Running PostgreSQL in a Kubernetes Cluster: CloudNativePG

Nick Ivanov

Solutions Architect

EnterpriseDB

Nick Ivanov

Solutions Architect

EnterpriseDB

nick.ivanov@enterprisedb.com

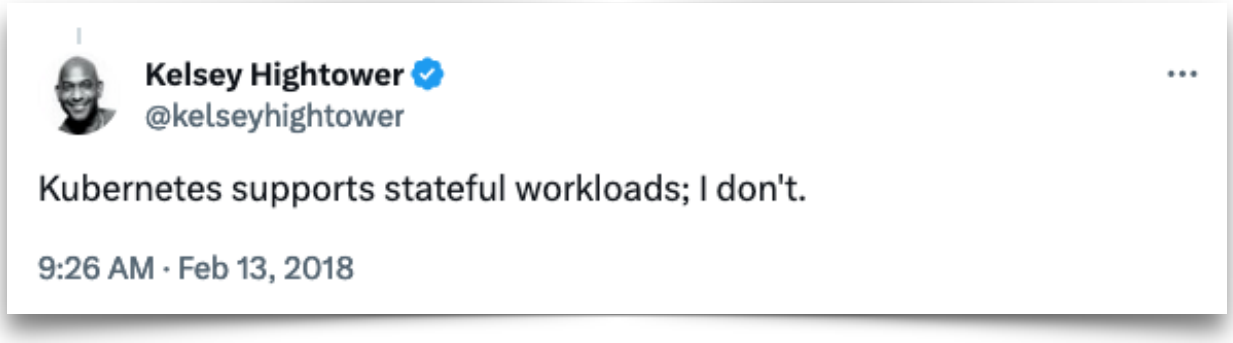
https://www.linkedin.com/in/nick-ivanov-toronto/

Before joining EnterpriseDB in 2022, Nick had been working at IBM Canada for more than 10 years as a database and cloud application architect. He has experience with database design, performance tuning, HA&DR implementation, migration on multiple database platforms, including Postgres, Db2, SQL Server, Oracle, MySQL, and others.

He's based in Toronto, Canada.

PASS Data Community SUMMIT

Can you run databases on K8s?



CloudNativePG

Documentation Blog Support End Users

Star 2,212

Run PostgreSQL.

The Kubernetes way.

CloudNativePG is the Kubernetes operator that covers the full lifecycle of a highly available PostgreSQL database cluster with a primary/standby architecture, using native streaming replication.

View on GitHub

Why operator is needed

- K8s built-in controllers only handle built-in resources
  - ReplicaSet - no PVC templates
  - StatefulSet - doable, but too complex to configure
- CNPG controller incorporates Postgres knowledge
  - Manages custom resources

