

PGMQ

Message Queue on Postgres

<https://github.com/tembo-io/pgmq>

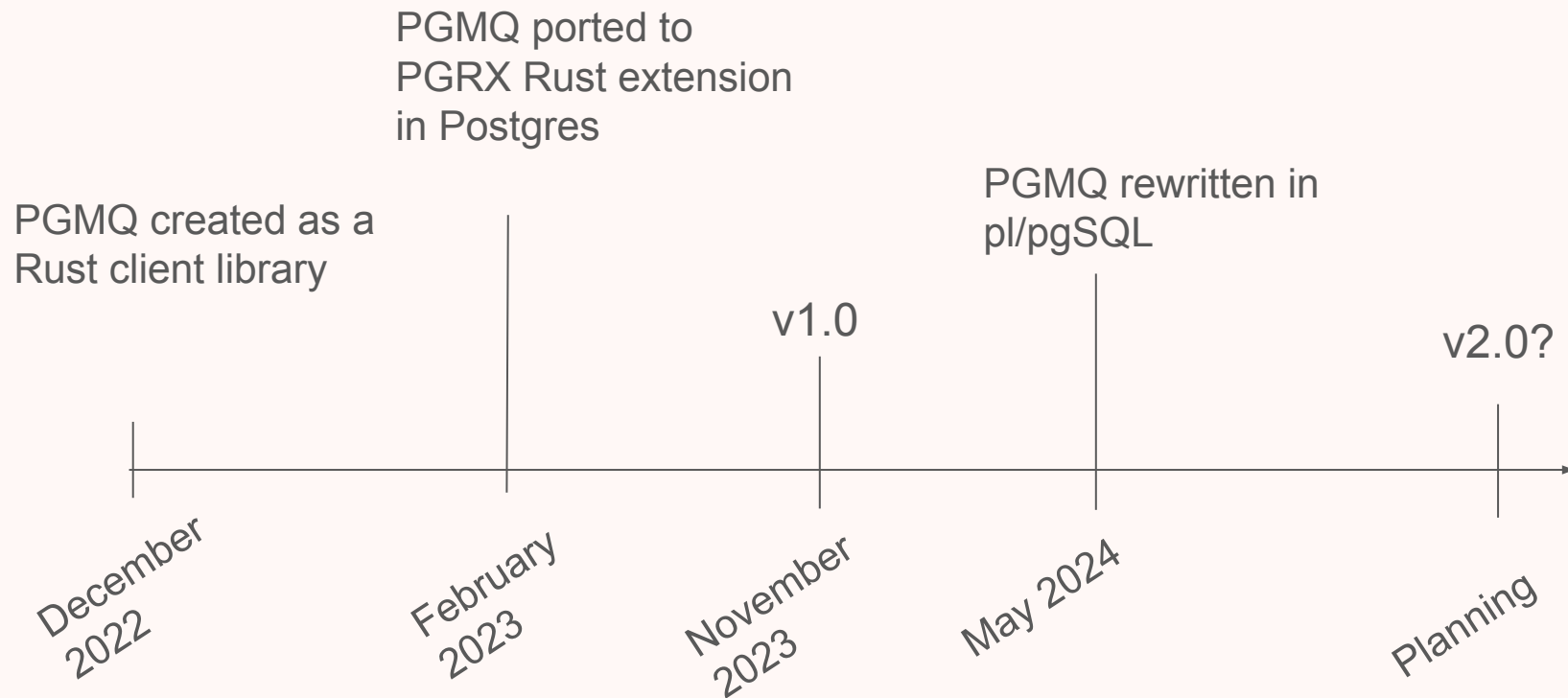
Me - Adam Hendel

- Founding engineer at Tembo
- Postgres user for 10+ years
- Extensions for 2 yr

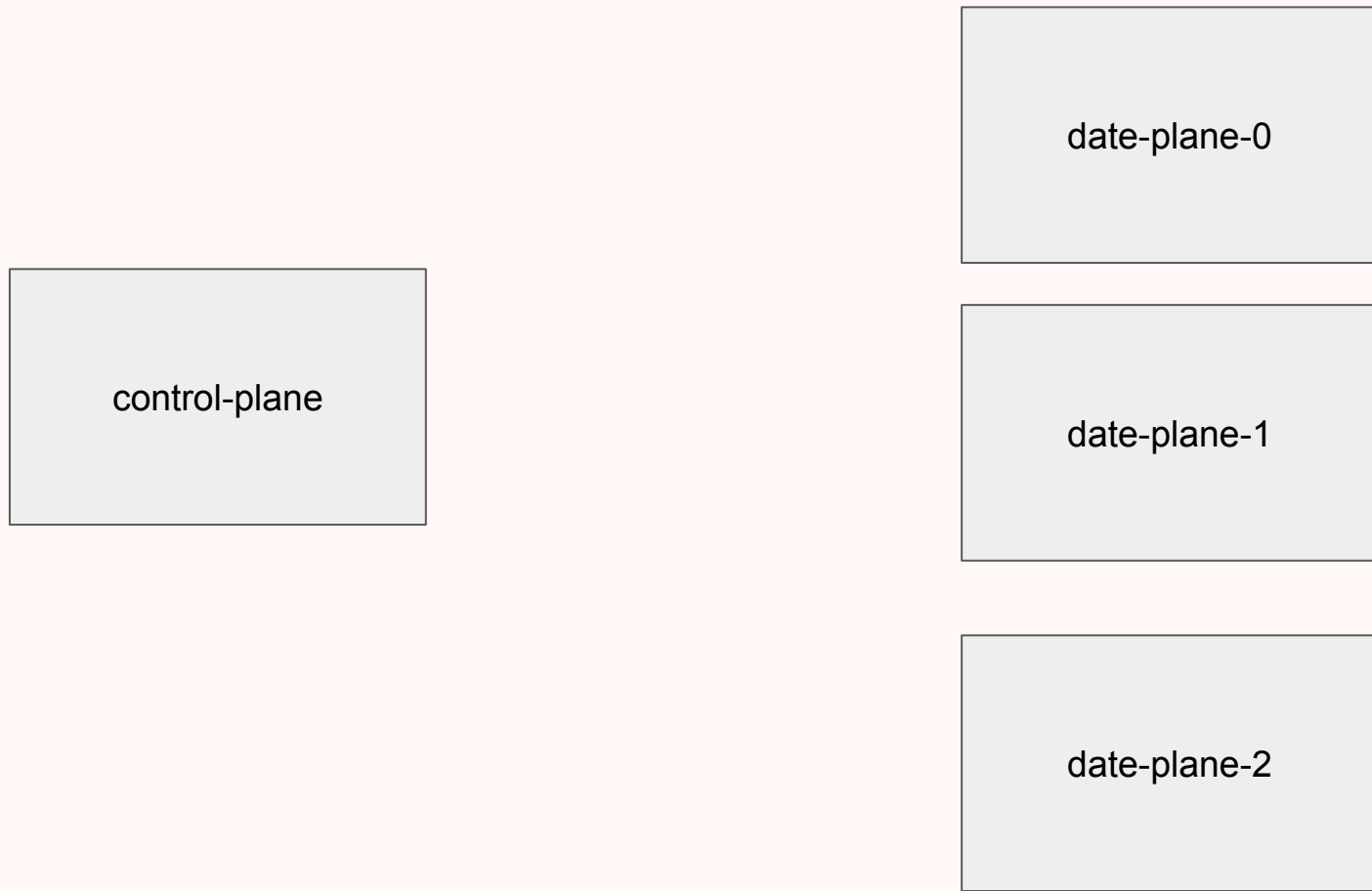
About PGMQ

- Message queue built as a Postgres extension
- Pure SQL, stateless no background worker or dependency outside of Postgres
- Delivery semantics like SQS & RSMQ
 - **Visibility timeout** - a period where a message is “in Postgres but unable to be read”
 - **Ack** messages by deleting (forever) OR archiving (moving message to an archive table)
- Low level API - Not a replacement for Celery but there is some ongoing work to make Celery run on top of PGMQ
- API is designed to be **SIMPLE**

