

# Replicación PostgreSQL

## **Alta disponibilidad.**

Con alta disponibilidad nos referimos a la creación de un protocolo para nuestro sistema que pueda proporcionar una protección completa de datos y su disponibilidad para poder asegurar un cierto grado de continuidad operacional durante un período de medición dado.

Con PostgreSQL disponemos de varias herramientas para poder implementar la alta disponibilidad y escalar PostgreSQL en varios servidores, como son:

- Almacenamiento compartido.
- Redundancia de datos por replicación.

## **Balanceo.**

En nuestro caso, es una técnica para repartir las peticiones a base de datos entre varias BD, de forma que se mejore el rendimiento del sistema.

## **Replicación de base de datos.**

Es el proceso de copiar y mantener actualizados los datos de una base de datos en otra, ya que sea que una base de datos reciba actualizaciones de una base maestra o que varias bases de datos modifiquen una misma base común.

## **Introducción a PostgreSQL.**

PostgreSQL es un sistema de gestión de bases de datos desarrollado en los 80's. Desde entonces ha ido variando, y aún sigue activo su desarrollo tanto por parte de un equipo específico de desarrollo como por contribuidores de código abierto. Es considerado el sistema de gestión de BD de código abierto más avanzado del mundo.

Diferentes versiones de PostgreSQL a lo largo del tiempo:

- Ingres (1977-1985)
- Postgres (1986-1994)
- Postgres95 (1995)
- PostgreSQL 6 (1996-1999)
- PostgreSQL 7 (1999-2004)
- PostgreSQL 8 (2005)

### **Soporte y prestaciones.**

Soporta diferentes juegos de caracteres internacionales, bases de datos de tamaño ilimitado, tablas de hasta 32TB, filas de 1,6TB, campos de 1GB, filas ilimitadas por tabla, de 250 a 1600 columnas por tabla, dependiendo del tipo de dato, e ilimitados índices por tabla.

Funciona tanto en Unix como en Windows.

### **Replicación.**

Uno de los métodos más usados para conseguir que nuestro servidor Postgres alcance el 99,9% de disponibilidad al año es la implementación de la replicación.

La replicación consiste en la transmisión de información derivada de las operaciones DML (INSERT, UPDATE, DELETE) de una base de datos a otra. Así, al tener la misma información ambas bases de datos, se consigue la redundancia de datos.

PostgreSQL dispone de varias herramientas y métodos para implementar.

#### **Tipos:**

Basada en disparadores.

Se implementa un disparador para las operaciones DML. Se usa una cola para almacenar los cambios y enviarlos a la base de datos remota de forma asíncrona. La ventaja de esta opción es que se puede implementar los disparadores para un subconjunto de tablas de las bases de datos.

Basada en archivos Write-Ahead-Log (WAL).

En un fichero se lleva un registro con los INSERTS, UPDATES y DELETES que se han realizado en una de las bases de datos. Estos ficheros se generan para garantizar la durabilidad de la base de datos.

### Técnicas de replicación:

Archivado continuo.

Se envía la información de los registros WAL mediante un archivado continuo entre el servidor maestro y el servidor esclavo.

Replicación por streaming.

Se envían los archivos WAL con las operaciones DML de una base de datos a otra a través de la red (puede ser de forma síncrona o asíncrona).

### Forma de transmisión:

Asíncrona.

Se realiza sin que ambas bases de datos estén activas al mismo tiempo.

Síncrona.

Se realiza en tiempo real, con ambas bases de datos activas al mismo tiempo.

### Herramientas:

‘Maestro-esclavo’.

Hay una base de datos que tiene la potestad de actualizar los datos, mientras que hay otra que se limita a recibir y aceptar las actualizaciones, sin posibilidad de alterar nada.

‘Multi maestro’.

Se habilitan los datos para ser distribuidos en un grupo de bases de datos, y que todas las bases partícipes puedan actualizar los datos.

### **Archivado continuo.**

El archivado continuo es una opción muy recomendable, y por ello, vamos a verlo en más detalle.

Es una técnica para enviar registros de WAL de un servidor maestro postgres a otro esclavo, de forma que el esclavo está continuamente reproduciendo los segmentos WAL, según llegan, y por lo tanto contienen una copia de la base de datos maestra. Esta copia puede ser utilizada tanto como copia de seguridad, como para implementar la replicación.

La réplica está continuamente en modo recuperación, aplicando los ficheros de WAL que reciba.

En Postgres existen dos modos de enviar ficheros WAL a las bases de datos de las replicas:

- Con un directorio común
- A través de streaming (red)

### **Inconvenientes.**

La gran desventaja es la pérdida de datos, debido a que no se copiará y procesará al esclavo hasta que no se modifique el fichero WAL. Además, requiere de un proceso externo para realizar la copia de los ficheros WAL, ya sea de forma síncrona o asíncrona.

El tiempo total de pérdida de datos está definido por la suma del tiempo que se tarde en modificar los los ficheros WAL y la duración del proceso de copia o transmisión del fichero nuevo y aplicado a la base de datos esclava.

**La solución** llega con la versión 9.0 de PostgreSQL, en la cual se introdujo la replicación por streaming, que copia los ficheros WAL según se produzcan y no cuando roten mediante la red. El modo de transmisión de estos ficheros es asíncrono, el modo síncrono legó con la versión 9.1.

### *Bibliografía:*

<https://todopostgresql.com/replicacion-postgresql/>