



Actividades UD 7. -Sistemas Distribuidos Conceptos Básicos



Concurrencia y Sistemas Distribuidos

Actividad I

- Dadas las siguientes afirmaciones, indique si son Verdaderas (V) o Falsas (F):

	Un sistema distribuido es también un sistema concurrente.
	En un sistema distribuido, sus nodos comparten memoria, así como la señal de reloj.
	Un sistema distribuido está formado por máquinas autónomas que ofrece a sus usuarios la imagen de un sistema único, haciendo uso de algoritmos distribuidos.
	En los sistemas distribuidos, al igual que en los sistemas en red, los usuarios acceden a los ordenadores de la red utilizando la dirección de red de la máquina o el nombre asignado a dicha máquina.
	Un programa concurrente formado por varios hilos dentro de un único proceso, es un ejemplo de sistema no distribuido.

Actividad I

► Dadas las siguientes afirmaciones, indique si son Verdaderas (V) o Falsas (F):

V	Un sistema distribuido es también un sistema concurrente.
F	En un sistema distribuido, sus nodos comparten memoria, así como la señal de reloj.
V	Un sistema distribuido está formado por máquinas autónomas que ofrece a sus usuarios la imagen de un sistema único, haciendo uso de algoritmos distribuidos.
F	En los sistemas distribuidos, al igual que en los sistemas en red, los usuarios acceden a los ordenadores de la red utilizando la dirección de red de la máquina o el nombre asignado a dicha máquina.
V	Un programa concurrente formado por varios hilos dentro de un único proceso, es un ejemplo de sistema no distribuido.



Actividad 2

- ▶ Explique, con sus propias palabras, cómo se reflejan las cuatro características principales de los sistemas distribuidos (i.e. transparencia, disponibilidad, escalabilidad, seguridad) para las siguientes aplicaciones:
 - a) La aplicación distribuida PoliformaT.
 - b) Los servicios de mensajería en Internet (email).

- ▶ **Características de los Sistemas Distribuidos:**
 - ▶ **Transparencia:** se oculta el hecho en sí mismo de la distribución, las diferencias entre las diferentes máquinas y la complejidad de los mecanismos de comunicación.
 - ▶ **Disponibilidad:** los servicios que ofrecen deben estar siempre disponibles. Se trata de sistemas intrínsecamente tolerantes a fallos y donde las tareas de mantenimiento no deben interrumpir el servicio.
 - ▶ **Escalabilidad:** deben ser relativamente sencillos de ampliar, tanto en usuarios, como en recursos y en número de nodos, sin que sus prestaciones se vean mermadas.
 - ▶ **Seguridad:** los recursos y usuarios deben coexistir respetando las reglas y restricciones de acceso y políticas de seguridad que dependerán de cada sistema. Ha de tener en cuenta aspectos de autenticación, integridad, confidencialidad y disponibilidad.

Actividad 2

- ▶ Explique, con sus propias palabras, cómo se reflejan las cuatro características principales de los sistemas distribuidos (i.e. transparencia, disponibilidad, escalabilidad, seguridad) para las siguientes aplicaciones:
 - a) La aplicación distribuida PoliformaT.
 - b) Los servicios de mensajería en Internet (email).

La transparencia de distribución implica ocultar la complejidad de los algoritmos que se ejecutan, los fallos que puedan suceder, el número de ordenadores involucrados, su ubicación y todas aquellas características que sean irrelevantes para el usuario final.

En los ejemplos, cuando nos conectamos al sistema no nos importa (ni podemos ver) sobre qué máquinas, sistemas operativos, etc. se está ejecutando la aplicación.

Actividad 2

- ▶ Explique, con sus propias palabras, cómo se reflejan las cuatro características principales de los sistemas distribuidos (i.e. transparencia, disponibilidad, escalabilidad, seguridad) para las siguientes aplicaciones:
 - a) La aplicación distribuida PoliformaT.
 - b) Los servicios de mensajería en Internet (email).