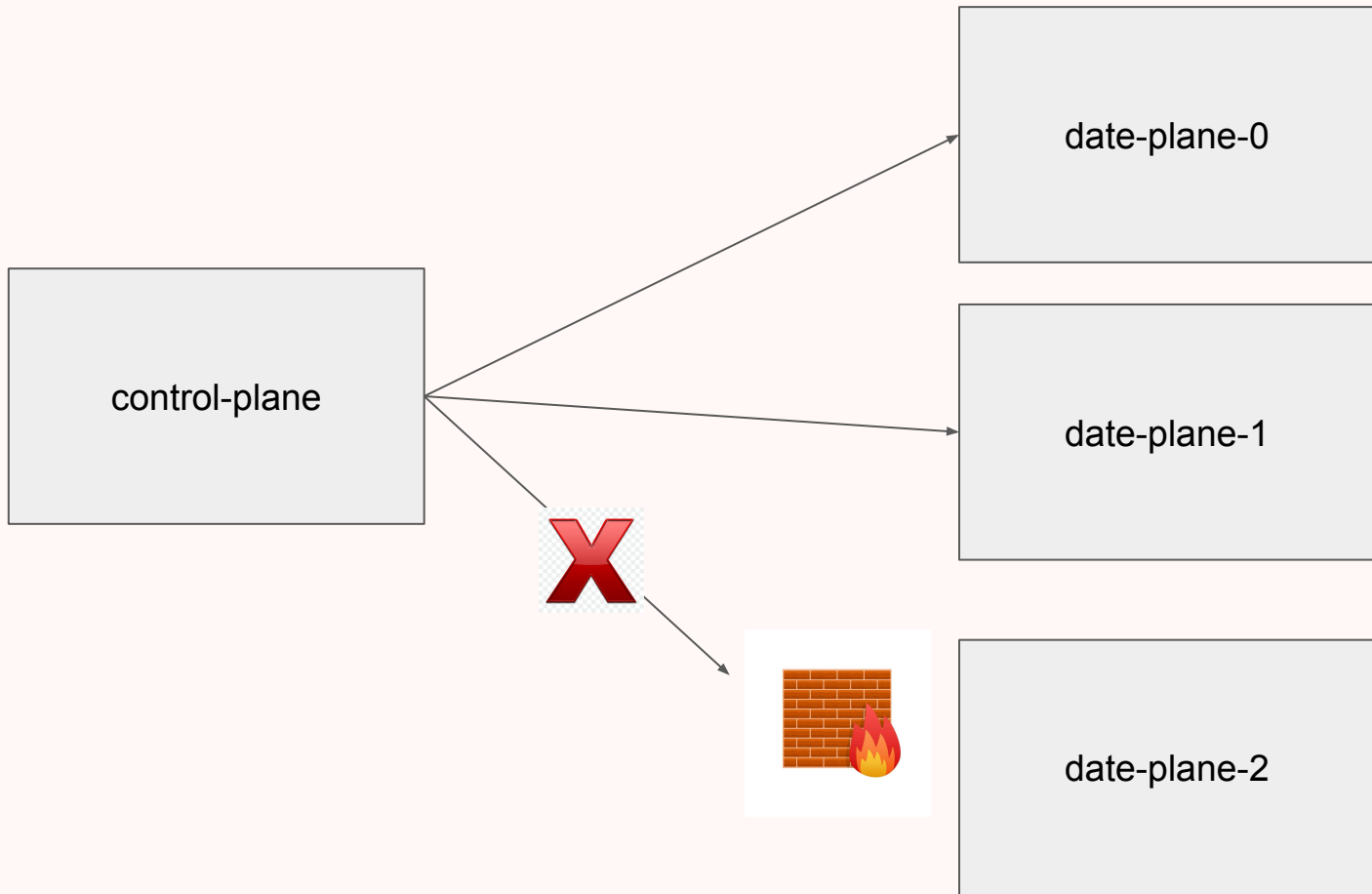
Timeline

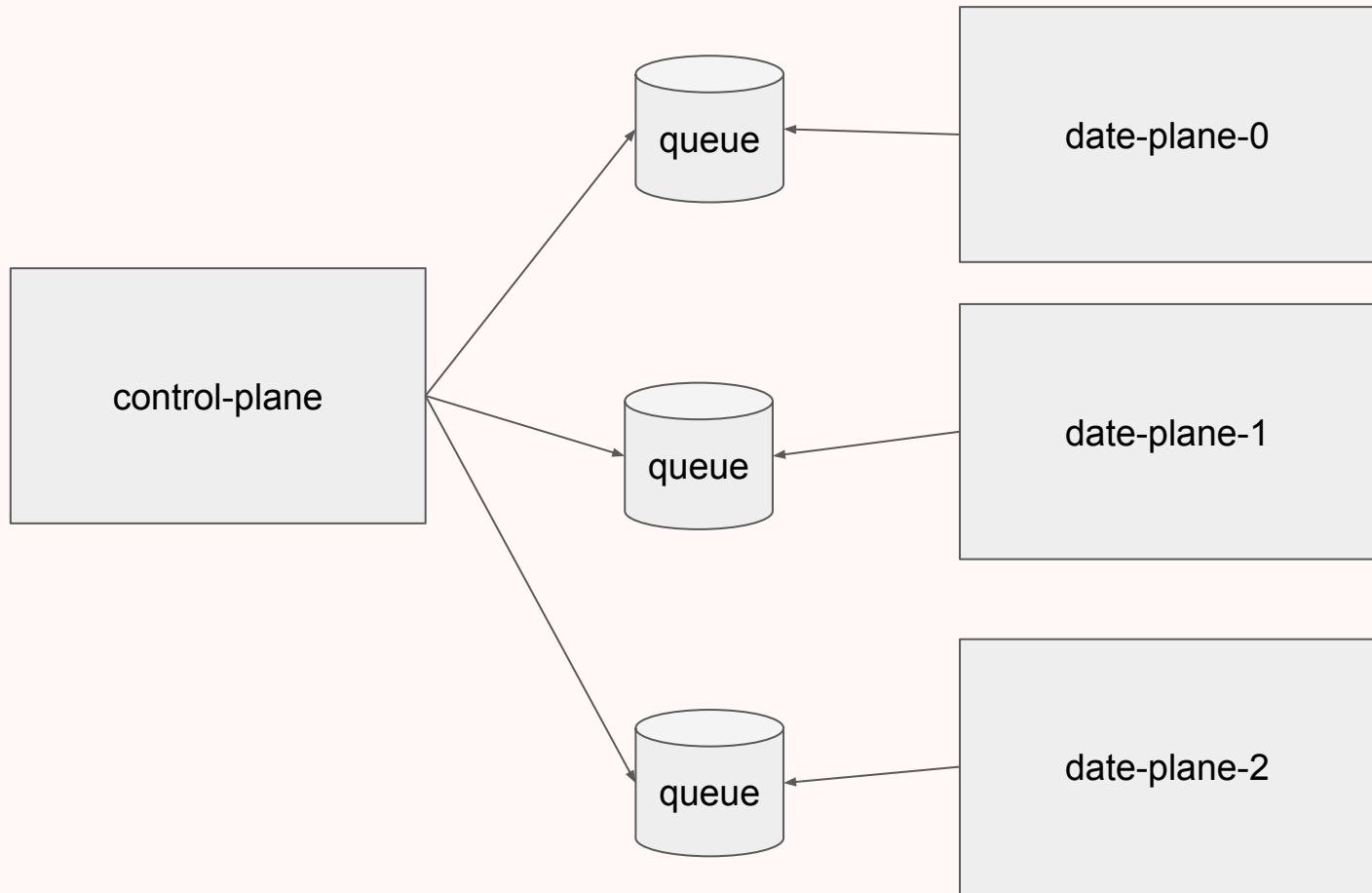