What you need

- A Kubernetes cluster
  - `kind` works just fine
- `kubectl`
- `cnpg` plugin

Postgres CRD

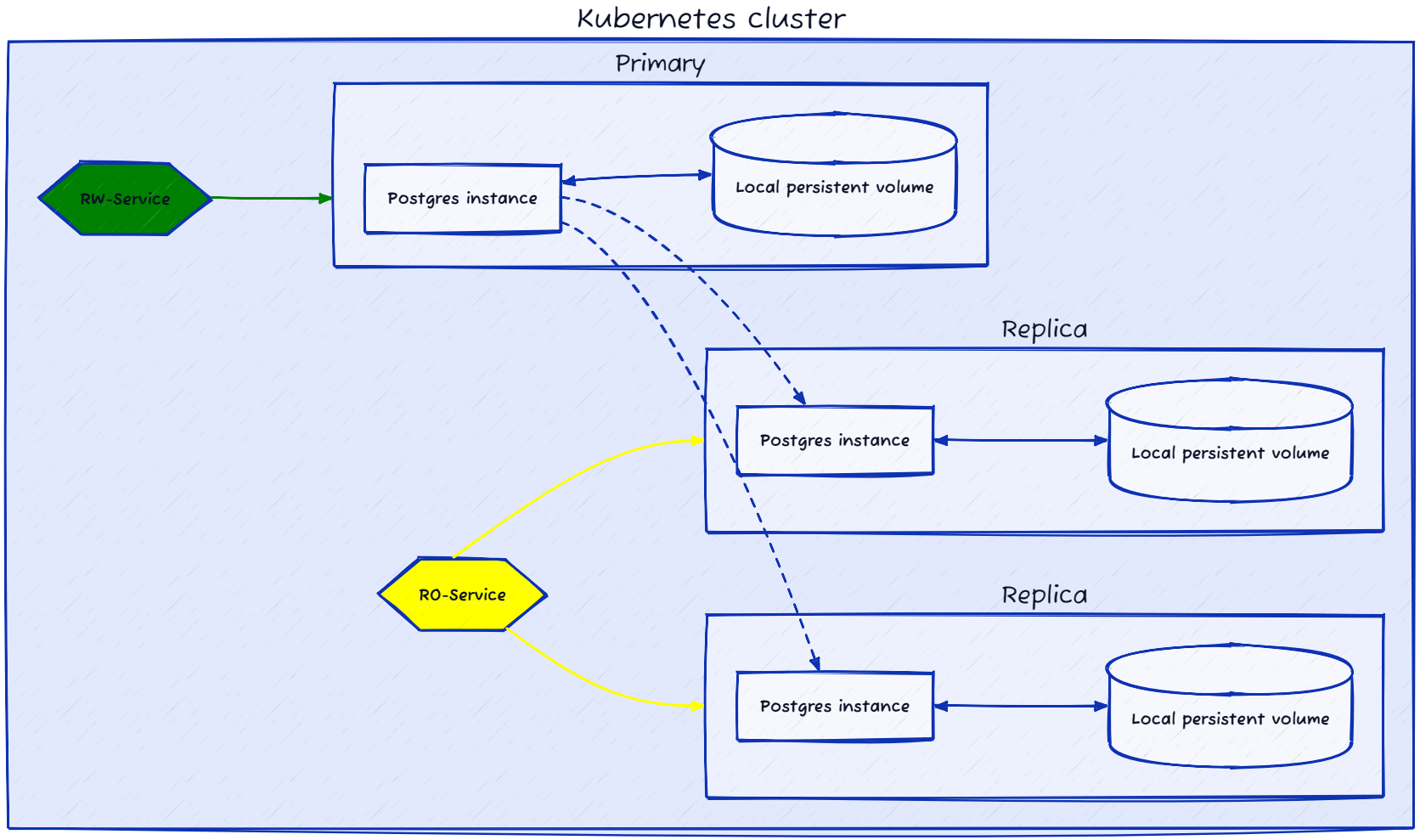
```yaml
apiVersion: postgresql.cnpg.io/v1
kind: Cluster
metadata:
  name: pg15
spec:
  instances: 3
  primaryUpdateStrategy: unsupervised
  storage:
    size: 1Gi
```

- All you need to create a three-node cluster
- Uses all defaults
- Demo

What's created

- Postgres pods as requested by the manifest
- Persistent volumes
- Three services
  - `<cluster name>-rw` — for the primary node
  - `<cluster name>-ro` — for the standby nodes
  - `<cluster name>-r` — for all nodes
- Secrets
  - Authentication credentials for database users
  - TLS certificates

What's created



Replication

- Physical replication streaming WAL records
- All databases in the instance
- Replicas are read-only
- Asynchronous or synchronous
- Automatic management of replication slots

Cluster initialisation methods

- New (`initdb`)
- From a backup (Barman Cloud)
- From another Postgres instance (`pg_basebackup`)
- Using import (`pg_dump` & `pg_restore`)
- Demo

High Availability and Failure Modes

- Highly reliable streaming replication
- No external failover managers
  - Postgres Instance Manager + K8s
- PVs reused if possible to start new pods
  - Backup of a primary otherwise
- Demo

Scheduling and resources

- Use dedicated worker nodes if possible
  - `nodeSelector` and `tolerations`
- Anti-affinity by default
- Scheduling is based on the resource `requests`
  - It is counterproductive to set `limits` much higher
- Use `Guaranteed` QOS