La disponibilidad implica que los servicios ofrecidos por un sistema estén siempre disponibles.

PoliformaT suele funcionar generalmente sin problemas, aunque se produzcan fallos en sus servidores. Como mucho, en algunos momentos se puede observar cierto retraso en la respuesta.

En los servicios de mensajería en internet, aunque fallen algunos servidores intermedios, sabemos que el mensaje que enviamos siempre llegará a su destino.

Actividad 2

- ▶ Explique, con sus propias palabras, cómo se reflejan las cuatro características principales de los sistemas distribuidos (i.e. transparencia, disponibilidad, escalabilidad, seguridad) para las siguientes aplicaciones:
 - a) La aplicación distribuida PoliformaT.
 - b) Los servicios de mensajería en Internet (email).

La escalabilidad implica que no resulte difícil incorporar más ordenadores para atender a un mayor número de usuarios.

En los ejemplos, se podrán incorporar servidores para poder atender al incremento de demanda de usuarios.

Actividad 2

- ▶ Explique, con sus propias palabras, cómo se reflejan las cuatro características principales de los sistemas distribuidos (i.e. transparencia, disponibilidad, escalabilidad, seguridad) para las siguientes aplicaciones:
 - a) La aplicación distribuida PoliformaT.
 - b) Los servicios de mensajería en Internet (email).

La seguridad implica que los recursos y usuarios deben coexistir respetando las reglas y restricciones de acceso y las políticas establecidas en cada sistema.

En los ejemplos, para poder utilizar los servicios, los usuarios deben autenticarse proporcionando sus credenciales. En PoliformaT puede observarse que según el tipo de usuario, puede utilizar recursos/servicios diferentes (aquellos para los que está autorizado).

Actividad 3

- ▶ Indique con qué ejes de la transparencia de distribución se corresponden las siguientes características de un sistema:
 - a) Un usuario no percibe que el sistema esté siendo utilizado por múltiples usuarios al mismo tiempo.
 - b) El sistema permite la movilidad de recursos sin afectar al sistema.
 - c) Se permite acceder a los recursos sin conocer el dispositivo exacto en el que se encuentran.
 - d) El fallo de un nodo no es percibido por los usuarios.
 - e) Se almacenan diversas copias de un mismo recurso, cada una en un nodo diferente, sin que el usuario lo perciba.

Actividad 3

- ▶ Indique con qué ejes de la transparencia de distribución se corresponden las siguientes características de un sistema:
 - a) Un usuario no percibe que el sistema esté siendo utilizado por múltiples usuarios al mismo tiempo.
Transparencia de concurrencia
 - b) El sistema permite la movilidad de recursos sin afectar al sistema.
Transparencia de migración, reubicación
 - c) Se permite acceder a los recursos sin conocer el dispositivo exacto en el que se encuentran.
Transparencia de ubicación
 - d) El fallo de un nodo no es percibido por los usuarios.
Transparencia de fallos
 - e) Se almacenan diversas copias de un mismo recurso, cada una en un nodo diferente, sin que el usuario lo perciba.
Transparencia de replicación

Actividad 4

- ▶ Relacione los ejes de la transparencia de distribución con los siguientes mecanismos para lograrlos. Hay mecanismos que podrían relacionarse con más de un eje de transparencia.
 - 1) Servicios de nombres
 - 2) Replicación
 - 3) Detectores de fallos
 - 4) Identificación de los recursos con nombres simbólicos únicos
 - 5) Algoritmos tolerantes a fallos
 - 6) Algoritmos de balanceo de carga
 - 7) Algoritmos para ofrecer consistencia

Actividad 4

- Relacione los ejes de la transparencia de distribución con los siguientes mecanismos para lograrlos. Hay mecanismos que podrían relacionarse con más de un eje de transparencia.

