PERCONA

What does moving to Pull Requests look like

The NGINX story

Why this presentation?

- Should Postgres use Pull Requests
  - In addition to
  - Instead of

  mailing list patches

- Is there anything to be learned from other projects' experience
  - Did they gain anything?

- NGINX made the move in September 2024
  - Dual run until the end of December

©2025 Percona

Who is telling this story?

- Alastair Turner
  - Technical Evangelist at Percona
- Reformed presales techie
- Database things since the 90s
- Postgres things since 2002
- Interest in how open source projects work with each other and can learn from each other
  - Drupal/Postgres, (PHP, Python, Go)/Postgres, Python wrapping c/Rust, RabbitMQ, Valkey…

©2025 Percona

What are we going to cover?

- Why this presentation?
- Who is telling this story?
- The NGINX open source project

- By the numbers
- It’s not Conway, is there a name for this law?
- Closing thoughts

- Q&A

©2025 Percona

The NGINX Open Source Project

- Web server and reverse proxy
  - 2 clause BSD license
  - Event loop rather than prefork process
  - Open core of a product from F5
- Hosted their own
  - Mercurial
  - Patches accepted via mailing list
- Adopted all things GitHub

©2025 Percona

By the Numbers

	NGINX		Postgres
	Before	After	
Installbase	???		???
LoC	170k, 27k		1.7M, 68k
Change Events	52, 60	66, 121	1.4k, ???
Participants	150	180	3k
Proposers	± 10	± 25	312 (229 + 83)
Contributors	± 10	± 15	463

©2025 Percona

New Contributions and Contributors

- Accepted
  - Docs, comments, message strings
  - Smaller, deep, but very narrow patches
  - Multi-project contributors

- Rejected or in limbo
  - Docs, comments, message strings
  - Nit picking over (enforcing) standards vs (deprecating) usage in the wild

©2025 Percona

The tools shaping the process?

- Just not used to the tools
  - PRs withdrawn and resubmitted with squashed commits

- Tools and culture
  - Large volume of comments, and updates, on PRs for stylistic issues
    - Would have been fixed by the committer while pushing in Postgresland
  - Where committers have force-pushed changes to the patch branch before merging the diffs are unreadable
    - Polluted by changes between patch submission and merge

©2025 Percona

Closing Thoughts

- Some evidence for acceptance of GitHub PRs increasing first or one-off contributions
  - Git in general, or specifically GitHub?
  - Flows for bigger, hand polished, incrementally accepted patches don't seem well supported
- Some noise generated through tool friction
  - "Please read the history of this topic" will become common noise
- The noise stays front-and-centre for ever
  - Cleanup runs on long idle items?
- Do the review flow and statuses create a new role?
  - Analogous to a commitfest manager?

©2025 Percona

Thoughts / Questions?

PERCONA

Thank you!