



EDB Postgres, Cloud and Kubernetes Day



“**PostgreSQL e Kubernetes** si complementano a vicenda ed insieme riescono a creare la **miglior esperienza** di un database PostgreSQL”



Running PostgreSQL in Kubernetes

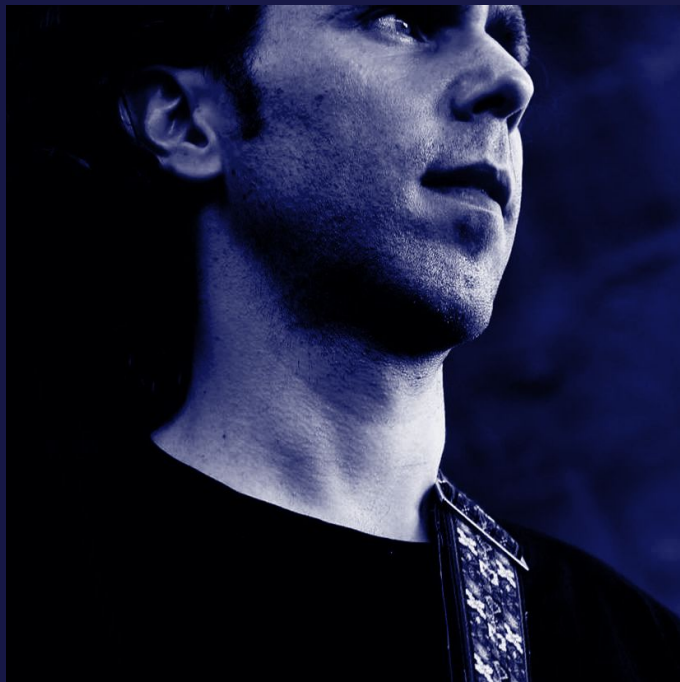
from day 0 to day 2 with CloudNativePG

Gabriele Bartolini, VP & CTO of Cloud Native at EDB

@EDBPostgres @CloudNativePG @_GBartolini_

Milano, 7 ottobre 2022

About me



- VP/CTO di Cloud Native at EDB
 - In precedenza a 2ndQuadrant
- Uso PostgreSQL dal ~2000
 - Membro della comunità dal 2006
 - Co-fondatore di PostgreSQL Europe
- Appassionato di DevOps
- Contributor Open source
 - Barman (2011)
 - CloudNativePG (2022)

Voi e Kubernetes

1. Non mi interessa
2. Non lo uso, ma mi interessa
3. Lo uso in sviluppo e/o pre-produzione
4. Lo uso in produzione per applicazioni stateless (DB fuori)
5. Lo uso in produzione con database
6. La nostra azienda lo integra nella catena di produzione

Messaggio importante

Questa presentazione assume che conosciate almeno le basi di Kubernetes.

Tuttavia, questa presentazione può essere utile anche per chi non le ha, stimolando nuove idee e prospettive nel medio termine.

Kubernetes richiede una preparazione specifica e approfondita. **Non si impara in poche ore di un corso.** Le basi si possono ottenere in un periodo intensivo e ricco di esercizi pratici che varia da 6 a 12 settimane (secondo la mia esperienza).

Consiglio personale: ottenere la certificazione CKA della CNCF.

Soltanto una parte delle slide è stata tradotta in italiano.