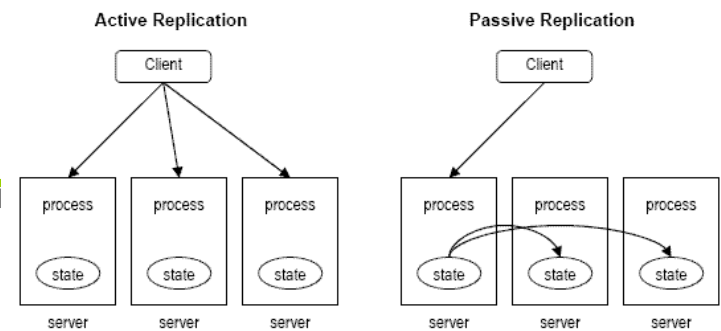
- 1) Servicios de nombres Transparencia de ubicación y de replicación
- 2) Replicación Transparencia de replicación y de fallos
- 3) Detectores de fallos Transparencia de fallos
- 4) Identificación de los recursos con nombres simbólicos únicos
Transparencia de ubicación y de replicación
- 5) Algoritmos tolerantes a fallos Transparencia de fallos
- 6) Algoritmos de balanceo de carga Transparencia de replicación
- 7) Algoritmos para ofrecer consistencia Transparencia de replicación

Actividad 5

- ▶ Responda brevemente a las siguientes preguntas :
 - a) ¿Qué diferencias existen entre replicación activa y replicación pasiva?
 - b) ¿Qué es la "consistencia" entre réplicas? ¿Qué tipos de consistencia existen?

▶ Replicación

- ▶ Cada servicio se configura con más de una réplica, de forma que ante fallos, las réplicas que permanecen “vivas”, se reconfiguran y continúan ofreciendo servicio.
- ▶ Los clientes del servicio acceden al servicio con **transparencia de replicación**, observando como única diferencia, una mayor disponibilidad.



a) ¿Qué diferencias existen entre replicación activa y replicación pasiva?

► Replicación pasiva

- ❑ Entre el grupo de réplicas, una única réplica será la réplica **primaria**. El resto de réplicas son las **secundarias**.
- ❑ El nombre de “replicación pasiva” viene derivado del rol de las réplicas secundarias, que no procesan peticiones. Son pasivas.
- ❑ La réplica primaria recibe todas las peticiones por parte de los clientes del servicio, las procesa y responde a los clientes.
- ❑ Ante cada petición que implique un cambio de estado, la réplica primaria difundirá un mensaje de actualización de estado a las secundarias.

► Replicación activa

- ❑ Todas las réplicas son iguales. Todas reciben las peticiones, todas procesan las peticiones y todas contestan al cliente.

Actividad 5

- b) ¿Qué es la "consistencia" entre réplicas? ¿Qué tipos de consistencia existen?
- ▶ La replicación introduce el problema de la consistencia: el grado de similitud/diferencia entre las diferentes réplicas.
 - ▶ Sistemas con mucha consistencia, o **consistencia fuerte**:
 - Idealmente todas las réplicas son iguales entre sí en todo momento.
 - Este objetivo es imposible de lograr, pero se pueden lograr sistemas con consistencia bastante fuerte, donde los usuarios del servicio obtienen la misma respuesta independientemente de la réplica que les atienda.
 - ▶ Sistemas con poca consistencia, o **consistencia débil**:
 - Las réplicas pueden divergir, de forma que cada una de ellas puede dar respuestas diferentes en cierto instante.

