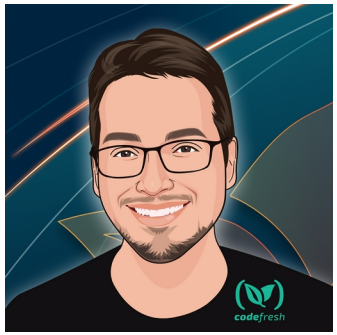
codefresh

# How We Securely Scaled Multi-Tenancy with vcluster, Crossplane, and Argo CD

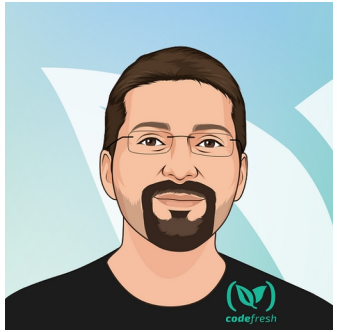
Kubecon 2023 Amsterdam

# Your hosts



Ilia Medvedev

DevOps Engineer - Codefresh



Kostis Kapelonis

Developer Advocate - Codefresh

codefresh

Let’s set the stage

# Provide hosted Argo CD to the masses?

# About Codefresh

- ## Modern Deployment Platform

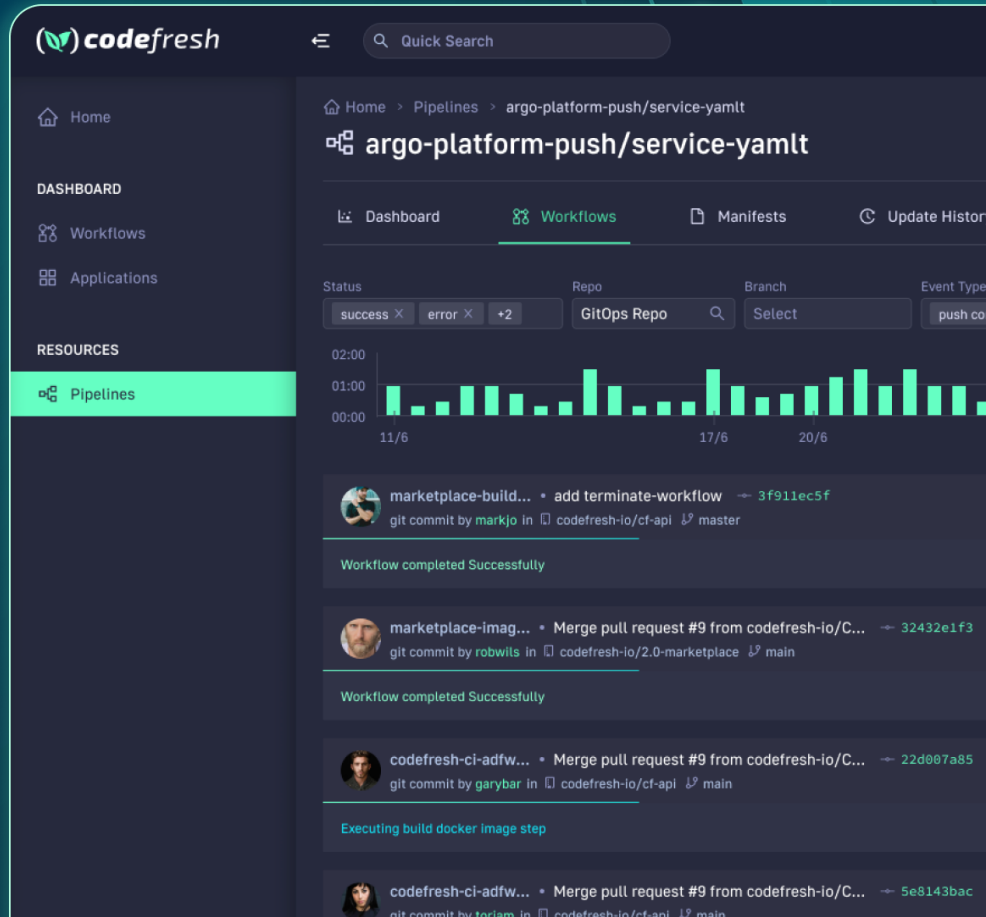
  Comes with CI, CD and GitOps modules

- ## Enterprise Ready

  Code-to-cloud visibility across apps and clusters

- ## Continuous Delivery

  Progressive delivery without compromising stability powered by Argo CD and Argo Rollouts



# Goals

- Allow customers to sign-up
- Offer an Argo CD instance to EACH customer account
- SAAS platform is open to anybody
- We don’t know the number of users in advance
- Essentially we want hosted Argo CD instances

codefresh

# Meme without a picture™

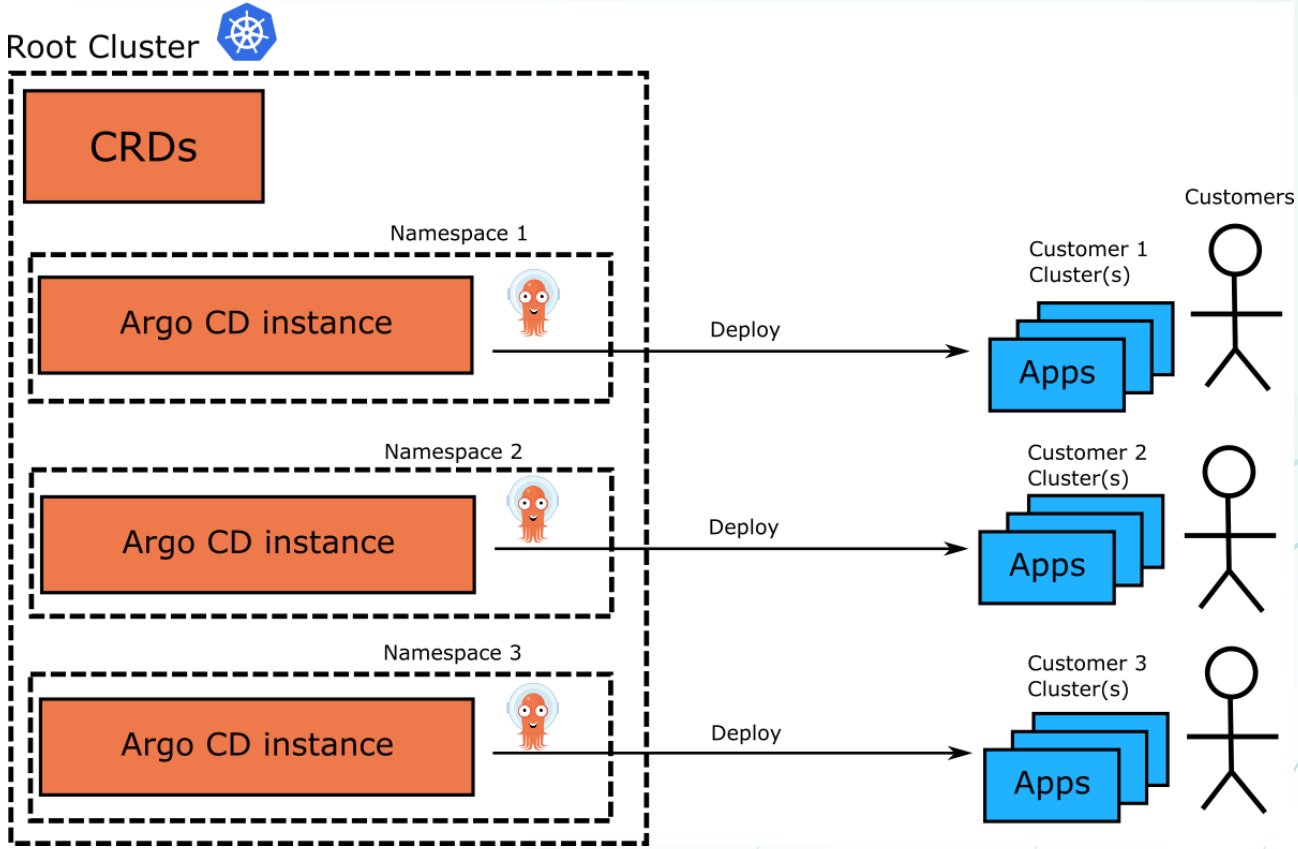
“ You get an Argo CD instance, you get an Argo CD instance, everybody gets their own Argo CD instance”

