



GitLab

PostgreSQL at [GitLab.com](https://gitlab.com)

Speaker — Jose Cores Finotto

- I work with the Infrastructure team at GitLab.
- I have been a part of the GitLab team since September 2018.
- Background in large organizations with extensive experience in Infrastructure, especially in relational databases.



Speaker — Alexander Sosna

- Senior Database Reliability Engineer in the GitLab infrastructure team
- Just joined GitLab in October 2021
- Strong background in Open Source Infrastructure with a focus on databases and PostgreSQL
- [<alexander@sosna.de>](mailto:alexander@sosna.de)



Agenda

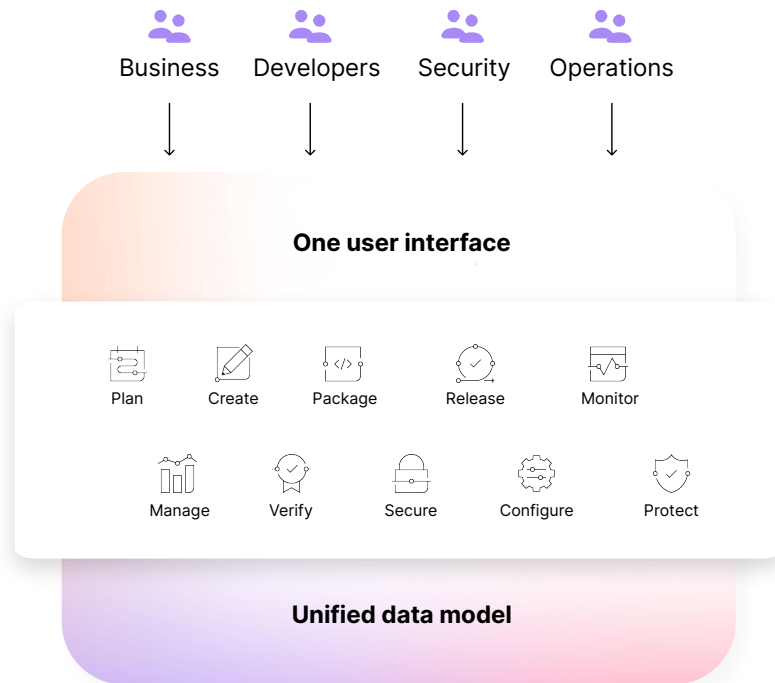
- GitLab
- Key Specs
- Architecture
- Decomposition
- Links and Resources





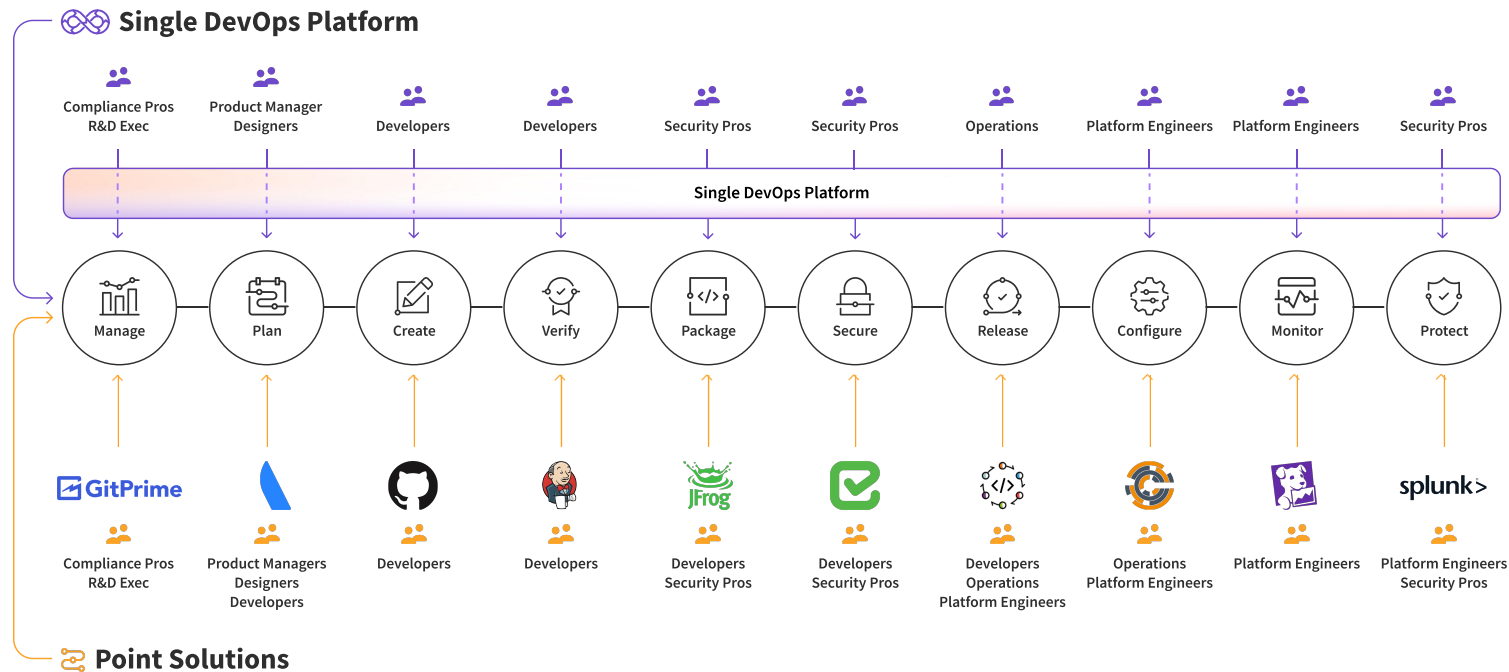
The One DevOps Platform for software innovation

- Project planning
- Source code management
- Continuous integration
- Infrastructure configuration
- Incident monitoring
- Application security
- And so much more...



