

REPLICACIÓN Y CONSISTENCIA

Material basado en el libro:

Andrew S. Tanenbaum. *Sistemas distribuidos:
paradigmas y principios.*



Objetivos

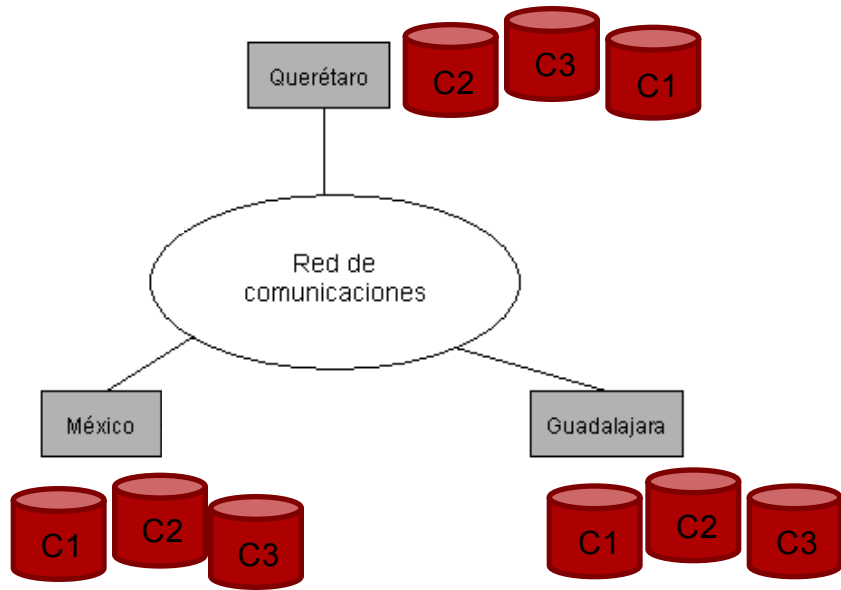
- Explicar el concepto de replicación, las mejoras y los requisitos para que ésta se lleve a cabo.
- Identificar los componentes de un sistema de réplicas.
- Identificar y explicar las fases de un sistema de réplicas.
- Explicar los tipos de ordenamiento para la ejecución de peticiones en el gestor de réplicas.

Replicación

- **Replicación:** mantenimiento de copias de la información en múltiples máquinas: replicación total o parcial.
- Es un recurso utilizado ampliamente en los sistemas distribuidos:
 - Servidores web y servidores de base de datos.
 - DNS: copias de los mapeos URL-IP
 - Google: Google Data Centers
 - Replicación del reloj del sistema

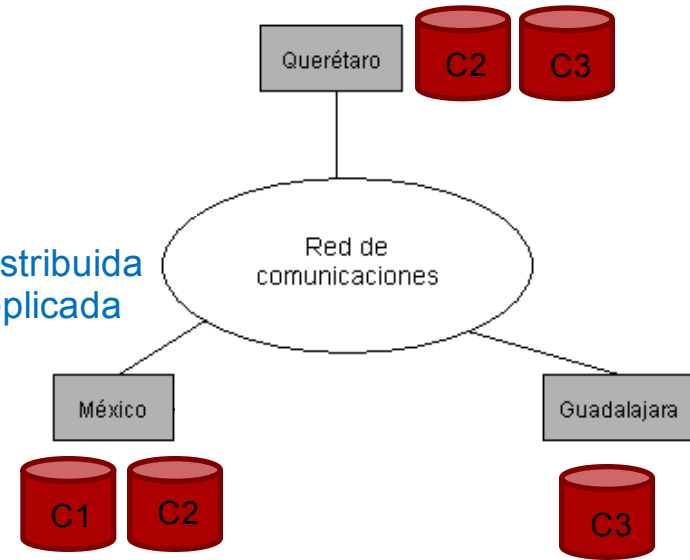
Tipos de replicación: escenarios

- **Total:** mantener copias de todos los objetos/recursos (base de datos, archivos, etc.)
- **Parcial:** mantener copias sólo de ciertos fragmentos de los objetos/recursos del sistema distribuidos (base de datos, archivos, etc.).
 - ciertos elementos están duplicados y algunos que no lo están.
- **No replicada** guarda cada objeto / recurso un solo sitio.