codefresh

Options, Options, Options

# Possible solutions

# Argo CD per namespace



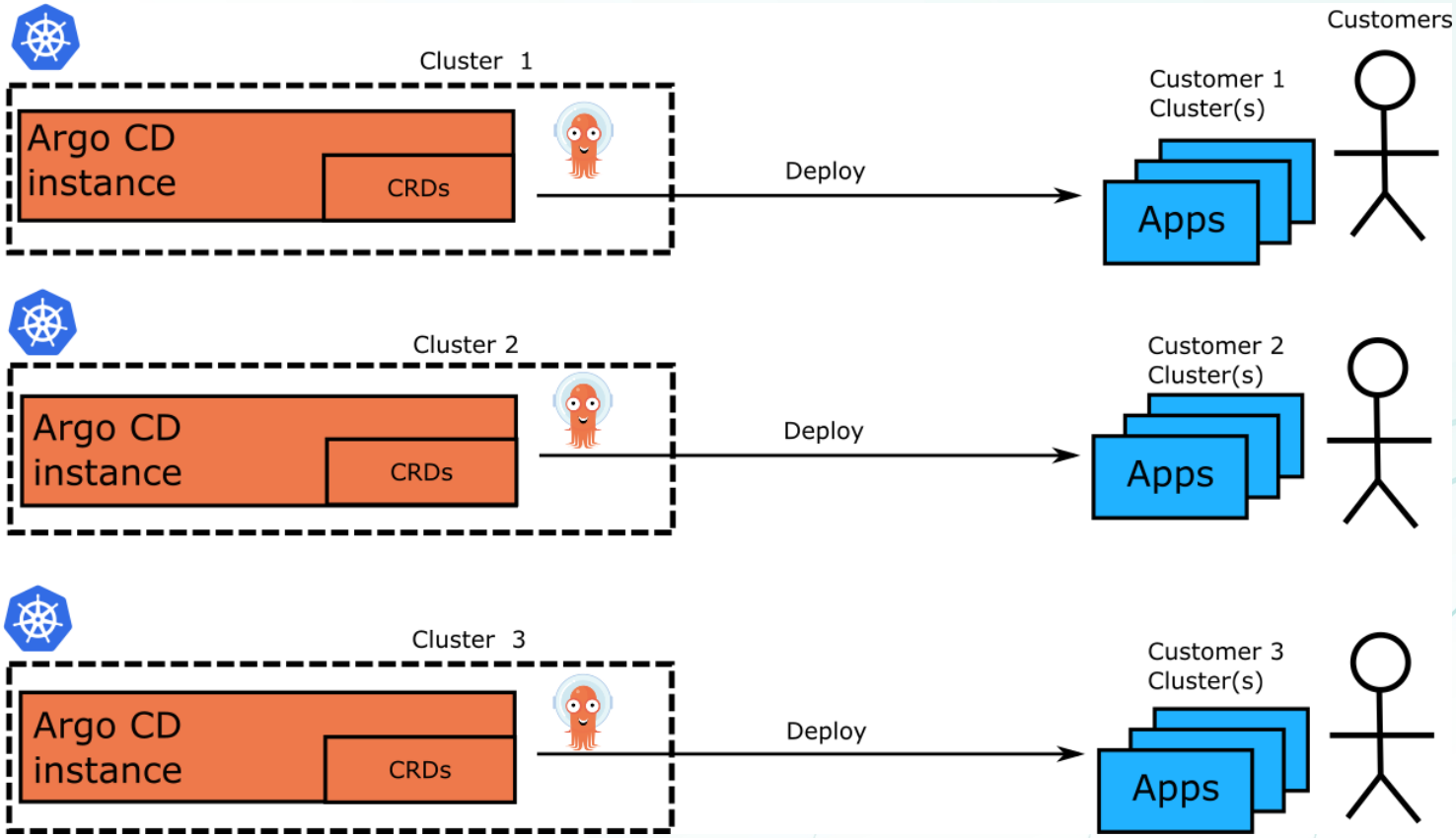

codefresh

# Single cluster to house all Argo CD instances

- Centralized management ✓
- Resource efficient ✓
- Fast setup (namespace based) ✓
- Need to setup policies/quotas/isolation ✗
- “Noisy neighbor” issues ✗
- Argo CD itself has CRDs ✗
- Same cluster version for everybody ✗

codefresh

# Argo CD per cluster




codefresh

# New cluster per account

- Total isolation ✔
- Different cluster version per customer ✔
- No issues with Argo CD CRDs ✔
- Cloud cost issues ✖
- Slow to setup (wait for new cluster) ✖
- Difficult management ✖

codefresh

# Single cluster/multiple ns

✅ - Centralized management
✅ - Cost Effective
✅ - Common resources
✅ - Fast init

❌- No isolation
❌- Tenant confined to namespace
❌- Same K8s version for everybody
❌- CRDs hard to handle
❌- Resource starvation

# Multiple clusters

- Great isolation
- Full cluster access for tenant
- No issues with CRDs
- K8s version flexibility

❌ - Complex management
❌ - Expensive
❌ - No Resource sharing
❌ - Slow init

# What about Security?

- Argo CD has network access to target deployment clusters (including production)
- Compromising Argo CD could compromise production
- Tenants should never get access to other Argo CD instances than their own

codefresh

New kid on the block

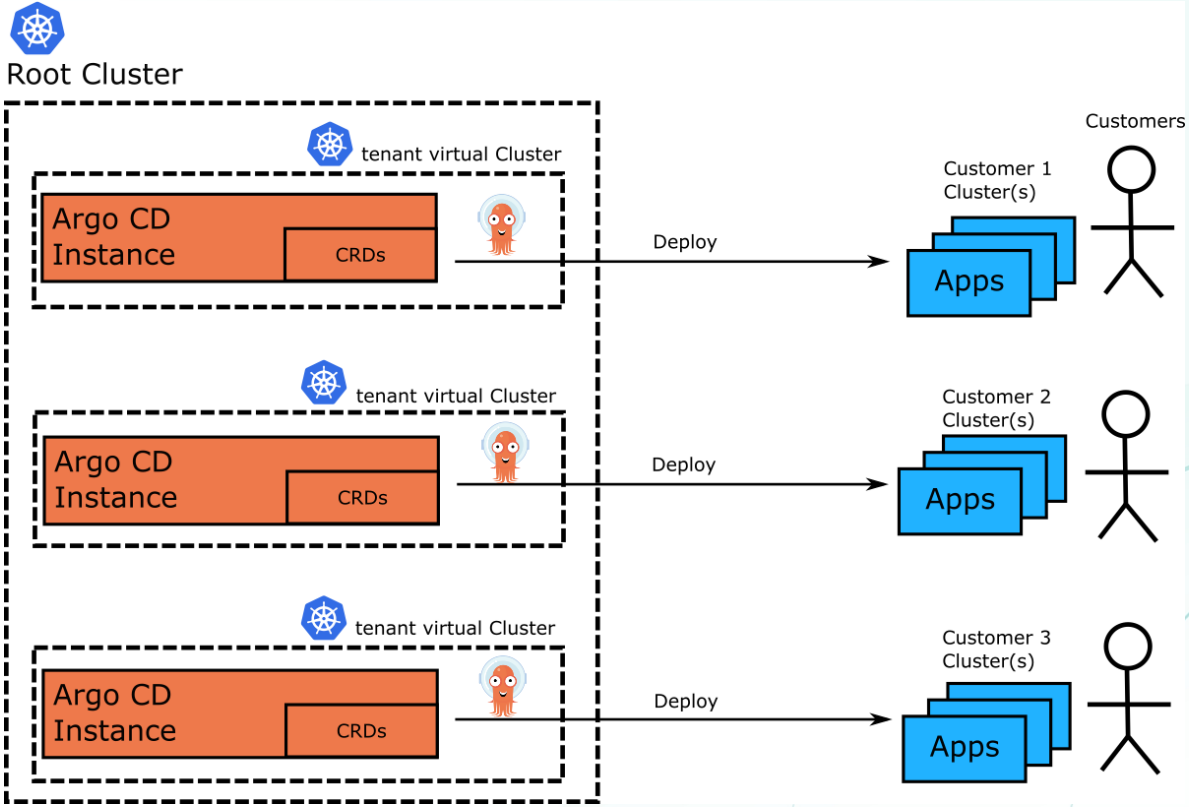
# Enter Virtual cluster

# Virtual Kubernetes clusters

- vcluster.com
- Open source project by Loft Labs
- Cluster within a cluster
- Fully Kubernetes compliant

