

# TWWADI

*"The Way We've Always Done It"*

**Kubernetes**

**threat?  
opportunity?**

*The* CROSSROADS

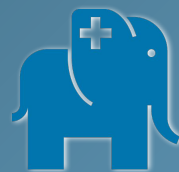


**EDB**

Postgres® for the AI Generation

# Unleashing the Power of PostgreSQL in Kubernetes

Gabriele Bartolini  
VP, Cloud Native at EDB  
28 June 2024



**PG Day**



# Gabriele Bartolini

VP, Cloud Native at EDB

PostgreSQL user since ~2000

Ex 2ndQuadrant (co-founder)

PostgreSQL Contributor

DoK Ambassador

DevOps evangelist

Open source contributor

- Barman (2011)
- CloudNativePG (2022)

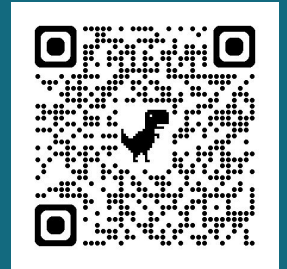


Blog: [gabrielebartolini.it](https://gabrielebartolini.it) @\_GBartolini\_

“Kubernetes has come a long way from  
'no way I would run a database on Kubernetes' to  
'I'm running databases at petabyte scale with  
automated rolling upgrades.’”

Michelle Au - Google Engineer, Kubernetes Storage

**June 25, 2024**



# Agenda

- Kubernetes and T-shaped skills
- Introduction to CloudNativePG
- What changes for HA
- Quick Demo
- What changes for DR
- Conclusions



# Kubernetes and T-shaped skills



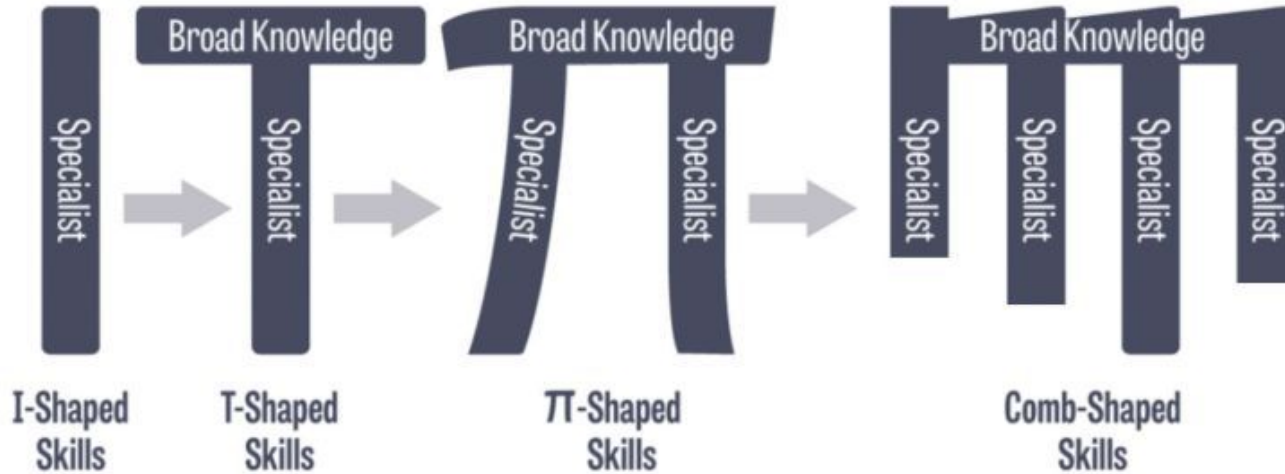
# The move to Kubernetes

- A decision made by CIOs and/or infrastructure teams
- Often assuming databases are left out of Kubernetes:
  - DBaaS (especially when there's no DBA in the organisation)
  - TWWADI (VMs and/or bare metal)
- DBAs have little influence in decision-making regarding Kubernetes adoption
  - Critical career crossroads: adopt it, avoid it (or deny it)
- **People** react differently to change, depending on the way it is introduced
  - **Enthusiasm -> Opportunity**
  - Fear (of losing your job) -> Threat and rejection (TWWADI)
- **The best time for a DBA to approach Kubernetes is now**



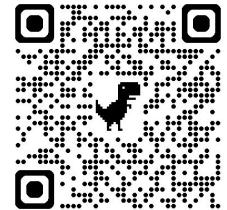
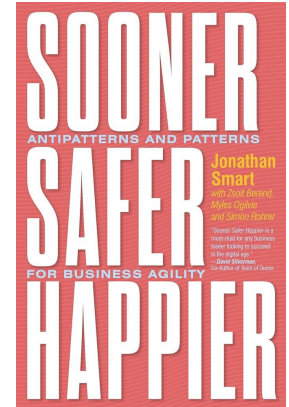


# Breaking the silos



**Figure 8.3: Skills Have Shapes**

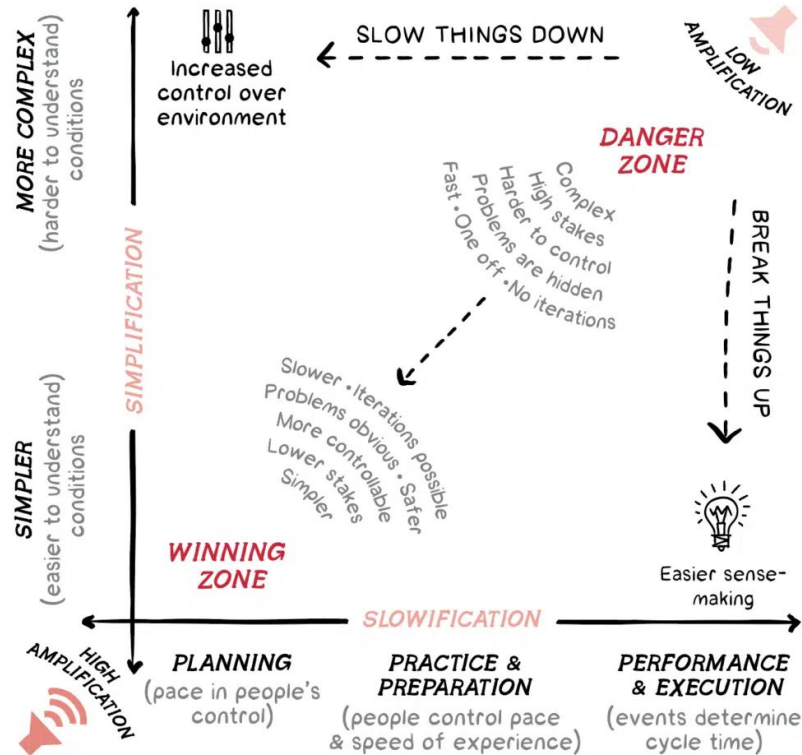
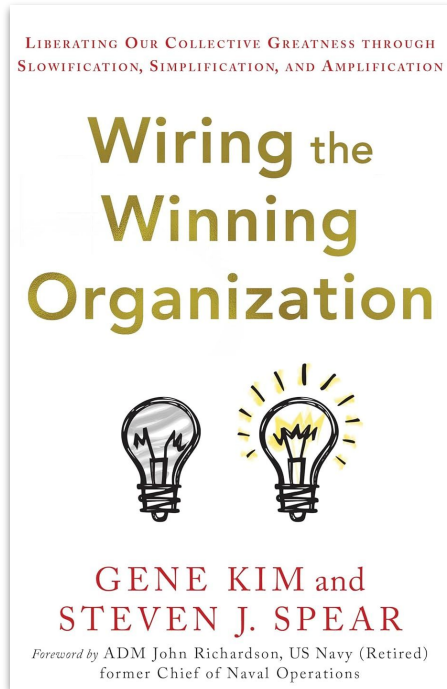
[soonersaferhappier.com](http://soonersaferhappier.com)