Tembo Needed a message queue

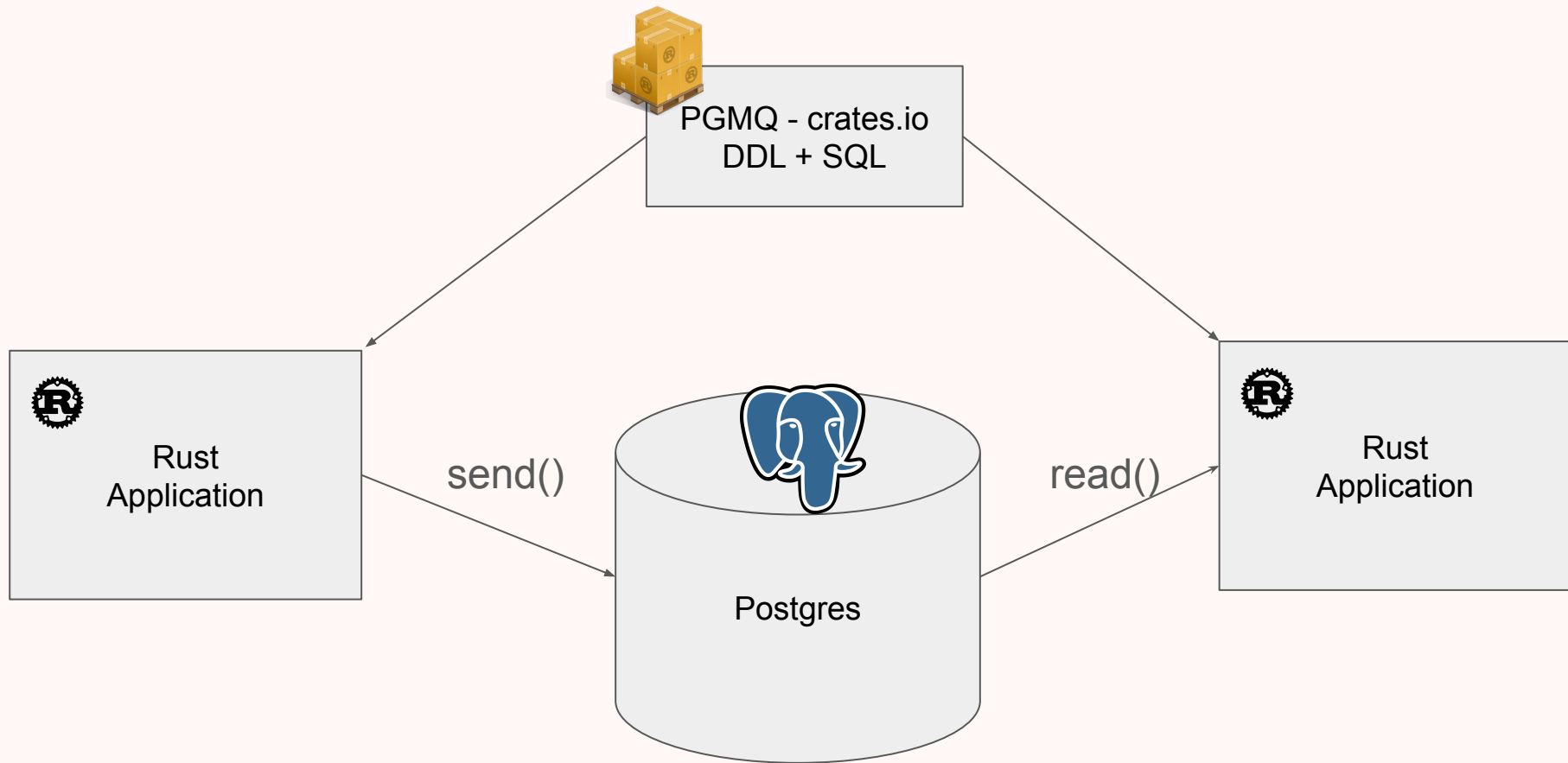