Actividad 6

- ▶ Explique, para cada tipo de replicación visto en clase, cómo se produce la reconfiguración en caso de fallo:
 - a) Si falla la réplica secundaria.
 - b) Si falla la réplica primaria.

a) Replicación pasiva: Reconfiguración en caso de fallos:

- ▶ En caso de fallo de una **réplica secundaria**, el trabajo de reconfiguración es pequeño → la réplica primaria la ignorará, por tanto enviará un mensaje menos de actualización.
- ▶ En caso de fallo de la **réplica primaria**, el trabajo de reconfiguración puede ser elevado:
 - Elegir a una réplica secundaria para que asuma el rol de réplica primaria. Todas las réplicas deberán estar de acuerdo.
 - Asegurarse que la réplica elegida tiene el estado más reciente posible.
 - Asegurarse que los clientes pueden encontrar adecuadamente al nuevo primario cuando detecten que el antiguo primario no responde.

b) Replicación activa: Reconfiguración en caso de fallos:

- ▶ En caso de fallo de una réplica no es necesario mucho trabajo, tan sólo eliminar las referencias a tal réplica de los clientes que traten de acceder al servicio.
- ▶ Todas las réplicas tienen el mismo estado, por tanto no es necesario trabajo para mantener la consistencia.

Actividad 7

- ▶ Responda brevemente a las siguientes preguntas
 - a) ¿Qué es la disponibilidad? ¿Qué factores afectan a la disponibilidad?
- ▶ Podemos definir **disponibilidad** como la probabilidad de que un determinado sistema ofrezca sus servicios a los usuarios.
 - ▶ Hay 3 factores que afectan a la disponibilidad:
 - ▶ Fallos
 - ▶ Tareas de mantenimiento
 - ▶ Ataques maliciosos

- ▶ Responda brevemente a las siguientes preguntas
 - b) ¿Qué son los sistemas tolerantes a fallos?
 - ▶ Todo sistema distribuido debe ser tolerante a fallos. El sistema debe seguir funcionando y proporcionando servicio en presencia de fallos.
 - c) ¿Qué tareas típicas de mantenimiento requiere cualquier sistema?
 - ▶ Todo sistema requiere mantenimiento.
 - Tareas tales como sustituir discos, alterar la configuración de los sistemas, realizar copias de seguridad, ampliar sistemas, o incluso cambiar el sistema por completo.
 - ▶ Se deben diseñar sistemas que permitan realizar el mantenimiento al mismo tiempo que los usuarios acceden a los servicios.

Actividad 8

- Dadas las siguientes afirmaciones, indique si son Verdaderas (V) o Falsas (F):

	En general, en un sistema distribuido los usuarios no deberían poder acceder a los servicios del sistema mientras se realizan copias de seguridad del mismo.
	Un ataque de denegación de servicio es un ataque malicioso al sistema.
	En un sistema con consistencia débil, los usuarios del servicio obtienen la misma respuesta independientemente de la réplica que les atienda.
	Los fallos compuestos, al afectar a varios nodos y canales simultáneamente, son indetectables.
	Los fallos de parada, de temporización y los fallos bizantinos son ejemplos de fallos simples detectables.
	Un fallo de respuesta detectable consiste en que un nodo tarda demasiado en responder y el nodo que espera su respuesta detecta, mediante un timeout, que la respuesta no llega.

Actividad 8

- Dadas las siguientes afirmaciones, indique si son Verdaderas (V) o Falsas (F):

F	En general, en un sistema distribuido los usuarios no deberían poder acceder a los servicios del sistema mientras se realizan copias de seguridad del mismo.
V	Un ataque de denegación de servicio es un ataque malicioso al sistema.
F	En un sistema con consistencia débil, los usuarios del servicio obtienen la misma respuesta independientemente de la réplica que les atienda.
F	Los fallos compuestos, al afectar a varios nodos y canales simultáneamente, son indetectables.
F	Los fallos de parada, de temporización y los fallos bizantinos son ejemplos de fallos simples detectables.
F	Un fallo de respuesta detectable consiste en que un nodo tarda demasiado en responder y el nodo que espera su respuesta detecta, mediante un timeout, que la respuesta no llega.