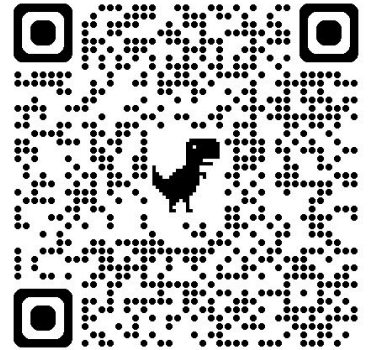
# “Slowification”, Simplification, Amplification

Kubernetes is a way to simplify complex IT problems through modularisation



# Kubernetes Just Turned Ten: Where Does PostgreSQL Stand?

Recommended reading from [gabrielebartolini.it](https://gabrielebartolini.it)



# Kubernetes vs Virtual Machines

## Kubernetes

Infrastructure and containerised apps

Containers share the same OS

Immutable Application Containers

Portable on any cloud, standard

Day 2 operations on applications

## VMs

Operating systems

VMs have their own OS

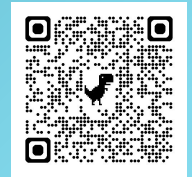
Mutable (“dnf update”)

Indirectly portable (Day 1, Terraform)

No Day 2 operations on applications



“The **status** of the  
**entire Kubernetes cluster**  
intrinsically **defines**  
an **organisation’s infrastructure**  
at any given time,  
**including applications**”



# From infrastructure to applications

Fundamental concepts for DBAs to better understand Kubernetes

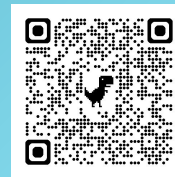
- **Region (Kubernetes Cluster)**
- **Data Centre (Availability Zone)**
- **Nodes:**
  - Lin
  - VM
- **Network**
  - The `Service` resource
- **Storage:**
  - Network vs Local
  - Container Storage Interface (CSI)
- **Resources:**
  - The `Deployment` resource
  - Databases are applications
    - The Operator pattern
      - Custom Resources

**Single authority for all applications  
(Postgres included)**

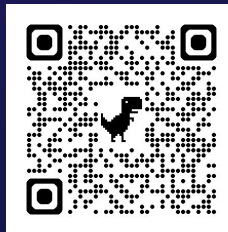


Regarding databases,  
**Kubernetes' standard resources are not sufficient.**

However, one of the strengths of Kubernetes is the **operator pattern**, a development pattern designed to manage **complex applications like a PostgreSQL database.**



# Run PostgreSQL. The Kubernetes way.





# Introduction to CloudNativePG



# The history of CloudNativePG

**Bringing PostgreSQL to Kubernetes** (not Kubernetes to PostgreSQL)

## 2019-2020: The 2ndQuadrant era

- MVP (fail-fast)
- Cloud Native BDR
- Cloud Native PostgreSQL

## 2021-2022: EDB Cloud Native PostgreSQL

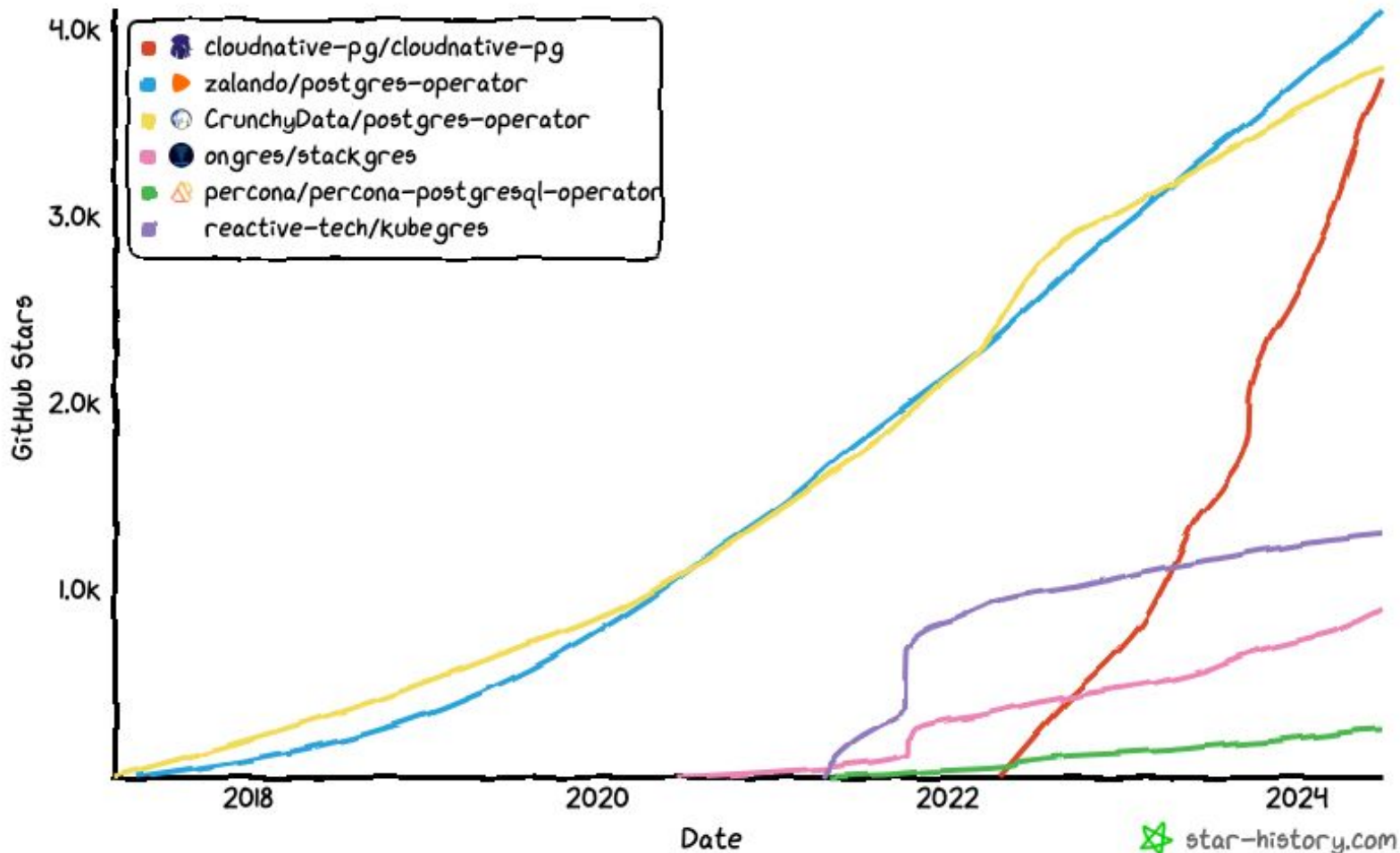
- Production ready (IBM Cloud Pak)
- Several features gradually introduced
  - HA + DR
  - Observability
  - Security
  - Pooling
  - ...

