

pg_ctl restart vs reload

in PostgreSQL

– Explained the Clement Way

**A friend asked me:
“What’s the difference between
pg_ctl restart and pg_ctl reload?”**

**So, I gave a tech answer... with a
little twist.**

pg_ctl restart – The Full Shutdown

Imagine your manager says:

“Everyone, log off, shut down, leave the building. We’re redoing everything.”

What it really means:

- Stops and restarts the PostgreSQL server
- Terminates all client connections
- Downtime is involved
- Needed for parameters like:

`shared_buffers`

`max_connections`

`wal_level`

Use it when: Configuration changes require a complete restart.

pg_ctl reload – Quiet Adjustments

Now imagine your manager says:

“Carry on. We’re just changing a few settings behind the scenes.”

What it really means:

- Reloads configuration files without restarting the server
- Active sessions continue uninterrupted
- No downtime
- Good for parameters like:

`work_mem`

`log_min_duration_statement`

`statement_timeout`

Use it when: You want changes without disturbing ongoing operations.

Command	Downtime	Session Termination	Typical Use Case
pg_ctl restart	Yes	Yes	Major config changes (memory, WAL, etc.)
pg_ctl reload	No	No	Minor tweaks (logging, timeouts, etc.)

Even databases need a break sometimes.

But if you can solve it with a reload, don't force a restart.

Tech tip shared. Smile delivered.

Thanks to my friend for asking the right question!