codefresh

# Virtual Kubernetes clusters




codefresh

# Get the best of both worlds

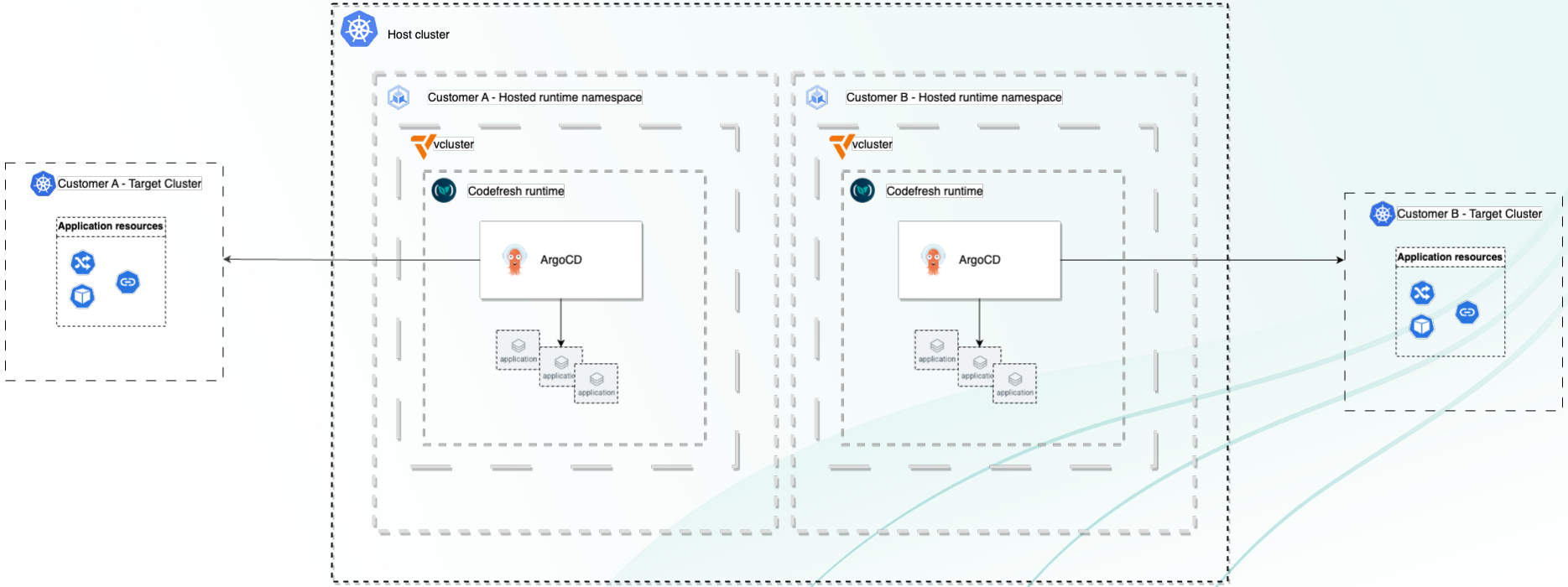
- Good isolation
- Full cluster access for tenant
- Cost effective
- No issues with CRDs
- Centralized management
- Common resources
- Fast init
- K8s Version flexibility

✖ - Some hardening required
✖ - Host cluster is SPF

codefresh

# Implementation

# Solution Architecture

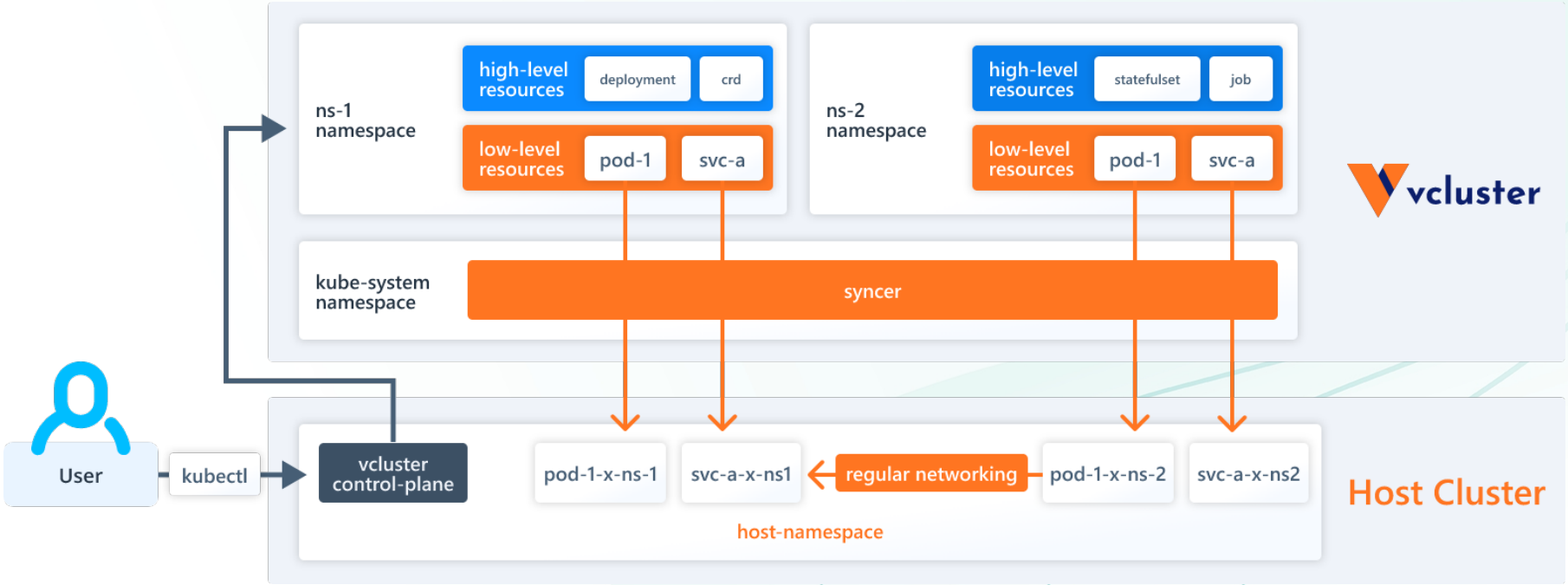



codefresh

# vcluster concepts

- vclusters are deployed on the host cluster just like any other workload- using plain manifests or a Helm chart.
- vclusters are entirely namespace scoped hence their installation does not require cluster admin privileges.
- High level resources are virtual (Deployments, Statefulsets,CRD’s)
- Low level resources that are required for workloads to run are synced to the host (Pods, Secrets, etc)

codefresh




https://www.vcluster.com/docs/architecture/basics

codefresh

# Scaling and automation

- To deploy a single Argo CD instance (Codefresh runtime) we need to:
  - Deploy  vcluster Helm chart
  - Deploy the runtime workloads onto the vcluster - Also using Helm.

- The challenge - Since vcluster has its own Kube API, it’s not as simple as deploying workloads to the same cluster.

- Maybe we should treat vcluster as a piece of infrastructure, and consider tools that belong to infrastructure provisioning domains?

codefresh

# Enter Crossplane

- Kubernetes native.
- Manages non-Kubernetes resources using Kubernetes CRD’s. Even pizza orders!
- Crossplane utilizes Kubernetes control loops to serve as a general purpose control plane that among other things can be used for infrastructure provisioning and lifecycle.


codefresh

# Provisioning infra with crossplane

To provision infrastructure with crossplane we use providers and resources:

- Resources - Are represented using Kubernetes CRD’s and describe the resource we want to provision.
- Providers - Are Kubernetes controllers that manage those resources and provision the infrastructure by invoking 3rd party API’s
- Provider config - Defines how the provider should create resources. For example which credentials to use against the infrastructure provider.

codefresh

# Crossplane example - AWS VPC

```yaml
apiVersion: pkg.crossplane.io/v1
kind: Provider
metadata:
  name: aws-provider
spec:
  package: crossplane/provider-aws:alpha
```

```yaml
apiVersion: aws.crossplane.io/v1beta1
kind: ProviderConfig
metadata:
  name: awsconfig
spec:
  credentials:
    source: Secret
    secretRef:
      namespace: crossplane-system
      name: aws-secret-creds
      key: creds
```

```yaml
apiVersion: ec2.aws.crossplane.io/v1beta1
kind: VPC
metadata:
  name: production-vpc
spec:
  forProvider:
    region: us-east-1
    cidrBlock: 192.168.0.0/16
    enableDnsSupport: true
    enableDnsHostNames: true
    tags:
    - key: Environment
      value: Production
    - key: Owner
      value: Pavan
    - key: Name
      value: production-vpc
    instanceTenancy: default
  providerConfigRef:
    name: awsconfig
```

codefresh

# Crossplane composition

- One of the most powerful features of Crossplane is the ability to create composite resources. Composite resources may utilize multiple provisioners.