Collaborate across personas

Deliver faster, more efficiently, with reduced risk

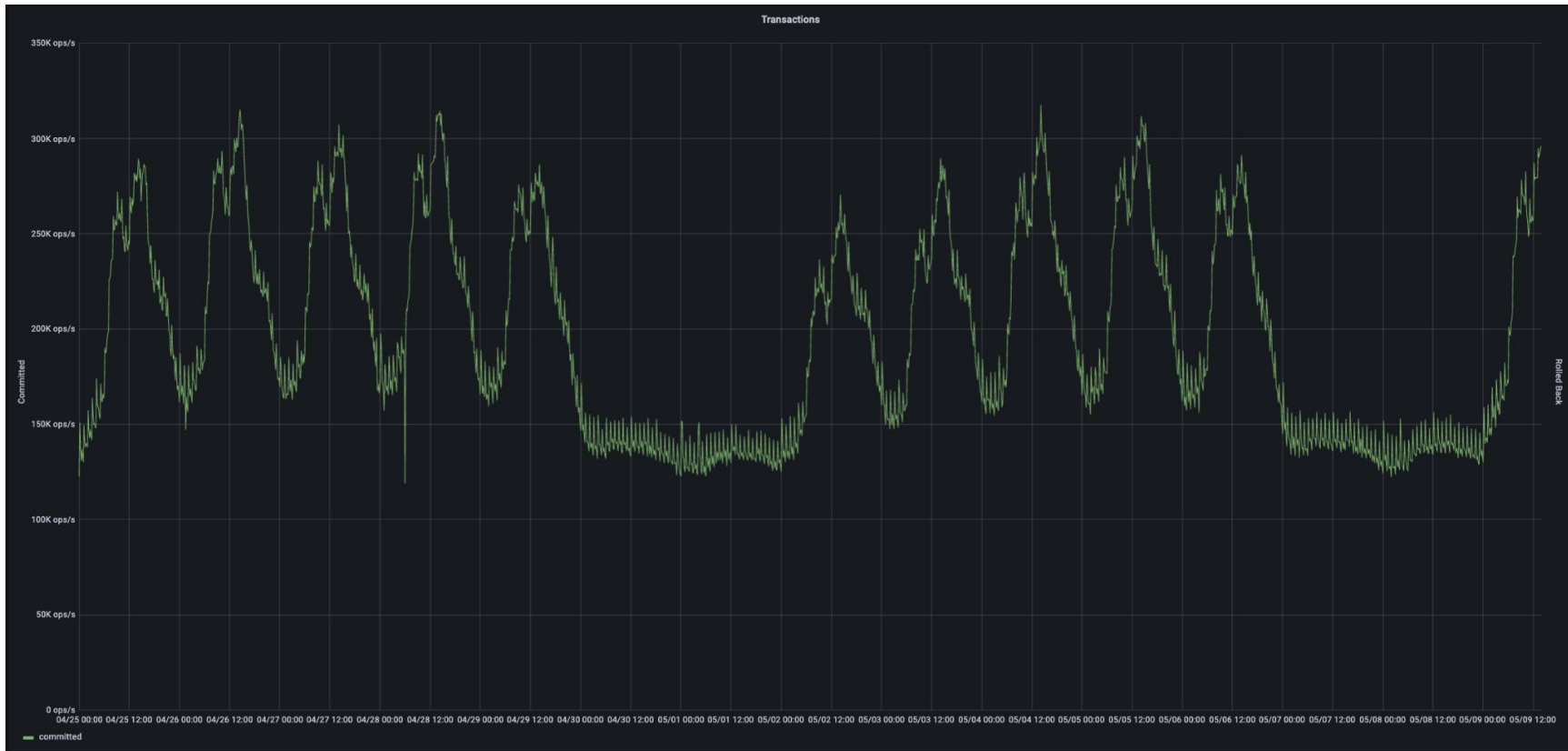




Key Specs

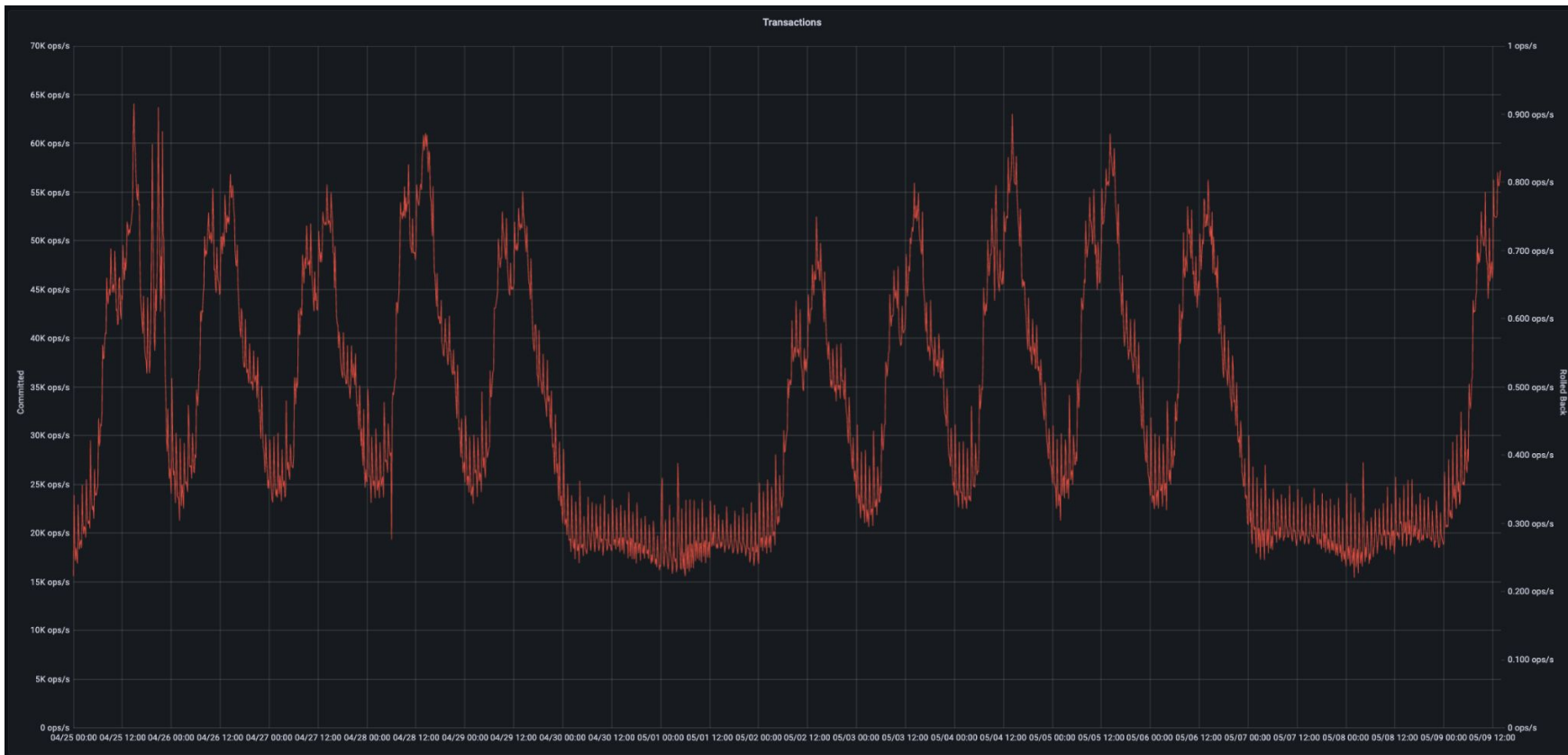
Key Specs - Read Transactions per Second

