

## Índice

Ejemplo examen → Temas 1, 2, 3 Sistemas Distribuidos .....	1
Pregunta 1. (repetida 3 veces) .....	2
Pregunta 2. (repetida 4 veces) .....	2
Pregunta 3. (repetida 4 veces) .....	3
Pregunta 4. (repetida 4 veces) .....	4
Pregunta 5. (repetida 3 veces) .....	4
Pregunta 6. (repetida 2 veces) .....	5
Pregunta 7. (repetida 2 veces) .....	5
Pregunta 8. (repetida 2 veces) .....	5
Pregunta 9.....	6
EXAMEN GRUPO ARA 2014-11-25 .....	6
PREGUNTA 1 .....	6
EXAMEN JULIO 2013/14.....	7
PREGUNTA 1 .....	7
PREGUNTA 2 .....	8
Noviembre 2013 A .....	9
Pregunta 1.....	9
Pregunta 2.....	9
Pregunta 4.....	9
Noviembre 2013 B .....	9
Pregunta 3.....	9
Pregunta 4.....	9
TIPO TEST .....	10

*\*El presente documento está realizado a través de una recopilación de documentos de diferentes fuentes. Debe servir como complemento y apoyo para el estudiante, en ningún caso ha de ser empleado para otros fines. Este documento es de libre circulación, por lo que queda prohibida la comercialización del mismo.*

## Ejemplo examen → Temas 1, 2, 3 Sistemas Distribuidos

### Pregunta 1. (repetida 3 veces)

**Explica el paradigma de computación distribuida denominado MOM y sus tipos (1).**

**Enumera de forma general qué diferencias y coincidencias hay entre MOM y SOA (0,5).**

**Compara las funcionalidades del intermediario en cada una de ellas (0,5). ¿Qué mecanismo de comunicación distribuida (RPC, RMI, ORB, Servicios Web) utilizarías para implementar MOM. Justifica tu respuesta (0,5).**

- El modelo MOM (Middleware Orientado a Mensajes) es una evolución del sistema de paso de mensajes convencionales. Este permite que la comunicación entre el emisor y el receptor se produzca de una manera completamente desacoplada. Se utiliza en sistemas asíncronos, es decir, donde el emisor no necesita una respuesta inmediata del receptor tras enviar el mensaje y puede continuar con su funcionamiento. Existe un intermediario, conocido como proceso intermediario, un emisor, proceso emisor y un receptor, proceso consumidor. El proceso intermediario es el encargado de almacenar los mensajes del emisor, generalmente este es un middleware que incluye ciertos servicios para el tratamiento de los mensajes. Puede poseer las siguientes características: gestión de prioridades de mensajes, temporizadores para la gestión de mensajes, gestión de formatos de mensajes, gestión de seguridad, gestión de persistencia de los mensajes.

- Es un modelo adecuado para aplicaciones y sistemas donde se quiere establecer un único punto de entrada a una comunicación asíncrona de los mismos.

Existen dos variantes de este modelo:

- Punto a punto (1:1) → cada mensaje enviado por el emisor únicamente será procesado por un proceso consumidor. El agente intermediario obtiene el mensaje, lo procesa y el mensaje es eliminado del intermediario.
- Publicación/Suscripción (1:M) → un mensaje publicado por un emisor será procesado por todos los agentes consumidores que se hayan suscrito a dicho proceso intermediario.

- Las principales diferencias entre SOA y MOM son:

- MOM es usado en comunicación asíncrona, mientras que SOA es petición/respuesta (síncrona).
- MOM consigue desacoplar el escenario completo gracias al proceso intermediario, en SOA únicamente se desacopla la localización de los servicios.
- MOM puede ser usado para la implementación de la arquitectura SOA.

- Debido a que MOM se apoya en un proceso intermediario para la gestión de los mensajes, una comunicación adecuada sería ORB. ORB dispone de un agente intermediario para la gestión de objetos ofreciéndonos la abstracción del acceso a objetos heterogéneos. Por otro lado, la arquitectura de servicios web también nos podría dar un enfoque MOM con una mayor interoperabilidad a más alto nivel.