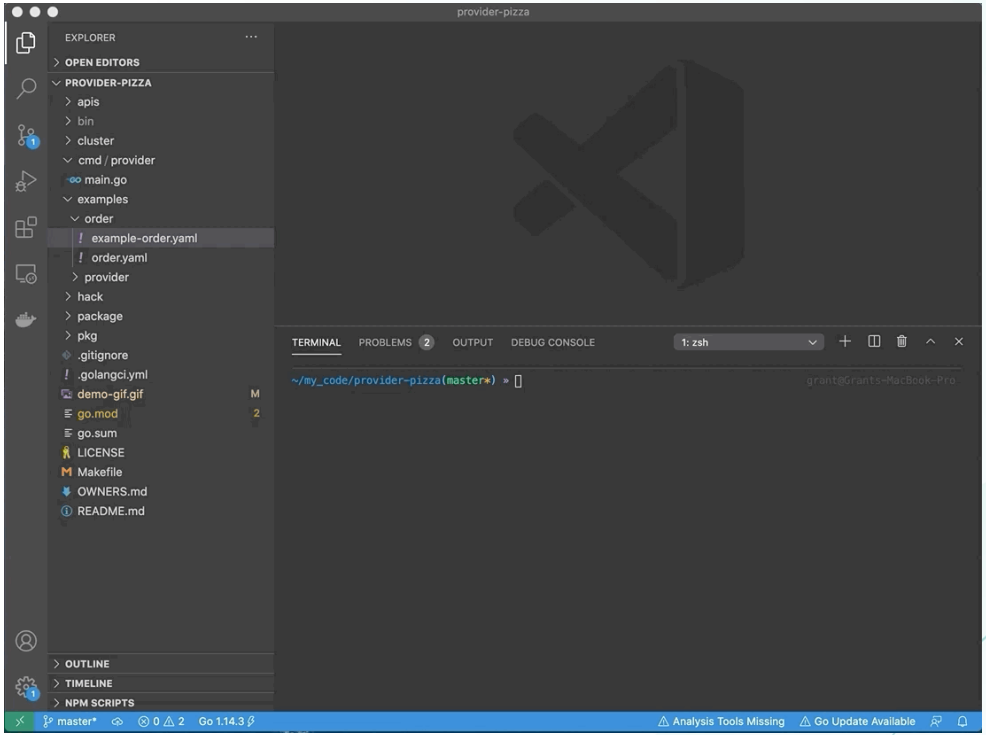
- Create your own CRD and reuse existing controllers.

- An example of such use case would be to provision a GKE cluster using GCP provider and once the cluster is deployed use Kubernetes provider to deploy ArgoCD  onto the cluster.

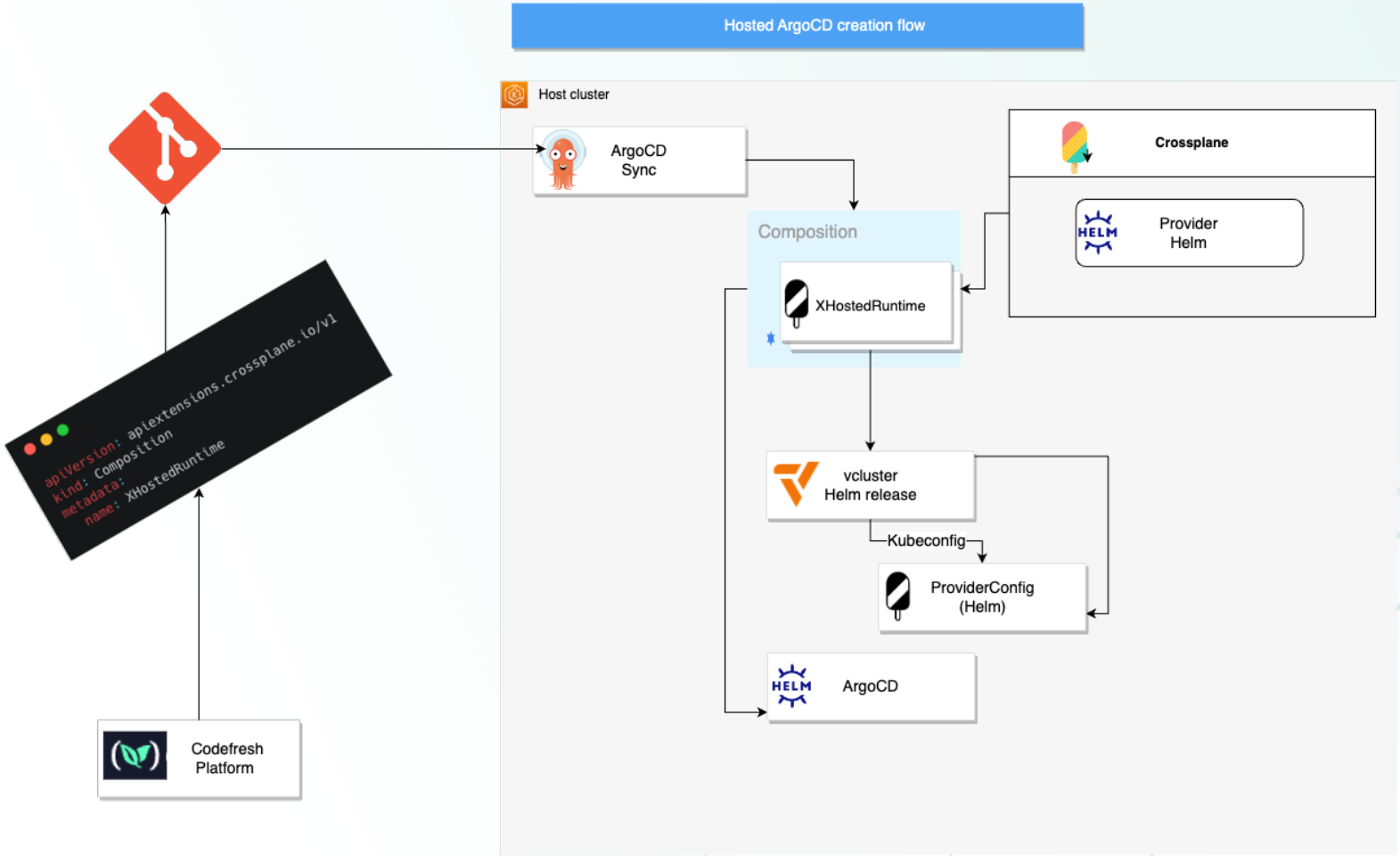
https://github.com/crossplane-contrib/provider-kubernetes/blob/main/examples/in-composition/composition.yaml