Between ~130.000 and ~300.000 TPS on the standbys



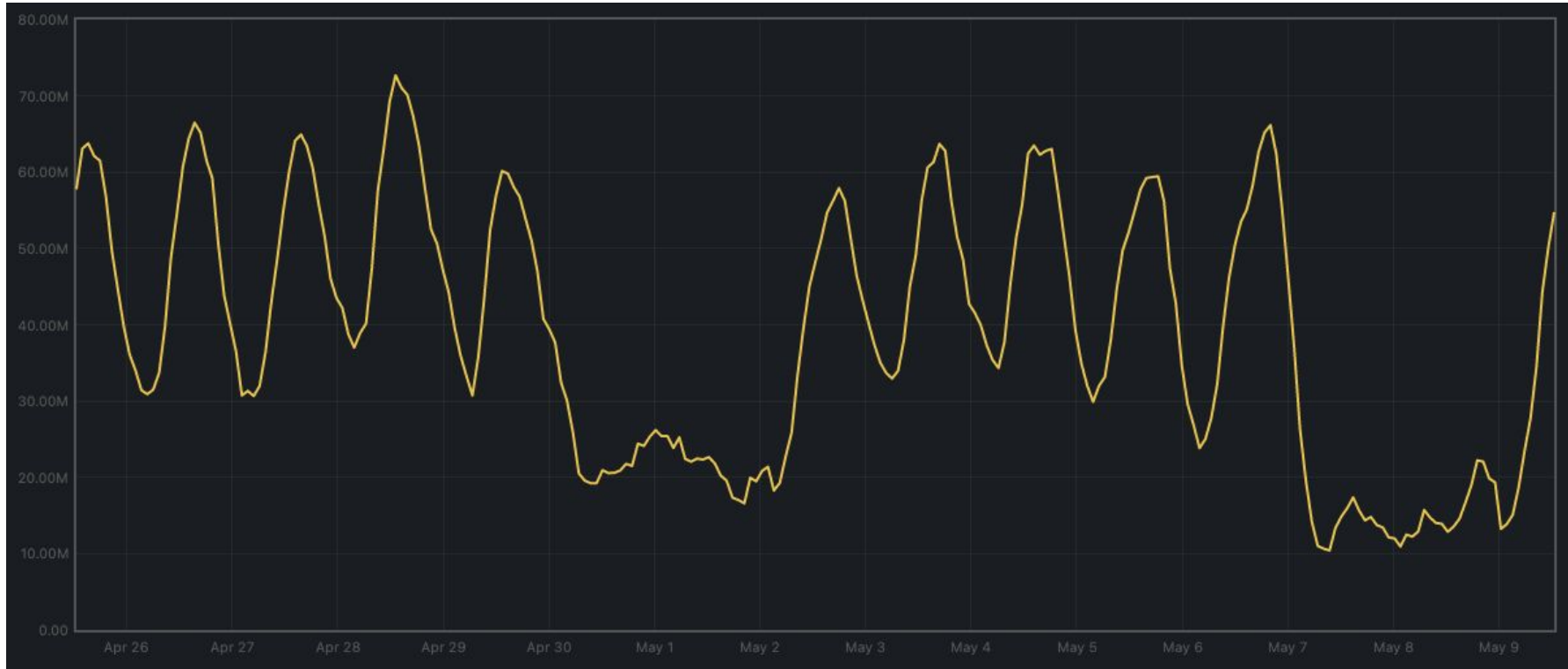
Key Specs - R/W Transactions per Second

Between ~20.000 and ~60.000 TPS on the primary



Key Specs - WAL creation

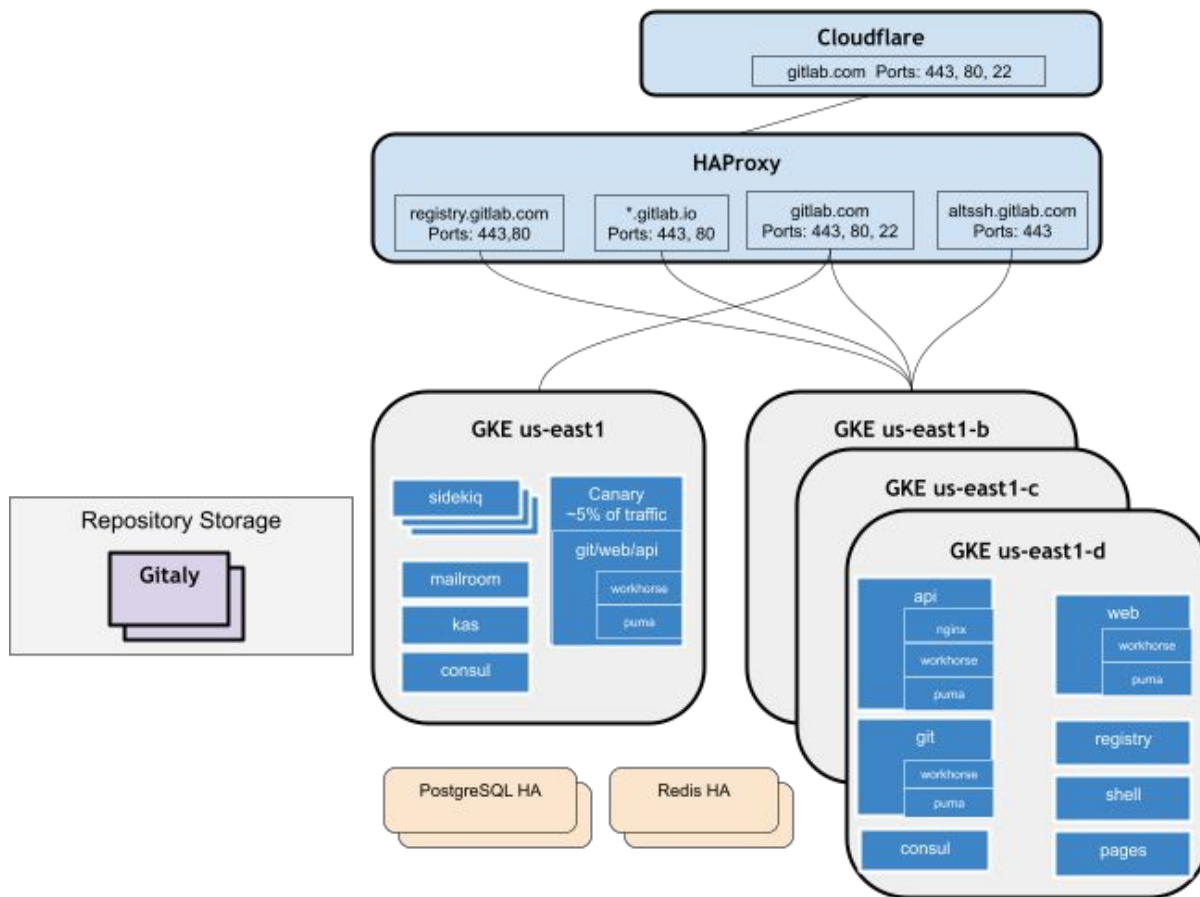
rate(pg_xlog_position_bytes, [6h]) between ~15MB/s and ~65MB/s WAL creation at all times