### Pregunta 2. (repetida 4 veces)

**Explica y describe la funcionalidad dentro de la tecnología de Servicios Web de WSDL, UDDI y SOAP. Explica y describe las partes de un documento WSDL. ¿Qué relación hay entre WSDL, UDDI y SOAP?**

- Los servicios web son procesos o funciones de negocio significativas, con una interfaz bien definida y accesible a través de Internet, basada en el intercambio de mensajes en XML, pudiendo ser estos procesos o funciones combinados entre sí.

- WSDL (Web Service Description Language) es un formato XML que es utilizado para describir los mensajes SOAP que definen un servicio web en particular. Es un IDL (Interface Definition Language) para la comunicación con el servicio y definir de manera abstracta una interfaz

## Parcial 1 SD (II - UA)

publica del mismo. Describe principalmente el protocolo a utilizar y la implementación concreta del servicio.

### • Anatomía WSDL:

- <definitions> → definición de uno o más servicios.
- <messages> y <portType> → operaciones que provee el servicio.
- <binding> → como se invocan las operaciones.
- <service> → donde se ubica el servicio.
- <documentation> → puede contener información del servicio para el usuario.

### • Estructura WSDL:

- Types → definiciones de los tipos de datos para describir los mensajes intercambiados.
- Message → definición abstracta de los datos que se transmiten. Un mensaje es dividido en partes lógicas, cada una asociada a una definición de sistema de tipos.
- PortType → operaciones abstractas que hacen referencia a un mensaje de entrada y uno de salida.
- Binding → especifica el protocolo concreto y las especificaciones del formato de datos, de los mensajes definidos por un portType concreto.
- Service → unir un conjunto de puertos relacionados.
- Port → dirección para un Binding, para definir un único nodo de comunicación.

- UDDI (Universal Description, Discovery and Integration) es un mecanismo que nos permite localizar saber con quién comunicarse para acceder a un servicio web concreto y donde hacerlo. Su objetivo es ser accedido por los mensajes SOAP y dar paso a documentos WSDL. Posee dos categorías de API:

- De publicación → para que los proveedores de servicios se registren (ellos y sus servicios).
- De consulta → permite al subscriptor buscar los servicios disponibles y obtenerlos una vez localizados.

- SOAP (Simple Object Access Protocol) es un protocolo ligero basado en XML para el intercambio de la información en un entorno descentralizado y distribuido, digamos que es el protocolo sobre el que hablan los servicios. Este protocolo deriva de XML-RPC. Nos facilita la comunicación entre objetos de cualquier tipo, sobre cualquier lenguaje y plataforma. Permite la comunicación máquina a máquina débilmente acoplada y el intercambio de mensajes a través de firewalls. Consta de las partes: Sobre o envoltura, reglas de codificación y representación RPC/Document.

- La relación entre las tecnologías WSDL, SOAP y UDDI radica en que son los estándares o protocolos que permiten la comunicación entre los negocios a nivel de aplicaciones. Constando de las partes de: Publicación, Búsqueda, Descubrimiento y Consumo.

## Pregunta 3. (repetida 4 veces)

**Explica qué es un servicio de nombres y qué es un servicio de directorios y cuáles son sus principales características. De forma breve, clasifica y separa DNS, LDAP y UDDI en servicios de nombre o directorio y si son de propósito general o específico. Justifica la respuesta.**

- Un servicio de nombres provee a los clientes información sobre elementos de un sistema distribuido de manera legible para el ser humano, con el fin de identificar los mismo. Es usado para referencia recursos y usuarios, así como comunicar y compartir recursos. Este servicio almacena colecciones de pares <nombre, atributo> y busca atributos a partir de nombres. Podemos hacer un símil con las paginas blancas.

Las características principales de los servicios de nombre son:

## Parcial 1 SD (II - UA)

- Utilizan el paradigma cliente/servidor.
- Es un servicio independiente fácilmente escalable.
- Independencia de su ubicación.
- Alta disponibilidad.
- La información se almacena jerárquicamente.
- Débil consistencia de replicación.
- Flexibilidad.
- BD optimizada: orientada a la lectura de información, datos de una entrada en un único registro, no necesita transacciones y tampoco bloqueos.

