Embracing Cloud Native Landscape in Drupal Development

Beyond Docker and Kubernetes

Ricardo Amaro @acquia.com

DRUPAL IBERIA

ASOCIACIÓN ESPAÑOLA DE DRUPAL

Drupal Portugal

Embracing Cloud Native Landscape in Drupal Development

Beyond Docker and Kubernetes

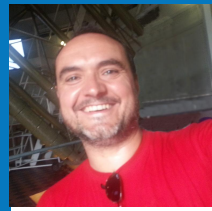
Ricardo Amaro @acquia.com

DRUPAL IBERIA

ASOCIACIÓN ESPAÑOLA DE DRUPAL

Drupal Portugal

Who am I?



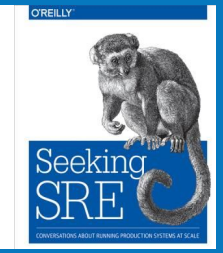
@ricardoamaro

Published author

ACM Computing Surveys

IEEE

Information & Management

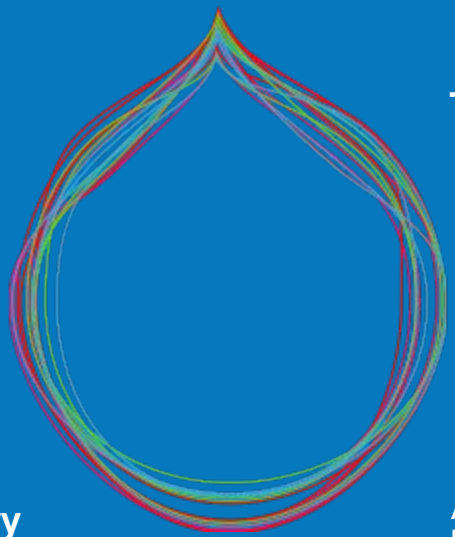


DevOps PhD Research

Drupal Portugal

DevOps Lisbon

Kubernetes OpenTelemetry



+14 years Drupal (DevOps & Infra)

90's Linux & FreeSoftware

12 years at Acquia

Senior Engineering Manager

Applying Platform Automation Engineering, Machine Learning and Cloud Native

https://drupal.org/user/666176

Session Overview

- Current Challenges
- Impact for Drupal Devs & Ops
- Cloud Native Principles
- Key Projects and Products
- Drupal in a Cloud-Native environment overview
- Successful Drupal Transition Use-case
- Conclusion and Key Takeaways

The Challenge at Hand

- The need for scalability, resilience, and flexibility.
- Modernizing Drupal environments and applications.
- Modernizing Drupal applications to leverage distributed systems and cloud-native architectures.

The Impact on Drupal Devs & Ops

- Performance improvements.
- Cost reduction at scale.
- improved reliability.
- “It also works on my machine!”

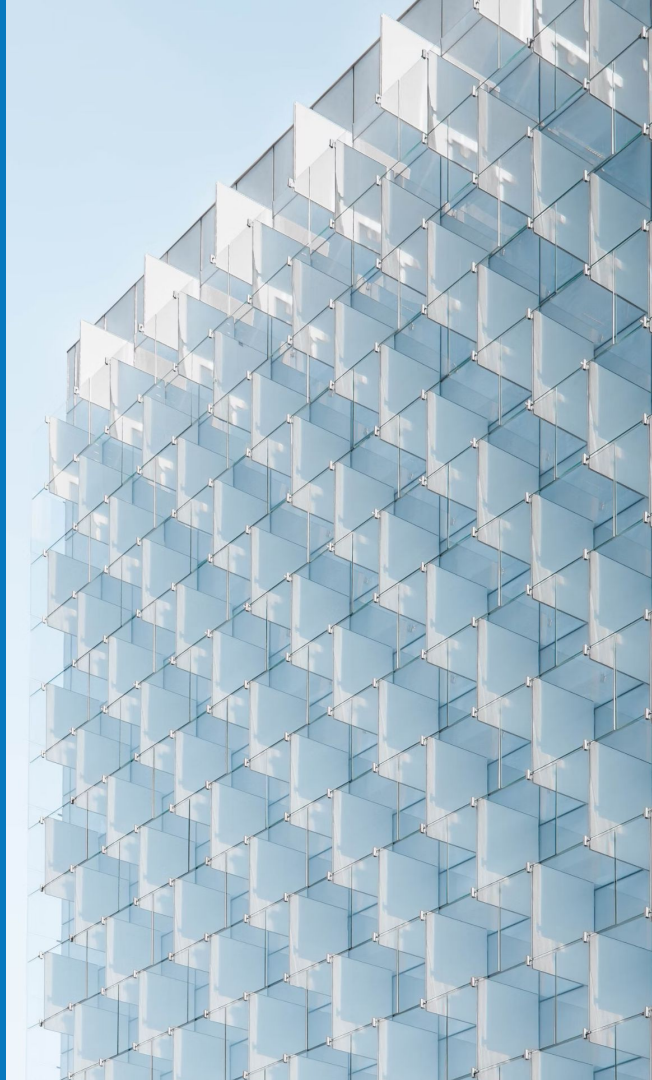
Why Cloud Native?