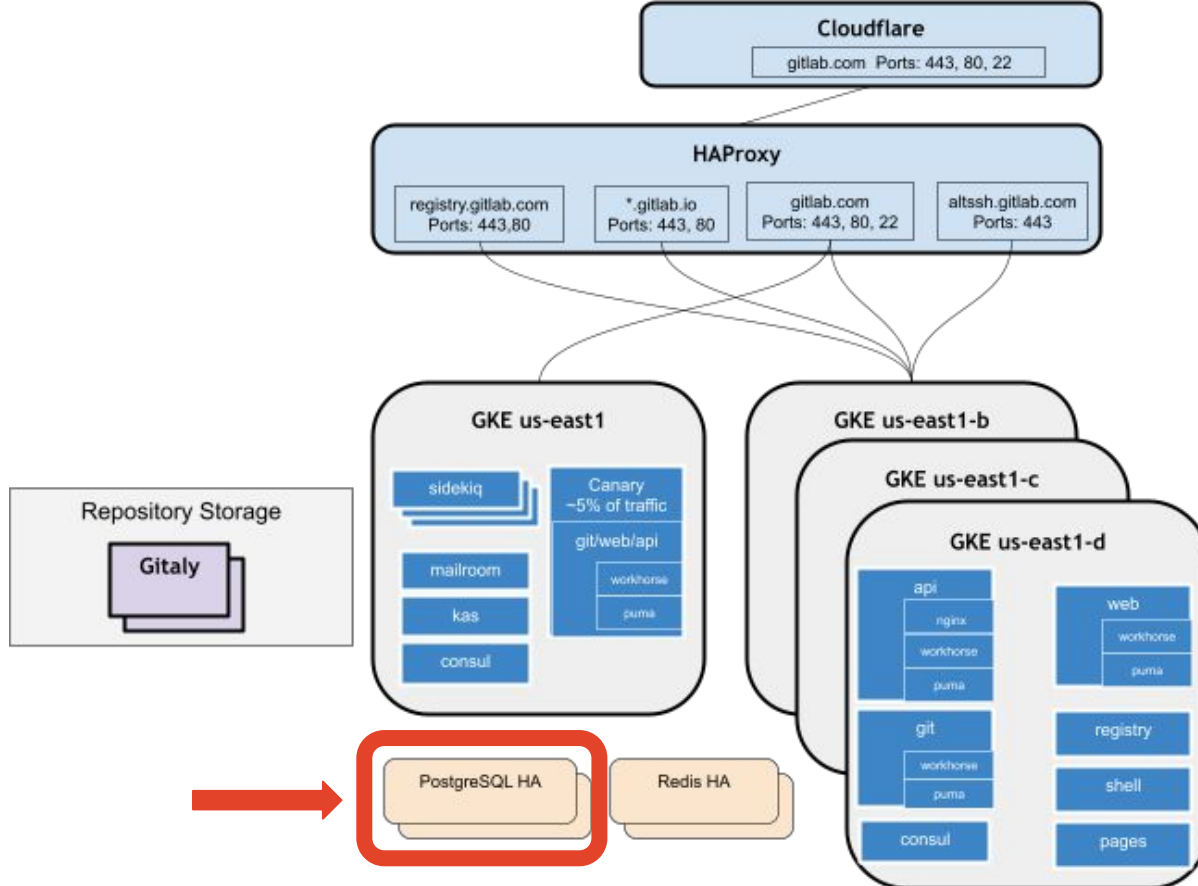


Architecture

GitLab.com Architecture

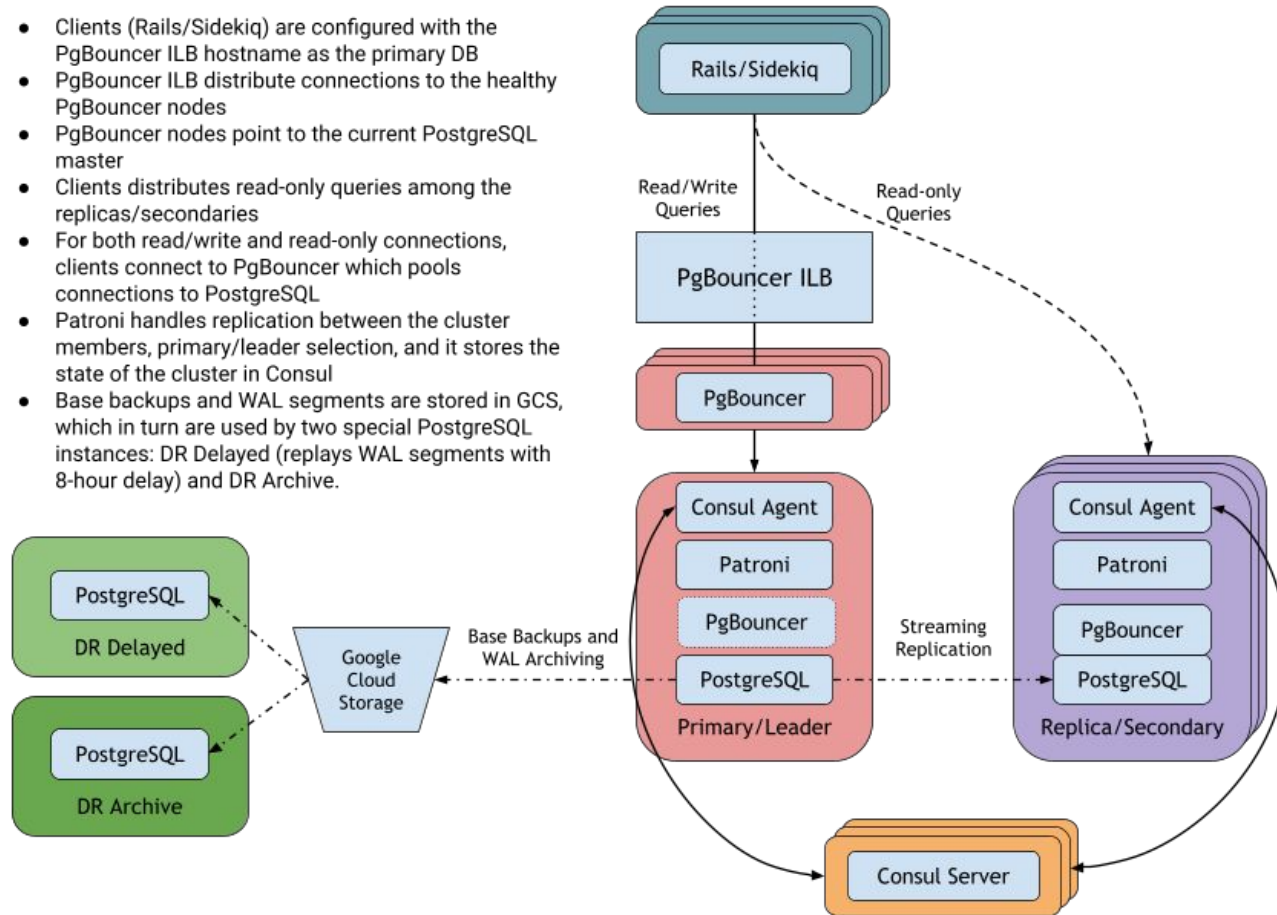


GitLab.com Architecture

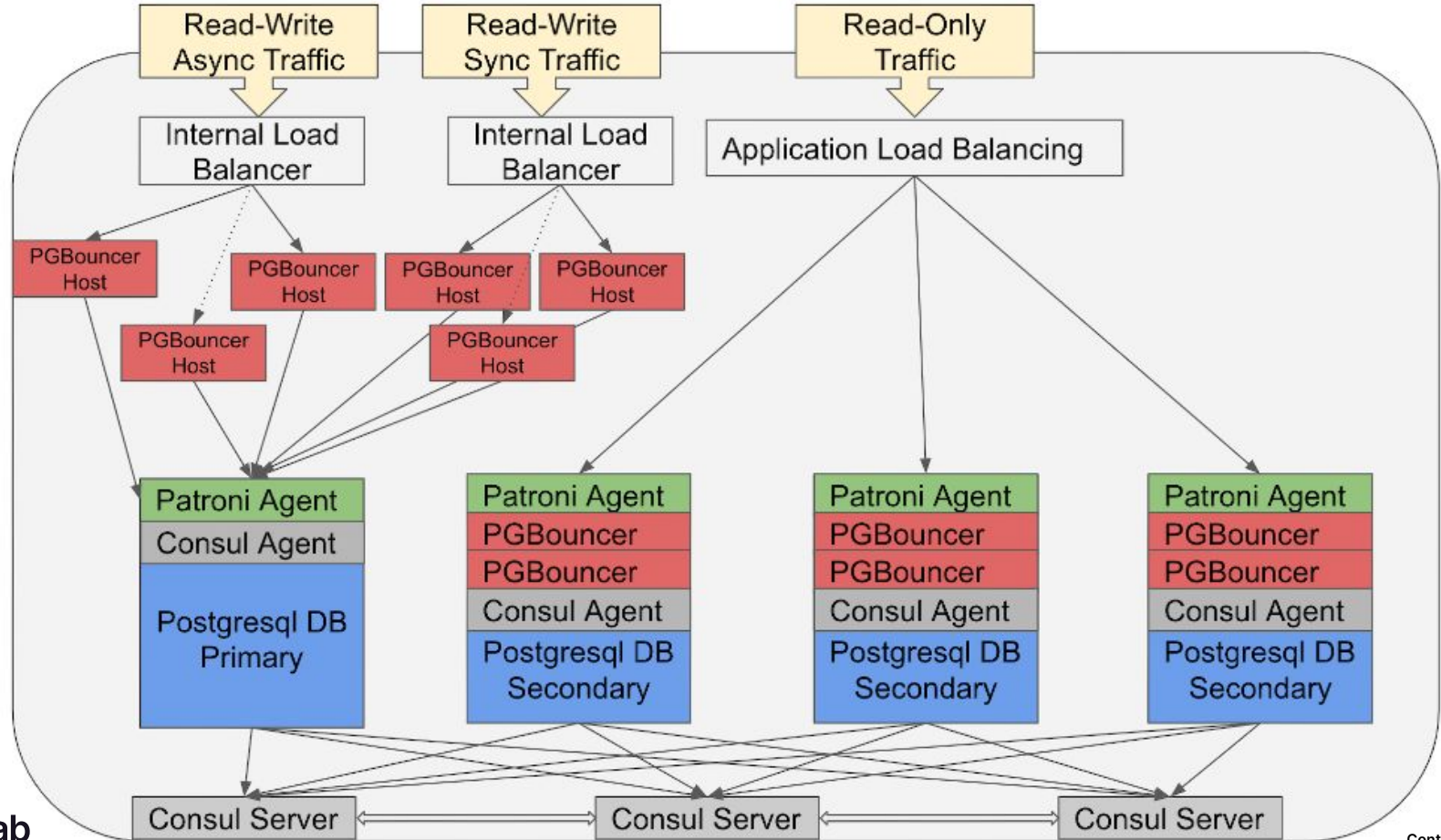


PostgreSQL Architecture

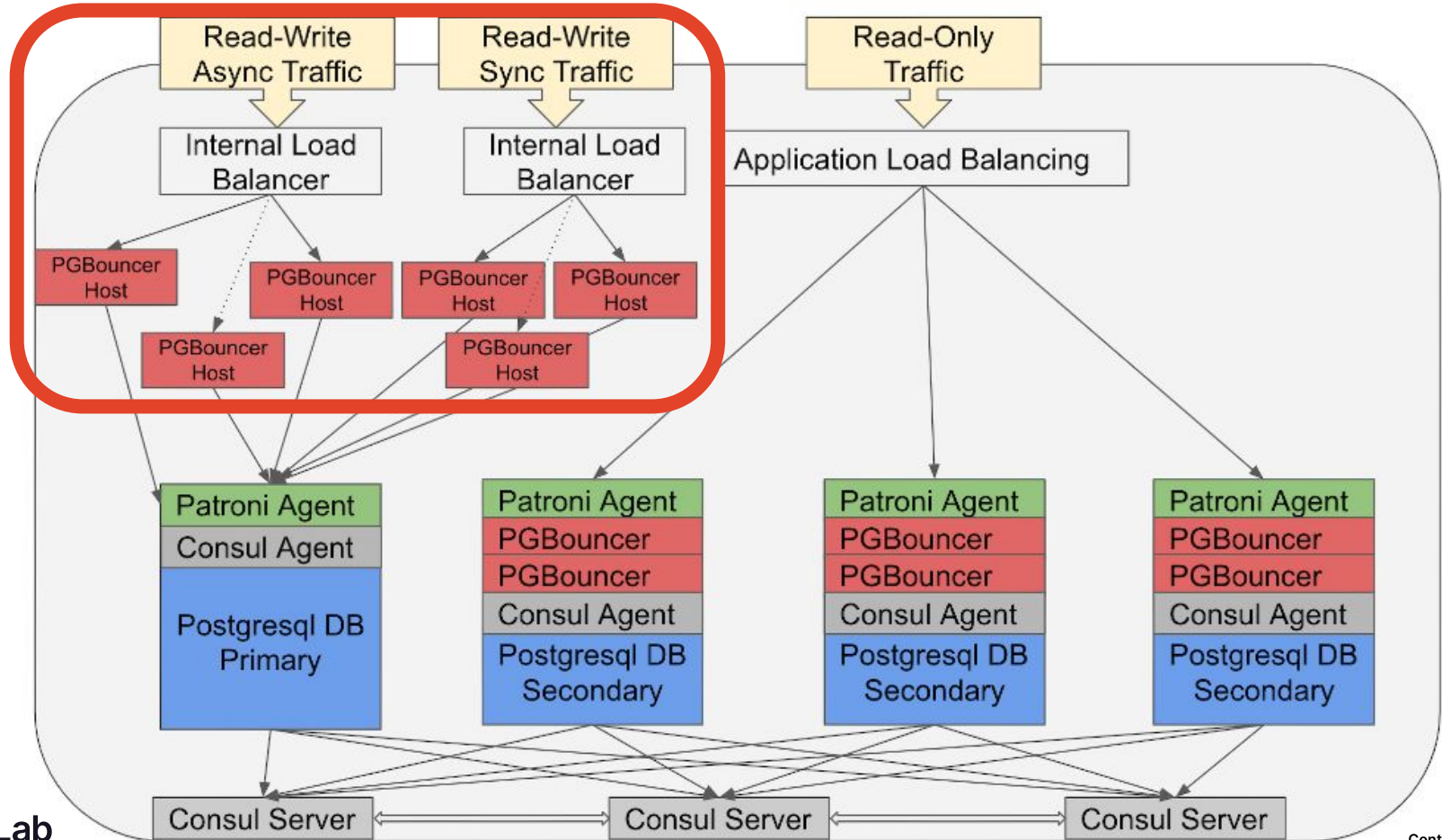
- Clients (Rails/Sidekiq) are configured with the PgBouncer ILB hostname as the primary DB
- PgBouncer ILB distribute connections to the healthy PgBouncer nodes
- PgBouncer nodes point to the current PostgreSQL master
- Clients distributes read-only queries among the replicas/secondaries
- For both read/write and read-only connections, clients connect to PgBouncer which pools connections to PostgreSQL
- Patroni handles replication between the cluster members, primary/leader selection, and it stores the state of the cluster in Consul
- Base backups and WAL segments are stored in GCS, which in turn are used by two special PostgreSQL instances: DR Delayed (replays WAL segments with 8-hour delay) and DR Archive.