## 2022-: CloudNativePG Revolution

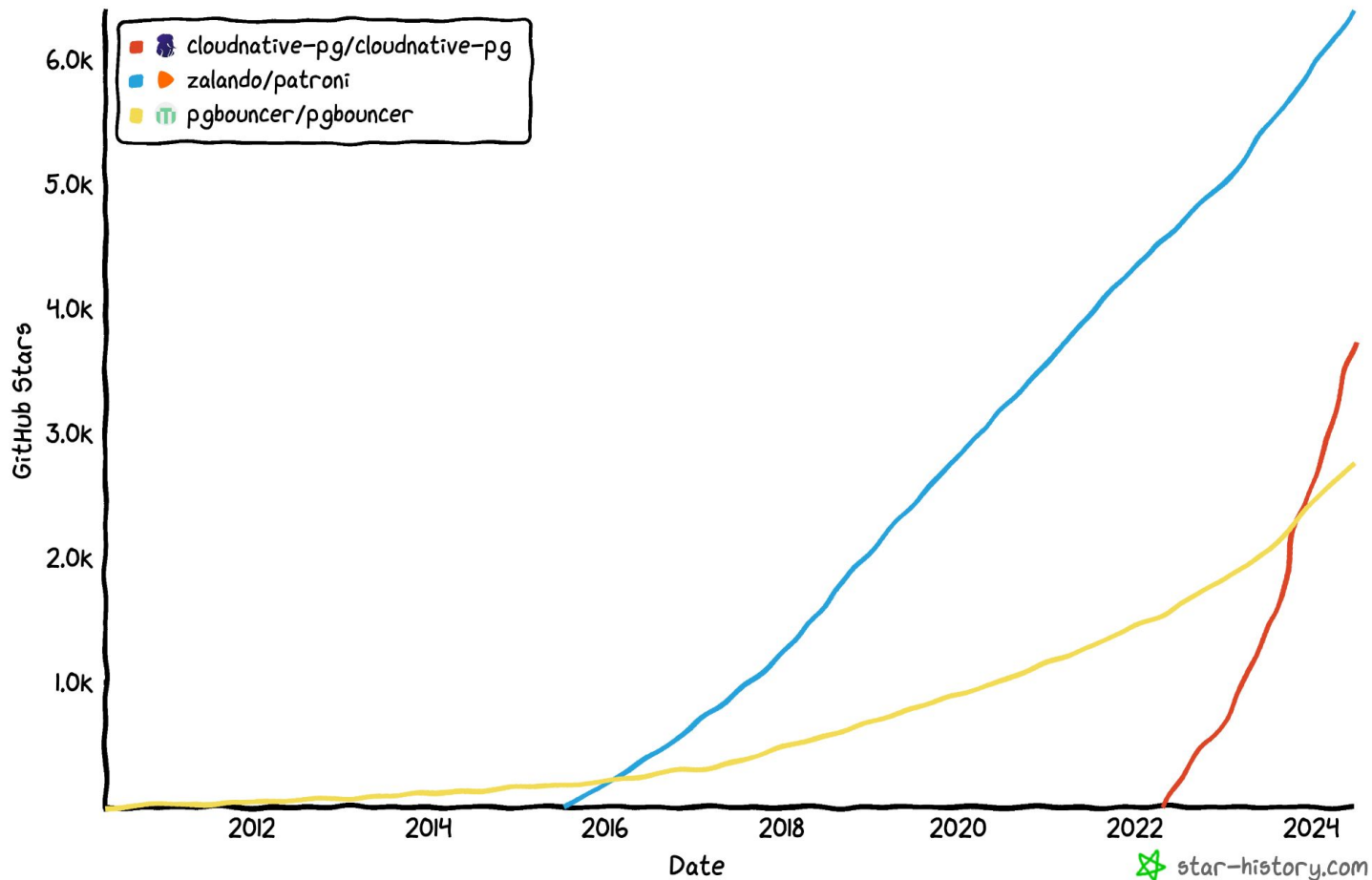
- Launched in May 2022
- Consistent commitment from EDB
  - 9 releases in 2 years
  - 3.2k commits
  - 100+ contributors, 260 forks
- Major features:
  - Declarative Fencing (2022)
  - Volume for WALs
  - Failover of physical Replication Slots
  - Backup from a Standby (2023)
  - Declarative Hibernation
  - Declarative Role Management
  - Volumes for Tablespaces
  - Volume Snapshot backup and recovery



# Star History

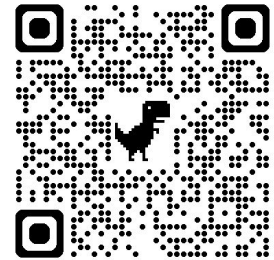


# Star History



# About CloudNativePG

- “Level 5”, Production ready
  - EDB Big Animal, IBM Cloud Pak, Google Cloud, Tembo, ...
- Open source (May 2022)
  - Apache License 2.0
  - Vendor neutral, openly governed, always free
  - Originally created by EDB (2019 by 2ndQuadrant)
- Multiple installation methods:
  - K8s manifests
  - Helm chart
  - OperatorHub.io (OLM)
- Most popular Postgres operator in 2023 (TimeScale survey)
- 3.7k stars and 51M downloads on Github (2 years)



[github.com/cloudnative-pg](https://github.com/cloudnative-pg)



# CloudNativePG's 4 Pillars

More than an operator, it's a "Kubernetes native database"

## Seamless Kubernetes API Integration

Unlock the power of the Kubernetes API to effortlessly manage PostgreSQL databases. Utilize the operator pattern and leverage standard resources for streamlined operations.

## Advanced Observability

Gain crucial insights through integrated monitoring and logging, enhancing diagnostics for operations, supporting strategic decision-making, and empowering autopilot control.

## Declarative Configuration

Deploy, scale, and maintain databases using automation that optimizes operational workflows and facilitates Infrastructure as Code.

## Secure by Default

Safeguard your sensitive data with robust security protocols and industry-leading features across all layers, including source code, containers, Kubernetes clusters, and the cloud.

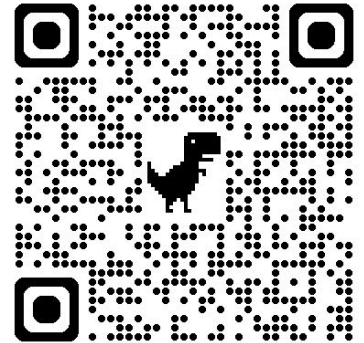


# The PostgreSQL community debates ALTER SYSTEM

April 2024: "Sometimes the smallest patches create the biggest discussions"



**LWN.net**  
News from the source



My contribution to PostgreSQL 17 thanks to Jelte Fennema-Nio and Robert Haas





What changes for HA



# The PostgreSQL `Cluster` resource

```
apiVersion: postgresql.cnpg.io/v1
```

```
kind: Cluster
```

```
metadata:
```

```
  name: swiss-pgday-24
```

```
spec:
```

```
  instances: 3
```

```
  minSyncReplicas: 1
```

```
  maxSyncReplicas: 1
```

```
  affinity:
```

```
    nodeSelector:
```

```
      workload: postgres
```

```
  storage:
```