- Un servicio de directorio provee a los clientes información sobre objetos que satisfacen una determinada descripción. Es similar al servicio de nombre, pero este nos permite buscar nombres a través de atributos. Podemos hacer un símil con las páginas amarillas.

Además de las características de los servicios de nombre, los servicios de directorio también poseen las siguientes:

- Información acerca de objetos relacionados (recursos de red, personas.).
- Refuerza la seguridad para proteger a los objetos de intrusos.

- Servicios de nombres (páginas blancas): DNS (específico)

• DNS es un servicio de nombre ya que nos permite buscar un atributo a través de un nombre, y específico debido a que nos permite crear un espacio y extender la funcionalidad, guardando la información en la estructura que se crea.

- Servicio de directorio (páginas amarillas): UDDI (específico) y LDAP (general).

- UDDI es un servicio de directorio ya que nos permite realizar búsquedas por una descripción dada, además es específico por lo mismo que lo es el DNS.
- LDAP es un servicio de directorio ya que nos permite realizar búsquedas por atributos y/o distinta información, además es de propósito general ya que solo nos permite crear un espacio de nombres.

## Pregunta 4. (repetida 4 veces)

**Explica el modelo arquitectónico denominado middleware orientado a mensajes (MOM). Enumera y describe sus elementos. ¿Cuáles son las principales similitudes y diferencias entre MOM y el modelo cliente-servidor convencional?**

- Mientras que el modelo C/S actúa sobre el paradigma de paso de mensajes, la arquitectura MOM es una elaboración más extensa del mismo. En el modelo MOM nos encontramos con un proceso intermediario, mientras que en la arquitectura C/S solo existen dos procesos (cliente y servidor). Ambos nos permiten la abstracción del acceso a recursos de red.

## Pregunta 5. (repetida 3 veces)

**Uno de los principales problemas en la comunicación entre entidades de un sistema distribuido es la representación de la información. Enumera y describe el funcionamiento de las tres posibles tácticas que tratan de resolver esta problemática.**

- Debido a la heterogeneidad de la información (diferentes representaciones de estas) surgen las siguientes tácticas para resolver esta problemática:

- Emisor se adapta a la representación del receptor: esta es la técnica menos utilizada, debido a que tiene una gran carga sobre la red debido a la traducción de los mensajes.
- Receptor adapta la representación del emisor: esta técnica, aunque es más usada que la anterior, tampoco es ampliamente aceptada. El mensaje que recibe el receptor debe contener información sobre cómo tratar dicha información y tiene la misma problemática que la anterior táctica.
- Representación externa común: esta técnica es la más utilizada actualmente, ambos se comunican de la misma manera, “usando el mismo lenguaje”.

### Pregunta 6. (repetida 2 veces)

**Comenta brevemente el modelo CGI indicando sus ventajas y desventajas. Justifica su aparición partiendo del modelo Web básico**

- CGI (Common Gateway Interface) especifica una manera de transferir datos entre un cliente y un programa alojado en un servidor. El servidor web pasara las solicitudes que reciba del cliente a un programa externo (CGI) y la salida de dicho programa será enviada al cliente.

Sus principales ventajas son:

- Capacidad de respuesta dinámica.
- Libertad de elección del lenguaje de programación.

Y sus principales desventajas:

- No existe relación entre el programa CGI y el servidor web, por lo que no existe un control sobre la ejecución de este, ni del resultado.
- Se crea una nueva instanciación del CGI por cada solicitud produciendo una sobrecarga de recursos.

La principal justificación de su aparición es la necesidad de ofrecer contenido web dinámico, capaz de interactuar con el usuario a través de una aplicación web sobre el protocolo HTTP.