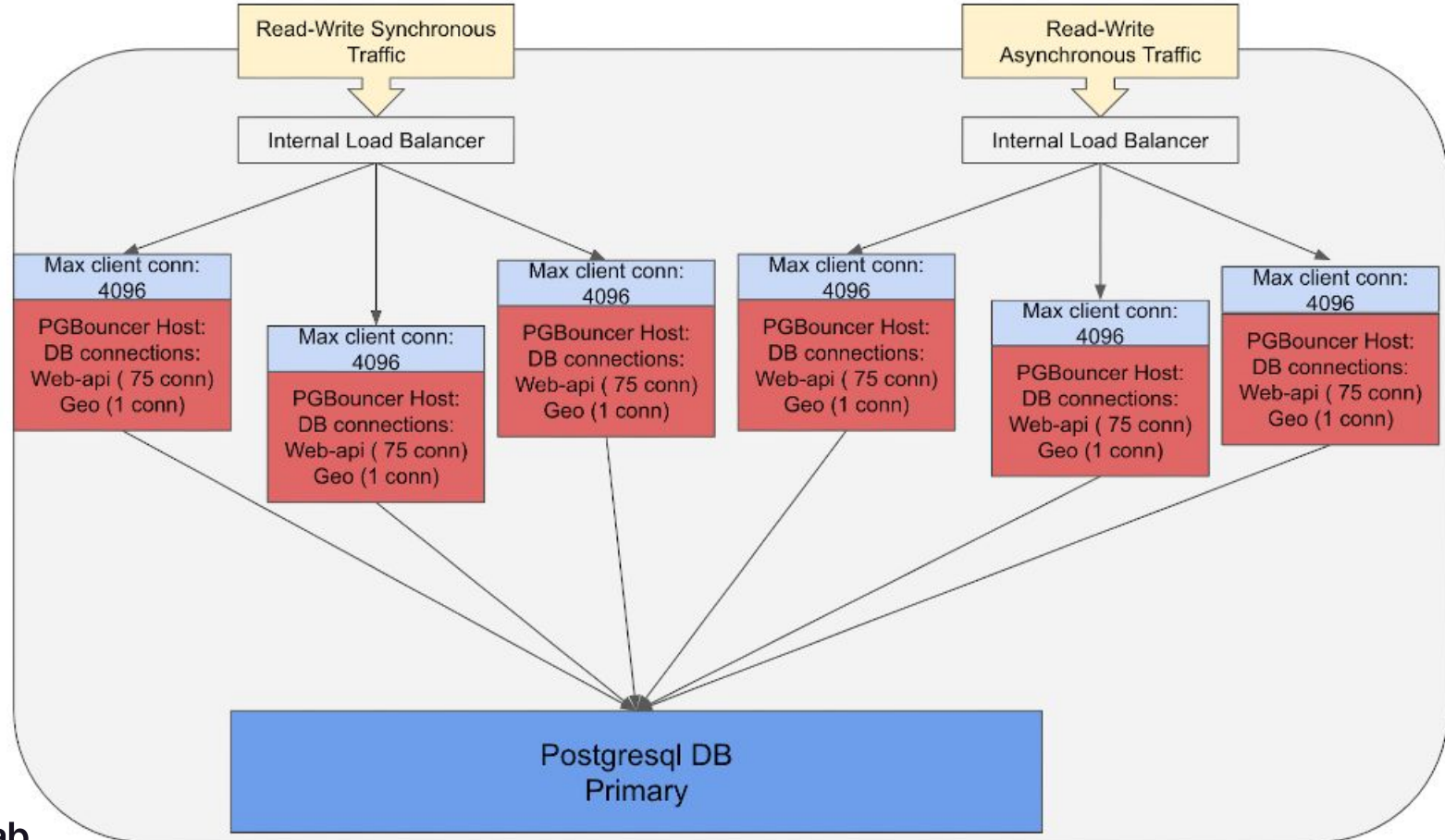
PostgreSQL Architecture - Data Diagram



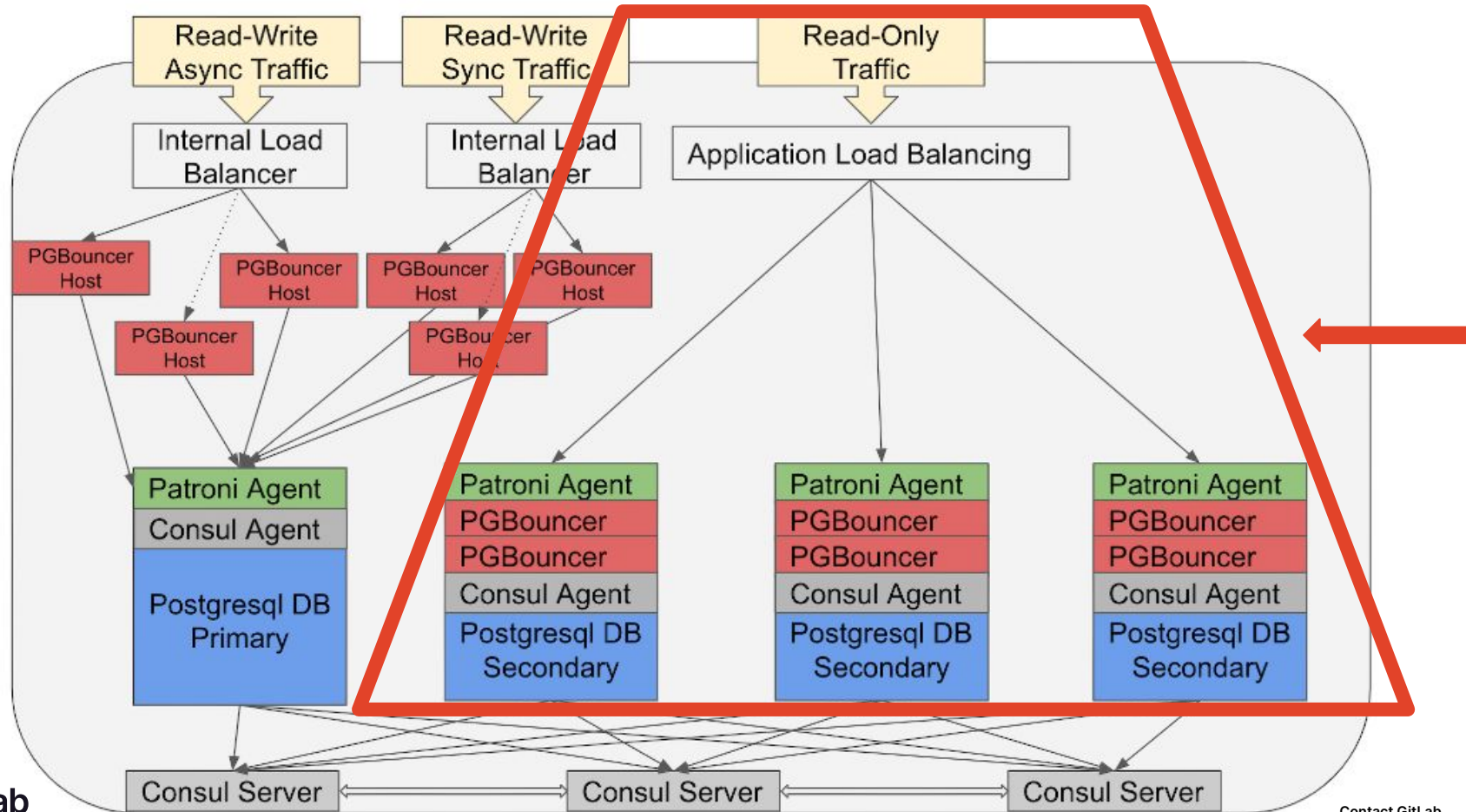
PostgreSQL Architecture - Data Diagram



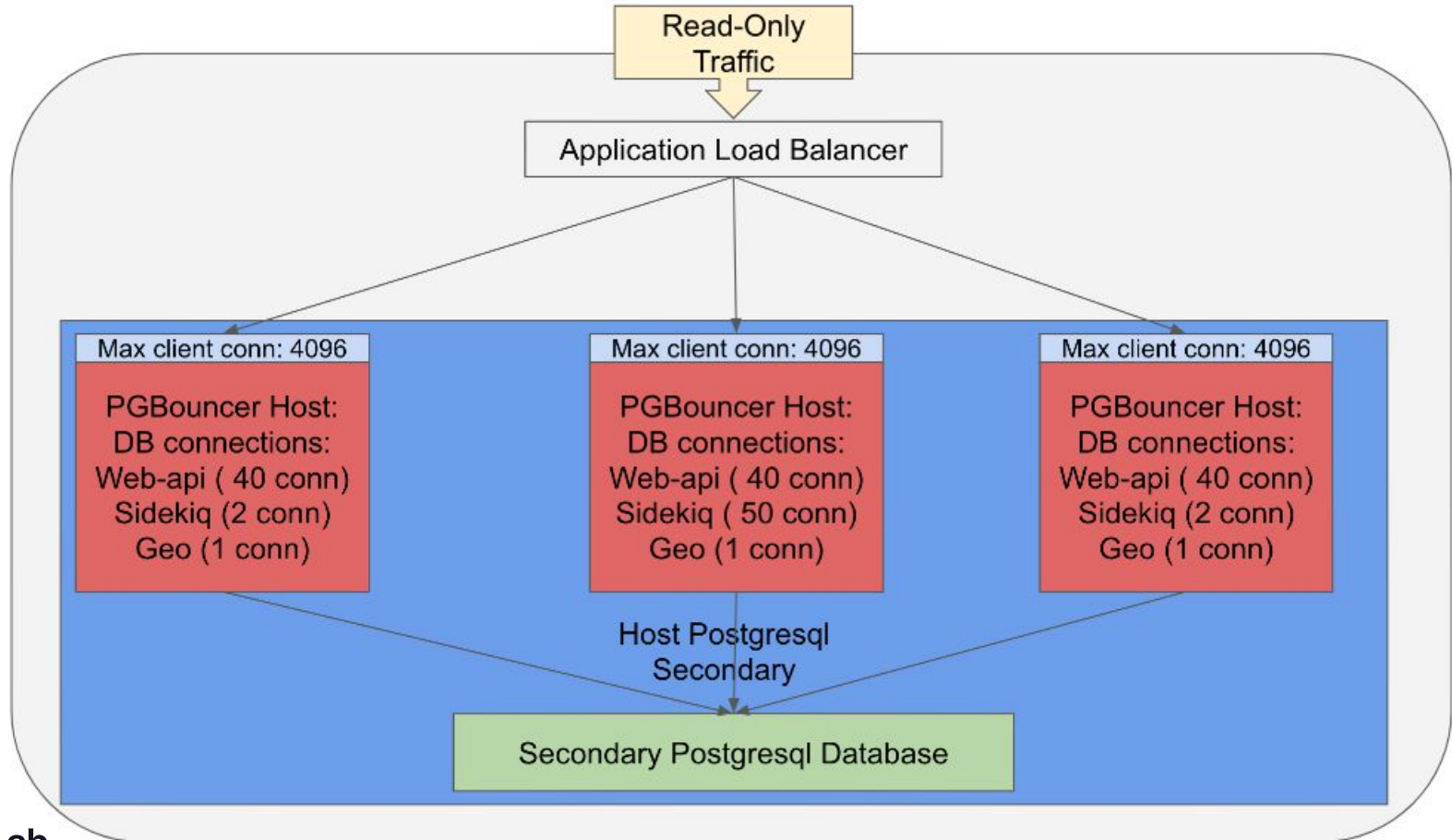
Read and Write Requests



PostgreSQL Architecture - Data Diagram