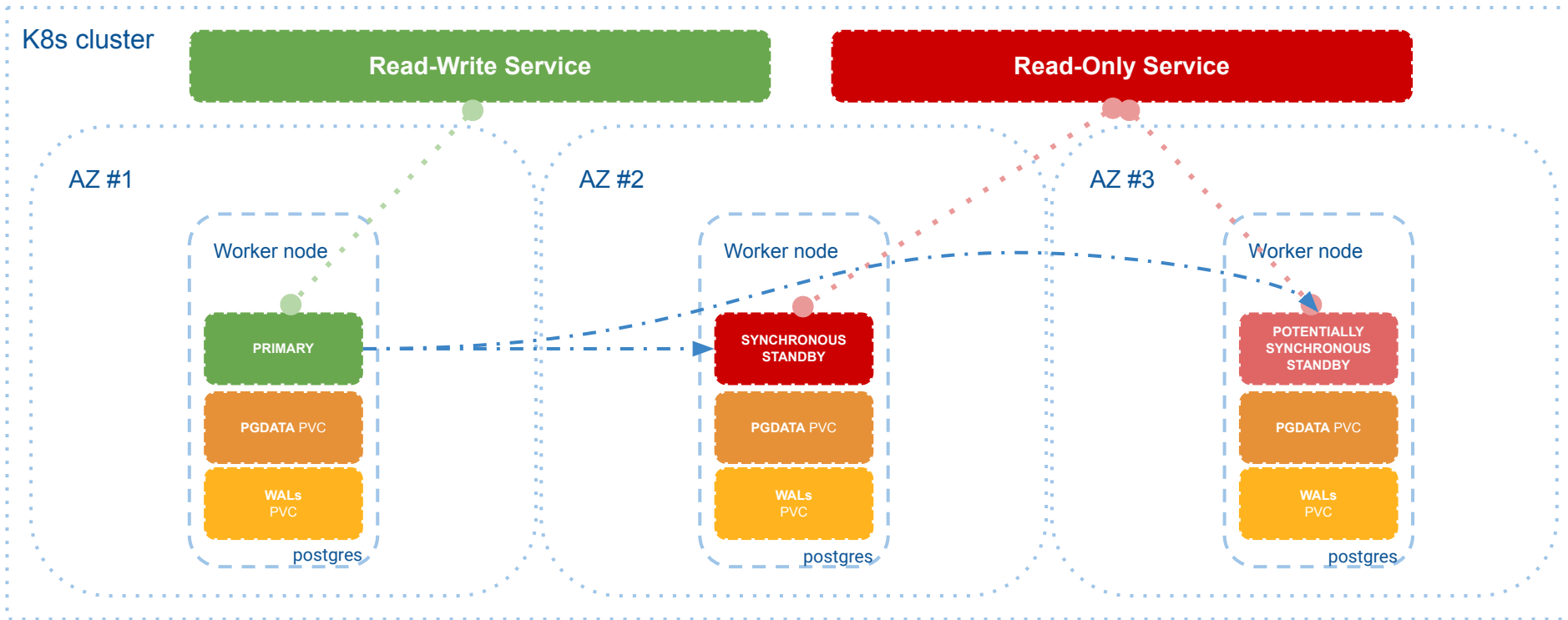
```
    size: 40Gi
```

```
  walStorage:
```

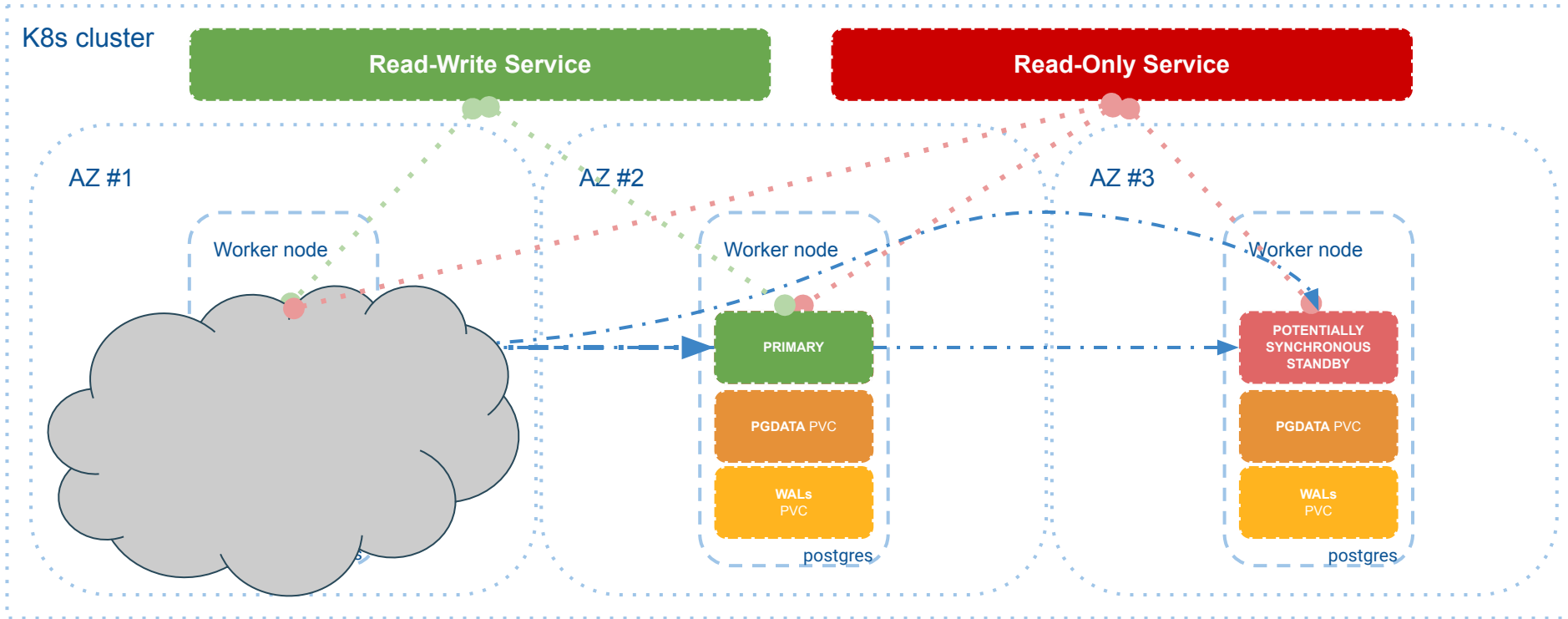
```
    size: 10Gi
```



# Highly Available PostgreSQL Cluster



# Automated failover (HA with very low RTO)



Quick Demo



Questions on the demo



What changes for DR





# Continuous backup

Guaranteed RPO from the first available backup to the latest archived WAL

## WAL archive

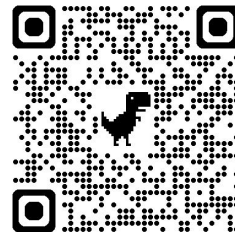
- Currently only via object store (Barman Cloud)
- WAL files are archived every 5 minutes maximum (RPO  $\leq$  5 min)

## Physical base backups

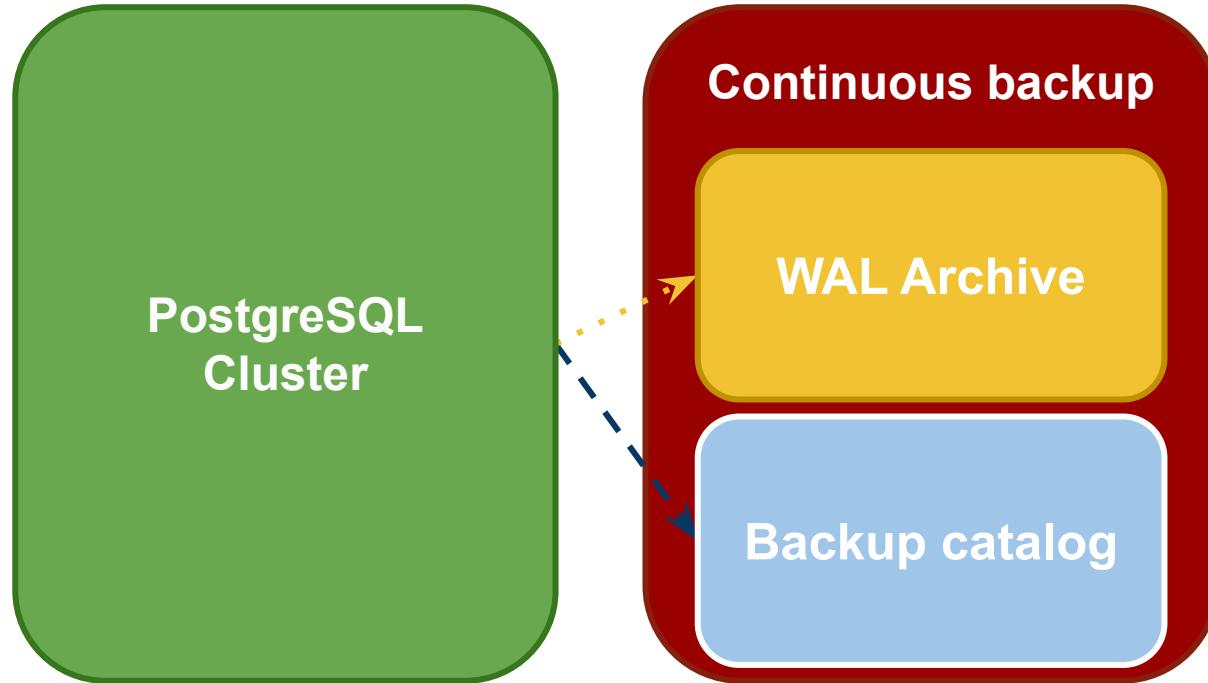
- From the primary or a standby
- Scheduled or on demand
- On object stores (Barman Cloud)
  - Hot (online)
- As Kubernetes Volume Snapshots
  - Hot (online) and cold (offline)
  - Transparent incremental/differential backup

