tamaño infinito.

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



Suponemos que los servidores están en estado estacionario. Este caso la tasa que tendrán a la salida será la misma que la que tienen a la entrada. Como la llegada de peticiones sigue un proceso Poisson, la probabilidad de salir del sistema es mayor que cero, los tiempos están distribuidos de forma exponencial y las colas tienen tamaño infinito, podemos usar el teorema de Jackson y ver el sistema como una red de colas abierta. Usando el teorema de Jackson, podemos describir cada servidor como un sistema M/M/1.

1.2 (2 puntos) Calcular la tasa de llegadas efectiva a la entrada de cada servidor.

Al estar en estado estacionario los sistemas a la salida tendrán la misma tasa que a la entrada, por lo que se ha de cumplir que:

$$\begin{aligned}
 \lambda_{\text{balanceador}} &= \alpha + 0.25\lambda_{\text{ap1}} + 0.25\lambda_{\text{ap2}} \\
 \lambda_{\text{sw1}} &= 0.5\lambda_{\text{balanceador}} = \lambda_{\text{sw2}} \\
 \lambda_{\text{ap1}} &= 0.5\lambda_{\text{sw1}} + 0.25\lambda_{\text{ap1}} \\
 \lambda_{\text{ap2}} &= 0.5\lambda_{\text{sw2}} + 0.25\lambda_{\text{ap2}}
 \end{aligned}$$

De donde

$$\begin{aligned}
 \lambda_{\text{ap1}} &= \frac{0.5}{0.75} \lambda_{\text{sw1}} = \lambda_{\text{ap2}} \\
 \lambda_{\text{balanceador}} &= \alpha + 0.5\lambda_{\text{ap1}} = \alpha + 0.5 \cdot \frac{0.5}{0.75} \lambda_{\text{sw1}} \\
 \lambda_{\text{sw1}} &= 0.5 \left(\alpha + 0.5 \cdot \frac{0.5}{0.75} \lambda_{\text{sw1}} \right) = 0.5\alpha + \frac{0.5^3}{0.75} \lambda_{\text{sw1}} \\
 \lambda_{\text{sw1}} &= \frac{0.5\alpha}{1 - \frac{0.5^3}{0.75}} = \frac{1}{1 - \frac{0.125}{0.75}} = 1.2 \text{ p/s} = \lambda_{\text{sw2}} \\
 \lambda_{\text{balanceador}} &= 2 + \frac{0.5^2}{0.75} \lambda_{\text{sw1}} = 2 + \frac{0.5^2}{0.75} 1.2 = 2.4 \text{ p/s}
 \end{aligned}$$



Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....
 Apellidos..... Nombre.....
 Ejercicio del día **19 de mayo de 2017**.....

$$\lambda_{ap1} = \frac{0.5}{0.75} \lambda_{sw1} = \frac{0.5}{0.75} 1.2 = 0.8 \text{ p/s}$$

Las capacidades de los servidores son:

$$\begin{aligned} \mu_{balanceador} &= 10 \text{ p/s} \\ \mu_{sw1} &= \mu_{sw2} = 5 \text{ p/s} \\ \mu_{ap1} &= \mu_{ap2} = 4 \text{ p/s} \end{aligned}$$

Luego se cumple que los sistemas estén en estado estacionario.

1.3 (1 puntos) Calcular el número medio de peticiones en el sistema total.

El nº medio de peticiones en el sistema será la suma del nº medio de peticiones en cada servidor. Empleando el teorema de Jackson estos números vendrán dados por las fórmulas del modelo M/M/1.

$$\rho_{balanceador} = \frac{2.4}{10} = 0.24$$

$$\rho_{sw1} = \rho_{sw2} = \frac{1.2}{5} = 0.24$$

$$\rho_{ap1} = \rho_{ap2} = \frac{0.8}{4} = 0.2$$

$$L_{total} = L_{balanceador} + L_{sw1} + L_{sw2} + L_{ap1} + L_{ap2}$$

$$L_{balanceador} = \frac{\rho_{balanceador}}{1 - \rho_{balanceador}} = \frac{0.24}{1 - 0.24} = 0.3158 = L_{sw1} = L_{sw2}$$

$$L_{ap1} = \frac{\rho_{ap1}}{1 - \rho_{ap1}} = \frac{0.2}{1 - 0.2} = \frac{0.2}{0.8} = 0.25 = L_{ap2}$$

$$L_{total} = 3 \cdot 0.3158 + 2 \cdot 0.25 = 1.447 \text{ clientes}$$

1.4 (1 puntos) Calcular justificadamente el tiempo medio de respuesta de todo el sistema.