Control-plane not guaranteed to reach data-plane

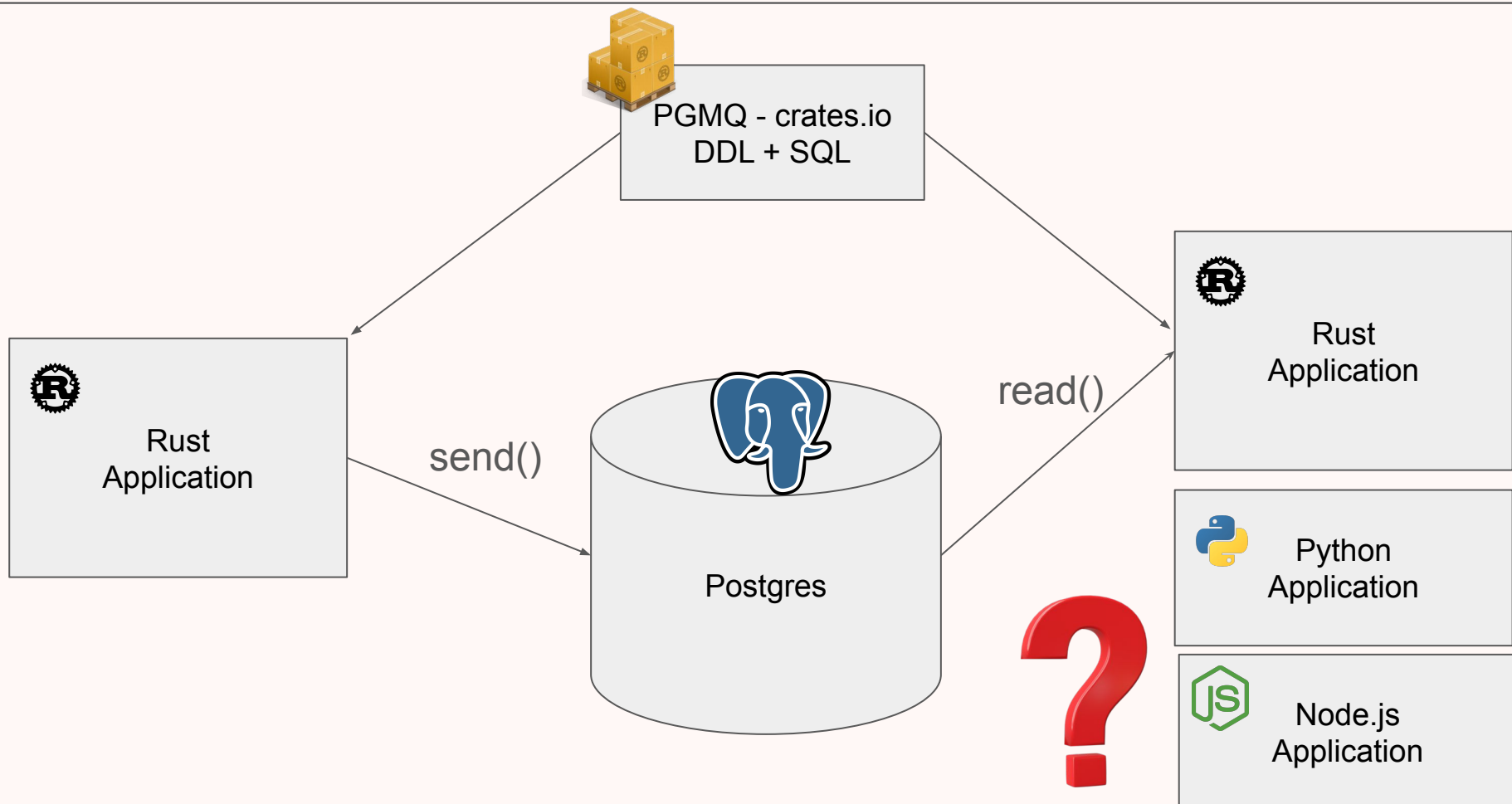




Rust crates work well with...Rust apps



But what if you aren't using Rust?