**Future: generic plug-in interface (CNPG-I)**

Scan here to watch  
"Disaster Recovery with  
Very Large Postgres Databases  
by me and Michelle Au  
**KubeCon NA 2023 - Chicago**



# Continuous backup



Scan here to watch  
"Disaster Recovery with  
Very Large Postgres Databases  
by me and Michelle Au  
**KubeCon NA 2023 - Chicago**



# Recovery

Full or to a specific point

## Bootstrap method

- Copy from an available physical base backup
- Apply WAL files (REDO logs) until you reach the target

## Full recovery

- Without a target (until the end of the WAL)
- Can be used to setup a physical replica
  - Replica cluster, even delayed
  - “Continuous recovery”

## Point-In-Time Recovery

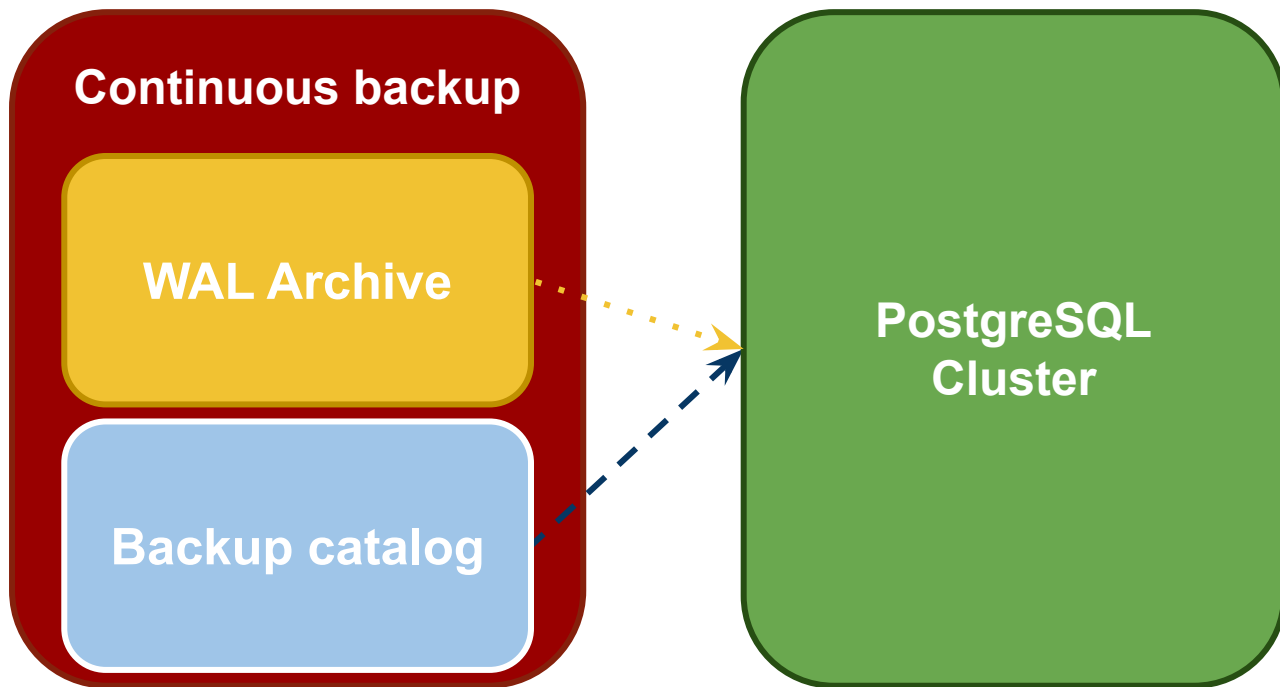
- Up to a given time or transaction
- When reaching the target, the cluster promotes itself

Scan here to watch  
“Disaster Recovery with  
Very Large Postgres Databases  
by me and Michelle Au  
**KubeCon NA 2023 - Chicago**