Premessa

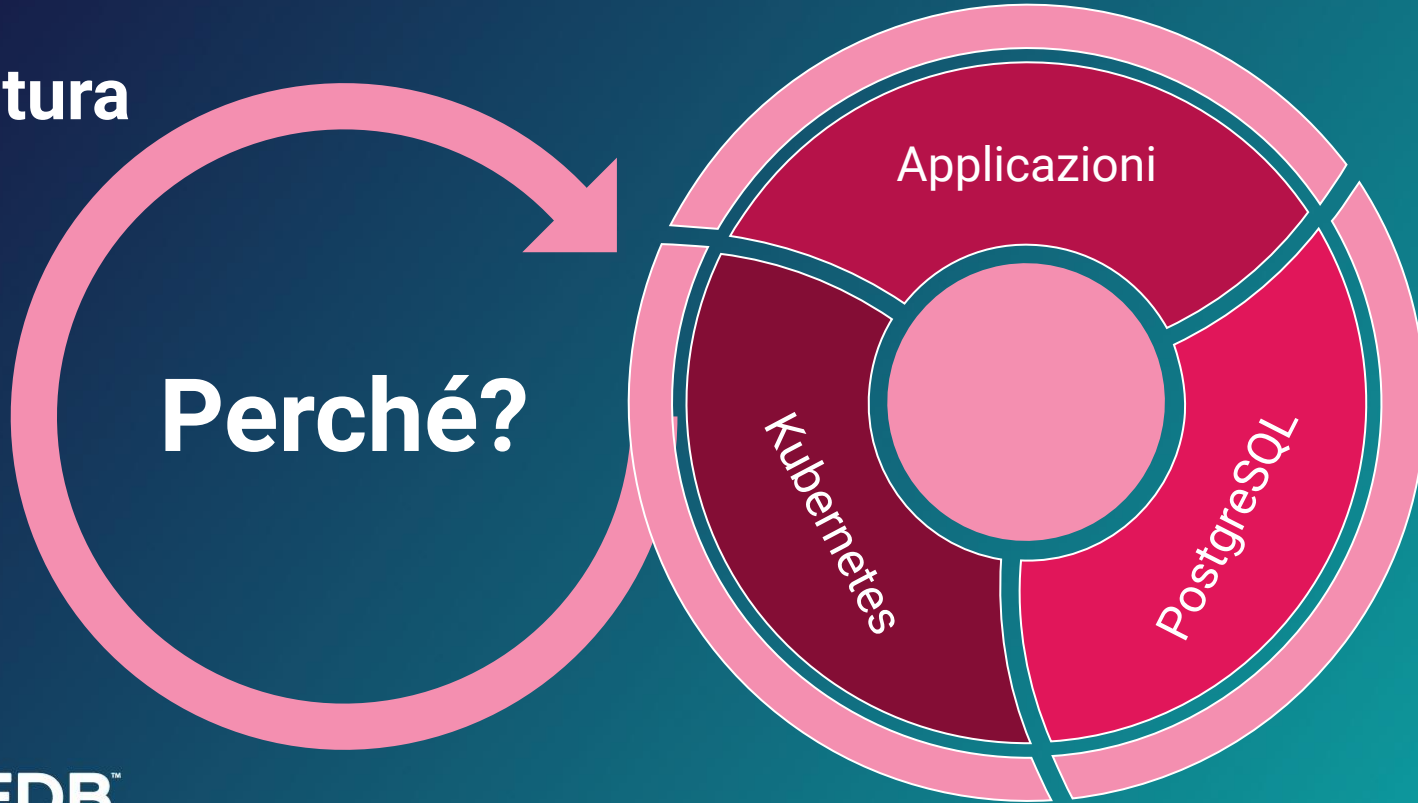
Alcuni falsi miti
su Kubernetes

Kubernetes
non è
una **moda passeggera**

Kubernetes non è magia

cultura

Perché?





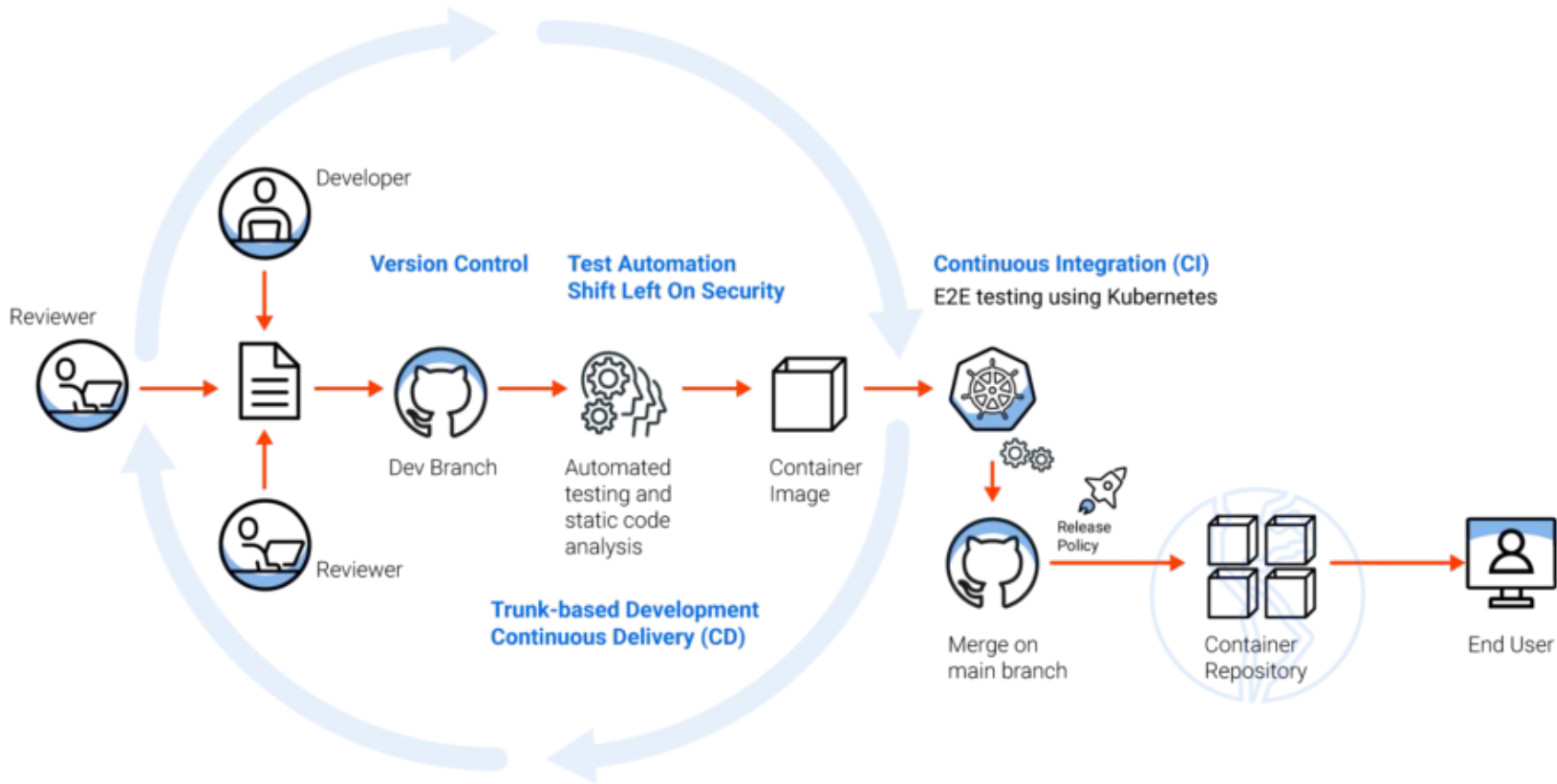
Why Run Postgres in Kubernetes?