Maintenance: Rolling Updates

- Triggered automatically
- Unsupervised update is fully automatic
- Supervised update requires intervention prior to the final switchover
- Only minor version upgrade possible (currently)

Maintenance: Backup & Recovery

- Hot only, on-demand or scheduled, plus WAL archiving
- Optional compression & encryption
- Uses Barman Cloud
  - Any S3-compatible service
  - MinIO Gateway option offers many alternatives
- Recovery — instantiate a new cluster from backup

Maintenance: Volume Snapshots

- Alternative backup & recovery method
- Hot or cold
- Allows incremental and delta backup
  - If supported by the storage class
- Better choice for large databases

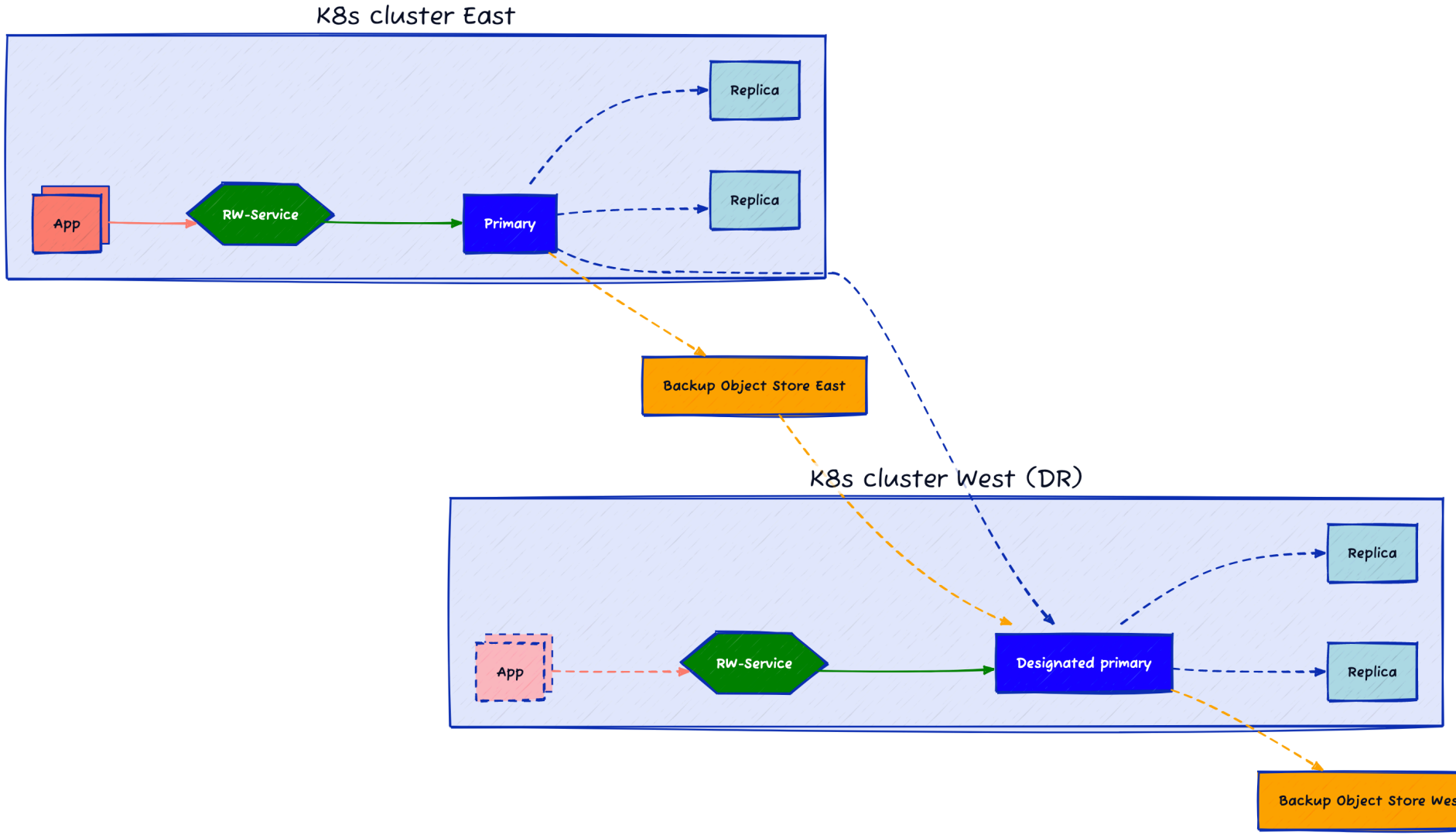
Maintenance: More

- Fencing nodes
- Hibernation
- TLS certificate management

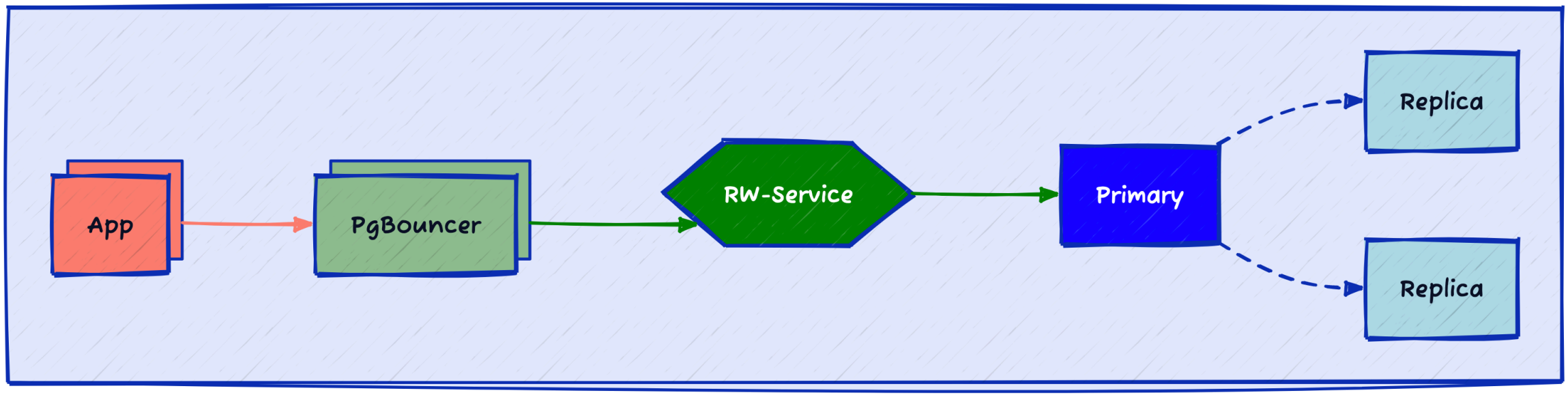
Cluster Monitoring

- Prometheus — “default” on K8s
- Exporters are set up upon deployment
- Create PodMonitor resources using the cluster spec