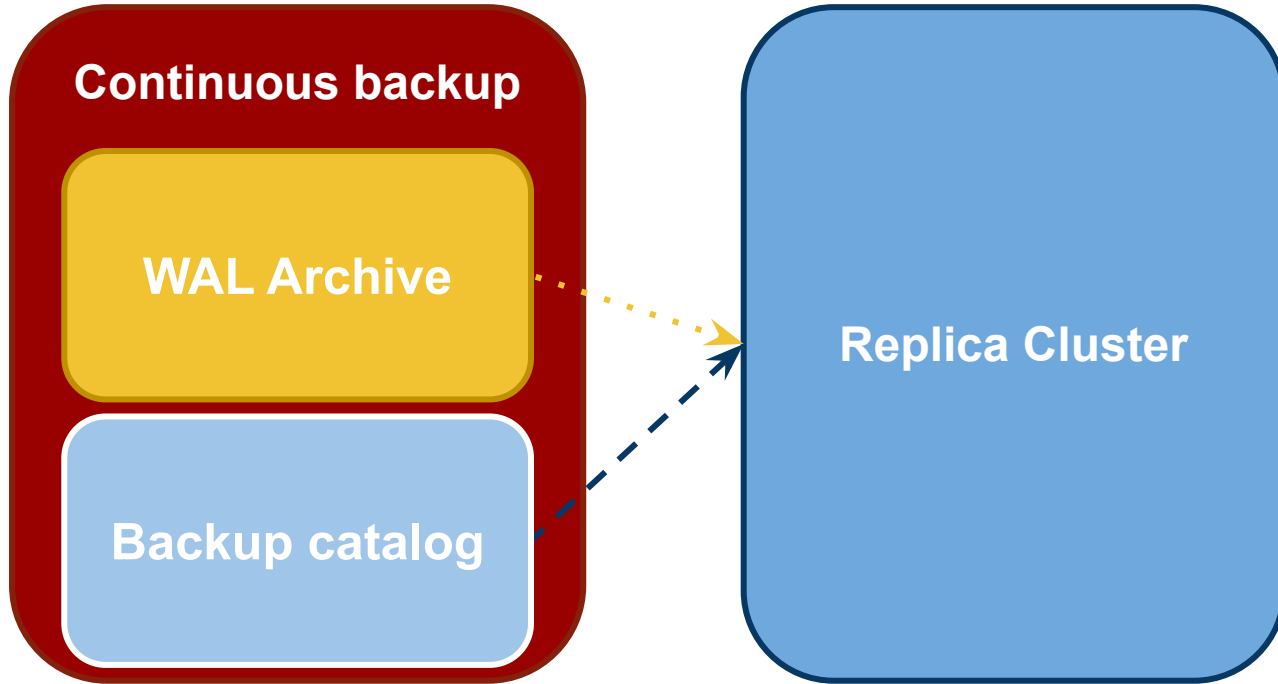
# Full recovery or PITR

PostgreSQL is promoted when the recovery target is reached

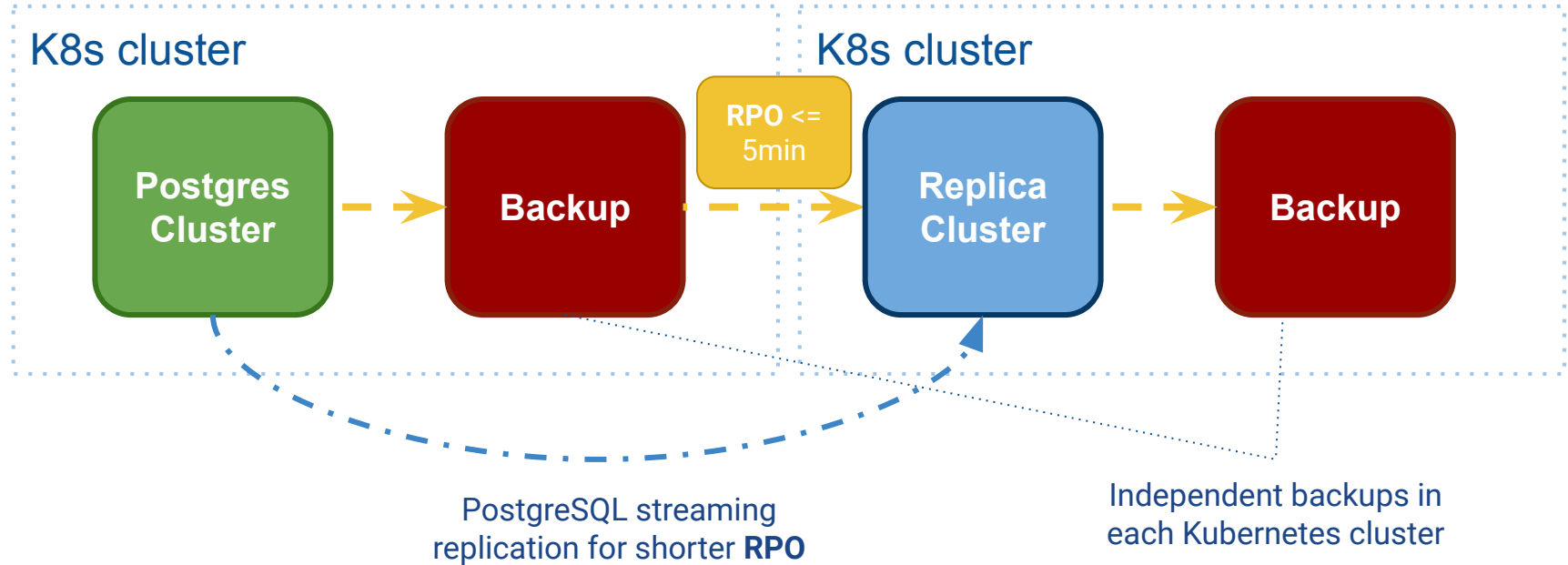


# Replica Cluster (for DR or Reporting)

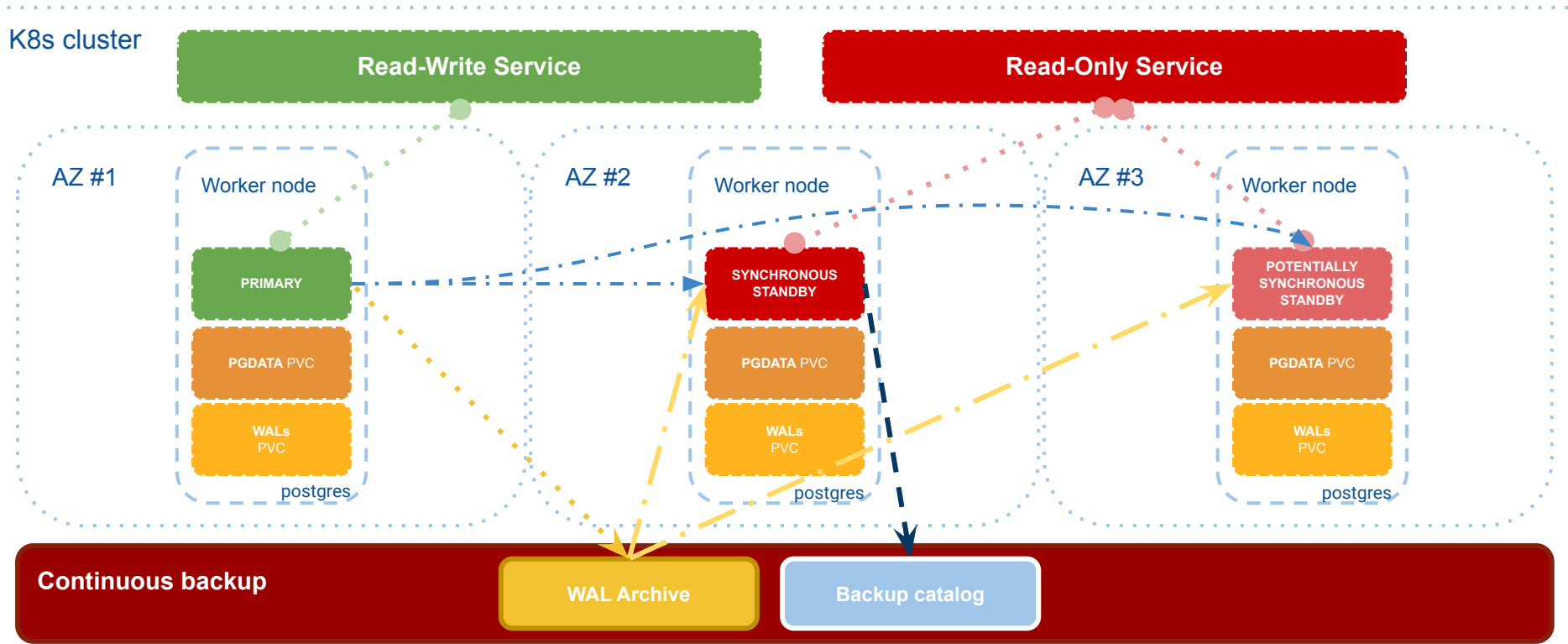
PostgreSQL remains in continuous recovery (read-only replica)