📅 May 5, 2022 🏷️ data storage, database, KubeCon, Postgres, SQL

 by Gabriele Bartolini

PostgreSQL is an amazing open source project that has concretely contributed to innovation in the database management industry for at least the last two decades. Built on the solid foundations that were conceived by one of the luminaries of database science, Michael R. Stonebraker, over time PostgreSQL has become one of the most popular database management systems in the world, especially in virtualized and bare-metal installations.





Kubernetes funziona **ovunque**

Infrastruttura standard

IaC

Container immutabili

Da macchine virtuali a data center virtuali

Kubernetes è pronto per i **database** come **Postgres**

90% credono sia pronto per stateful workload (DoK)
Scoglio principale nell'adozione sono gli operatori



Database come PostgreSQL
sono **applicazioni stateful
complesse.**

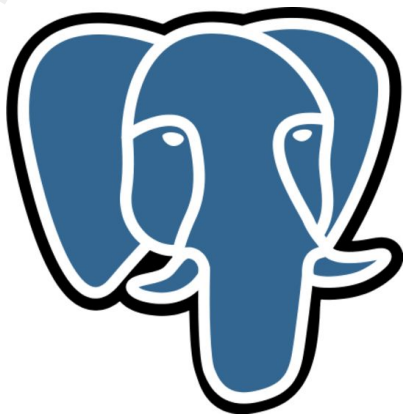
Il pattern “Operator”!

CloudNativePG

Run PostgreSQL.
The Kubernetes way.



CloudNativePG



Il progetto CloudNativePG

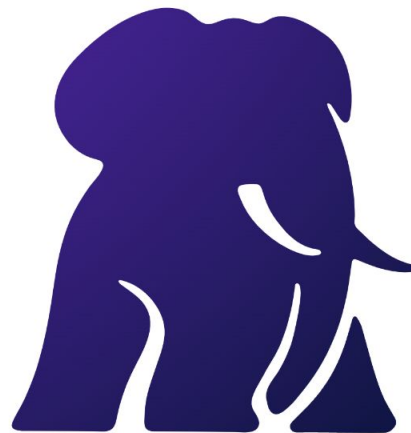
- **Organizzazione “CloudNativePG” su GitHub: github.com/cloudnative-pg**
 - Operatore CloudNativePG (progetto principale): github.com/cloudnative-pg/cloudnative-pg
 - Immagini di operandi contenenti PostgreSQL e PostGIS
 - Chart per helm (operatore, cnpg-sandbox, pgbench)
- **Licenza: Apache 2.0**
- **IP: “The CloudNativePG Contributors”**
 - Abbiamo fatto domanda per la Sandbox della CNCF per donare il progetto alla CNCF
 - Primo progetto su PostgreSQL/Kubernetes a provarci

A proposito di CloudNativePG

- Operatore per Kubernetes
- “Day 1 & 2 operation” di un database PostgreSQL
- Attività solitamente riservate a essere umani
- Open source
 - Creato e sviluppato originariamente da EDB
 - Comunità “Vendor neutral/openly governed”
 - Licenza Apache 2.0
 - Submitted to the CNCF Sandbox
- Pronto per la produzione
 - BigAnimal
 - Clienti di EDB
- Ultima minor version: 1.17
 - La 1.18 è programmata per KubeCon NA

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](#)

Autopilot

It automates the steps that a human operator would do to deploy and to manage a Postgres database inside Kubernetes, including automated

Data persistence

It doesn't rely on statefulsets and uses its own way to manage persistent volume claims where the PGDATA is stored.

Designed for Kubernetes

It's entirely declarative, and directly integrates with the Kubernetes API server to update the state of the cluster — for this reason, it does not



main

16 branches

38 tags

Go to file

Add file

Code

About



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

[cloudnative-pg.io](#)
[kubernetes](#)
[postgres](#)
[devops](#)
[sql](#)

3 authors doc: FAQ about cluster impact during Operator outage (#779) ... ✓ e7c9384 10 hours ago 1,751 commits

.github	ci: fix 'ok to merge' labeling (#763)	6 days ago
api/v1	feat: cluster-managed replication slots for High Availability (#740)	5 days ago
cmd	feat: add destroy command to the cnpg plugin (#643)	26 days ago

Mettete una



licenses/go-licenses	chore: update go-licenses files (#173)	4 months ago
pkg	fix: honour MAPPEDMETRIC and DURATION metric types conversion ...	5 days ago
releases	Version tag to 1.17.0 (#701)	20 days ago
tests	test: ensure tabby writer cache is always flushed before returning the...	5 days ago
.dockerignore	feat: add multiarch build for the operator	12 months ago
.gitignore	ci: make test now installs envtest	9 months ago
.golangci.yml	chore: Update to golangci-lint 1.49 (#658)	28 days ago
.goreleaser.yml	feat: add support for armv8 and armv7 (#160)	4 months ago
.spellcheck.yml	docs: Add pgaudit extensions	7 months ago

