Clase del día - 25/05/2021

En la clase anterior vimos que la replicación de los datos es una estrategia utilizada para la recuperación de un sistema en caso de desastre, así mismo, la replicación de los datos tiene un efecto positivo en el rendimiento de un sistema, debido a que es más rápido acceder a datos cercanos.

Para mantener consistentes las diferentes copias de los datos, es necesario implementar un modelo de consistencia, el cual puede considerar diferente granularidad de los datos.

Mantener la consistencia de los datos suele ser complicado y costoso en tiempo de comunicación.

**Respaldos incrementales**

En el pasado, la replicación de los datos mediante respaldos era una solución medianamente buena para garantizar la continuidad de un sistema. Si un sistema fallaba, entonces se restauraba el último **respaldo**con el fin de recuperar el estado del sistema hasta un cierto punto del tiempo.

Se solía realizar **respaldos incrementales**, lo que significa que se debe diseñar una estrategia para realizar respaldos anuales, mensuales, semanales, diarios, cada hora, etc. En estas condiciones, el respaldo más frecuente solo deberá guardar los cambios realizados desde el último respaldo y no todo el almacén de datos.

No obstante, aún realizando respaldos cada hora, si el sistema falla se perderán los cambios efectuados los últimos minutos.

Para mitigar los riesgos debidos a errores humanos o el secuestro de datos (*ransomware*), los proveedores de cómputo en la nube ofrecen servicios escalables de respaldo de datos, los cuales permiten respaldar y restaurar máquinas virtuales completas o bien archivos, carpetas o bases de datos.

**Respaldos continuos**

Los sistemas manejadores de bases de datos (DBMS) implementan una estrategia de **respaldos continuos** de las transacciones.

A partir del inicio de una transacción (begin work), se va guardando copias de los registros que son actualizados dentro de la transacción, si el DBMS falla o la computadora se apaga, es posible recuperar la base de datos hasta un estado consistente a partir de los registros contenidos en el respaldo continuo.

El respaldo continuo de los registros modificados en una transacción también se utiliza para mantener un estado consistente de la base de datos. Si la transacción termina con **rollbak**, entonces se restablecen todos los registros modificados  al estado anterior a la transacción.

Por otra parte, si la transacción termina con **commit**, entonces los cambios quedan firmes y se desecha el respaldo continuo de los registros modificados dentro de la transacción.

A la funcionalidad del DMBS que garantiza la consistencia de la base de datos se le llama **ACID** (Atomicity, Consistency, Isolation, & Durability).

En el caso de bases de datos distribuidas, el control de las transacciones distribuidas se realiza mediante un protocolo llamado two-phase commit (2PC). En este caso el respaldo continuo también se realiza en forma distribuida.

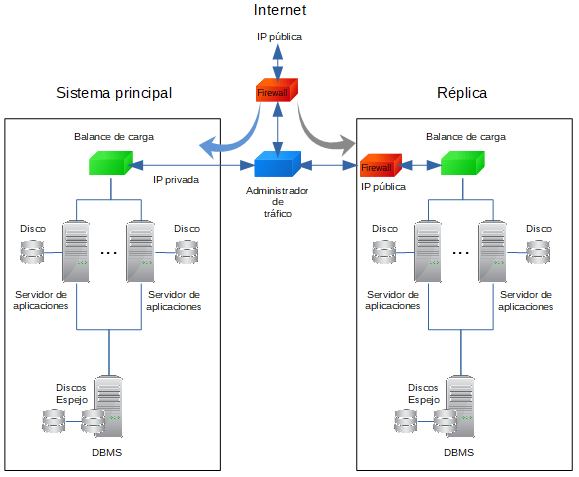
**Replicación del sistema completo**

En la actualidad el cómputo en la nube nos permite realizar el aprovisionamiento dinámico de recursos de forma fácil (automática), rápida y a bajo costo.

Entonces, ¿por qué no replicar el sistema completo?

**Arquitectura de un sistema replicado**

La siguiente figura muestra la arquitectura general de un sistema replicado:



En este caso se supone que el sistema consta de dos o más servidores de aplicaciones conectados a un servidor de bases de datos, el cual tiene implementada la replicación de datos mediante discos espejo.

El administrador de tráfico funciona como un servidor proxy transparente subrogado (*reverse transparent proxy*) el cual recibe las conexiones y peticiones de los clientes y las envía al sistema principal con copia a la réplica.

El sistema principal procesa las peticiones del cliente y envía las respuestas al administrador de tráfico, el cual las re-envía al cliente. Notar que el administrador de tráfico deberá ignorar las respuestas de la réplica.

Para garantizar la recuperación en caso de desastre catastrófico en el *site* dónde ejecuta el sistema principal, la réplica deberá ejecutar en una locación diferente, por lo tanto el administrador de tráfico y la réplica deberán tener cada uno una IP pública.

Mientras el sistema principal puede estar ejecutando en la nube o en una instalación propia de la empresa (*on-premise*), la réplilca estaría funcionando en la nube, en alguna locación geográfica diferente a la locación dónde ejecuta el sistema principal.

El administrador de tráfico puede ser un programa o un *appliance*localizado en mismo *site* dónde ejecuta el sistema principal, de manera que el administrador de tráfico se pueda conectar al sistema principal mediante una red privada.

Como puede observarse, tanto el sistema principal como la réplica realizarán las mismas transacciones sobre la base de datos (y/o el sistema de archivos), por lo que la consistencia de los datos está garantizada. Si se produce un error en la comunicación entre el administrador de tráfico y la réplica, el cliente deberá recibir un de error de comunicación.

En caso de falla del sistema principal y/o falla del administrador de tráfico, solo habrá que re-definir el dominio del sistema a la IP pública de la réplica, entonces los clientes estarán conectados directamente al sistema de respaldo. Desde luego, posteriormente será necesario: 1) sacar de producción la réplica, 2) realizar una copia espejo de la réplica, y 3) restaurar el sistema principal y el administrador de tráfico.

**Azure Site Recovery**

Azure Site Recovery (ASR) es un servicio que permite la replicación de máquinas virtuales entre diferentes regiones de Azure. La configuración de la replicación se lleva a cabo directamente en el portal de Azure.

ASR permite realizar la recuperación de una aplicación multi-capa (servidor web + servidor de aplicaciones + servidor de bases de datos) ejecutando en varias máquinas virtuales.

Es posible realizar pruebas de recuperación de desastres, sin impactar el ambiente de producción y a los usuarios. Así mismo, es posible mantener las aplicaciones disponibles durante el proceso de recuperación.

La clase siguiente veremos cómo replicar una máquina virtual utilizando Azure Site Recovery.

Actividades individuales a realizar

Revisar las páginas:

1. [Azure Traffic Manager](https://azure.microsoft.com/en-us/services/traffic-manager/)

2. [Load Balancer](https://azure.microsoft.com/en-us/services/load-balancer/)

3. [Azure Backup](https://azure.microsoft.com/en-us/services/backup/)