### Pregunta 7. (repetida 2 veces)

**Justifica la gestión distribuida de LDAP, define las ventajas sobre otros sistemas de información tradicionales y enumera y describe los motivos que pueden llevar a esta gestión distribuida. ¿Qué técnica es utilizada para relacionar las diferentes partes del espacio de nombres cuando es distribuido?**

- La justificación de porque la gestión es distribuida es que de esa forma puede dividirse en subárboles por motivos de rendimiento, localización geográfica (q ayuda al rendimiento, porque esta más cerca y a los contenidos) y por cuestiones administrativas.

A diferencia del sistema tradicional (x500):

- LDAP utiliza TCP/IP en lugar de protocolos OSI
- El modelo funcional de LDAP es más simple y ha eliminado opciones raramente utilizadas en X.500. LDAP es más fácil de comprender e implementar.
- LDAP representa la información mediante cadenas de caracteres en lugar de complicadas estructuras ASN.1.

- La técnica para referenciar las diferentes partes del espacio de nombres es el uso del objeto ObjectClass:referral. Atributo obligatorio para almacenar la url de acceso a los subárboles.

### Pregunta 8. (repetida 2 veces)

**Explicar arquitectura SOA**

- La arquitectura SOA (Service Oriented Architecture) es un paradigma de arquitectura para desarrollar sistemas distribuidos. Permite la abstracción de acceso a actividades de negocio, conocidas como servicios.

Sus principales características son:

- Localización, descubrimiento y publicación.
- Interoperabilidad.
- Composición.
- Autonomía y autocontenidos.
- Reusabilidad y desacoplamiento.
- Contrato bien definido.
- Sin estado

Sus principales componentes son un proveedor de servicios, un consumidor y un registro de los mismos.

## Parcial 1 SD (II - UA)

Los pasos para su utilización serían: Publicación, búsqueda, descubrimiento y consumo.

Las herramientas utilizadas en este paradigma serían los servicios web, JINI y UPnP entre otros.

### Pregunta 9.

**Explicar los Servlets, sus características y compararlos con los CGI y las paginas dinámicas.**

- Los Servlets son una tecnología utilizada para la ampliación del modelo web básico desde el lado del servidor. El uso más común de los Servlets es generar webs de forma dinámica a partir de parámetros de una petición enviada por un cliente.

Sus principales características son:

- Portabilidad → solo precisa de JVM.
- Rendimiento → única instanciación.
- Sesión → mantiene información en diferentes conexiones al servlet.
- Software Distribuido → permite comunicación entre servlets.
- Multithread → permite concurrencia

## EXAMEN GRUPO ARA 2014-11-25

### PREGUNTA 1

**Explica el paradigma de computación distribuida denominado MOM y sus tipos (1). Enumera de forma general qué diferencias y coincidencias hay entre MOM y SOA (0,5). Compara las funcionalidades del intermediario en cada una de ellas (0,5). ¿Qué mecanismo de comunicación distribuida (RPC, RMI, ORB, Servicios Web) utilizarías para implementar MOM? Justifica tu respuesta (0,5).**

MOM es la evolución del sistema de mensajes básico. Existe un sistema de mensajes intermediario que redirige todos los mensajes que se envían dentro de este sistema. Los mensajes se envían de forma asíncrona y desacoplada. El funcionamiento se basa en que el emisor envía un mensaje a la red que llega al sistema de mensajes. El sistema de mensajes redirigirá el mensaje a la cola del receptor correspondiente. Cuando el mensaje haya sido enviado, la cola quedará de nuevo vacía para que el receptor pueda seguir realizando otras tareas.

Hay dos tipos de MOM:

- Punto a punto: el sistema de mensajes redirigirá el mensaje a la cola de recepción directamente al receptor. Este tipo de MOM ofrece una abstracción que el sistema básico no ofrece.
- Publicación o suscripción: si el tipo utilizado es éste, cada mensaje tendrá una referencia a un evento o tema. Los mensajes serán redirigidos solo a aquellos participantes que se hayan suscrito al tema. Es una herramienta que ofrece una potente abstracción.