- Architectural Design
- Operational Efficiency
- System Reliability

Cloud-Native Principles

Architectural Design

- Microservices Architecture
- Containerization
- Declarative APIs
- Loosely Coupled Services



Cloud-Native Principles

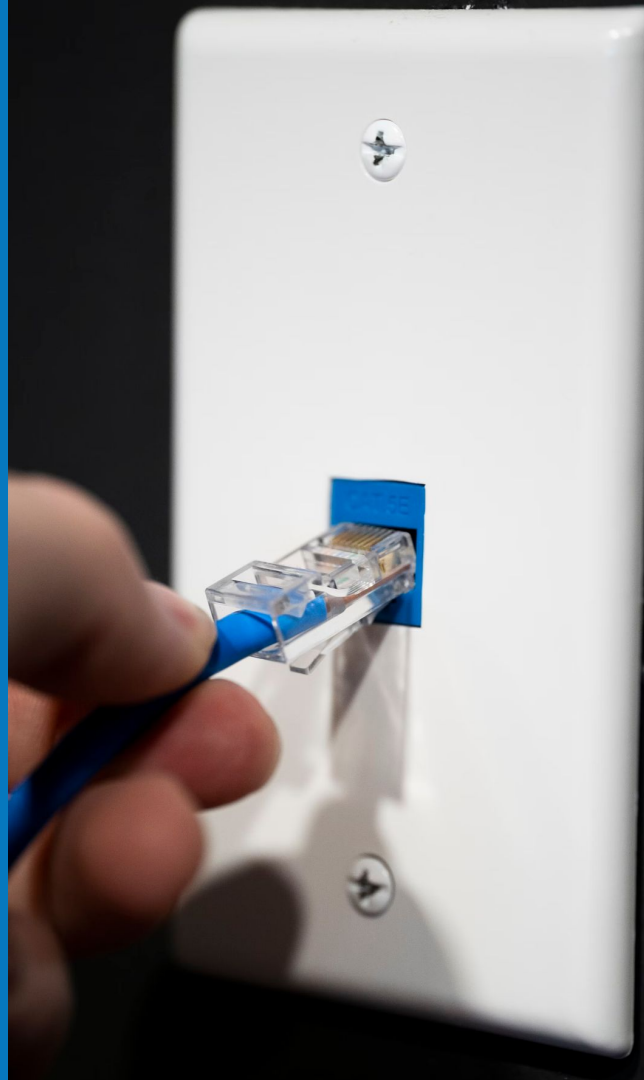
Operational Efficiency

- Orchestration
- Continuous Integration and Delivery
- Automation

Cloud-Native Principles

System Reliability

- Scalability
- Resiliency



184 Projects and Products

App Definition and Development

Orchestration & Management

Application Definition & Image Build

CNCF GRADUATED: HELM

CNCF INCUBATING: Backstage, Buildpacks.io, dapr, KubeVela, KubeVirt, Operator Framework