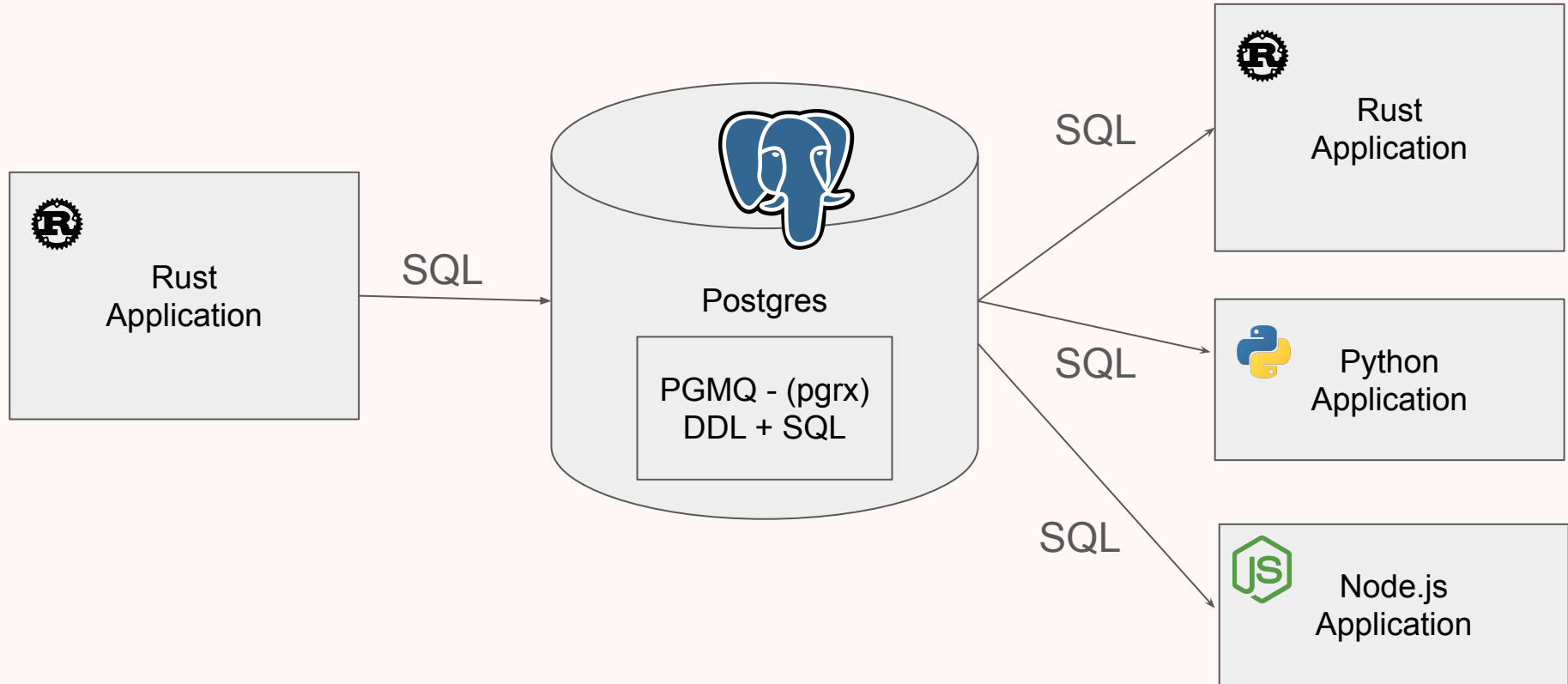


**PGRX was an incredibly convenient way to turn
the Rust crate into a Postgres extension**

<https://github.com/pgcentralfoundation/pgrx>

Give them a ★!

Works with any language when its an extension



Rust was not enough

1. It was difficult to install, compile times were slow.
Rust toolchain is large...
2. Batch read operations were slow due to
unnecessary deserialization in the SPI impl.

So.....