Las diferencias entre MOM y SOA son:

- MOM tiene un sistema de mensajes que es el que redirige el envío del mensaje al receptor, mientras que SOA conecta a emisor y receptor y el envío es directo.
- MOM manda mensajes de forma asíncrona y desacoplada totalmente y SOA sólo tiene desacoplamiento en el servicio de directorio.
- SOA tiene interoperabilidad, mientras que MOM no permite la conexión de aplicaciones en distintos lenguajes.
- La principal diferencia radica en el intermediario, ya que no funcionan de la misma forma.

Las similitudes entre MOM y SOA:

- Arquitecturas de red con gestor intermediario.

La funcionalidad del intermediario es distinta, ya que:

- En MOM el intermediario realiza la función de gestor del sistema, administrando el paso de mensajes redireccionando a las colas de recepción correspondientes según el mensaje. La comunicación siempre pasa por este intermediario.

-En SOA el intermediario es un servicio de registro al que el cliente acude para buscar un servidor. Su función es la de proporcionar al cliente los datos del proveedor para poder realizar la conexión. La comunicación solo pasa por este intermediario en el proceso de registro en el caso del proveedor de servicios y en el caso de búsqueda en el consumidor de servicios.

Para la implementación de MOM sería necesario la utilización de un gestor intermediario, por lo que ORB podría ser la tecnología adecuada para el sistema que buscamos, ya que no nos interesa llamar o invocar a métodos remotos, que es la funcionalidad que ofrecen RPC y RMI. Se busca realizar la conexión y el paso de mensajes entre emisor y receptor.

## EXAMEN JULIO 2013/14

### PREGUNTA 1

**Define y explica los tres enfoques de sistemas distribuidos (sistemas operativos en red, sistemas operativos distribuidos y middleware) realizando una comparativa entre ellos. ¿Qué mecanismos de comunicación distribuida (RPC, RMI, ORB, Servicios Web) utilizarías para implementar cada uno? Justifica la respuesta.**

#### SISTEMAS OPERATIVOS EN RED

Se realiza una abstracción en las comunicaciones del sistema operativo.

Implementar un sistema en red implica la aplicación de técnicas maduras tanto para la implementación como para el uso.

- Una de las ventajas es la flexibilidad, aprovechando el canal de red para potenciar esta característica.
- Las principales desventajas de tener implementado este sistema en red es la transparencia y el gran esfuerzo de integración que se ha de realizar por parte de los desarrolladores para el correcto funcionamiento del sistema.

No podemos aprovechar las ventajas del sistema distribuido tales como la seguridad o la tolerancia a fallos, pero tenemos la ventaja de que nuestro sistema operativo estará en un grado de madurez muy alto.

Para la implementación de un sistema operativo en red utilizaría RPC en caso de que nuestro sistema operativo esté en red local, por la facilidad de uso de Sockets. En caso de que nuestro sistema operativo estuviera fuera de una red local podríamos utilizar RMI para la compartición de recursos.

#### SISTEMAS OPERATIVOS DISTRIBUIDOS

La gran ventaja de implementar este tipo de sistemas es el aprovechamiento de todas las ventajas que nos ofrecen los sistemas distribuidos, aunque las más importantes son:

- Madurez
- Transparencia
- Escalabilidad
- Tolerancia a fallos

Las desventajas que se nos presentan son costosas y difíciles de arreglar:

- Es necesaria una conexión de alta velocidad para poder trabajar de una forma correcta y para que no haya problemas con los recursos.
- Técnicas complejas en la implementación del sistema distribuido.
- Alta competencia de mercado.

Un ejemplo de sistema distribuido es Amoeba.

## Parcial 1 SD (II - UA)

Con respecto a los sistemas operativos en red, no nos aprovecharemos de la flexibilidad que nos ofrece la capa de Internet, pero es muy importante la tolerancia a fallos, ya que siempre será más difícil que nuestro sistema operativo caiga totalmente con el uso de la computación distribuida.

Para la implementación de un sistema distribuido podríamos utilizar RMI por su facilidad de uso en Java y las invocaciones remotas a métodos.

### MIDDLEWARE

El middleware es una capa que se encuentra por encima del sistema operativo y que sirve para realizar una comunicación en el nivel más bajo de la capa de red.