Readme

Apache-2.0 license

Code of conduct

481 stars

10 watching

32 forks

Releases 9

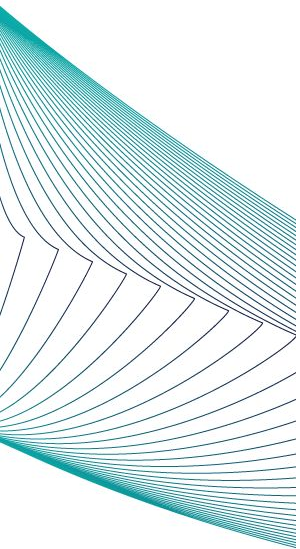
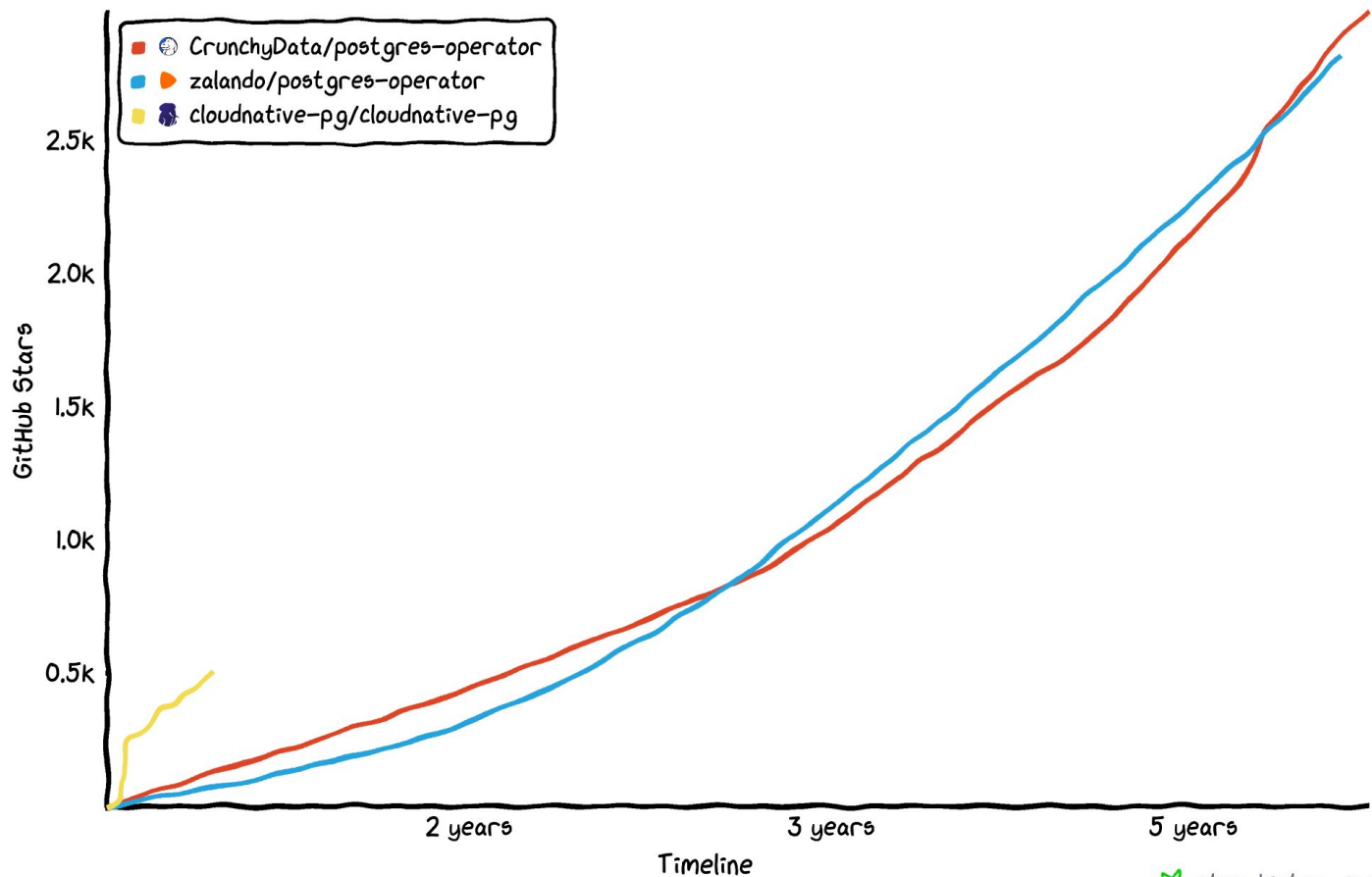
v1.17.0 (Latest)

20 days ago

+ 8 releases



Star History



Versioni supportate dalla comunità di CloudNativePG

Version	Currently Supported	Release Date	End of Life	Supported Kubernetes Versions	Tested, but not supported
1.17.x	Yes	September 6, 2022	~ February 12, 2023	1.22, 1.23, 1.24	1.19, 1.20, 1.21
1.16.x	Yes	July 7, 2022	~ November 25, 2022	1.22, 1.23, 1.24	1.19, 1.20, 1.21
1.15.x	No	April 21, 2022	October 6, 2022	1.21, 1.22, 1.23	1.19, 1.20, 1.24
main	No, development only				