Continuous Integration & Delivery

CNCF GRADUATED: argo, flux

CNCF INCUBATING: keptn, OpenKruise

Database

CNCF GRADUATED: KV, Vitess

Streaming & Messaging

CNCF GRADUATED: cloudevents

CNCF INCUBATING: NATS, STRIMZI

Scheduling & Orchestration

CNCF GRADUATED: KEDA, kubernetes

CNCF INCUBATING: Crossplane, KARMADA, Knative, VOLCANO

Service Mesh

CNCF GRADUATED: Istio, LINKERD

Remote Procedure Call

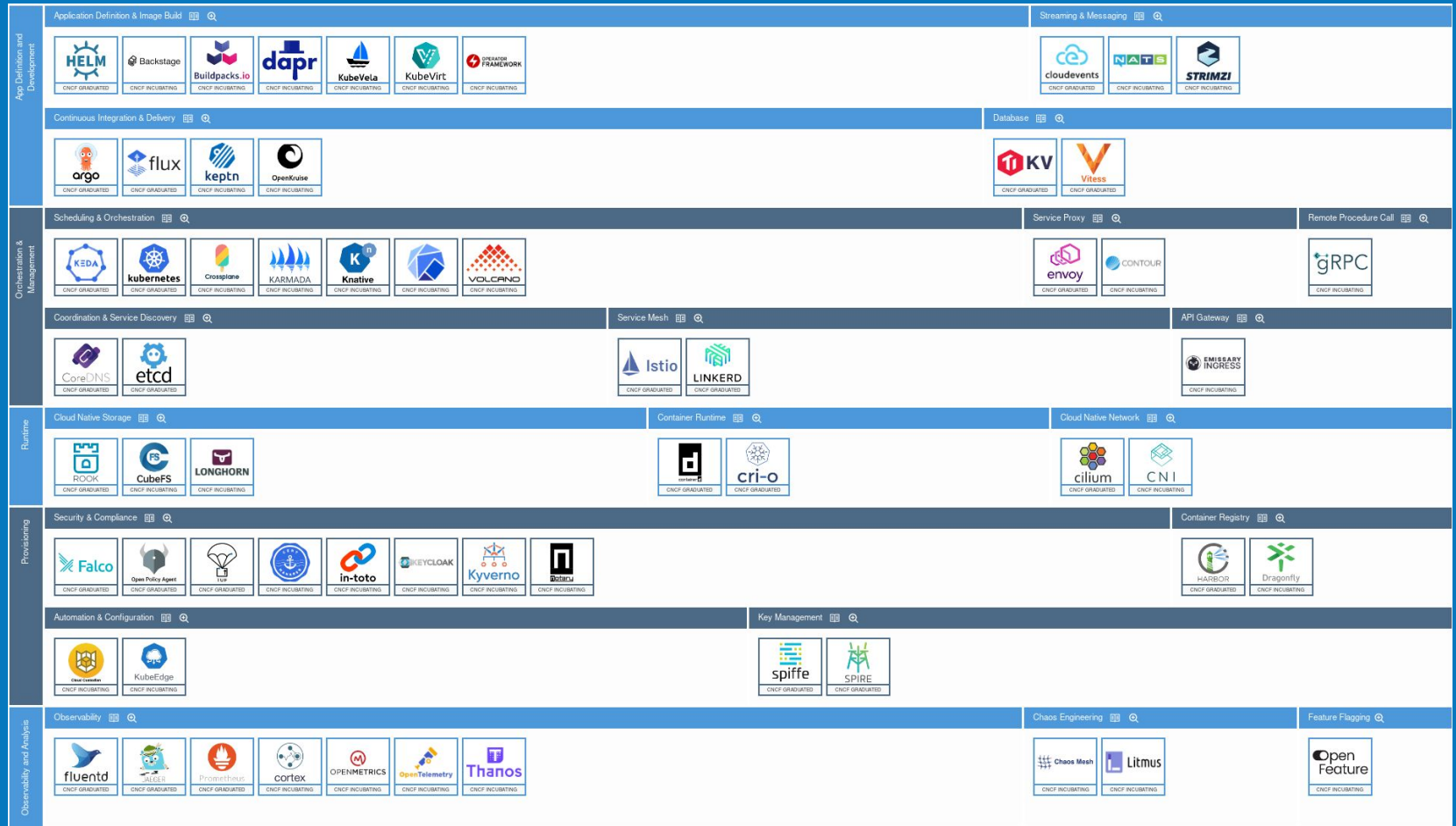
CNCF INCUBATING: gRPC

Service Proxy

API Gateway

Coordination & Service Discovery

62 Graduated + Incubating Projects and Products



Drupal in a Cloud-Native environment

- Development and Management
- Hosting and Provisioning
- Monitoring
- Continuous Integration and Delivery (CI/CD)
- Orchestration
- Security and Compliance