Tiene un enfoque mixto, el modelo conceptual se encuentra en SOD y las infraestructuras en SOR.

Las ventajas que nos ofrece son:

- Transparencia.
- Escalabilidad
- Integración
- Madurez

La desventaja que nos ofrece un sistema middleware:

- Heterogeneidad de las plataformas.
- Necesidad de estandarización. Esto es debido a que al realizar la comunicación entre emisor y receptor, es necesario un estándar para la representación de datos.

Para la implementación de un sistema Middleware utilizaría ORB, ya que muchos sistemas middleware utilizan este tipo de tecnología de manera integrada en su capa de red.

## PREGUNTA 2

**Explica el modelo arquitectónico básico y su funcionamiento de la tecnología de servicios web explicando cada una de las tecnologías que intervienen. ¿Qué relación existe con el modelo básico del paradigma SOA?**

El modelo arquitectónico básico es cliente-servidor. Se trata de un modelo de asignación de roles a sus participantes. Diferenciamos dos roles principales:

- Proceso servidor: será el encargado de hacer el papel de proveedor de servicios. Permanecerá en espera pasiva la llegada de peticiones y se encargará de devolver respuestas.
- Proceso cliente: será el encargado de realizar peticiones al servidor. Realizará peticiones y permanecerá a la espera de respuestas.

En este modelo hay una fuerte abstracción y una gran simplificación en la comunicación, ya que al final ambos roles estarán a la espera: el servidor permanecerá a la espera de peticiones y el cliente permanecerá a la espera de respuestas.

La conexión entre servidor y cliente es directa y punto a punto. El envío de mensajes se realiza de forma acoplada y síncrona.

La relación con el paradigma SOA radica en que la comunicación que se realiza en SOA entre cliente y servidor también se realiza de forma síncrona y acoplada. Una vez se ha realizado el registro por parte del proveedor y el consumidor de servicios ha realizado una búsqueda en el servicio de registro y ha encontrado al proveedor que buscaba, la conexión que se realiza es punto a punto.



## Noviembre 2013 A

### Pregunta 1.

**Explica el paradigma de computación distribuida denominado SOA (Service Oriented Architecture) (0,5). Enumera y describe sus elementos (0,5). ¿Cuáles son las principales similitudes y diferencias entre SOA y el modelo cliente-servidor convencional (básate en los principios de SOA) (0,5)? ¿Qué relación existe entre el modelo arquitectónico MatchMaker de SOA y el de Servicios Web (0,5)? ¿Qué mecanismos de comunicación distribuida (RPC, RMI, ORB, Servicios Web) utilizarías para implementar SOA? Justifica la respuesta (0,5). [2,5 puntos]**

### Pregunta 2.

**Dentro de las tecnologías web describe el modelo servlet (0,5) indicando sus principales características (0,5). Realiza una breve comparativa con el modelo CGI (0,5) y el de páginas activas atendiendo, en este último, a las recomendaciones de uso (0,5). [2 puntos]**

### Pregunta 4.

**Dentro de la práctica no guiada de sockets y RMI que habéis realizado explica esquemáticamente tu arquitectura desde el controlador hasta los sensores y**

## Noviembre 2013 B

### Pregunta 3.

**En referencia a la capacidad de distribución de la información y su gestión de un servicio de directorio (básate en el caso de LDAP), enumera y explica, de forma breve, los principales motivos que pueden llevar a dicha distribución (0,9). ¿Qué técnica es utilizada para relacionar las diferentes partes del espacio de nombres cuando es distribuido (0,5) y cómo funciona en relación con la resolución de peticiones (0,6)? [2 puntos]**

### Pregunta 4.

**Dentro de la práctica no guiada de sockets y RMI que habéis realizado, explica esquemáticamente tu arquitectura desde el navegador hasta el controlador (0,75). ¿Qué importancia tiene el controlador? (0,25)**

## TIPO TEST

**1. Dentro de los tres enfoques de sistemas distribuidos, ¿cuál o cuáles de ellos son apropiados para entornos heterogéneos como Internet?**

c. Los Sistemas Operativos en Red por su flexibilidad e independencia y el middleware al estar basado en los Sistemas Operativos en red.