Day 0

Pianifica la tua infrastruttura K8s per workload Postgres

- **La prima impressione è quella che conta**
 - L'infrastruttura K8s è spesso pensata solo per stateless workload
 - Scelta comune: database fuori da Kubernetes - DBaaS
- **Database e Kubernetes è un binomio vincente**
 - trasformazione agile (con la 't' e la 'a' minuscole)
 - Architetture condivise o di tipo "shared nothing"
 - Settore storage in K8s in crescita
- **Scegli il tuo storage in modo saggio**
 - Seguendo gli stessi principi per VM e bare metal

KubeCon NA 2022 - talk con Chris Milsted (Ondat)

Thursday, October 27 • 4:30pm - 5:05pm

[Back To Schedule](#)

Data On Kubernetes, Deploying And Running PostgreSQL And Patterns For Databases In a Kubernetes Cluster. - Chris Milsted, Ondat & Gabriele Bartolini, EDB

[Sign up](#) or [log in](#) to save this to your schedule, view media, leave feedback and see who's attending!

<https://sched.co/182>



In this talk we are going to focus on the newly open sourced Cloud Native PostgreSQL operator. From work in the Data on Kubernetes community, we are seeing databases becoming first class citizens in our kubernetes clusters. This talk is going to look at how to combine the cloudnative-pg operator with a CSI-plugin, and how to leverage these two components to deliver databases which are as fast and resilient as their non-kubernetes equivalents. Wondering how to architect for a zero or low RTO and what the effect of replication has on recovery time for your database? Wondering how to use an object store as well to deliver point in time backups to protect from cyber attacks? In this talk we are going to answer these questions and more mapping to the patterns in the cloud native disaster recovery whitepaper and the data protection whitepaper. Like all good talks we will also have a live demo of this all in action. The time of Data on Kubernetes is now, come and find out how.

Speakers



Chris Milsted

Solution Architect, Ondat

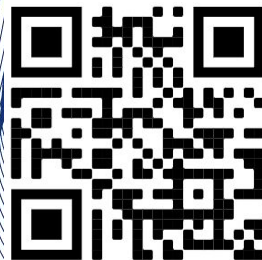
Chris has been working with kubernetes since pre 1.0 when it was the Beta for OpenShift version 3 at Red Hat. Since then he has moved, via VMware and Tanzu, to OnDat as a Solution Architect helping customers to realize the power of data in kubernetes. Outside of work, Chris plays... [Read More](#) →



Gabriele Bartolini

VP Cloud Native, EDB, EDB

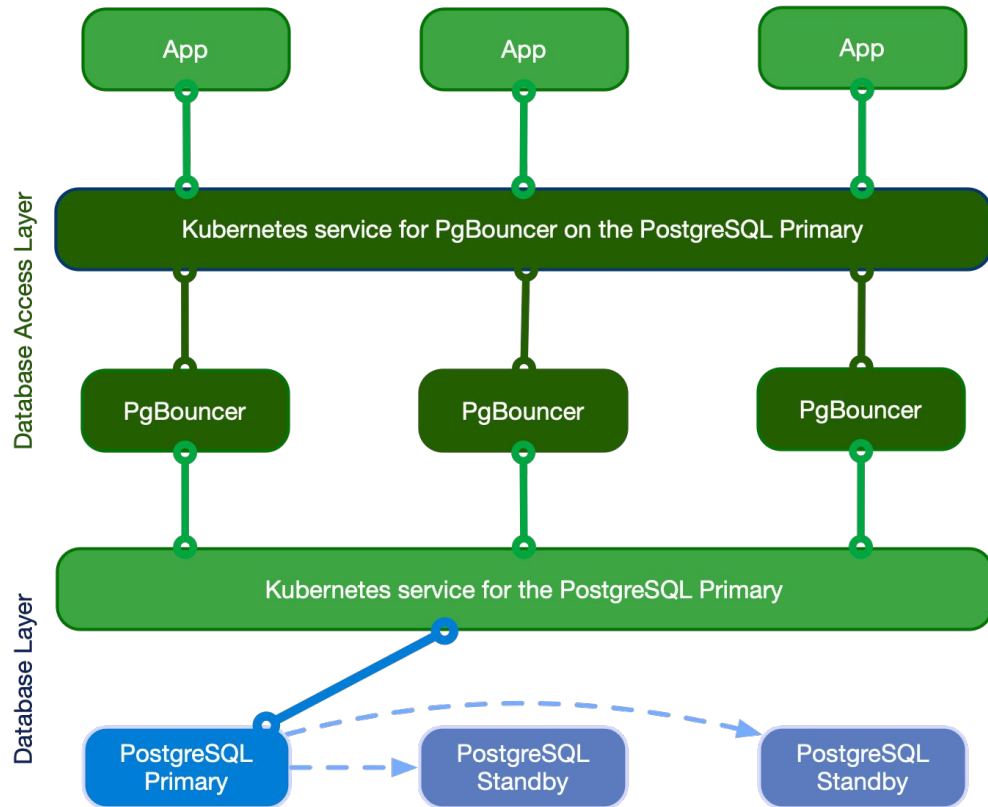
A long time open-source programmer and entrepreneur, Gabriele has a degree in Statistics from the University of Florence. After having consistently contributed to the growth of 2ndQuadrant and its members through nurturing a lean and devops culture, he is now leading the Cloud Native... [Read More](#) →