Development and Management

Helm

Hosting and Provisioning

Kubernetes

Terraform

Monitoring

Prometheus

Grafana

Observability

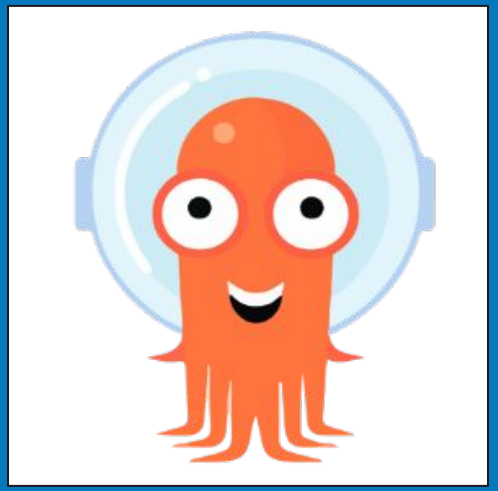
OpenTelemetry

Jaeger

Fluentd

CI/CD and Orchestration

TEKTON



ArgoCD

Security and Compliance

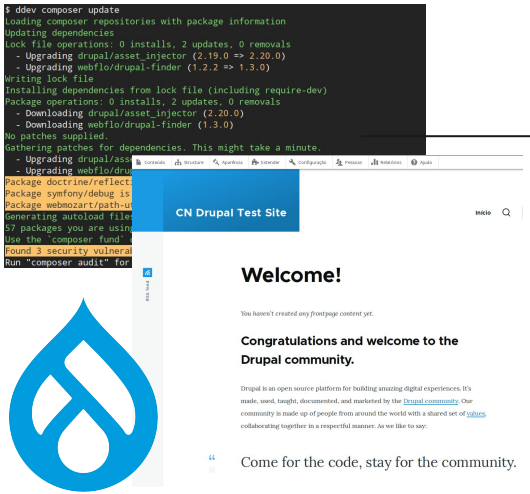
Open Policy Agent (OPA)