Read-Only Requests





Decomposition

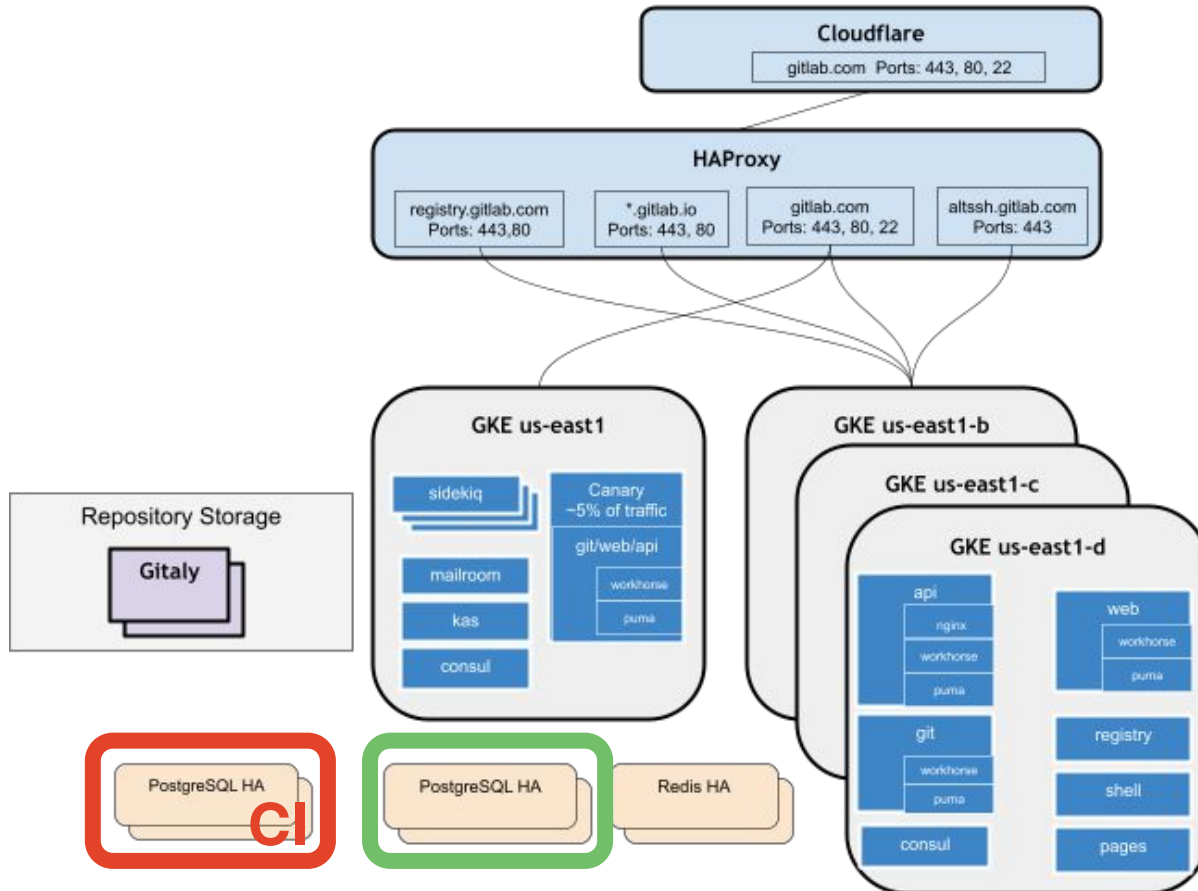
How can we improve?

- **Use bigger machines, currently: *n1-standard-96***
- **More hot standbys for read scaling**
- **Separate different workloads**
- ...