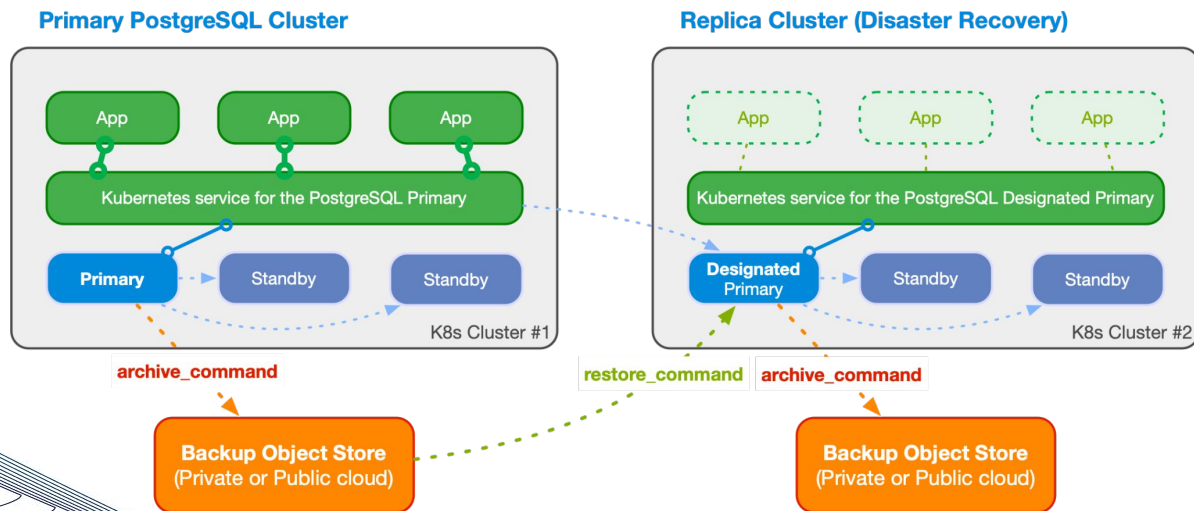
Non finisce qui ...

- Single vs Multi-region
- Availability zone
- Nodi dedicati vs condivisi
- Applicazioni e database
- Connection pooling
- Database “Microservice” o “Monolitico”
- Monitoraggio
- Log Management
- Certificati TLS

Connection pooling con PgBouncer



Cluster in replica fisica per la disaster recovery



Shared nothing architecture

Node

Availability zone 1

Node

Availability zone 2

Node

Availability zone 3



CloudNativePG

Kubernetes cluster



Installare CloudNativePG

```
kubectl apply -f \
```

```
https://raw.githubusercontent.com/cloudnative-pg/cloudnative-pg/main/releases/cnpg-1.17.1.yaml
```

Configurazione dichiarativa tramite manifesto YAML

Day 1

Obiettivo del giorno 1: cluster Postgres di 3 nodi

- Installare la ultima minor di PostgreSQL 14
- Creare un nuovo **Cluster** con PostgreSQL 14
- Un primario e due standby
- Autenticazione mTLS con le repliche
- 4GB di RAM, 8 core, 50Gb di storage
- 1GB di shared buffer
- Un modo per accedere sempre al primario via rete
- Un utente per l'applicazione
- Un database per l'applicazione

myapp-db.yaml

```
apiVersion: postgresql.cnpg.io/v1
kind: Cluster
metadata:
  name: myapp-db
spec:
  instances: 3

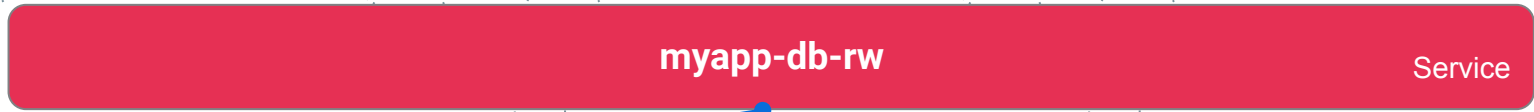
  postgresql:
    parameters:
      shared_buffers: "1GB"

  resources:
    requests:
      memory: "4Gi"
      cpu: 8
    limits:
      memory: "4Gi"
      cpu: 8

  storage:
    size: 50Gi
```

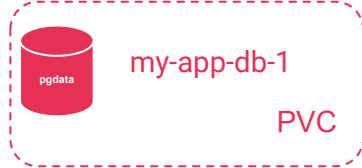
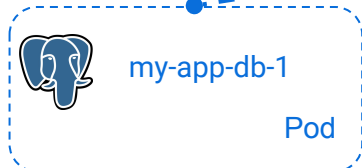
Come fare il deploy del cluster

```
kubectl apply -f myapp-db.yaml
```



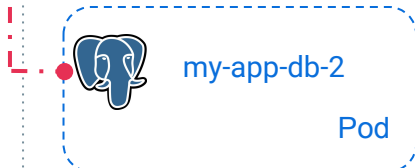
(m)TLS

mTLS



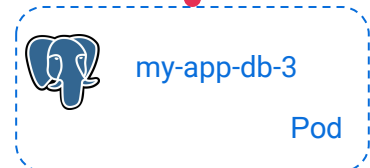
Node

Availability zone 1



Node

Availability zone 2



Node

Availability zone 3



CloudNativePG

Kubernetes cluster



There's more

- A service to access read-only replicas (myapp-db-ro)
- A service to access any instance for reads (myapp-db-r)
- Many other Kubernetes objects are created:
 - Secrets
 - ConfigMaps
 - Roles
 - RoleBindings
 - ServiceAccounts
 - ...
- Convention over configuration
- Separate volume for WAL files
- Import existing databases
 - Even outside Kubernetes
 - Performing major upgrades of Postgres