Calculamos el número total de clientes en todo el sistema y aplicamos Little.

$$W_T = \frac{L_T}{\alpha} = \frac{1.447}{2} = 0.7234 \text{ segundos}$$

1.5 (1 punto) Calcular justificadamente el tiempo medio de respuesta de las peticiones que no necesitan invocar ningún servlet en el sistema.

Esta es la suma de los tiempos de estancia en el balanceador de carga y el servidor web.

Asignatura **SISTEMAS INFORMÁTICOS II**

Grupo.....

Apellidos.....

Nombre.....

Ejercicio del día **19 de mayo de 2017**

$$W_{balanceador} = \frac{L_{balanceador}}{\lambda_{balanceador}} = \frac{0.3158}{2.4} = 0.1316 \text{ segundos}$$

$$W_{sw1} = \frac{L_{sw1}}{\lambda_{sw1}} = \frac{0.3158}{1.2} = 0.2632 \text{ segundos}$$

$$W = W_{balanceador} + W_{sw1} = 0.1316 + 0.2632 = 0.3948 \text{ segundos}$$

1.6 (1 punto) Calcular la probabilidad de que el número de peticiones en la cola del balanceador de carga exceda 6.

Mediante el teorema de Jackson podemos usar las fórmulas del modelo M/M/1.

$$p(n_q > 6) = p(n > 7) = \sum_{n=8}^{\infty} (1-\rho)\rho^n = (1-\rho) \sum_{n=8}^{\infty} \rho^n = \frac{(1-\rho)\rho^8}{(1-\rho)} = \rho_{balanceador}^8 = 1.1 \cdot 10^{-5}$$

2. PROBLEMA (3 puntos) Una empresa tiene un servidor que cuenta con una cola de espera de tamaño infinito. Tras monitorizar el servidor se ha observado que el tiempo medio de servicio es de **200ms** y su varianza es igual a **50ms²**. El servidor recibe tráfico que sigue un proceso Poisson con una tasa de 4 peticiones por segundo.

2.1 (1 punto) Justificar el uso de un modelo de colas para analizar el sistema descrito.

El sistema sería M/G/1 debido a que el tiempo entre llegadas está distribuido de forma exponencial (proceso Poisson), el tamaño de la cola es infinito, solo hay un servidor y no se conoce la distribución del tiempo de servicio.

2.2 (1 punto) Calcular el tiempo medio de respuesta del sistema.

$$\begin{aligned} T_s &= \frac{1}{\lambda} = 5 \text{ p/s} \\ \lambda &= 4 \text{ p/s} \\ \rho &= \frac{4}{5} = 0.8 \\ E[S^2] &= 0.050 + 0.200^2 = 0.09 \\ L &= \frac{\lambda^2 E[S^2]}{2(1-\rho)} + \rho = 16 \cdot \frac{0.09}{2(1-0.8)} + 0.8 = 4.4 \\ W &= \frac{L}{\lambda} = \frac{4.4}{4} = 1.1 \text{ s} \end{aligned}$$

2.3 (1 punto) Calcular el número medio de peticiones en cola.



Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....
 Apellidos Nombre.....
 Ejercicio del día **19 de mayo de 2017**.....

$$L_q = \lambda W_q = \lambda(W - E[S]) = 4(1.1 - 0.2) = 3.6 \text{ clientes}$$



Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....
 Apellidos Nombre.....
 Ejercicio del día **19 de mayo de 2017**.....

Formulario:

Modelo M/M/1

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

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

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

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

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

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

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:

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 **19 de mayo de 2017**.....

1.1 (1.25)	1.2 (1.25)	1.3 (1.25)	1.4 (1.25)	Total Parte III Teoria (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). Describe en qué consiste un enlace EtherChannel, sus ventajas, sus inconvenientes, y su tipo de redundancia según la nomenclatura vista en clase.

1.2. (1.25 puntos). Cluster Activo-Pasivo cruzado o simétrico de dos nodos. Describe cómo funciona, que ventajas aparentes tiene, y que inconvenientes.



Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....
 Apellidos Nombre.....
 Ejercicio del día **19 de mayo de 2017**.....