codefresh

# Learn how to order a pizza with Crossplane!



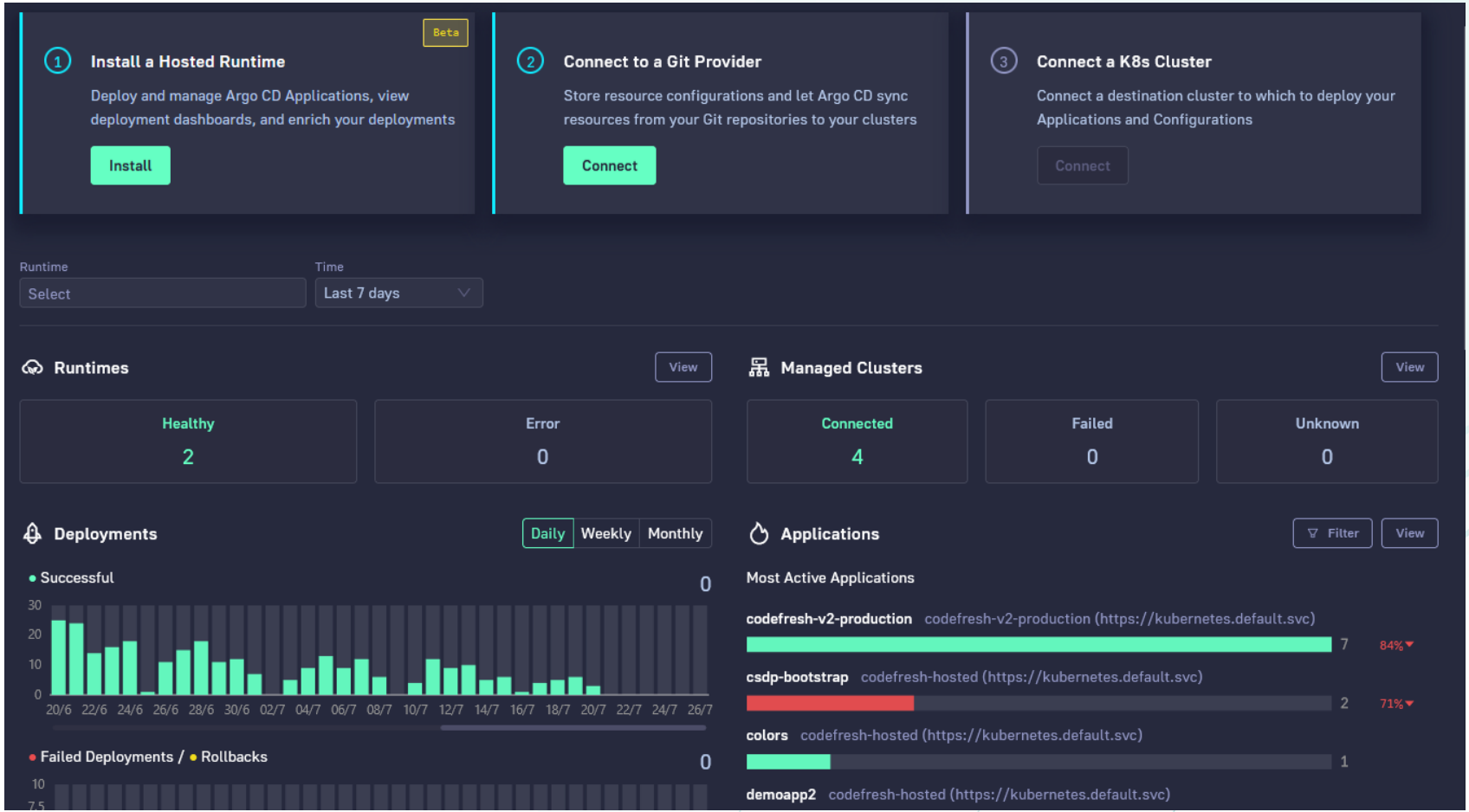

codefresh


Hosted ArgoCD creation flow
Host cluster
ArgoCD
Sync
Crossplane
Provider
Helm
Composition
XHostedRuntime
apiVersion: apiextensions.crossplane.io/v1
kind: Composition
metadata:
name: XHostedRuntime
vcluster
Helm release
Kubeconfig
ProviderConfig
(Helm)
ArgoCD
Codefresh
Platform

codefresh

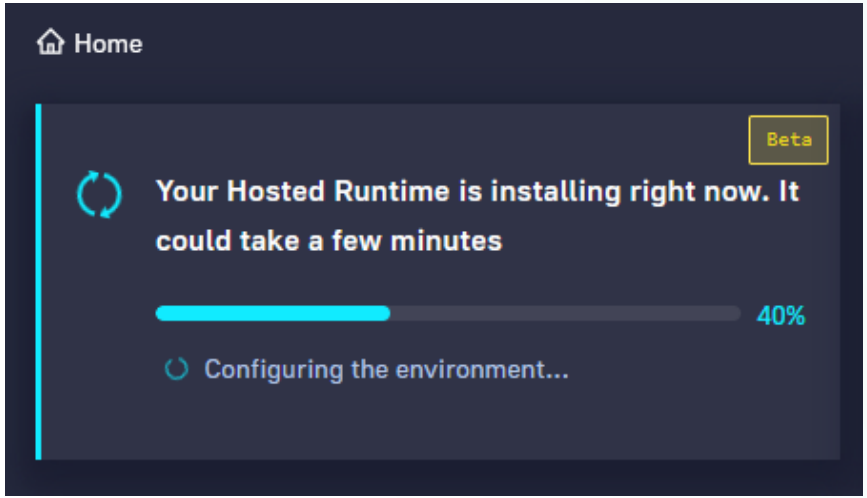
How it looks

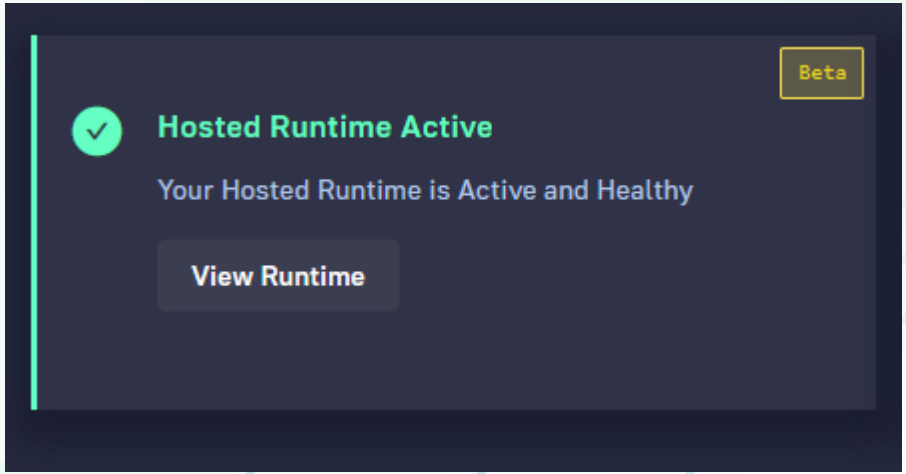
# End user experience


Beta
1
Install a Hosted Runtime
Deploy and manage Argo CD Applications, view deployment dashboards, and enrich your deployments
Install
2
Connect to a Git Provider
Store resource configurations and let Argo CD sync resources from your Git repositories to your clusters
Connect
3
Connect a K8s Cluster
Connect a destination cluster to which to deploy your Applications and Configurations
Connect
Runtime
Select
Time
Last 7 days
Runtimes
View
Healthy
2
Error
0
Managed Clusters
View
Connected
4
Failed
0
Unknown
0
Deployments
Daily
Weekly
Monthly
Successful
0
30
20
10
0
20/6 22/6 24/6 26/6 28/6 30/6 02/7 04/7 06/7 08/7 10/7 12/7 14/7 16/7 18/7 20/7 22/7 24/7 26/7
Failed Deployments / Rollbacks
0
10
7.5
Applications
Filter
View
Most Active Applications
codefresh-v2-production codefresh-v2-production (https://kubernetes.default.svc)
7
84%
csdp-bootstrap codefresh-hosted (https://kubernetes.default.svc)
2
71%
colors codefresh-hosted (https://kubernetes.default.svc)
1
demoapp2 codefresh-hosted (https://kubernetes.default.svc)