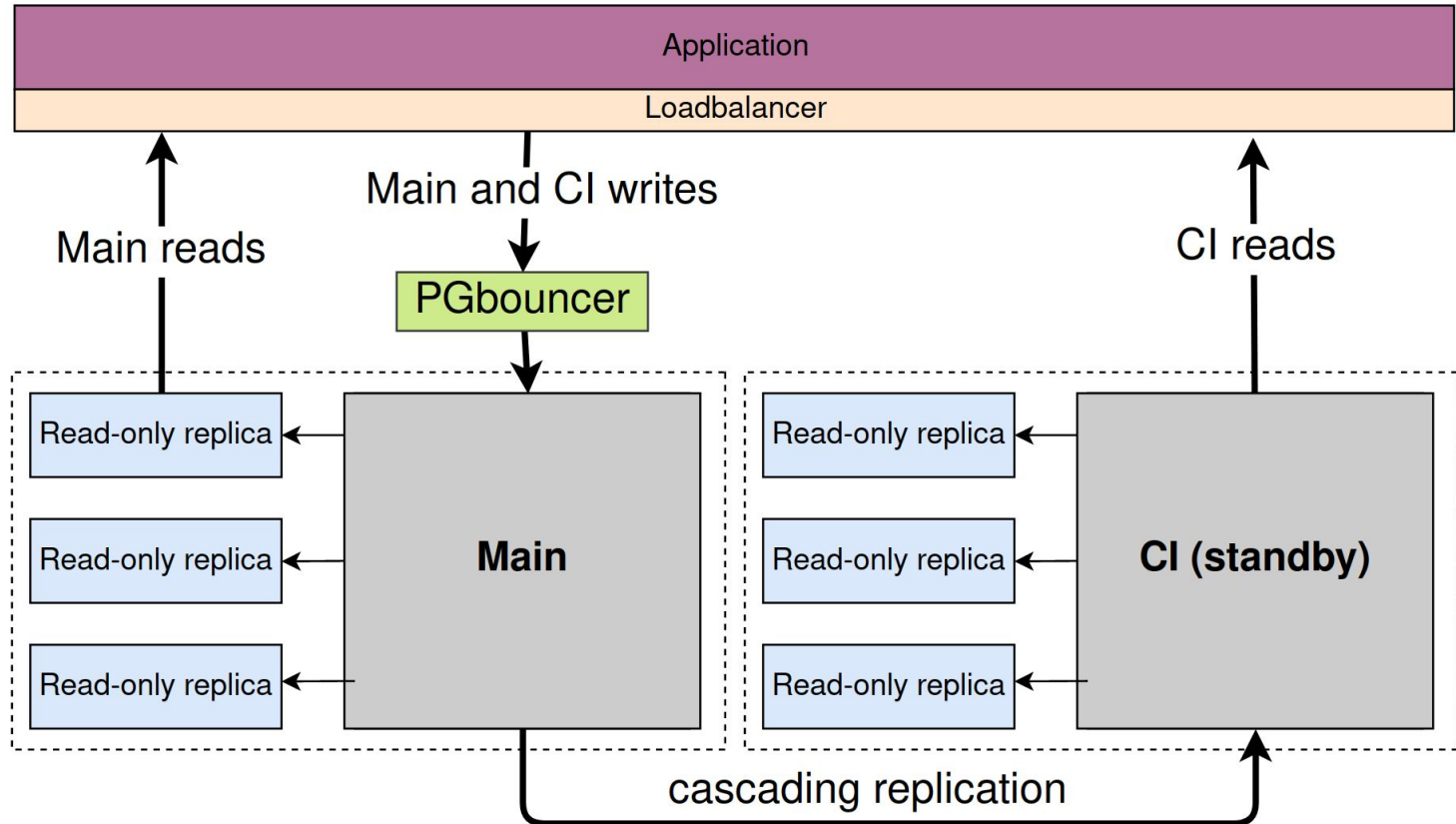
GitLab.com Architecture



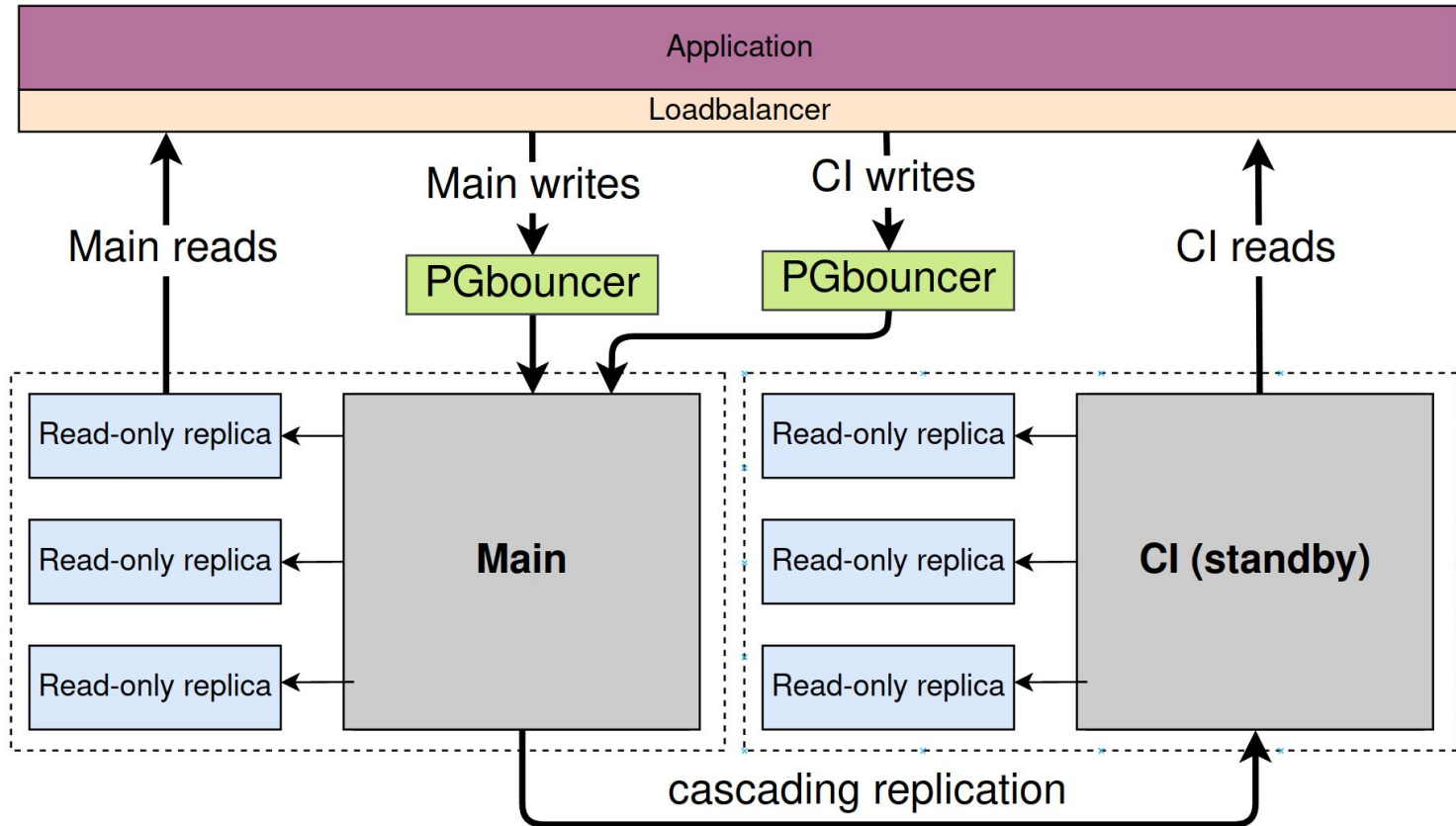
Decomposition Benefits

- **Better write scaling through dedicated masters**
- **Better tuning for the specific workload**
- **Significant less updates on the main instance**
 - **Faster backup**
 - **Faster restore in case of disaster**
 - **Less stress on replication and archive**
 - **New standbys quicker to create and catch up**
 - **Lesser TXID consumption and fewer wraparounds**

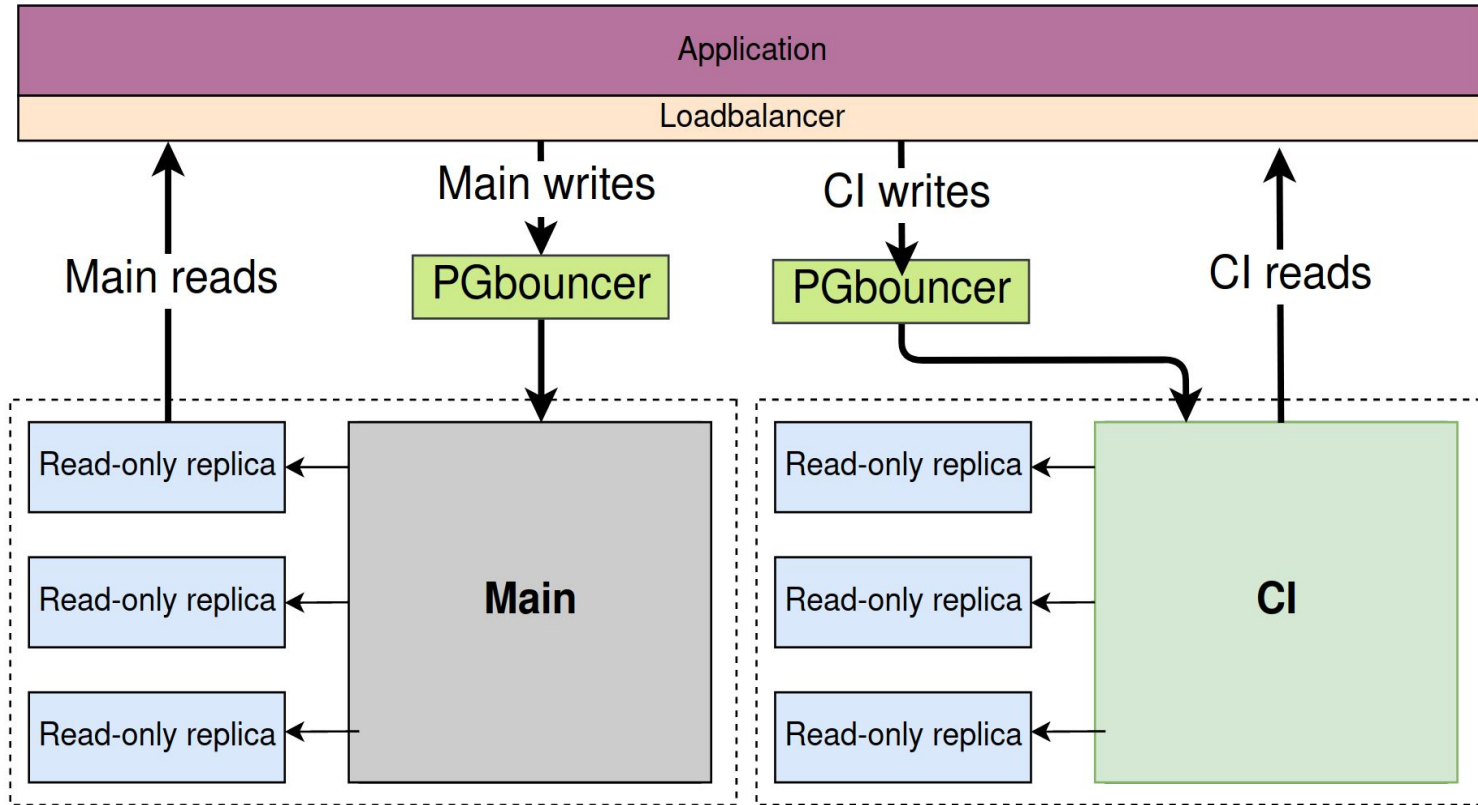
Decomposition - Current State



Decomposition - CI Writes Endpoint



Decomposition - Promotion and Switchover





Resources

Resources

- GitLab: about.gitlab.com
- The Handbook: about.gitlab.com/handbook
- Our RDBMS: about.gitlab.com/handbook/engineering/infrastructure/database
- Configuration: gitlab.com/gitlab-com/gl-infra/chef-repo
- Decomposition: gitlab.com/gitlab-com/gl-infra/production/-/issues/6440
- Jose Cores Finotto: about.gitlab.com/company/team/#Finotto
- Alexander Sosna: about.gitlab.com/company/team/#alexander-sosna



Questions?!