Actividad 8

	Los fallos simples detectables se tratan utilizando replicación y, en su caso, se ignora a la réplica que ha fallado.
	En los fallos bizantinos los nodos proporcionan una respuesta errónea, por ejemplo por un error en el software.
	Para tratar los fallos bizantinos se utilizan algoritmos de quorum.
	El teorema CAP implica que se debe ofrecer a la vez Consistencia fuerte, Disponibilidad elevada y permitir que ocurran Particiones.
	El servicio de pertenencia a grupo se encarga de llegar a un consenso entre los nodos sobre qué nodos han fallado.
	Todos los fallos simples detectables, que no causen particiones, se tratan como fallos de parada, de modo que si se detecta que un nodo ha fallado, se expulsa y el resto de nodos lo ignora.

Actividad 8

V	Los fallos simples detectables se tratan utilizando replicación y, en su caso, se ignora a la réplica que ha fallado.
V	En los fallos bizantinos los nodos proporcionan una respuesta errónea, por ejemplo por un error en el software.
V	Para tratar los fallos bizantinos se utilizan algoritmos de quorum.
F	El teorema CAP implica que se debe ofrecer a la vez Consistencia fuerte, Disponibilidad elevada y permitir que ocurran Particiones.
V	El servicio de pertenencia a grupo se encarga de llegar a un consenso entre los nodos sobre qué nodos han fallado.
V	Todos los fallos simples detectables, que no causen particiones, se tratan como fallos de parada, de modo que si se detecta que un nodo ha fallado, se expulsa y el resto de nodos lo ignora.

Actividad 9

- Indique el orden de los pasos para lograr tolerancia a fallos, en el caso de fallos simples detectable que no causen particiones:

	La mayoría de los nodos del sistema acuerda que el nodo X ha fallado.
	El servicio de pertenencia a grupo inicia una fase de acuerdo para determinar si el nodo X ha fallado.
	Los nodos se reconfiguran para funcionar sin el nodo "X".
	El servicio de pertenencia notifica al resto de nodos "vivos" que el nodo X ha sido expulsado.
	El módulo "detector de fallos" monitoriza a varios nodos.
	El módulo "detector de fallos" indica que sospecha del fallo del nodo X.
	El servicio de pertenencia expulsa al nodo X.
	El módulo "detector de fallos" notifica su sospecha de fallo del nodo X al servicio de pertenencia a grupo.
	Los nodos ignoran los mensajes del nodo "X".

Actividad 9

- Indique el orden de los pasos para lograr tolerancia a fallos, en el caso de fallos simples detectable que no causen particiones:

5	La mayoría de los nodos del sistema acuerda que el nodo X ha fallado.
4	El servicio de pertenencia a grupo inicia una fase de acuerdo para determinar si el nodo X ha fallado.
9	Los nodos se reconfiguran para funcionar sin el nodo "X".
7	El servicio de pertenencia notifica al resto de nodos "vivos" que el nodo X ha sido expulsado.
1	El módulo "detector de fallos" monitoriza a varios nodos.
2	El módulo "detector de fallos" indica que sospecha del fallo del nodo X.
6	El servicio de pertenencia expulsa al nodo X.
3	El módulo "detector de fallos" notifica su sospecha de fallo del nodo X al servicio de pertenencia a grupo.
8	Los nodos ignoran los mensajes del nodo "X".

- Relacione los siguientes conceptos como corresponda:

Particiones
Fallo de respuesta detectable
Fallo de temporización
Fallo de parada
Fallo bizantino

Fallo simple
Fallo compuesto
Detectable
No detectable

Replicación
Algoritmos de quorum
Teorema CAP

- Relacione los siguientes conceptos como corresponda:

Particiones
Fallo de respuesta detectable
Fallo de temporización
Fallo de parada
Fallo bizantino

Fallo simple
Fallo compuesto
Detectable
No detectable

Replicación
Algoritmos de quorum
Teorema CAP

- Relacione los siguientes conceptos como corresponda:

Particiones
Fallo de respuesta detectable
Fallo de temporización
Fallo de parada
Fallo bizantino