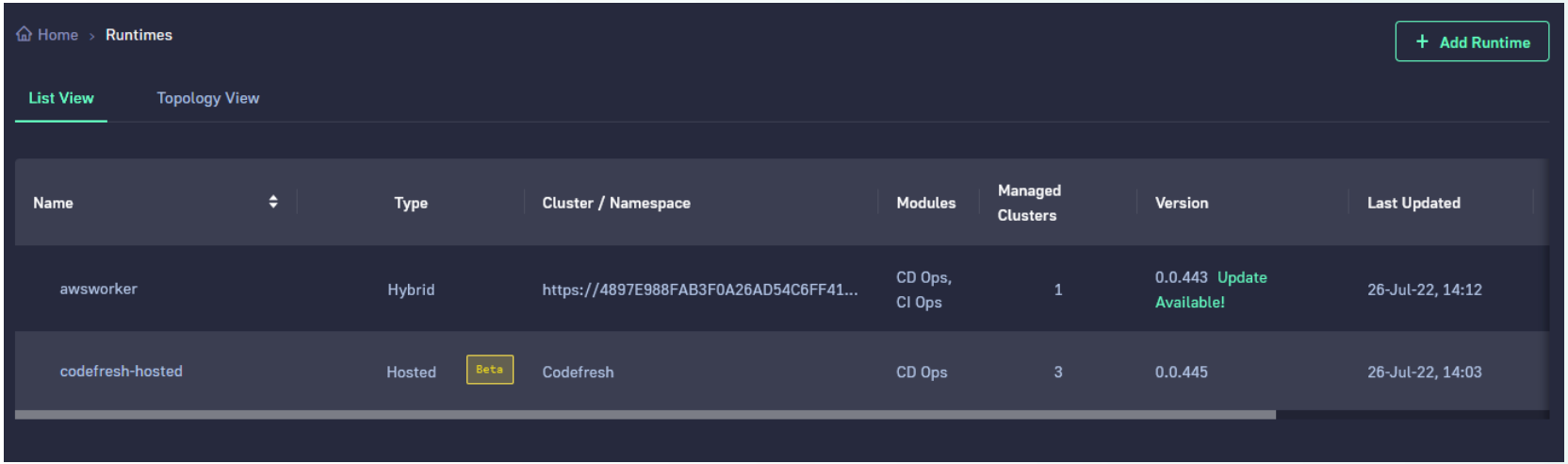
codefresh

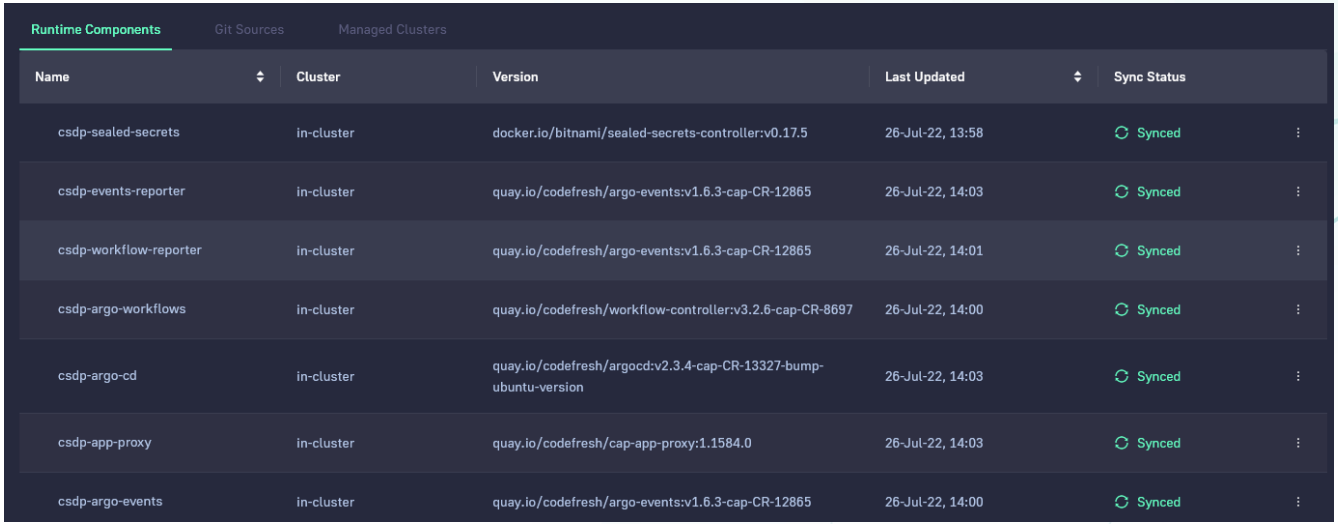
# GUI abstracts everything



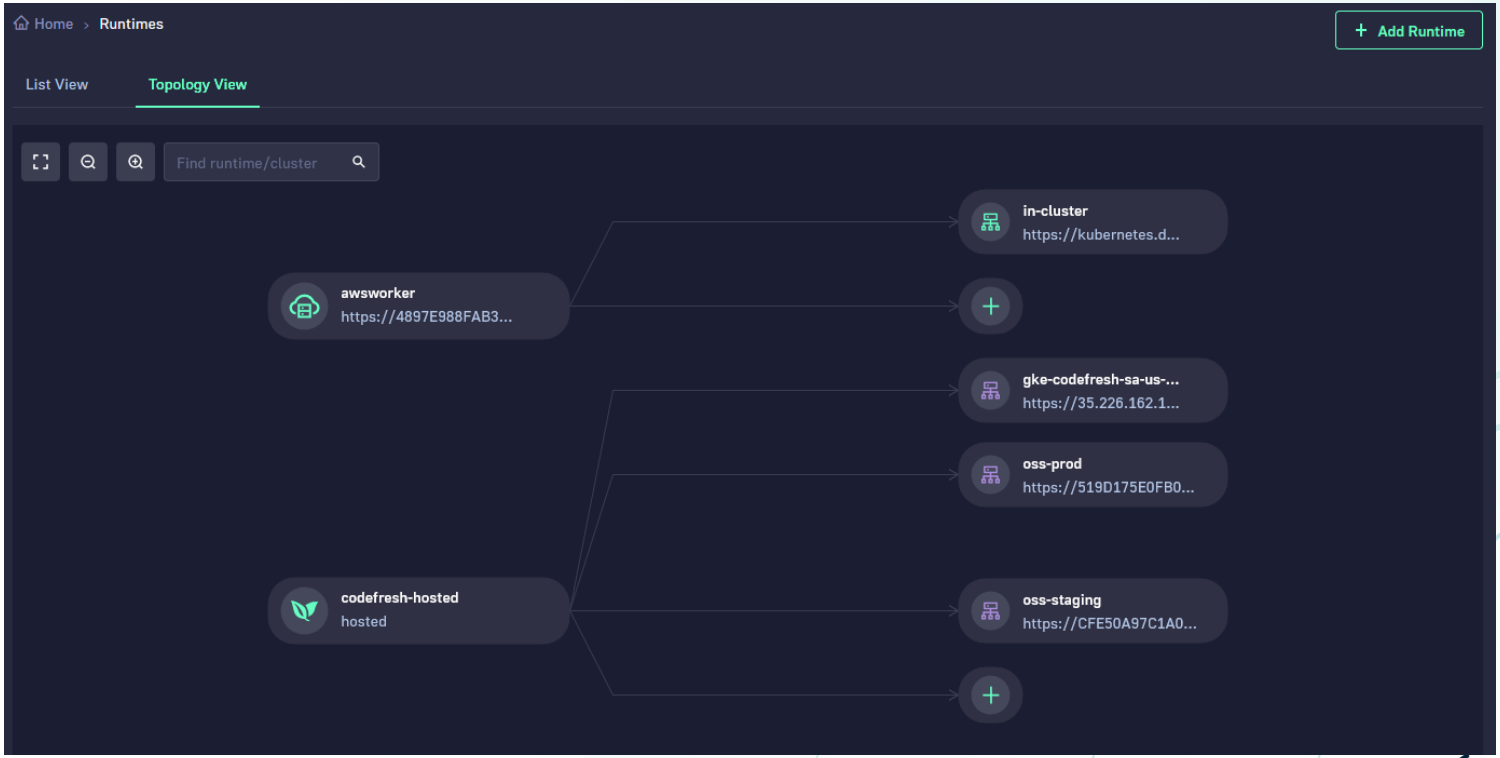

codefresh


Home
Runtimes
Add Runtime
List View
Topology View
Name
Type
Cluster / Namespace
Modules
Managed Clusters
Version
Last Updated
awsworker
Hybrid
https://4897E988FAB3F0A26AD54C6FF41...
CD Ops, CI Ops
1
0.0.443 Update Available!
26-Jul-22, 14:12
codefresh-hosted
Hosted
Beta
Codefresh
CD Ops
3
0.0.445
26-Jul-22, 14:03


Runtime Components
Git Sources
Managed Clusters
Name
Cluster
Version
Last Updated
Sync Status
csdp-sealed-secrets
in-cluster
docker.io/bitnami/sealed-secrets-controller:v0.17.5
26-Jul-22, 13:58
Synced
csdp-events-reporter
in-cluster
quay.io/codefresh/argo-events:v1.6.3-cap-CR-12865
26-Jul-22, 14:03
Synced
csdp-workflow-reporter
in-cluster
quay.io/codefresh/argo-events:v1.6.3-cap-CR-12865
26-Jul-22, 14:01
Synced
csdp-argo-workflows
in-cluster
quay.io/codefresh/workflow-controller:v3.2.6-cap-CR-8697
26-Jul-22, 14:00
Synced
csdp-argo-cd
in-cluster
quay.io/codefresh/argocd:v2.3.4-cap-CR-13327-bump-ubuntu-version
26-Jul-22, 14:03
Synced
csdp-app-proxy
in-cluster
quay.io/codefresh/cap-app-proxy:1.1584.0
26-Jul-22, 14:03
Synced
csdp-argo-events
in-cluster
quay.io/codefresh/argo-events:v1.6.3-cap-CR-12865
26-Jul-22, 14:00
Synced

