AWS re:Invent

DEM118

RDS PostgreSQL configuration for humans [Spanish]

Álvaro Hernández

Founder & CEO
OnGres

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.

`whoami`

- Founder & CEO, OnGres
- 20+ years PostgreSQL user and DBA
- Mostly doing R&D to create new, innovative software on Postgres
- Frequent speaker at PostgreSQL, database conferences
- Principal Architect of ToroDB
- Founder and President of the NPO Fundación PostgreSQL
- AWS Data Hero

PostgreSQL Configuration

- Mainly **postgresql.conf** (here’s most of the meat).
- Authentication: **pg_hba.conf** (and **pg_ident.conf**).
- Replicas: **recovery.conf** (merged with postgresql.conf @ v12).
- Some **initdb** parameters.
- Per-object settings (eg. **fillfactor**).

Advanced stuff:

- Developer parameters.
- Compile-time #defines.

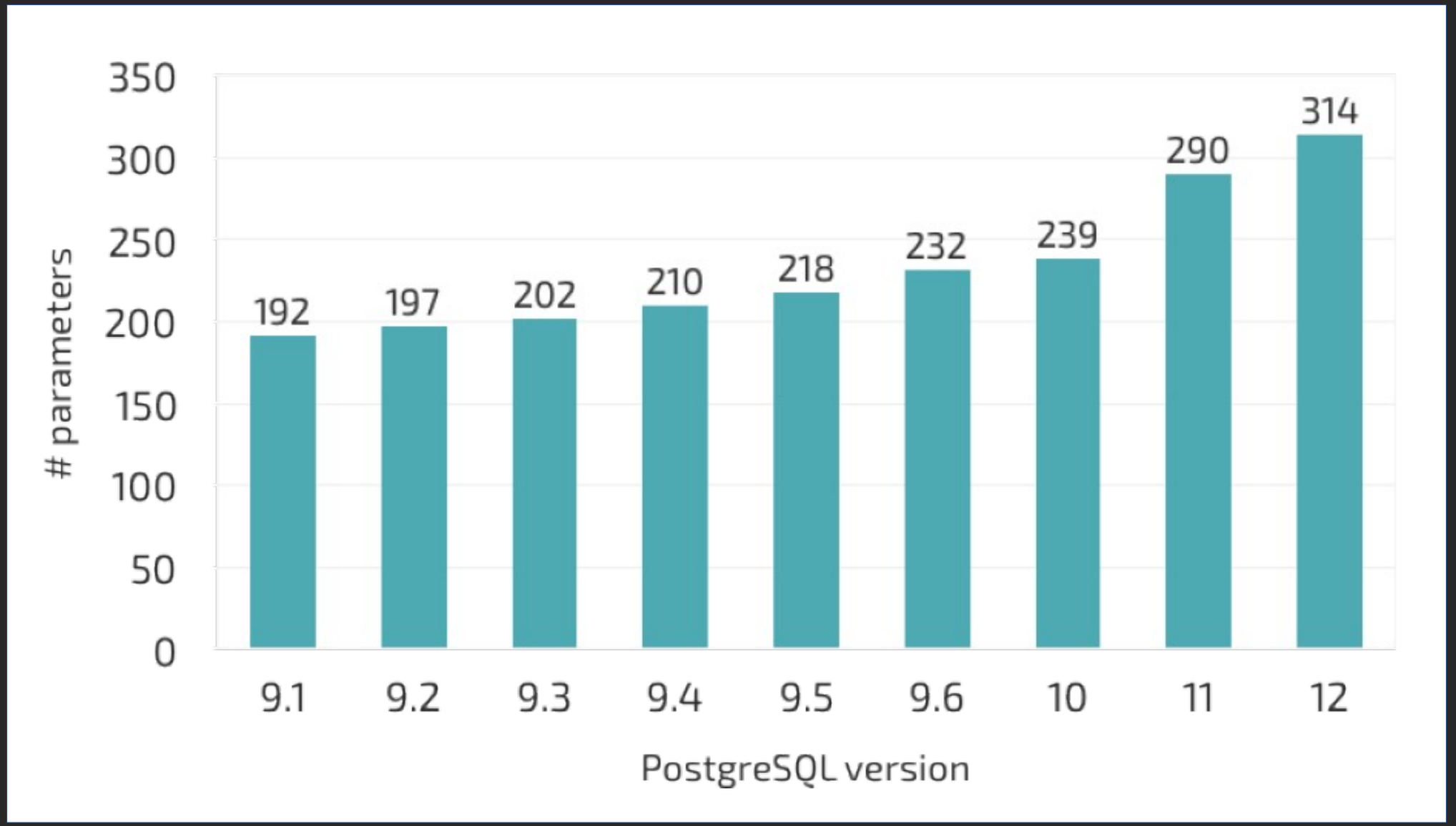
RDS Configuration

- **DB ParameterGroup** (contains postgresql.conf).
- **Authentication** (pg_hba.conf): provided.
- **Replication, HA**: provided.
- No **initdb** options.
- Per-object settings is still possible.

