

pgEdge, Inc.

Use Cases for Multi-Master Distributed Postgres

17 April 2024

Phillip Merrick
Co-founder & CEO, pgEdge
phillip@pgedge.com



Brief Bio

- Software developer by background → 25+ years as a CEO
- Pioneered web services and web APIs at webMethods
- Co-founded EnterpriseDB (EDB) with Denis Lussier & Andy Astor
 - Helped lead company as Chairman for first three years
- Also co-founded and/or led VisualCV, SparkPost and Fugue
- Now co-founder and CEO of pgEdge, with Denis Lussier (CTO)

Agenda

- 1 Demands on Today's Applications
- 2 pgEdge Distributed PostgreSQL
- 3 Use Cases for Multi-master Postgres
- 4 Other considerations
- 5 Q&A



Demands on Today's Applications



Always on



Always available



Serve a global user base



Meeting data residency requirements

*Multi-master distributed
Postgres is an effective way
of meeting these demands*

pgEdge Distributed PostgreSQL

- Asynchronous logical replication
- Configurable conflict resolution and conflict-free delta columns
- Enhanced for fault tolerance and greater performance
- Support for ultra high availability
- Anti Chaos Engine ensures databases remain in sync

pgEdge Customers



Use Cases for Multi-Master Postgres



Low Latency



Ultra High
Availability



Data Residency



Geographic
Scale Out

**And replacing
legacy databases!
E.g. Oracle
Goldengate**

Low latency: placing data closer to global users

- Reduce round trip response times for distant users
- Place presentation, application logic and the data closer to users
- Needed to take advantage of “edge platforms” like Cloudflare Workers or Vercel
- Even just 3 nodes makes a material difference
 - E.g. US East / US West / Europe or US / EU / Asia

Before ...



Low latency: placing data closer to global users

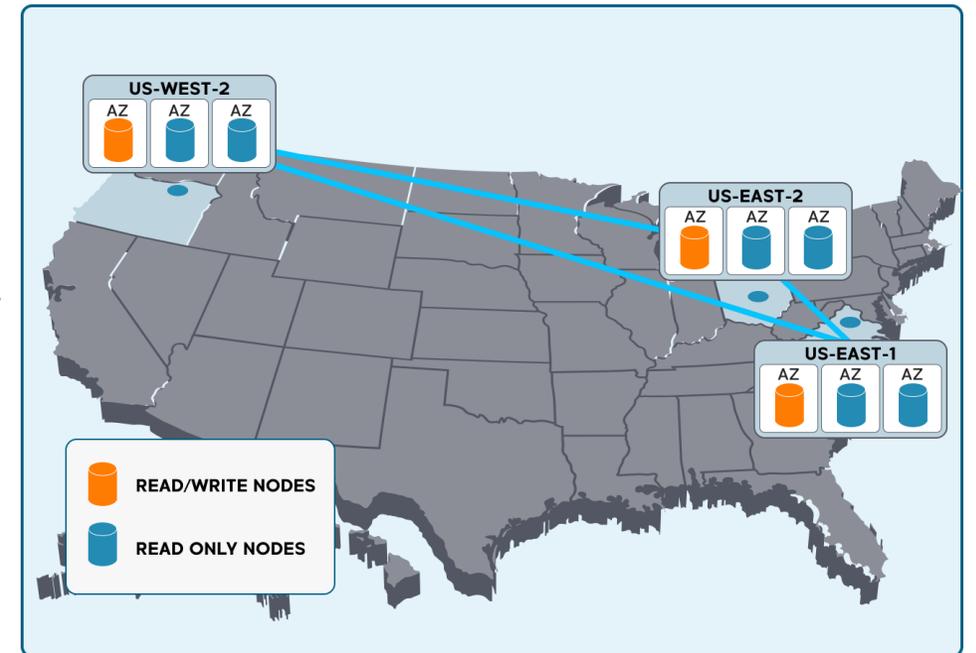
- Reduce round trip response times for distant users
- Place presentation, application logic and the data closer to users
- Needed to take advantage of “edge platforms” like Cloudflare Workers or Vercel
- Even just 3 nodes makes a material difference
 - E.g. US East / US West / Europe or US / EU / Asia