PostgreSQL configuration

- **Most GUCs are configurable**
 - `.postgresql.parameters` section
 - Some cannot be changed (e.g. `log_destination`)
 - Some have defaults
- **Host-Based Authentication can be configured**
 - `.postgresql.pg_hba` section
 - By default:
 - Requires TLS authentication for streaming replicas
 - Fallback sets sha-256/md5 authentication
- **CloudNativePG supports changes of configuration**
 - Reload
 - Rolling updates if restart is required
 - **Update of standby sensitive parameters**

Leverage a New Way to Import an Existing Postgres Database to Kubernetes

[Gabriele Bartolini](#) – 15/8/2022

Product updates Kubernetes

Are you thinking about moving your PostgreSQL databases to Kubernetes but wondering how to do it? What about importing databases from RDS for PostgreSQL or another database as a service?

Release 1.16 of the [CloudNativePG open source operator](#) introduces a new feature which makes it easier to import inside Kubernetes an existing Postgres database, from any location as long as it can be reached via the network.

The same feature also enables major version upgrades of PostgreSQL, as well as migrating to CloudNativePG any existing PostgreSQL database that you are already running inside Kubernetes with a different operator—or without one, using a pure statefulset based deployment.

This feature enhances the *initdb* bootstrap, by introducing a new subsection called import. Such a section is evaluated only if it is not empty, after the cluster has been initialized from scratch, and it defines which data to import from an existing Postgres instance. Such a Postgres instance can be running in a virtual machine, or on bare metal, or even as a service—like Amazon RDS. The important thing is that objects can be exported using logical backup from the source, and subsequently imported in the target instance.

Categories

Company

PostgreSQL E

Customer Suc

Postgres Tuto

Popular Posts

[Connecting PostgreSQL using psql and](#)



Esempio con volume dedicato per i WAL

```
apiVersion:  
postgresql.cnpg.io/v1  
kind: Cluster  
metadata:  
  name: myapp-db  
spec:  
  instances: 3  
  
  [...]  
  
  storage:  
    size: 50Gi  
  
  walStorage:  
    size: 5Gi
```


Day 2

The role of a Kubernetes operator for Postgres

- Simulate the work of a human DBA
- Do it in a programmatic and automated way
- Extend the Kubernetes API server
 - The only authority for the whole infrastructure
 - Single source of truth of the status of the infrastructure
 - Current status
 - Desired status
 - Reason why we don't rely on failover management tools like Patroni, Stolon, or repmgr
- Kubernetes controls both applications and databases

Rolling updates

- **Update of a deployment with ~zero downtime**
 - Standby servers are updated first
 - Then the primary:
 - supervised / unsupervised
 - switchover / restart
- **When they are triggered:**
 - Security update of Postgres images
 - Minor update of PostgreSQL
 - Configuration changes when restart is required
 - Update of the operator
 - Unless in-place upgrade is enabled

Backup and recovery

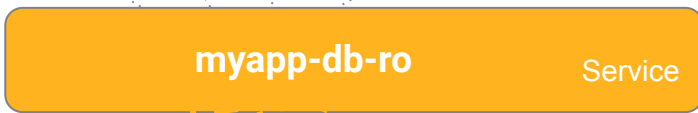
- **Continuous physical backup on “backup object stores”**
 - Scheduled and on-demand base backups
 - Continuous WAL archiving (including parallel)
- **Support for recovery window retention policies (e.g. 30 days)**
- **Recovery means creating a new cluster starting from a “recovery object store”**
 - Then pull WAL files (including in parallel) and replay them
 - Full (End of the WAL) or PITR
- **Both rely on Barman Cloud technology**
 - AWS S3
 - Azure Storage compatible
 - Google Cloud Storage

Synchronous replication

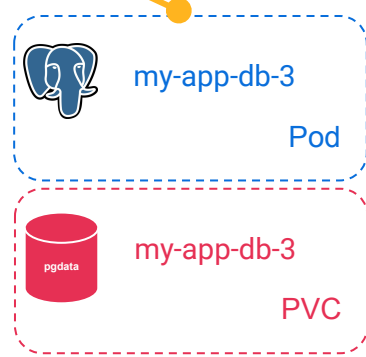
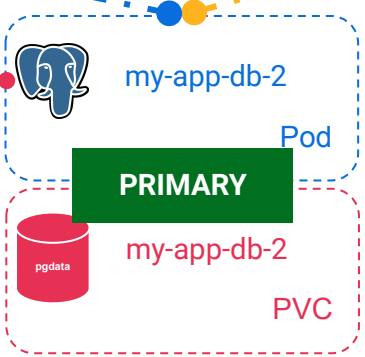
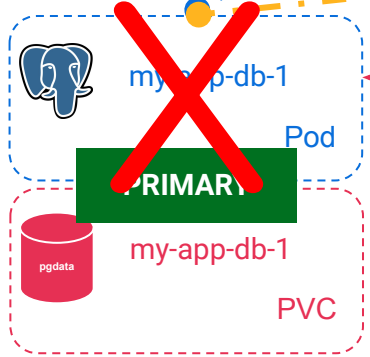
- Quorum-based synchronous streaming replication
- Controlled by two options:
 - `minSyncReplicas`
 - `maxSyncReplicas`
- CloudNativePG takes care of `synchronous_standby_names`
 - ANY `q` (`pod1, pod2, ...`)
 - Where:
 - `1 <= minSyncReplicas <= q <= maxSyncReplicas <= readyReplicas`
 - `pod1, pod2, ...` is the list of all PostgreSQL pods in the cluster
- Reduce risk of data loss