# Replica clusters: beyond the K8s cluster SPoF

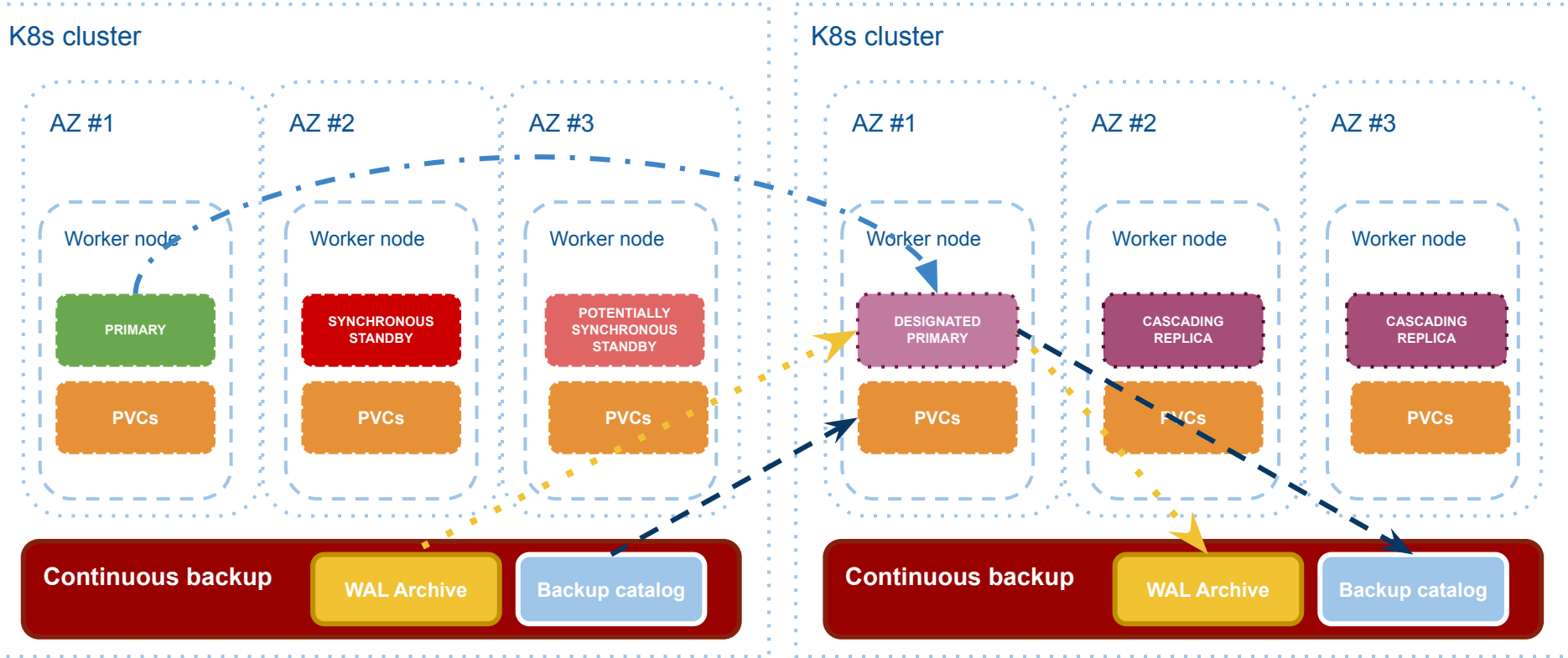


# Production cluster in a 3+ AZ Kubernetes cluster

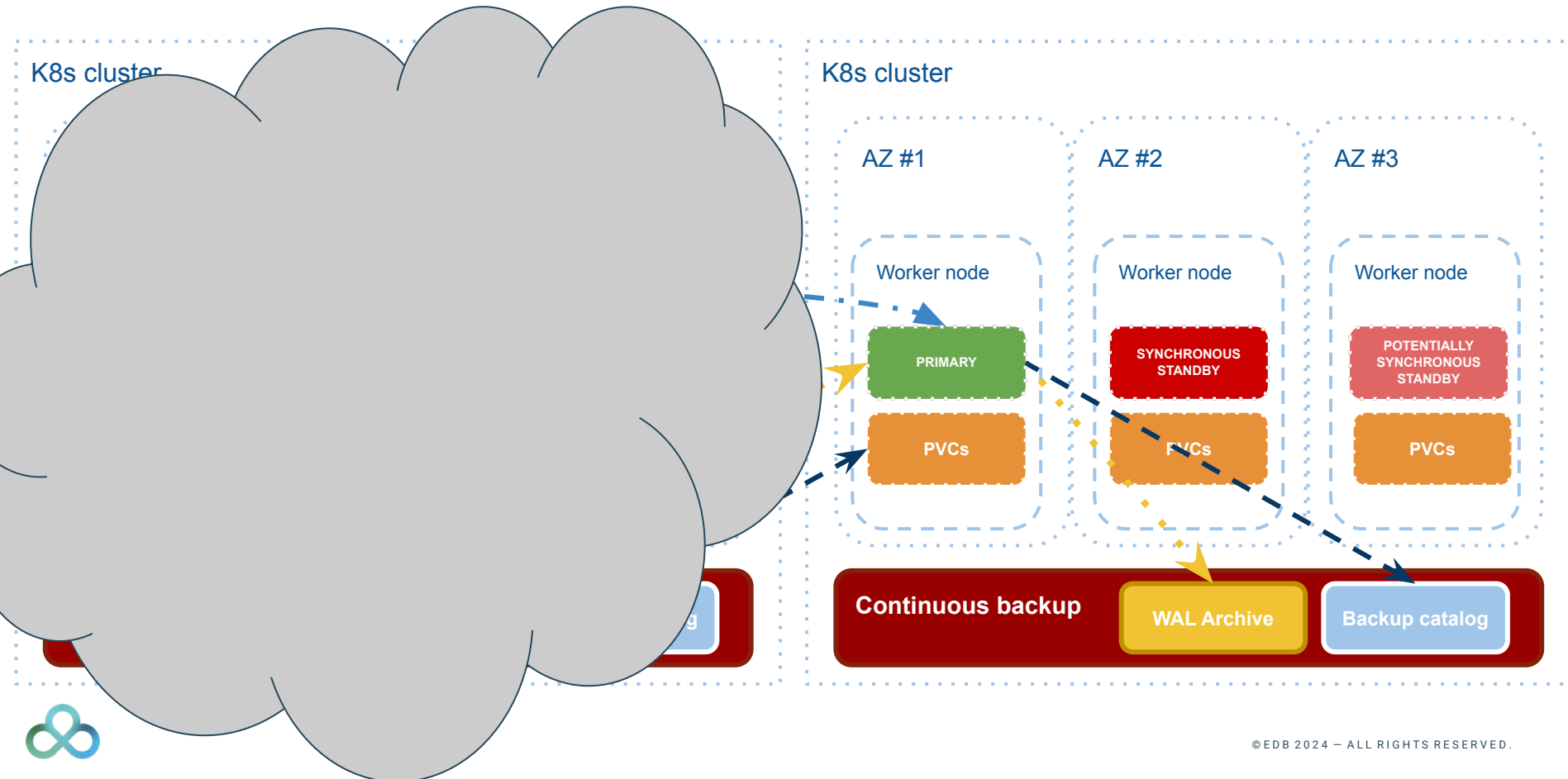




# Symmetric architecture on 2 Kubernetes clusters



# Controlled failover across Kubernetes clusters



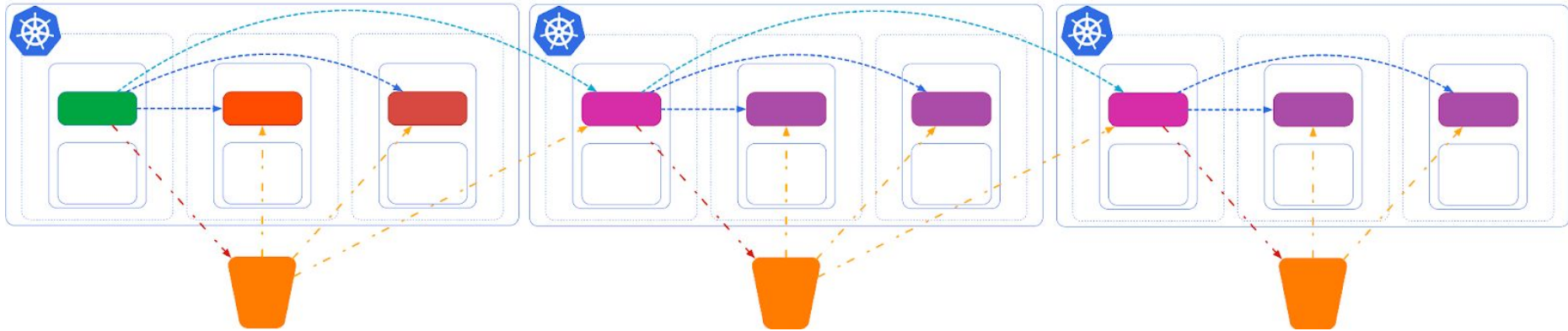
# Beyond the two Kubernetes clusters / regions