No Advanced stuff:

- ~~Developer parameters.~~
- ~~Compile-time #defines~~

Tunable postgresql.conf parameters



Tunable parameters in RDS

Excluding extensions and dedicated rds parameters:

```
aws --profile $PROFILE --region $REGION \
      rds describe-db-parameters \
      --db-parameter-group-name default.postgres10 \
      | jq -r '.Parameters[] | select ( .IsModifiable ) | .ParameterName' \
      | grep -v '\.' |wc -l

185
```

DB Connections 101 (max_connections)

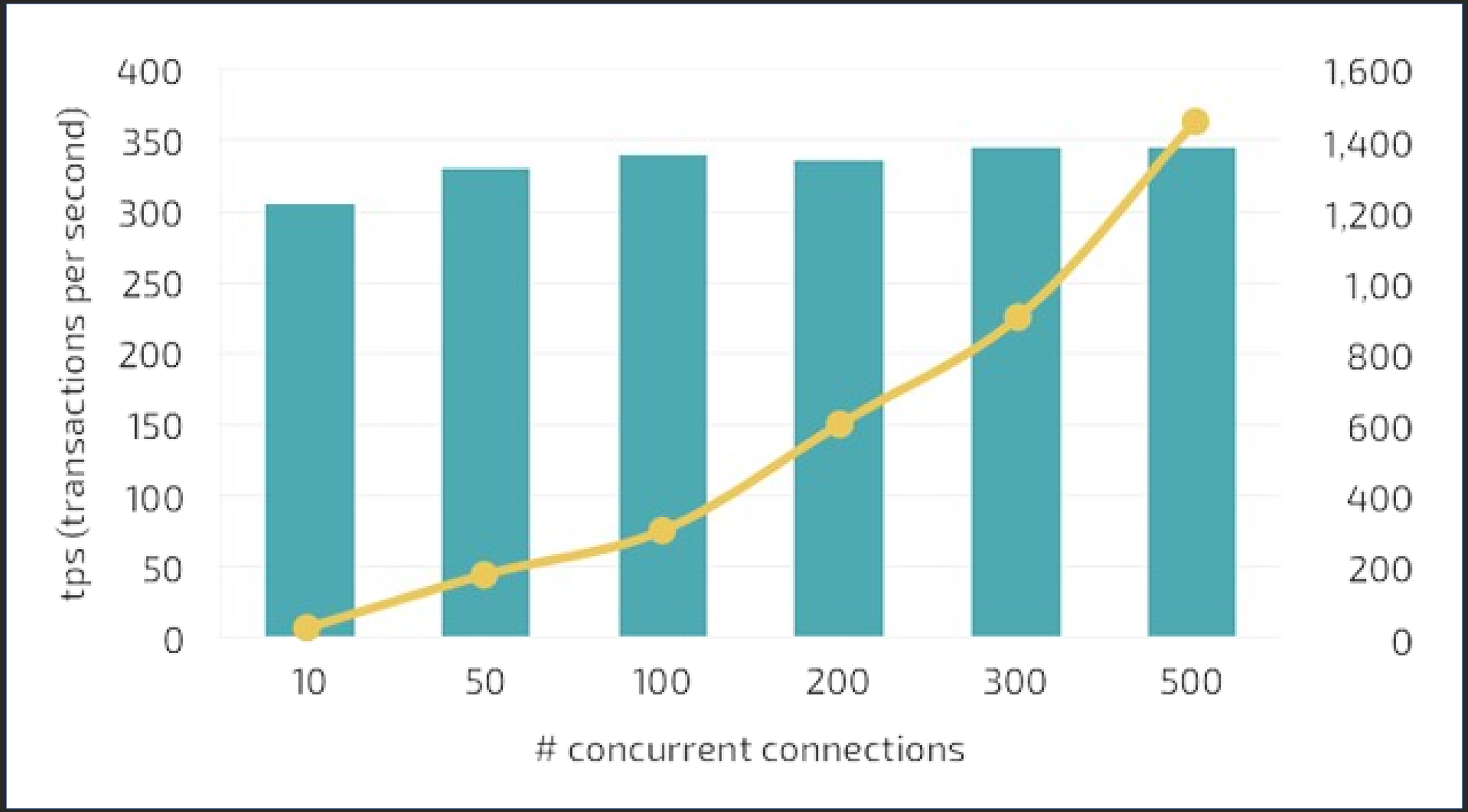
- max_connections is a hard limit.
- PostgreSQL will reject connections over this number.
- Unhappy users!
- Default is LEAST(GB_MEM / 9, 5000).

- **What if I want more than 5000 connections?**
- **Is 3-5K the appropriate sizing?**

DB Connections 101 (max_connections)

- One process per connection (excl. parallelism!)
- One process handled by one core.
- How many cores do you have?
- Sure, you have a multi-process, time-sharing OS but what is the scheduling overhead with many processes?
- Much worse: cache trashing!
- **Solution: use connection pooling (AWS: please include it in RDS!)**

DB Connections 101 (max_connections)



shared_buffers

- The first recommendation everybody tells you.
- Set it on RDS, by default to 1/4th of RAM.
- Done!
- ¼ too low on low memory, too high on high memory.
- **Benchmark, benchmark, benchmark.**
- How high is work_mem * max_connections, maintenance_work_mem, etc?

work_mem

- Max local process memory used for operations like sort and joins in queries.
- Not written in stone: users can SET it overriding its value.
- But if more memory is used, it spills to disk (and may use different algorithm) reducing performance.
- Not the same if you are OLTP, OLAP, DW (small to very large).
- Raise it from defaults, but don’t forget it could be times max_connections.