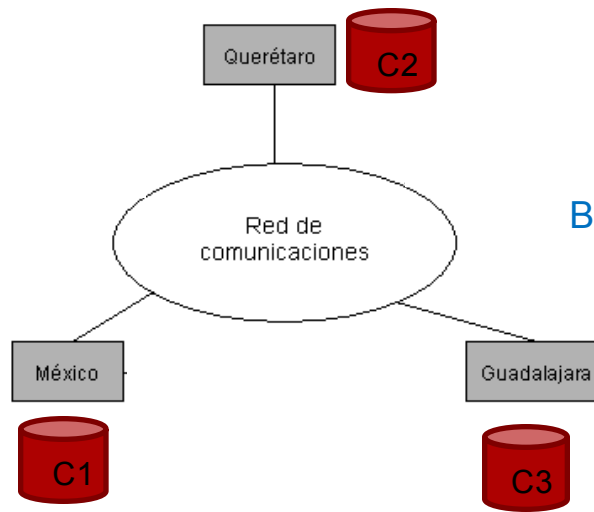


Base de datos distribuida totalmente replicada

Base de datos distribuida parcialmente replicada



Base de datos distribuida no replicada



Ventajas de la replicación

La replicación ayuda a mejorar un Sistema Distribuido en distintos aspectos:

- **Mayor disponibilidad.** El recurso puede encontrarse en varios sitios.
- **Confiabilidad.** Seguir respondiendo a pesar del fallo de un sitio.
- **Paralelismo incrementado.** En caso de que el acceso un recurso sea de sólo lectura, varios sitios pueden procesar, en paralelo, las lecturas que impliquen al recurso.

Desventajas de la replicación

- **Sobrecarga** durante la actualización. Se debe asegurar la consistencia mutua, en caso contrario, pueden producirse cálculos erróneos. Para ello, se debe propagar la actualización a todos los sitios que contienen réplicas.
- **Mayor tiempo de procesamiento:** el intercambio de mensajes hacia las réplicas y los cálculos adicionales suponen una forma de tiempo extra que debe ser considerada.
- **Costos**
 - Costos del desarrollo de software: es necesario un software para la administración de la replicación.
 - Costos de almacenamiento de las réplicas.
 - Costos de procesamiento.

Comparación de las características en las estrategias de replicación

	Replicación total	Replicación parcial	No replicada
Paralelismo	Muy alto	Alto	Nulo
Disponibilidad	Muy alta	Alta	Baja
Costo	Elevado	Moderado	Bajo
Riesgo de pérdida de datos	Muy bajo	Medio	Alta
Realidad	Aplicación posible	Realista	Aplicación posible

Alta disponibilidad

- La proporción de tiempo que un servicio está accesible con tiempos de respuesta razonables debe ser cercana al 100%.
- Factores de pérdida de disponibilidad
 - Fallos en el servidor
 - Particiones de red o desconexiones
 - Desconexión intencionada o no intencionada.

Alta disponibilidad

Fallos en el servidor

- Si un servidor tiene una posibilidad de fallo p del 5%, tendrá disponibilidad del 95%
- Si replicamos n veces el servidor, la disponibilidad será $1 - p^n$
- Con $n=2$ servidores: $1 - 0.05^2 = 99.75\%$

Alta disponibilidad

Considere una posibilidad de fallo del servidor de aplicaciones web del 20%. Calcular la disponibilidad de las aplicaciones en dos casos: si el servidor se replica dos veces y si se replica tres veces.

Alta disponibilidad

Explicar la fórmula: $1-p^n$

¿Cómo entienden la fórmula?

¿Por qué sirve para medir la disponibilidad

Por una participación en clase

Tolerancia a fallos

- Una alta disponibilidad no implica necesariamente datos correctos.
 - Puede haber datos no actualizados, o inconsistentes.
- Podemos utilizar replicación para ganar tolerancia a fallos
 - Parada o caída
 - Si tenemos n servidores/nodos, pueden fallar $n-1$ y se puede ofrecer el servicio

Requisitos

La replicación debe llevarse a cabo considerando

- **Transparencia:** los clientes no son conscientes de que hay múltiples copias del recurso al que acceden
- **Consistencia:** las operaciones sobre un conjunto de objetos replicados deben dar resultados que sigan la especificación de corrección definida para dichos objetos.
 - Las diferencias en la respuesta entre las réplicas pueden ser más o menos estricta según la aplicación.