Cascading deployments over multiple regions for world class resilience

Region 1

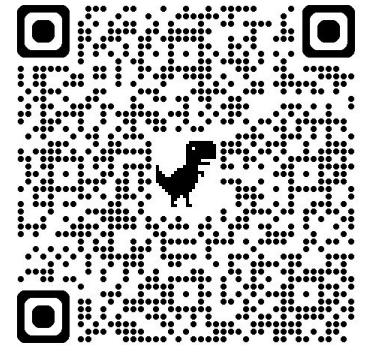
Region 2

Region 3



# Making the world my single point of failure with PostgreSQL

Recommended reading from [gabrielebartolini.it](https://gabrielebartolini.it)



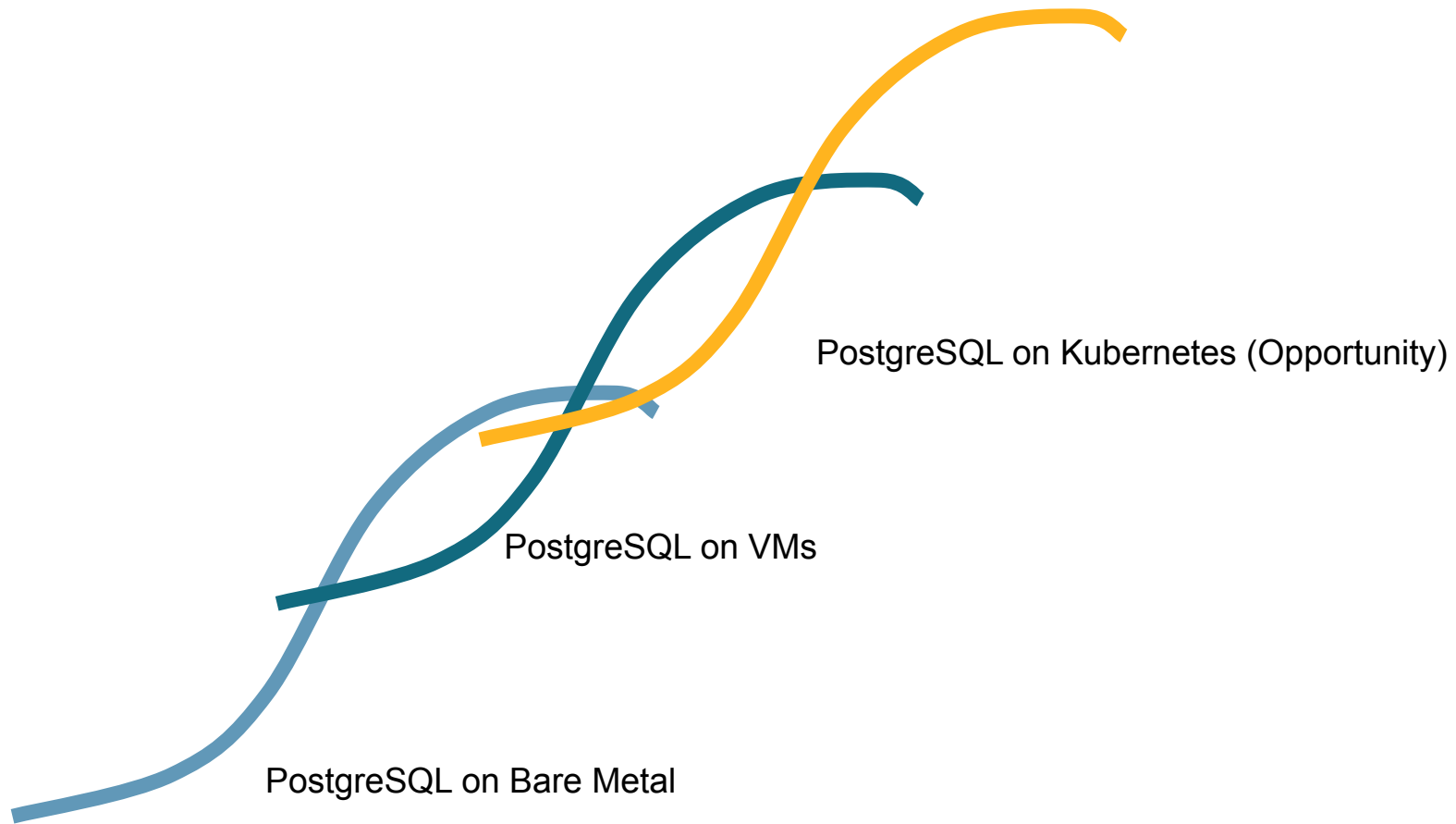
# Distributed topology configuration (upcoming 1.24)

```
apiVersion: postgresql.cnpg.io/v1
kind: Cluster
metadata:
  name: pg16-eu-central
spec:
  # <snip>
  externalClusters:
    - name: pg16-eu-central
      barmanObjectStore:
        destinationPath: s3://pg16-eu-central/
        # ...
    - name: pg16-eu-western
      barmanObjectStore:
        destinationPath: s3://pg16-eu-western/
        # ...
  replica:
    primary: pg16-eu-central
    source: pg16-eu-western
```



Conclusions





# Change is happening now

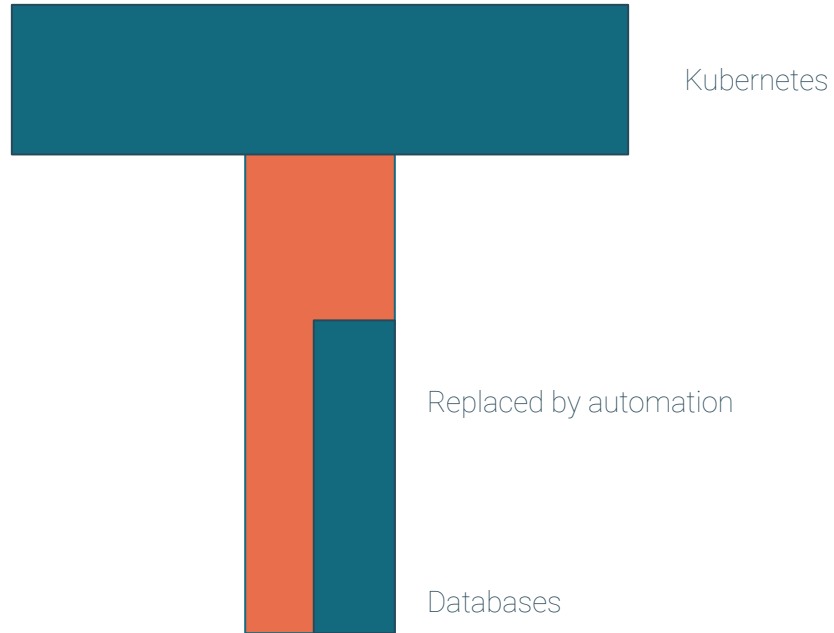
## Time to catch the wave and build T-skills on Kubernetes

- Running Postgres in Kubernetes is different from running it outside
- CloudNativePG manages HA clusters, not instances
- You don't need to know all Kubernetes (like with Linux):
  - CloudNativePG, Basic Kubernetes resources, Storage, Network basics
- Differences are primarily in the way we interact with a Cluster
  - Deployment, Configuration, Monitoring, Alerting, Logging, Access, ...
- It is still the same PostgreSQL you know inside
- Your skills (and yours only) are needed to improve running Postgres in Kubernetes
- At the crossroads, choose the Postgres in Kubernetes way
- **Join CloudNativePG NOW and meet other DBAs on the same journey!**





# Long live the Postgres DBA





Thank you!