1.3. (1.25 puntos). Indica en qué consiste el mecanismo de delayed binding en los balanceadores de carga y su principal utilidad.

1.4. (1.25 puntos). Copias remotas en servidores de disco. Tipos y características.



Asignatura **SISTEMAS INFORMÁTICOS II** Grupo.....
 Apellidos Nombre.....
 Ejercicio del día **19 de mayo de 2017**.....

2 (4)	3 (1)	Total Parte III Problemas (5)

PARTE III

(33,33% de la nota del examen)

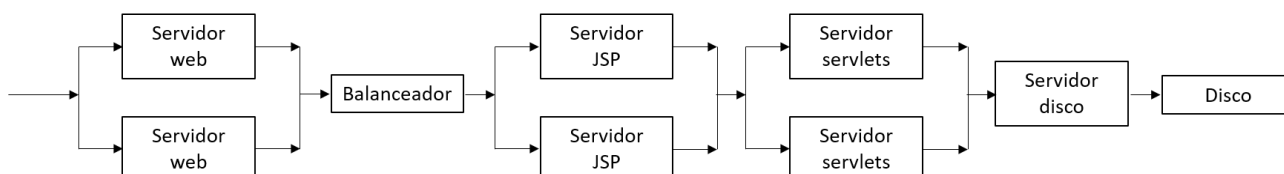
PROBLEMAS

2. PROBLEMA (4 puntos). Para ofrecer un determinado servicio, una empresa cuenta con los siguientes servidores para resolver las peticiones que recibe:

- Un servidor web, que entrega al cliente páginas estáticas e imágenes. El sistema posee dos de estos servidores de igual funcionalidad.
- Un distribuidor de carga, que reparte las peticiones no-estáticas recibidas por los servidores web entre los servidores JSP del sistema. El sistema posee uno sólo.
- Un servidor JSP, que resuelve las páginas JSP. El sistema posee dos de estos servidores, de igual funcionalidad. El distribuidor de carga puede enviar indistintamente sus peticiones a cualquiera de ellos.
- Un servidor de servlets para la ejecución de servlets cuando sea necesario. El sistema posee dos de estos servidores, de igual funcionalidad. Los servidores JSP pueden enviar indistintamente sus peticiones a cualquiera de los servidores servlets.
- Un servidor de disco, al cual acceden los servlets para recuperar los datos que necesitan para resolver las peticiones. El sistema posee uno de estos servidores.
- Un disco de 1TB al que se accede a través del servidor.

Es necesario, por tanto, que esté disponible al menos un elemento de cada una de las clases citadas anteriormente para que el sistema completo funcione.

2.1 (0.5 puntos) Dibujar el diagrama de disponibilidad del sistema teniendo en cuenta que es necesario que al menos uno de los servidores de cada tipo esté operativo para poder prestar el servicio.



2.2 (1 punto) Suponer fallos independientes. Sea el tiempo medio entre fallos de un servidor web de **800 h**, el tiempo medio hasta el fallo del distribuidor de carga de **1000 h**, el tiempo medio hasta el fallo de un servidor JSP de **2000 h**, el tiempo medio hasta el fallo de un servidor servlet de **2000 h** y el tiempo medio entre fallos del servidor de disco de **300h**. El disco tiene un tiempo medio entre fallos de **5000 h**. Suponer que tiene contratado un servicio de mantenimiento que garantiza la reparación de cualquier equipo en un tiempo medio de **48h**. Se pide calcular justificadamente la disponibilidad total del sistema. **Expresar los resultados redondeando a 4 decimales.**

Se calcula la disponibilidad para cada uno de los servidores teniendo en cuenta las métricas proporcionadas:

Servidor	MTTF	MTBF	MTTR	Disponibilidad
Web	-	800 h	48 h	$A_{SW} = \frac{MTBF - MTTR}{MTBF} = 0.94$
Balanceador	1000 h	-	48 h	$A_B = \frac{MTTF}{MTTF + MTTR} = 0.9542$
JSP	2000 h	-	48 h	$A_{JSP} = \frac{MTTF}{MTTF + MTTR} = 0.9766$
Servlets	2000 h	-	48 h	$A_{SERVLET} = \frac{MTTF}{MTTF + MTTR} = 0.9766$
Servidor de disco	-	300 h	48 h	$A_{SD} = \frac{MTBF - MTTR}{MTBF} = 0.84$
Disco	-	5000 h	48 h	$A_D = \frac{MTBF - MTTR}{MTBF} = 0.9904$