**2. Dentro de los dos paradigmas de computación distribuida de Cluster y Grid, ¿cuál es el más adecuado para ser utilizado en entornos heterogéneos como Internet?**

c. El de Grid puesto que su arquitectura es flexible y es una evolución del Cluster para entornos heterogéneos.

**3. ¿Qué es un stub en sistemas distribuidos?**

b. Es una representación del elemento remoto con la que interactúa el cliente a nivel local y abstrae al cliente de los problemas de comunicación.

**4. ¿Cuál o cuáles son los mecanismos de comunicación orientados a la interoperabilidad entre diferentes lenguajes y paradigmas de programación?**

d. Las respuestas A y C son correctas. (ORB y Servicios Web).

**5. ¿Cuál o cuáles de los siguientes modelos tecnológicos Web permiten la generación dinámica de contenidos?**

d. Las respuestas A y C son correctas. (Servlets y CGI y Páginas activas).

**6. ¿Qué técnica usa MIME para representar la información en un proceso de comunicación HTTP?**

c. Un lenguaje de representación externa.

**7. ¿Qué relación existe entre un documento WSDL y la generación de un mensaje SOAP?**

c. En la parte de enlace de un documento WSDL se indica mediante los atributos style y use cómo se estructurará el mensaje SOAP.

**8. ¿Cuál es el principal objetivo del middleware?**

a. Abstraer de la heterogeneidad existente del entorno a los participantes de un sistema distribuido mostrando una visión única del sistema.

**9. ¿Qué es la convención del nombrado de un sistema de nombres?**

c. Es la sintaxis para definir los nombres en un sistema de nombrado.

**10. ¿Qué tipo de resolución hace transparente al cliente (al servidor local DNS) del proceso de resolución de una petición DNS?, es decir el servidor local hace la petición si no puede resolverla él mismo y se olvida hasta que le devuelve la solución otro servidor de más alto nivel.**

c. La recursiva del servidor.

**11. De los tres tipos de arquitecturas P2P, ¿cuál es la que usaríais para tener mayor control de todas las transferencias realizadas?**

c. La centralizada.

**12. Si tu máquina fuera incluida en un Cluster o en un Grid, ¿en cuál de los dos paradigmas podrías continuar haciendo tus prácticas de Sistemas Distribuidos sin que influya en tu procesamiento?**

b. En el Grid.

**13. ¿Cuál de las tres técnicas de representación de la información utiliza el modelo básico Web?**

c. Utilización de una representación externa común.

**14. ¿Sobre qué mecanismo de comunicación se sustenta el mecanismo de llamadas a procedimientos remotos o RMI?**

a. Sobre el mecanismo de paso de mensajes implementado mediante el API de sockets.

**15. En una aplicación Web diseñada bajo patrón MVC en java, ¿qué papel sería más adecuado para el modelo de servlets y páginas activas JSP?**

c. Servlets como controlador y JSP como vista.

**16. ¿Qué modelo de los que implican una ampliación en el cliente (navegador Web) sobre el modelo Web debería utilizar para enviar datos confidenciales a un servidor Web?**

b. El modelo de applets o ActiveX.

**17. Desde el punto de vista de la arquitectura, el middleware**

b. Se centra en los servicios de infraestructura que ofrece a los componentes.

**18. El protocolo SOAP**

d. Las tres respuestas son correctas. (Fue creado para ser independiente del protocolo de transporte, está basado en XML, el protocolo de transporte que más utiliza es HTTP y es el único que define la especificación del W3C y es un protocolo extensible).

**19. Un servicio de nombres o de directorio**

d. Las respuestas A y C son correctas. (Está optimizado para lecturas y es un sistema escalable).

**20. Elige una de las respuestas**

d. Ninguna de las respuestas anteriores son correctas.

UDDI es un servicio de directorio de propósito general, FALSO.

DNS es un servicio de nombres de propósito general, FALSO.

LDAP es un servicio de directorio de propósito específico, FALSO.