codefresh

# Connect Deployment clusters to ArgoCD/vcluster




codefresh

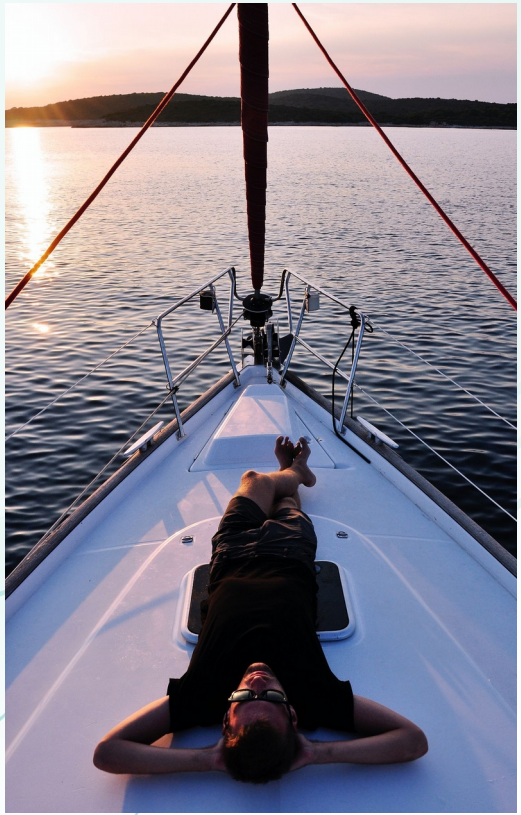
Is it worth it?

# Benefits

# Benefits for users

- Your own Argo CD instance on the cloud
- One-click installation
- (Almost) Instant setup
- Zero configuration, zero maintenance
- Flexibility on K8s/Argo CD version
- Friendly management UI with optional SSO

Photo by Alex Block on Unsplash



codefresh

# Benefits for Codefresh

- Centralized setup/monitoring
- Security isolation
- Cost effective/easy to scale
- Resource sharing
- Allow different combinations of K8s version/Argo CD version

Photo by Scott Blake on Unsplash

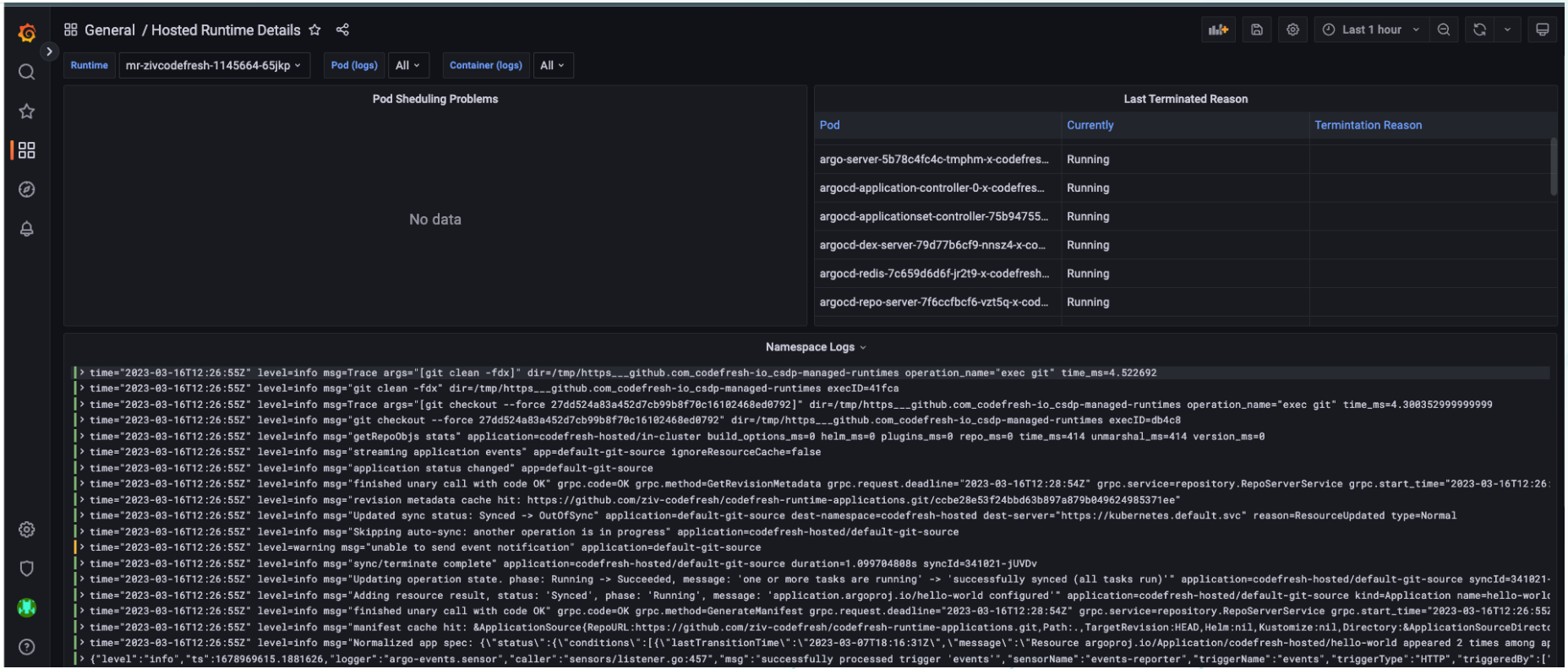
codefresh

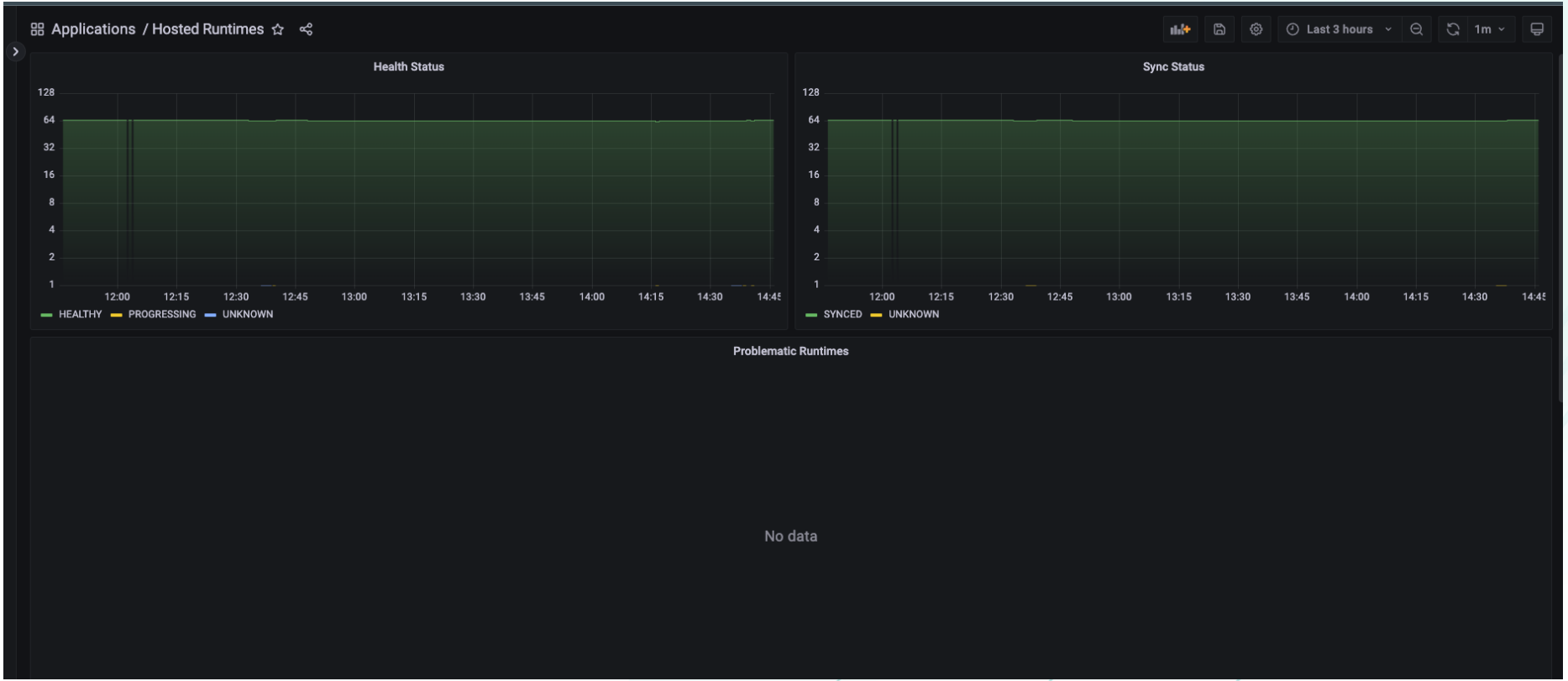
# Monitoring

- Since pods provisioned by workloads deployed on the vcluster are available on the host cluster API - we can use the same tools we use to monitor all other Kubernetes workloads.