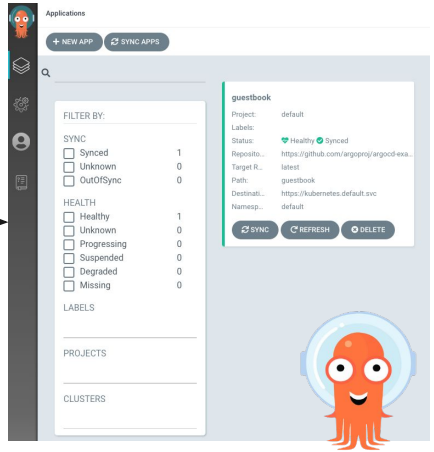
Falco

Cert Manager

Drupal Developer Workflow

DDEV



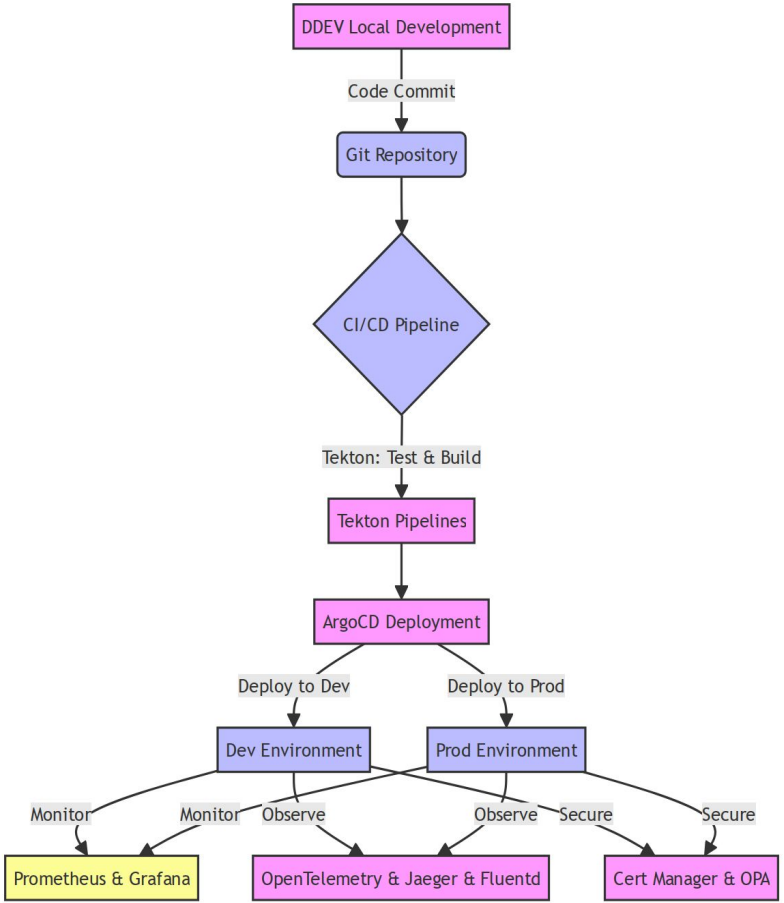


ArgoCD

Build ↔ Test → Release → Deploy to Dev/Prod

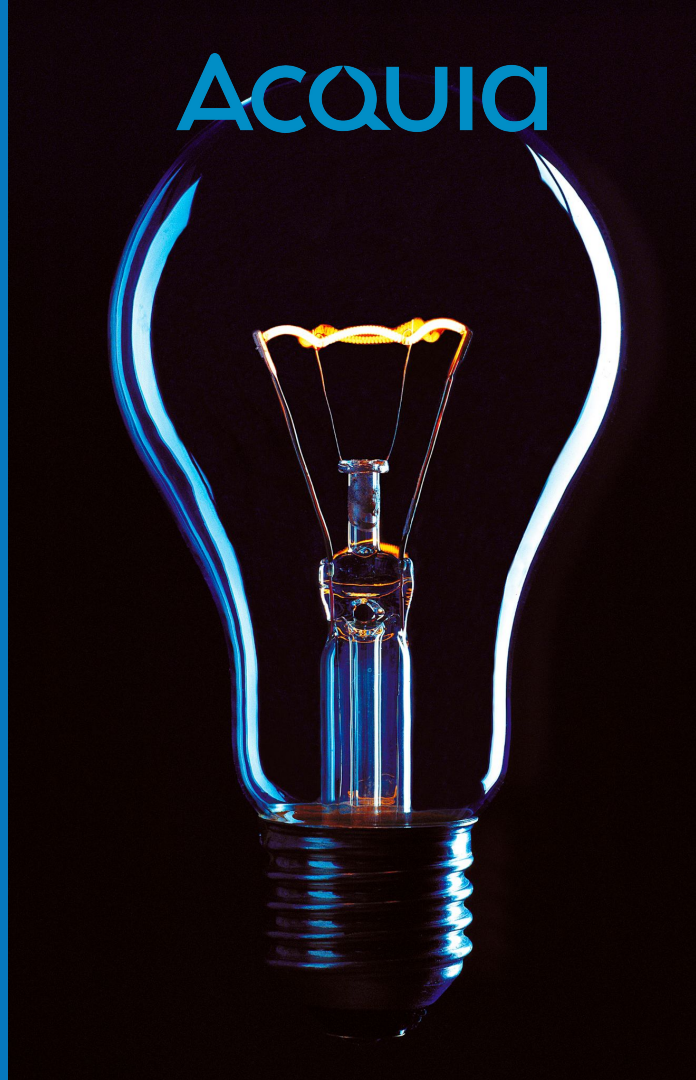
The Drupal CNCF Workflow

- **DDEV**: Docker-based local development for Drupal.
- **Git Repository**: Usual Gilab, GitHub, others.
- **Tekton Pipelines**: For CI/CD test and build process.
- **ArgoCD**: Deploys Drupal to Dev + Prod, ensuring the desired state matches the Git repository.
- **Prometheus & Grafana**: Monitoring and alerting. Collects, stores metrics, visible in Grafana.
- **OpenTelemetry, Jaeger, & Fluentd**: Provide observability through tracing (Jaeger), metrics (OpenTelemetry), and logs (Fluentd).
- **Cert Manager & OPA (Open Policy Agent)**: Cert Manager automates the management of TLS certificates, while OPA enforces security policies and practices.