Reglas en la replicación...

- Los datos replicados se deben someter a la **regla de consistencia mutua**, la cual requiere que todas las copias de fragmentos de datos sean idénticas.
- Para mantener la **consistencia de los datos** entre replicas, se debe garantizar la actualización de la base de datos en todos los sitios donde existen replicas.
- Una operación **read (lectura)** selecciona cualquier copia (de preferencia la más cercana) para satisfacer la transacción.
- Una operación **write (escritura)** requiere que todas las copias se seleccionen y actualicen para satisfacer la regla de consistencia muta.

Un sistema de réplicas

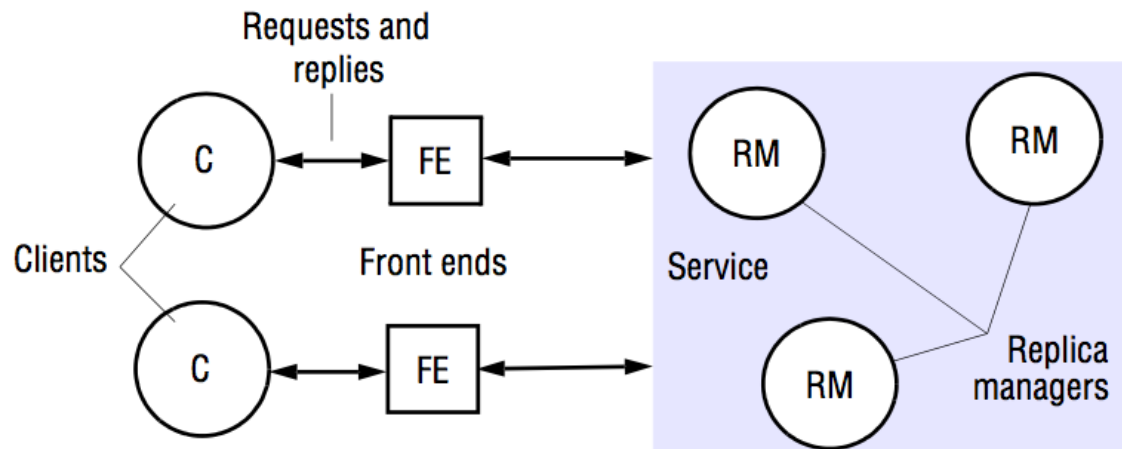
Componentes

Objetos

- **Objeto:** cualquier entidad de información a replicar
 - archivos, objetos Java, etc.
 - entidad de información visible por el usuario, implementada por distintas copias física o **réplicas**.
- Las réplicas de un objeto lógico no tienen por qué ser todas iguales en todo momento
 - Ejemplo: algunas pueden haberse actualizado y otras no

Componentes

- **Gestor de réplicas:** componentes que almacenan réplicas de un determinado objeto o servicio y operan sobre ellas.
- **Frontal (front end):** componente que atiende las llamadas de los clientes y se comunica con los gestores de réplicas



Fases de una petición

1. Petición: el frontal envía la petición a uno o más gestores.

- O bien envía la petición a un gestor y éste reenvía a otros
- O difunde la petición a varios gestores

2. Coordinación: los gestores se coordinan para ejecutar la petición de manera consistente:

- Ordenación FIFO

Fases de una petición

- 3. Ejecución:** se ejecuta la petición con base en un tipo de ordenamiento.
- 4. Acuerdo:** se llega a un consenso antes de consumir la ejecución
- 5. Respuesta:** uno o más gestores de réplicas responden al frontal

Ordenamiento para la ejecución en el gestor de réplicas

- **FIFO:** si un frontal envía la petición r y luego la petición r' , entonces los gestores procesan r antes que r'
 - La mayoría de las aplicaciones sólo requieren una ordenación FIFO

Resumen

- ¿Qué es replicación?
- ¿Cuáles es la diferencia entre replicación total y parcial ?
- ¿Cuáles son las mejores de la replicación en un sistema distribuidos?
- ¿Cuáles son los componentes de un sistema de réplicas?
- ¿Cuáles son las fases de una petición en un sistema de réplicas?
- ¿Cuáles son los tipos de ordenamiento para la ejecución de peticiones en el gestor de réplicas?