La disponibilidad total del sistema será:

$$A_T = (1 - (1 - A_{SW})^2) \cdot A_B \cdot (1 - (1 - A_{JSP})^2) \cdot (1 - (1 - A_{SERVLET})^2) \cdot A_{SD} \cdot A_D = 0.7902$$

2.3 (1 punto) Tras un tiempo en funcionamiento, se decide incorporar en el sistema servidores caché que atenderán directamente las peticiones entrantes. El tiempo hasta fallo de un servidor caché es de **200 h** y serán reparados bajo las mismas condiciones de mantenimiento anteriores. Determinar el número de servidores caché necesarios para garantizar una disponibilidad total del sistema de al menos el **70%** teniendo en cuenta que cualquier servidor caché puede atender cualquiera de las peticiones entrantes y que es necesario que al menos uno de los servidores de cada tipo esté operativo para poder prestar el servicio. Expresar los resultados redondeando a 4 decimales.

Del servidor caché nos dan la siguiente información:

MTTF = 200 h

MTTR = 48 h

Por tanto, la disponibilidad de un servidor caché será: $A_C = \frac{MTTF}{MTTF + MTTR} = \frac{200}{200 + 48} = 0.8065$

La disponibilidad del nuevo sistema vendrá dada por:

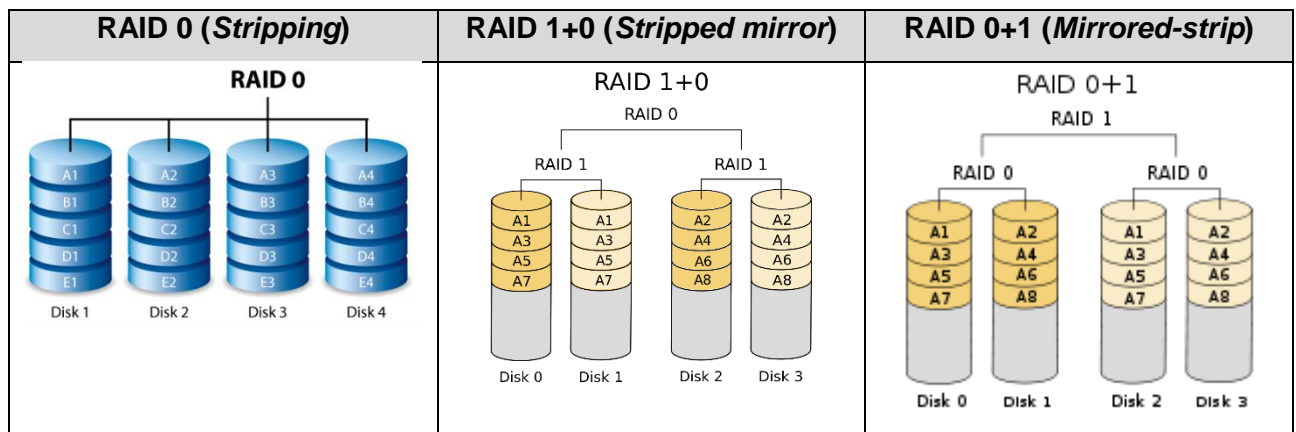
$$\tilde{A} = (1 - (1 - A_C)^N) \cdot A_T$$

donde N es el número de servidores caché y A_T la disponibilidad del sistema inicial calculada en el apartado 2.2. Por tanto, ha de verificarse $\tilde{A} \geq 0.7$:

$$(1 - A_C)^N \leq 1 - \frac{0.7}{A_T} \Leftrightarrow N \geq \frac{\log\left(1 - \frac{0.7}{A_T}\right)}{\log(1 - A_C)} = 1.3213$$

Por tanto, harán falta 2 servidores caché para satisfacer una disponibilidad igual o superior al 70% en el sistema total.

2.4 (1.5 puntos) La empresa decide comprar otros tres discos de 1TB idénticos al que ya disponía e incluidos en el contrato de mantenimiento. Se te pide analizar la capacidad de almacenamiento efectivo (cantidad máxima de información no redundante que el sistema es capaz de almacenar) y calcular la disponibilidad de tres configuraciones distintas para los cuatro discos idénticos disponibles:



En el caso de RAID 0+1 es necesario que los dos discos que forman un *strip* (elementos en RAID 0) estén disponibles para garantizar la disponibilidad de todo el *strip*.