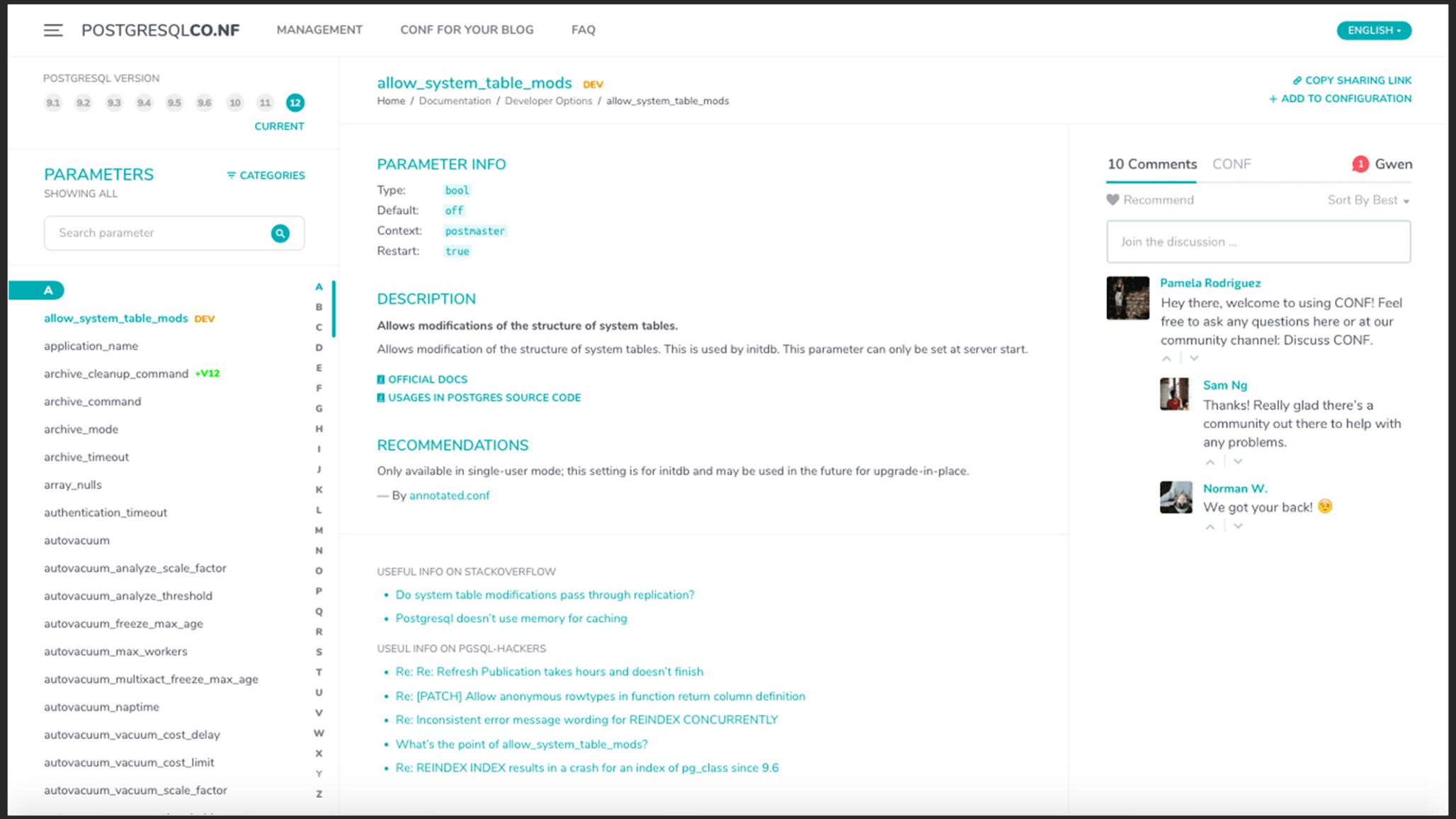
Vacuum and autovacuum

- Almost always too conservative.
- Bloat is one of the most frequent operational burdens.
- Hard to get it right: analyze and re-tune periodically.
- Some parameters are set to “-1” which means “look at these numbers from the vacuum parameters”.
- autovacuum_{vacuum,analyze}_scale_factor: you may override on a per-table level.

Vacuum and autovacuum

- General advice:
- Raise vacuum_cost_limit significantly.
- Reduce autovacuum_vacuum_cost_delay.
- Use more autovacuum_max_workers if you have many cores.

Get documentation from postgresqlco.nf



Thank you!

Álvaro Hernández
@ahachete

© 2019, Amazon Web Services, Inc. or its affiliates. All rights reserved.