Fallo simple
Fallo compuesto
Detectable
No detectable

Replicación
Algoritmos de quorum
Teorema CAP

- Relacione los siguientes conceptos como corresponda:

Particiones
Fallo de respuesta detectable
Fallo de temporización
Fallo de parada
Fallo bizantino

Fallo simple
Fallo compuesto
Detectable
No detectable

Replicación
Algoritmos de quorum
Teorema CAP

- Relacione los siguientes conceptos como corresponda:

Particiones
Fallo de respuesta detectable
Fallo de temporización
Fallo de parada
Fallo bizantino

Fallo simple
Fallo compuesto
Detectable
No detectable

Replicación
Algoritmos de quorum
Teorema CAP

- Relacione los siguientes conceptos como corresponda:

Particiones
Fallo de respuesta detectable
Fallo de temporización
Fallo de parada
Fallo bizantino

Fallo simple
Fallo compuesto
Detectable
No detectable

Replicación
Algoritmos de quorum
Teorema CAP

Actividad I I

- ▶ Responda brevemente a las siguientes preguntas
 - a) ¿Cuándo decimos que un sistema es escalable?
 - ▶ Cuando el servicio que ofrece no sufre alteraciones de rendimiento y disponibilidad (desde el punto de vista del usuario) al aumentar:
 - ▶ n° de usuarios
 - ▶ n° de recursos, n° de nodos
 - ▶ n° de peticiones simultáneas
 - b) ¿Cuáles son las técnicas más importantes para aumentar la escalabilidad?
 - ▶ Distribución de la carga, distribución de los datos, replicación y caching.

- ▶ Responda brevemente a las siguientes preguntas
 - c) ¿Qué es un sistema altamente escalable?
 - ▶ Sistemas con objetivos de crecimiento a nivel global: Google, Facebook ...
 - d) ¿Qué relación existe entre replicación y caching?
 - ▶ El caching es un caso particular de la replicación, donde disponemos de una copia del recurso en el propio cliente
 - d) ¿Son escalables los sistemas replicados con consistencia fuerte?
 - ▶ Si la mayoría de operaciones son de lectura, sí. Si hay operaciones de modificación de los datos, la replicación con consistencia fuerte no escala bien, ya que hay que actualizar todas las réplicas utilizando el algoritmos que bloquean el acceso al recurso replicado mientras se realizan los cambios.

Actividad 12

- Dadas las siguientes afirmaciones, indique si son Verdaderas (V) o Falsas (F):

	Con la distribución de los datos se distribuyen los recursos del sistema, de modo que cada nodo sirve una parte de esos recursos, como ocurre con la división de los DNS en zonas.
	La replicación permite aumentar la disponibilidad y la escalabilidad del sistema distribuido.
	Los sistemas altamente escalables ofrecen consistencia fuerte.
	La técnica de caching ofrece consistencia débil
	Los usuarios se autentican en un sistema al proporcionar sus credenciales.
	Los mecanismos de control de acceso gestionan si los usuarios autenticados tienen permiso para acceder a los recursos solicitados .
	Los ataques distribuidos de denegación de servicio se pueden controlar utilizando consistencia fuerte de los datos.

Actividad 12

- Dadas las siguientes afirmaciones, indique si son Verdaderas (V) o Falsas (F):

V	Con la distribución de los datos se distribuyen los recursos del sistema, de modo que cada nodo sirve una parte de esos recursos, como ocurre con la división de los DNS en zonas.
V	La replicación permite aumentar la disponibilidad y la escalabilidad del sistema distribuido.
F	Los sistemas altamente escalables ofrecen consistencia fuerte.
V	La técnica de caching ofrece consistencia débil
V	Los usuarios se autentican en un sistema al proporcionar sus credenciales.
V	Los mecanismos de control de acceso gestionan si los usuarios autenticados tienen permiso para acceder a los recursos solicitados .
F	Los ataques distribuidos de denegación de servicio se pueden controlar utilizando consistencia fuerte de los datos.