DR with Replica Clusters



Application Connection Pooling

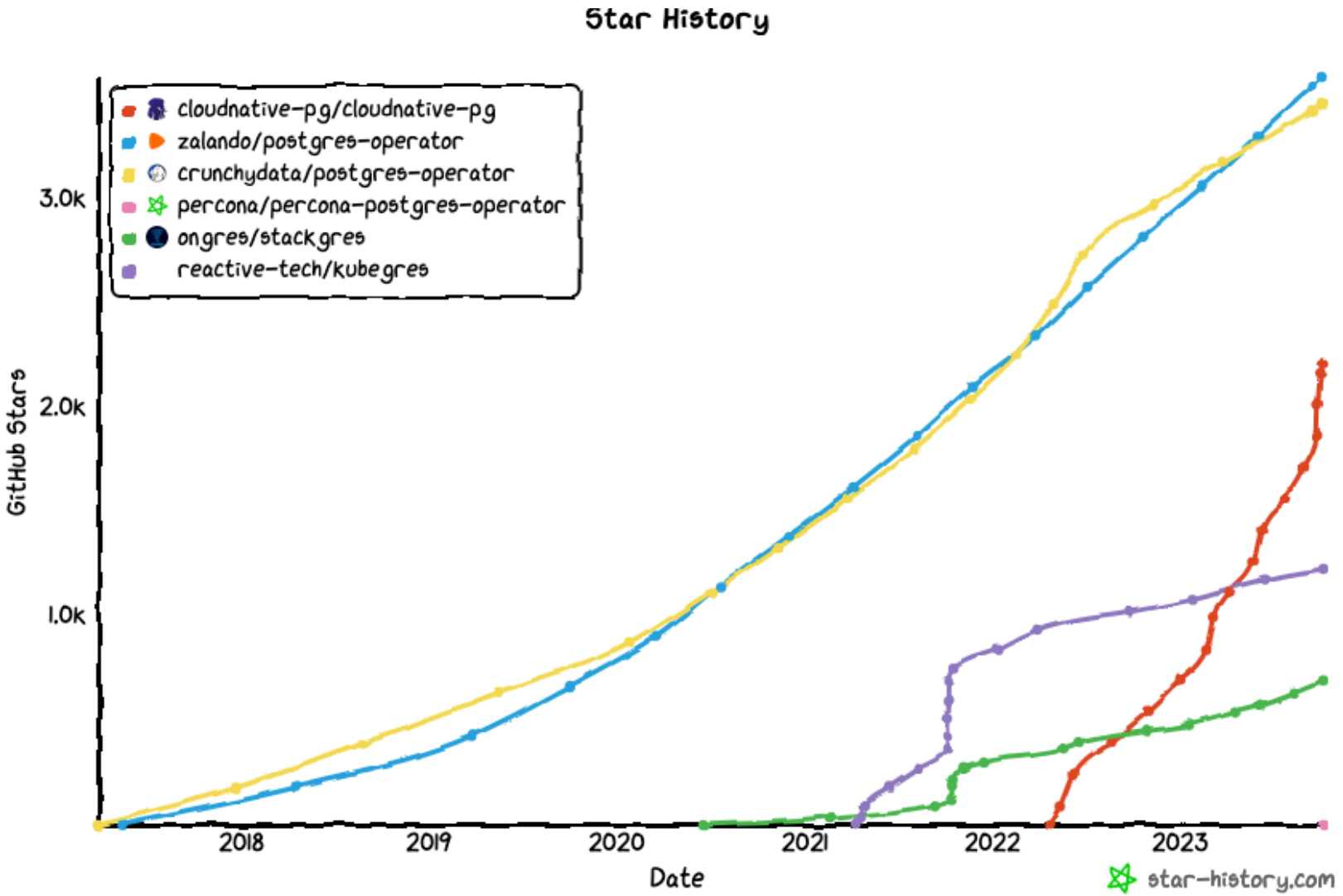


Final words

Operator capability levels

Source: https://sdk.operatorframework.io/docs/overview/operator-capabilities/

K8s Operators for Postgres



Links

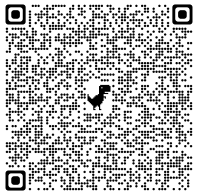
CloudNative Pg project home

Recommended architectures for PostgreSQL in Kubernetes

Cloud Native Disaster Recovery Whitepaper

More links

Github star history



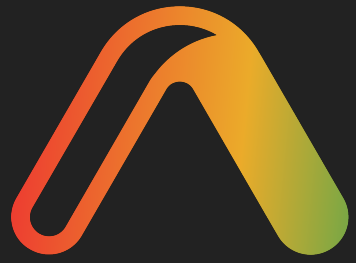
`kind` - tool for running K8s clusters using Docker

Session evaluation

Your feedback is important to us

Evaluate this session at:

www.PASSDataCommunitySummit.com/evaluation



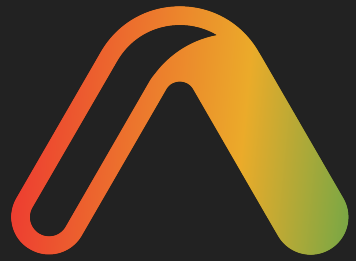
PASS
Data Community
SUMMIT
2023

Thank you

Nick Ivanov


nick.ivanov@enterprisedb.com

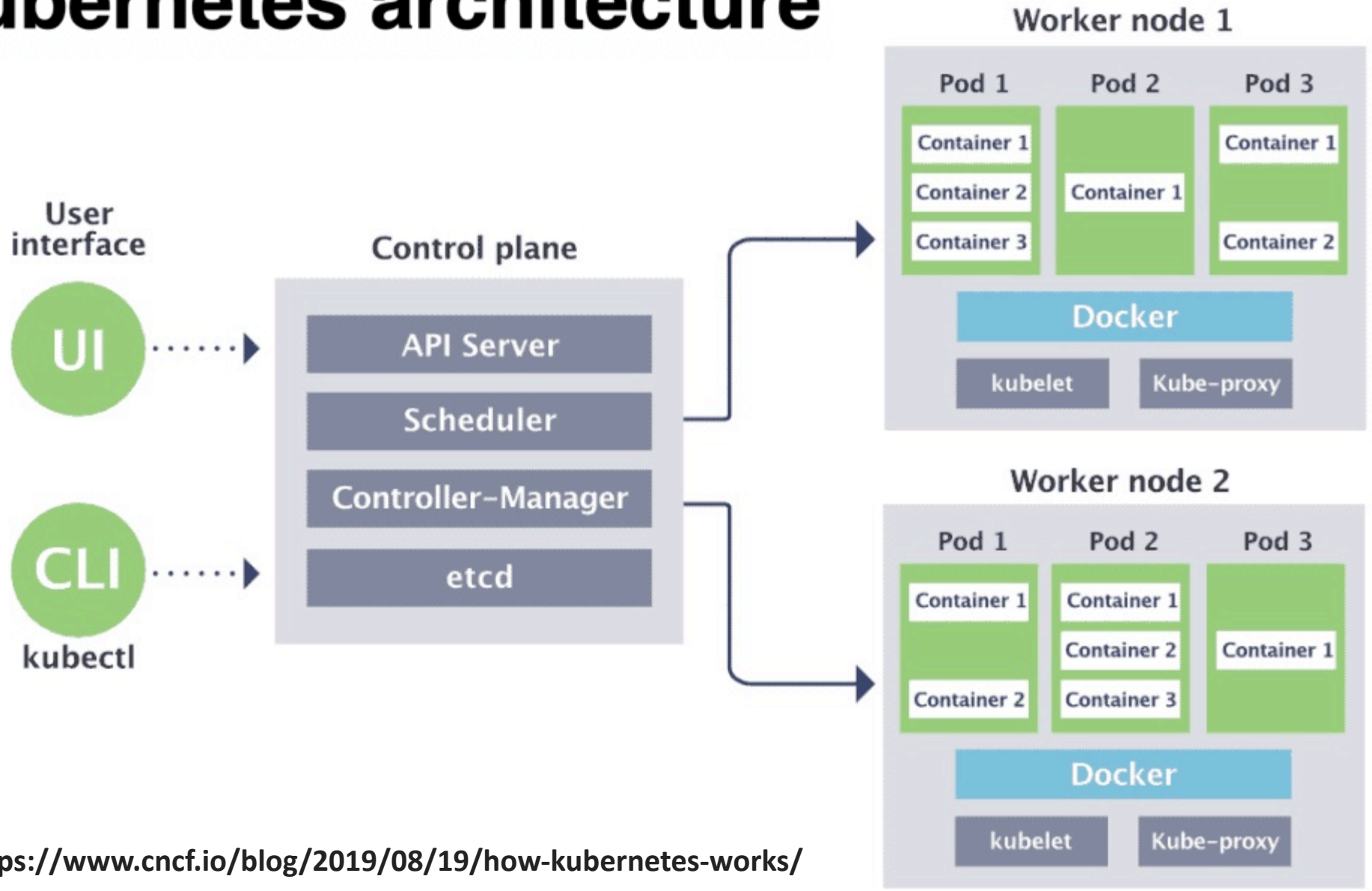

https://github.com/nick-ivanov-edb



PASS
Data Community
SUMMIT

Backup slides

Kubernetes architecture



Source: https://www.cncf.io/blog/2019/08/19/how-kubernetes-works/