

Configuración de PostgreSQL

info@todopostgresql.com



**Todo
PostgreSQL**
by Abatic

1. [Parámetros](#)
2. [Velocidad y Rendimiento](#)
3. [Mantenimiento](#)
4. [WAL](#)
5. [Checkpoint](#)

Parámetros

Los parámetros más importantes que controlan PostgreSQL están contenidos en el fichero “postgresql.conf”

La gran mayoría pueden modificarse en tiempo de ejecución.

Se pueden hacer ajustes por transacción/sesión/usuario/base de datos (SET).

Velocidad y Rendimiento

Principalmente hay dos parámetros que determina la velocidad y el rendimiento. ¡Pero dependen de la memoria disponible!

- `max_connections`
- `shared_buffers`

NOTA: `max_connections` y `shared_buffers` necesita reiniciar la base de datos (afectan al postmaster).

Velocidad y Rendimiento

shared_buffers:

- Asigna el número de buffers de memoria compartida que podrá usar el servidor de base de datos.
- Cada buffer es de 8 kibibyte de tamaño.
- El valor mínimo es 128KiB.

Velocidad y Rendimiento

shared_buffers:

- Al menos $16 \times \text{max_connections}$ (en KiB).
- En general, del 6% al 25% de la memoria RAM disponible del sistema, es un buen punto de partida (dependerá de la carga).

Velocidad y Rendimiento

max_connections:

- Es el número máximo de conexiones simultánea que puede soportar el servidor PostgreSQL.
- Por defecto, 100.
- Depende del tipo de uso del servidor.
- Se recomienda bajar las conexiones para mejorar el rendimiento.

Velocidad y Rendimiento

max_connections:

- Es típico aumentar mucho este parámetro para aumentar el rendimiento, pero lo normal es que lo baje (satura al servidor por sobrecarga).
- PG usa 1 proceso por conexión. Si el proceso está ocupado haciendo una consulta, usará 1 core. Luego max_connections en situación de uso máximo continuo no debería ser superior a n° cores.

Velocidad y Rendimiento

max_connections:

- La realidad nos dice que puede ser 4x o más porque no siempre todas las conexiones usan al máximo todo el tiempo y todas están todo el rato ocupadas.
- Para que no haya clientes que observen un “connection refused” a la base de datos debe usarse connection pooling (pgbouncer o PgPool).

Velocidad y Rendimiento

work_mem:

- Cantidad de memoria en KiB que se usará para realizar ordenaciones y Joins en memoria.
- El valor mínimo es 64 KiB y por defecto es 1MiB.
- Para sesiones pesadas (informes), se puede considerar aumentarlo a valores como 128 MiB ó incluso más.
- Se puede cambiar en la sesión, en la base de datos o en el usuario (SET).

Velocidad y Rendimiento

work_mem:

- El comando **EXPLAIN ANALYZE** revela la memoria usada por una consulta, y si se está usando el disco.

- Se utiliza la memoria work_mem

Sort Method: quicksort Memory: 861967kB

- Se utiliza el disco, y es necesario aumentar work_mem.

Sort Method: external merge Disk: 7526kB

Velocidad y Rendimiento

work_mem:

- PG puede usar diferentes algoritmos para ordenar según si tiene que usar RAM o disco, por lo que la cantidad a aumentar de RAM en work_mem no tiene porqué coincidir con la usada en disco.
- En éste caso, se debería de aumentar como mínimo 8MB.

Mantenimiento

maintenance_work_mem:

- Es la memoria máxima en KiB que se usa en operaciones de mantenimiento como **VACUUM**, **CREATE INDEX** y **ALTER TABLE ADD FOREIGN KEY**.
- El valor mínimo es 1024 KiB y por defecto es 16 MiB.
- Normalmente hay que aumentarlo notablemente (Dependiendo de la memoria del sistema a 2 GiB, 4 GiB).

Mantenimiento

maintenance_work_mem:

- El rendimiento para vacuuming y restauración de un volcado de base de datos mejorará al aumentar este valor. Puesto que se tienen más páginas en memoria para operar.

WAL

wal_level:

- Determina cuánta información se escribe en los ficheros WAL.
- El valor por defecto es “**replica**”, con el que se escribe la información necesaria para recuperarse de una caída o un apagado instantáneo, para realizar backup físicos en caliente y para implementar Alta disponibilidad.

WAL

wal_level:

- **minimal:** permite recuperarse de una caída o un apagado instantáneo.
- **replica:** permite crear nodos secundarios de replicación.
- **logical:** permite usar replicación lógica.

WAL

wal_buffers:

- Número de buffers de disco/página asignados en memoria compartida para datos WAL.
- Cada buffer es de 8 KiBytes (tamaño de página).
- Necesita ser suficientemente grande como para guardar los datos WAL de una transacción típica ya que los datos WAL se escriben en el discos después del commit de cada transacción.

WAL

wal_buffers:

- El valor mínimo permitido es 4KiB.
- El valor asignado por defecto es de -1 (Calcula automáticamente un 3 % del shared_buffers).

WAL

max_wal_size:

- Tamaño máximo directorio pg_wal.
- Cada fichero WAL es de 16 Mbytes.
- Se fuerza un checkpoint cuando el fichero está lleno.
- El valor asignado por defecto es de 1GB.

WAL

max_wal_size:

- Interesa subirlo en caso de tener mucha actividad de escritura.
- Si hay espacio en disco, tener varios GB es muy recomendable.
- Usar almacenamiento muy rápido, aunque sea pequeño y separado del resto de discos.

Checkpoint

checkpoint_timeout:

- Tiempo máximo entre los checkpoints automáticos antes de que se fuerce un checkpoint. Si no se generan suficientes segmentos, checkpoint incondicional.
- Asignar un valor grande implica menor número de checkpoints.
- Tiene un rango entre 30 segundos – 1 día.
- El valor asignado por defecto es 300 segundos (5 min).

Cursos de PostgreSQL para DBA y Developers



**Todo
PostgreSQL**
by Abatic

TODOPOSTGRESQL.COM