... After



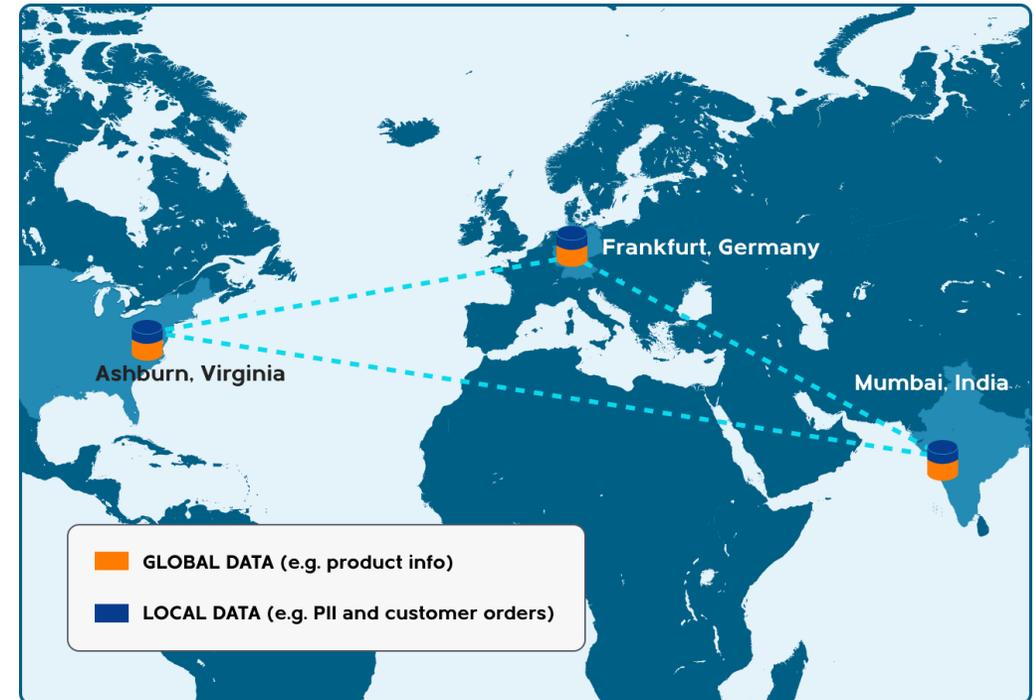
Ultra High Availability: Always on, always available

- Multi-region: Guard against regional cloud provider outages
- Asynchronous logical replication needed
- Support cross region and inter region failover
- Zero downtime maintenance/upgrades
- Can work multi-cloud and hybrid cloud
 - Enables cross cloud applications also
 - Addresses cloud concentration risk



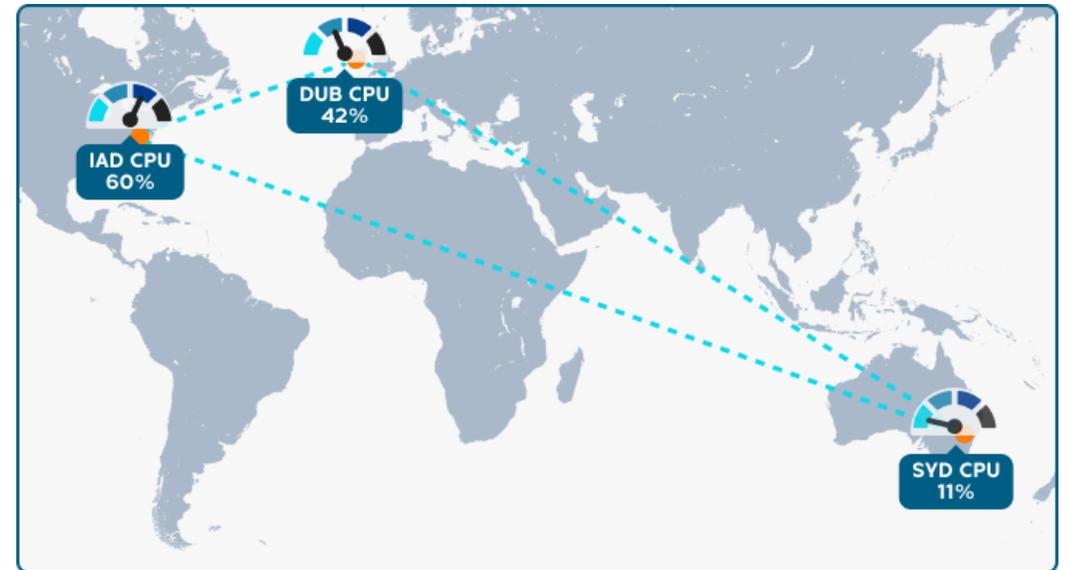
Data Residency: Keeping local data local

- Data generated by citizens of a country/region stays in that country/region
- Without intelligent replication requires awkward solutions
- Selective replication e.g. by table or partition
 - Replicate global tables e.g. product catalog, aggregated reporting tables
 - Don't replicate customer generated data e.g. customer PII, orders, etc.



Geographic Scale Out: Pushing past the limits

- Possible solution when in danger of exceeding available compute
- Use geographic sharding (“geosharding”) to break up workload
- Higher throughput and better performance
- Also get HA and latency benefits



Some other considerations

Application compatible with asynchronous logical replication?

- Ensure all replicated tables have primary keys
- Is workload relatively conflict-free?
- Do some transactions need to be synchronous?

Other components of application stack when going multi-region

Open versus proprietary?

Is the distributed database solution Postgres based, or "compatible"?

In Summary ...

- Multi Master Distributed Postgres is a strong potential solution for:
 - Low latency for more distant users
 - Ultra high availability inc. zero downtime maintenance
 - Data residency
 - Geographic scaleout
- Check us out at our table outside, or at www.pgedge.com
 - **We have some cool swag!**

Q&A

Thank You!