The community rewrote it in PL/pgSQL

Any interest in a pgrx-ectomy? #199



Closed

olirice opened this issue on Mar 5 · 6 comments



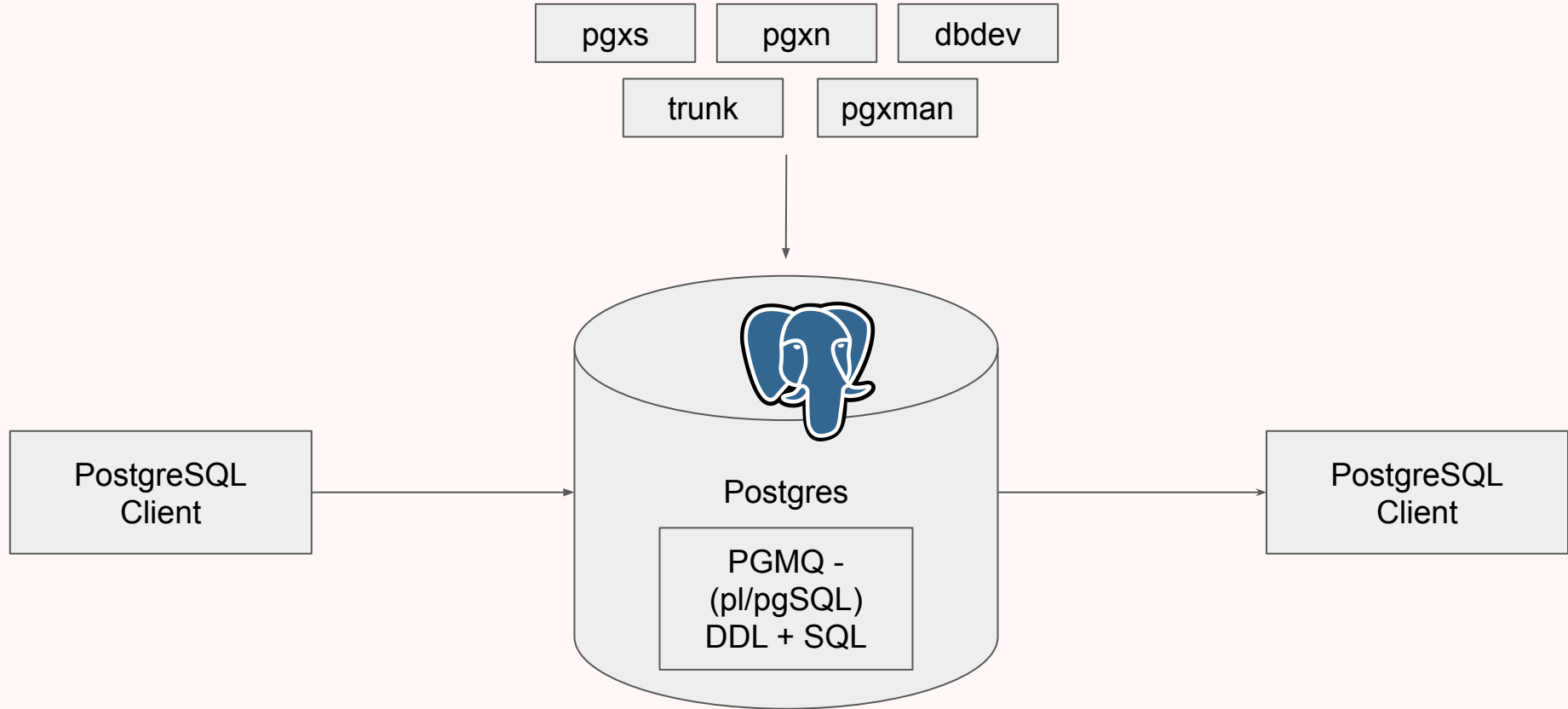
Implement all API functions in PL/pgSQL #212



Merged

v0idpwn merged 28 commits into `main` from `rewrite-sql` on May 17

easier to install



Demo!?



Contributing

1. 📖 Documentation, Benchmarking, Application Examples
2. 🚧 Work on a [Github Issue](#) - some bugs, some need design help
3. 📚 Client libraries
4. 🧪 Test coverage

Questions?

Give it a 

<https://github.com/tembo-io/pgmq>