Wal_paramter_info

1. wal_level

- Purpose: Determines how much information is written to WAL.
- Values:

minimal – Only enough WAL for crash recovery. Replication is not possible.

replica – Enough WAL to allow streaming replication.

logical – Enough WAL for logical replication.

- Example:

wal_level = replica

- This is needed if you have a standby server or streaming replication.

2. wal_buffers

- Purpose: Memory reserved for WAL before it's written to disk.
- Default: Usually 16MB (can depend on PostgreSQL version).
- Example:

wal_buffers = 16MB

Bigger workloads may benefit from increasing this.

3. wal_writer_delay

- Purpose: How often the WAL writer process flushes WAL buffers to disk.

Default: 200ms

Example:
-
wal_writer_delay = 100ms

Reducing this improves durability but may increase disk writes.

4. checkpoint_timeout

- Purpose: Maximum time between automatic checkpoints.

Default: 5min

- Example:

checkpoint_timeout = 10min

PostgreSQL creates a checkpoint (flushes all dirty pages to disk) every 10 minutes.

5. checkpoint_completion_target

- Purpose: Fraction of checkpoint_timeout that PostgreSQL spreads the checkpoint writes over.
- Default: 0.5 (50%)

Example:

checkpoint_completion_target = 0.7

This spreads writes over 70% of checkpoint interval, reducing I/O spikes.

6. max_wal_size

- Purpose: Maximum total size of WAL files before a checkpoint is forced.
- Default: 1GB
- Example:

max_wal_size = 2GB

If WAL grows beyond 2GB, PostgreSQL triggers a checkpoint to free space.

7. min_wal_size

- Purpose: Minimum WAL files kept even after a checkpoint.
- Default: 80MB
- Example:

min_wal_size = 1GB

Prevents frequent creation/deletion of WAL files.

8. wal_log_hints

- Purpose: Needed for pg_basebackup with hot_standby and certain replication setups.
- Default: off
- Example:

wal_log_hints = on

9. archive_mode & archive_command

- Purpose: Enable WAL archiving for backups.

- Example:

```
archive_mode = on
archive_command = 'cp %p /backup/pg_wal/%f'
```

Every completed WAL segment is copied to backup directory.

10. synchronous_commit

- Purpose: Determines when a transaction is considered “committed”.
- Values:

on – Waits until WAL is safely on disk.

off – Doesn’t wait; faster but less durable.

- Example:

```
synchronous_commit = on
```

11. max_wal_senders

- Purpose: Number of concurrent WAL sender processes allowed (needed for replication).
- Default: 10
- Example:

```
max_wal_senders = 5
```

Use: If you have 2 standby servers, set it at least to 2.

12. wal_keep_size

- Purpose: Minimum WAL files to keep for standby servers. Prevents standbys from falling behind.
- Default: 0 (disabled)
- Example:

```
wal_keep_size = 2GB
```

Use: Useful when network replication lag can happen.

13. hot_standby

- Purpose: Enable read-only queries on standby servers.
- Default: off
- Example:

```
hot_standby = on
```

Use: Allows reporting queries on standby without affecting replication.

14. wal_receiver_status_interval

- Purpose: How often the standby sends feedback to primary about received WAL.
- Default: 10s
- Example:

wal_receiver_status_interval = 5s

Use: Shorter interval = more accurate replication status.

15. wal_receiver_timeout

- Purpose: Timeout for standby to wait for WAL data from primary.

Default: 60s

- Example:

wal_receiver_timeout = 30s

- Use: If network is slow, may need higher value to prevent disconnect.

16. max_slot_wal_keep_size

- Purpose: Limit WAL retained by replication slots.
- Default: -1 (no limit)
- Example:

max_slot_wal_keep_size = 5GB

Use: Prevents WAL from growing indefinitely if standby is down.

17. wal_compression

- Purpose: Compress WAL to save disk space.
- Default: off
- Example:

wal_compression = on

Use: Reduces disk usage but adds CPU overhead.

18. archive_timeout

- Purpose: Maximum time before forcing WAL segment to archive, even if not full.
- Default: 0 (disabled)
- Example:

archive_timeout = 60s

Use: Ensures WAL segments are regularly archived for backups.

19. synchronous_standby_names

- Purpose: List of standby servers for synchronous replication.
- Example:

synchronous_standby_names = 'standby1, standby2'

Use: Transactions won't commit until WAL is confirmed on these standbys.

20. wal_init_zero

- Purpose: Initialize WAL files with zeros instead of random data for security.
- Default: on
- Example:

wal_init_zero = on

Use: Optional, mostly security-related.

Quick tip

WAL-related parameters generally control performance, replication safety, and disk usage. For a primary server: focus on wal_level, max_wal_size, checkpoint_timeout, archive_mode. For standby servers: focus on hot_standby, wal_keep_size, max_slot_wal_keep_size.