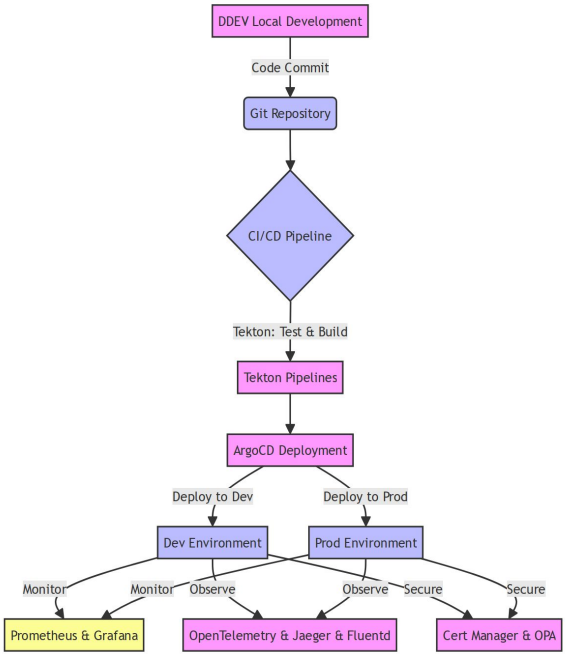
A Successful Transition

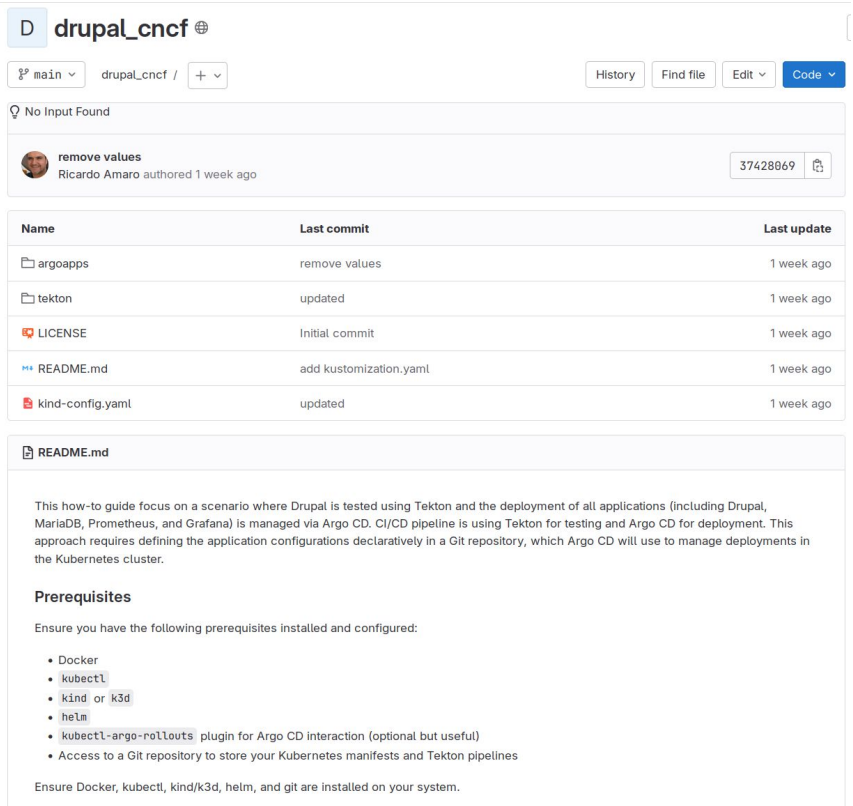
- Drupal Cloud to Cloud-Native transition.
- Key benefits achieved.
  - Scalability
  - Resilience
  - Flexibility
  - Automation
- ~50 clusters and ~68k pods total.
- COGS reduction using Machine Learning.
- Less operational toil, more releases.



The Drupal CNCF project

https://gitlab.com/ricardoamaro/drupal_cncf





D drupal_cncf

main drupal_cncf / + History Find file Edit Code

No Input Found

remove values
Ricardo Amaro authored 1 week ago 37428069

Name	Last commit	Last update
argoapps	remove values	1 week ago
tekton	updated	1 week ago
LICENSE	Initial commit	1 week ago
README.md	add kustomization.yaml	1 week ago
kind-config.yaml	updated	1 week ago

README.md

This how-to guide focus on a scenario where Drupal is tested using Tekton and the deployment of all applications (including Drupal, MariaDB, Prometheus, and Grafana) is managed via Argo CD. CI/CD pipeline is using Tekton for testing and Argo CD for deployment. This approach requires defining the application configurations declaratively in a Git repository, which Argo CD will use to manage deployments in the Kubernetes cluster.

Prerequisites

Ensure you have the following prerequisites installed and configured:

- Docker
- `kubectl`
- `kind` or `k3d`
- `helm`
- `kubectl-argo-rollouts` plugin for Argo CD interaction (optional but useful)
- Access to a Git repository to store your Kubernetes manifests and Tekton pipelines

Ensure Docker, kubectl, kind/k3d, helm, and git are installed on your system.

Conclusion and Key Takeaways

- Core principles of cloud-native development.
- Practical overview of the Drupal use case.
- Tools for effective Cloud-Native adoption.
- Example of a successful migration.
- Open source repository for collaborating on the Drupal CNCF project.

Time for some Q&A

Questions & Answers

Thank you!

Embracing Cloud Native Landscape in Drupal Development

Ricardo Amaro @acquia.com

DRUPAL IBERIA

ASOCIACIÓN ESPAÑOLA DE DRUPAL

Drupal Portugal