Monitoring

- Native support for Prometheus
- Built-in metrics at the operator level
- Built-in metrics at the Postgres instance level
- Customizable metrics at the Postgres instance level
 - Via ConfigMap(s) and/or Secret(s)
 - Syntax compatible with the PostgreSQL Prometheus Exporter
 - Auto-discovery of databases
 - Queries are:
 - transactionally atomic and read-only
 - executed with the `pg_monitor` role
 - executed with `application_name` set to `cnp_metrics_exporter`
- Support for `pg_stat_statements` and `auto_explain`



downtime



Node

Node

Node

Availability zone 1

Availability zone 2

Availability zone 3

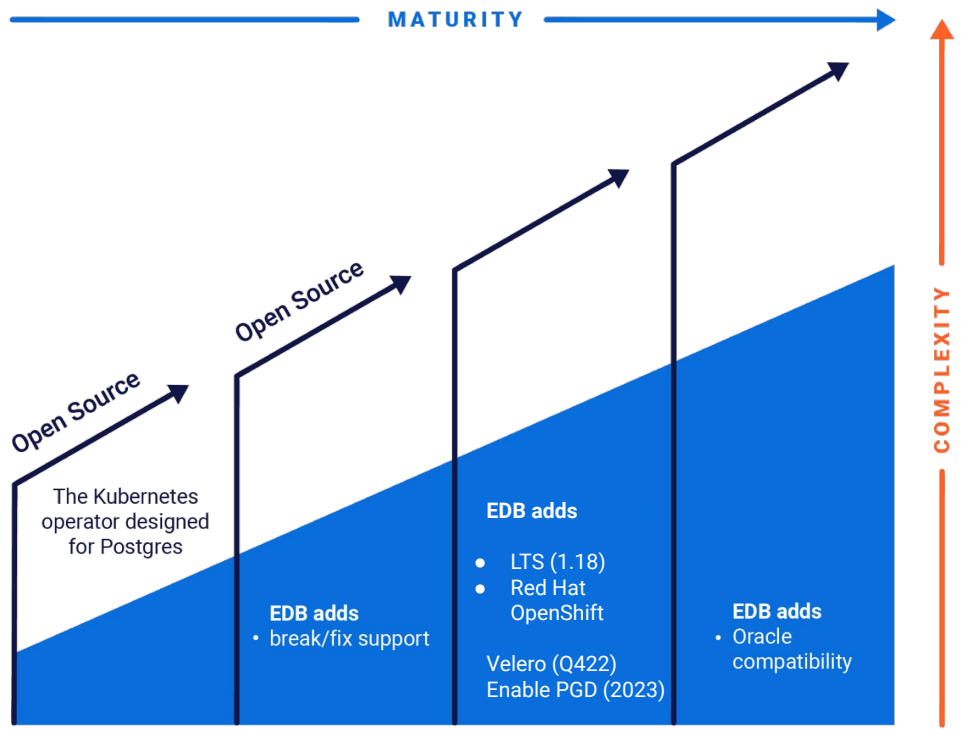


Kubernetes cluster



Part IV

EDB Postgres for Kubernetes



CloudNativePG

EDB Postgres for Kubernetes



Self Supported

EDB Community 360

EDB Standard Plan

EDB Enterprise Plan



	Community	Community 360	Standard	Enterprise
	For organizations that need PostgreSQL expertise and 24/7/365 support	For organizations that need break/fix support for a large PostgreSQL deployment	For businesses that need break/fix support for PostgreSQL, and enterprise tooling	For enterprises that need enterprise features including compatibility with Oracle and enhanced security
EDB Postgres Advanced Server Enterprise-ready, Oracle-compatible PostgreSQL database				✓
PostgreSQL Open source database supported by EDB		✓	✓	✓
Postgres-BDR The most advanced replication solution available for PostgreSQL			Optional	Optional
EDB Tools EDB management, backup, failover, migration, and replication			✓	✓
Open Source Tools Open source management, backup, and failover		✓	✓	✓
Technical Support 24x7 expert technical support	✓	✓	✓	✓
Technical Account Manager Proactive assistance from a trusted advisor		✓	Optional	Optional





Futuro e oltre

Future plans for CloudNativePG

- **Version 1.18 (~ 25 October 2022):**
 - Kubernetes 1.25 support
 - PostgreSQL 15 support
 - Cluster managed physical replication slots for HA
 - Cluster hibernation
- **Beyond 1.18:**
 - Declarative roles
 - Declarative databases
 - Declarative tablespaces

Freedom

It's your choice!



Own your data

Retain full control of your databases and infrastructure



devops

“Generative culture” - more than tools or processes





Grazie! Domande?



github.com/cloudnative-pg



@_GBartolini_

@CloudNativePG

@EDBPostgres



CloudNativePG

Originally created and sponsored by