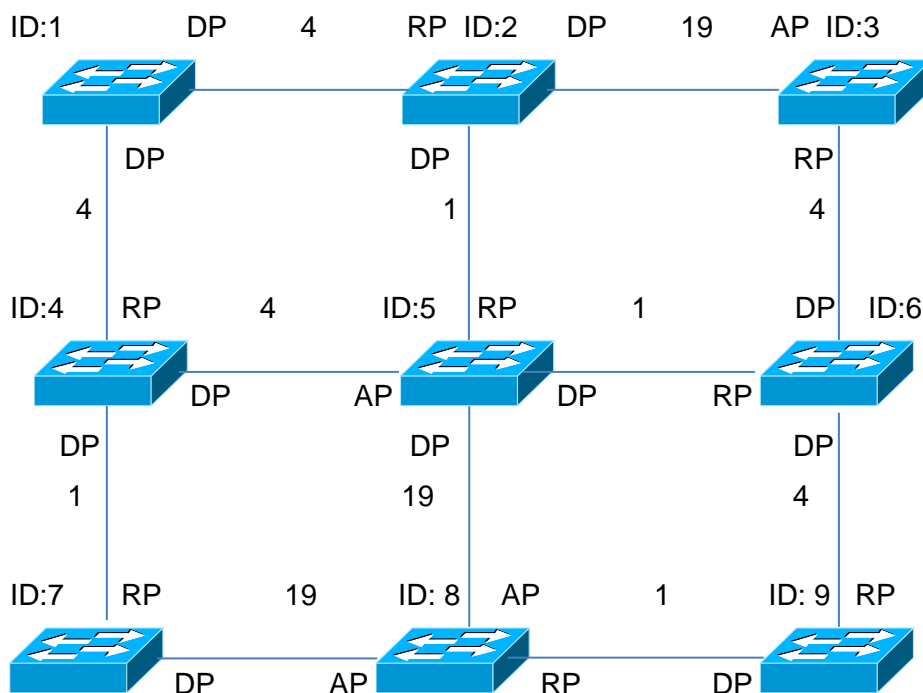
Completar la siguiente tabla con la información correspondiente.

Configuración	Capacidad de almacenamiento efectivo	Diagrama disponibilidad	Disponibilidad
RAID 0	4 TB	<pre> graph LR In(()) --> D1[D1] D1 --> D2[D2] D2 --> D3[D3] D3 --> D4[D4] D4 --> Out(()) </pre>	$(A_D)^4 = (0.9904)^4 = 0.9621$
RAID 1+0	2 TB	<pre> graph LR In(()) --> Split(()) Split --> D0[D0] Split --> D1[D1] D0 --> Join1(()) D1 --> Join1 Join1 --> D2[D2] D2 --> D3[D3] D3 --> Out(()) </pre>	$(1 - (1 - A_D)^2)^2 = 0.9998$
RAID 0+1	2 TB	<pre> graph LR In(()) --> Split(()) Split --> D0[D0] Split --> D2[D2] D0 --> D1[D1] D2 --> D3[D3] D1 --> Join1(()) D3 --> Join1 Join1 --> Out(()) </pre>	$(1 - (1 - (A_D)^2)^2) = 0.9996$

3. PROBLEMA (1 punto).

3.1 (0.5 puntos). Determinar el conmutador raíz y el estado de cada uno de los puertos de los conmutadores como puerto raíz (RP), puerto designado (DP) o puerto alternativo (AP) tras aplicar el RSTP para eliminar los bucles de la red mostrada más abajo. El identificador asignado a cada conmutador aparece marcado con un número a su lado (menor identificador es mayor prioridad).

Solución:



3.2 (0.5 puntos) Al detectarse un fallo en el switch 1, indicar justificadamente qué Root Ports cambiarían tras este fallo.

Al fallar el switch raíz (Switch 1), el switch raíz pasaría a ser el switch alternativo (Switch 2), y por tanto, ya no habría root switch en el switch 2. De todos los caminos mínimos al switch raíz en la configuración inicial (aparatado 3.1), únicamente los caminos desde los switches 4 y 7 NO pasaban por el switch 2 y, por tanto, los Root Ports de estos switches podrían ser los que cambien tras el fallo. Es suficiente con cambiar el RP del switch 4.