- In addition we built our own Prometheus exporter to monitor the hosted runtime health from the platform side

codefresh


General / Hosted Runtime Details
Last 1 hour
Runtime
mr-zivcodefresh-1145664-65jkp
Pod (logs)
All
Container (logs)
All
Pod Sheduling Problems
No data
Last Terminated Reason
Pod
Currently
Termintation Reason
argo-server-5b78c4fc4c-tmphm-x-codefres...
Running
argocd-application-controller-0-x-codefres...
Running
argocd-applicationset-controller-75b94755...
Running
argocd-dex-server-79d77b6cf9-nnsz4-x-co...
Running
argocd-redis-7c659d6d6f-jr2t9-x-codefresh...
Running
argocd-repo-server-7f6ccfbcf6-vzt5q-x-cod...
Running
Namespace Logs
time="2023-03-16T12:26:55Z" level=info msg=Trace args="[git clean -fdx]" dir=/tmp/https___github.com_codefresh-io_csdp-managed-runtimes operation_name="exec git" time_ms=4.522692
time="2023-03-16T12:26:55Z" level=info msg="git clean -fdx" dir=/tmp/https___github.com_codefresh-io_csdp-managed-runtimes execID=41fca
time="2023-03-16T12:26:55Z" level=info msg=Trace args="[git checkout --force 27dd524a83a452d7cb99b8f70c16102468ed0792]" dir=/tmp/https___github.com_codefresh-io_csdp-managed-runtimes operation_name="exec git" time_ms=4.300352999999999
time="2023-03-16T12:26:55Z" level=info msg="git checkout --force 27dd524a83a452d7cb99b8f70c16102468ed0792" dir=/tmp/https___github.com_codefresh-io_csdp-managed-runtimes execID=db4c8
time="2023-03-16T12:26:55Z" level=info msg="getRepoObjs stats" application=codefresh-hosted/in-cluster build_options_ms=0 helm_ms=0 plugins_ms=0 repo_ms=0 time_ms=414 unmarshal_ms=414 version_ms=0
time="2023-03-16T12:26:55Z" level=info msg="streaming application events" app=default-git-source ignoreResourceCache=false
time="2023-03-16T12:26:55Z" level=info msg="application status changed" app=default-git-source
time="2023-03-16T12:26:55Z" level=info msg="finished unary call with code OK" grpc.code=OK grpc.method=GetRevisionMetadata grpc.request.deadline="2023-03-16T12:28:54Z" grpc.service=repository.RepoServerService grpc.start_time="2023-03-16T12:26
time="2023-03-16T12:26:55Z" level=info msg="revision metadata cache hit: https://github.com/ziv-codefresh/codefresh-runtime-applications.git/ccbe28e53f24bbd63b897a879b049624985371ee"
time="2023-03-16T12:26:55Z" level=info msg="Updated sync status: Synced -> OutOfSync" application=default-git-source dest-namespace=codefresh-hosted dest-server="https://kubernetes.default.svc" reason=ResourceUpdated type=Normal
time="2023-03-16T12:26:55Z" level=info msg="Skipping auto-sync: another operation is in progress" application=codefresh-hosted/default-git-source
time="2023-03-16T12:26:55Z" level=warning msg="unable to send event notification" application=default-git-source
time="2023-03-16T12:26:55Z" level=info msg="sync/terminate complete" application=codefresh-hosted/default-git-source duration=1.099704808s syncId=341021-jUVDv
time="2023-03-16T12:26:55Z" level=info msg="Updating operation state. phase: Running -> Succeeded, message: 'one or more tasks are running' -> 'successfully synced (all tasks run)'" application=codefresh-hosted/default-git-source syncId=341021-
time="2023-03-16T12:26:55Z" level=info msg="Adding resource result, status: 'Synced', phase: 'Running', message: 'application.argoproj.io/hello-world configured'" application=codefresh-hosted/default-git-source kind=Application name=hello-world
time="2023-03-16T12:26:55Z" level=info msg="finished unary call with code OK" grpc.code=OK grpc.method=GenerateManifest grpc.request.deadline="2023-03-16T12:28:54Z" grpc.service=repository.RepoServerService grpc.start_time="2023-03-16T12:26:55Z
time="2023-03-16T12:26:55Z" level=info msg="manifest cache hit: &ApplicationSource{RepoURL:https://github.com/ziv-codefresh/codefresh-runtime-applications.git,Path:.,TargetRevision:HEAD,Helm:nil,Kustomize:nil,Directory:&ApplicationSourceDirecto
time="2023-03-16T12:26:55Z" level=info msg="Normalized app spec: {\"status\":{\"conditions\":[{\"lastTransitionTime\":\"2023-03-07T18:16:31Z\",\"message\":\"Resource argoproj.io/Application/codefresh-hosted/hello-world appeared 2 times among ap
{"level":"info","ts":1678969615.1881626,"logger":"argo-events.sensor","caller":"sensors/listener.go:457","msg":"successfully processed trigger 'events'","sensorName":"events-reporter","triggerName":"events","triggerType":"HTTP","triggeredBy":[
Grafana loki
codefresh



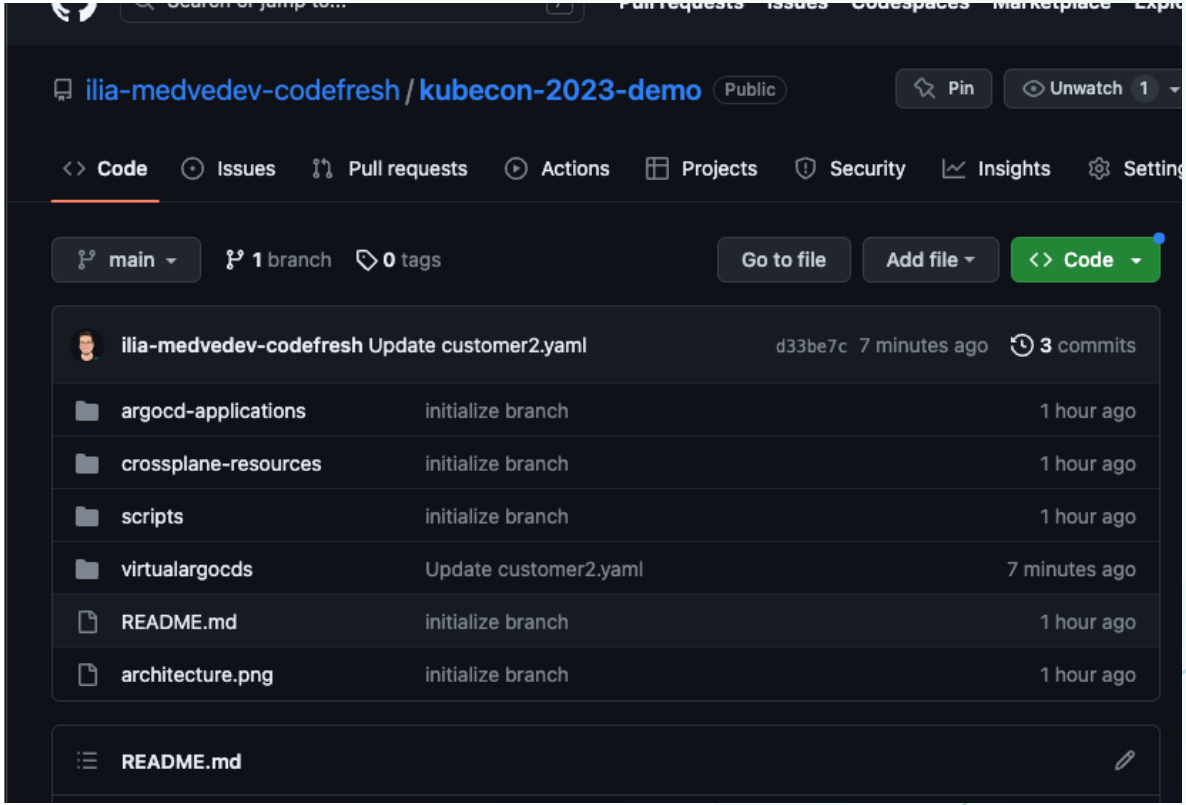

Platform custom exporter to aggregate data on all hosted runtimes health

codefresh

See it in action

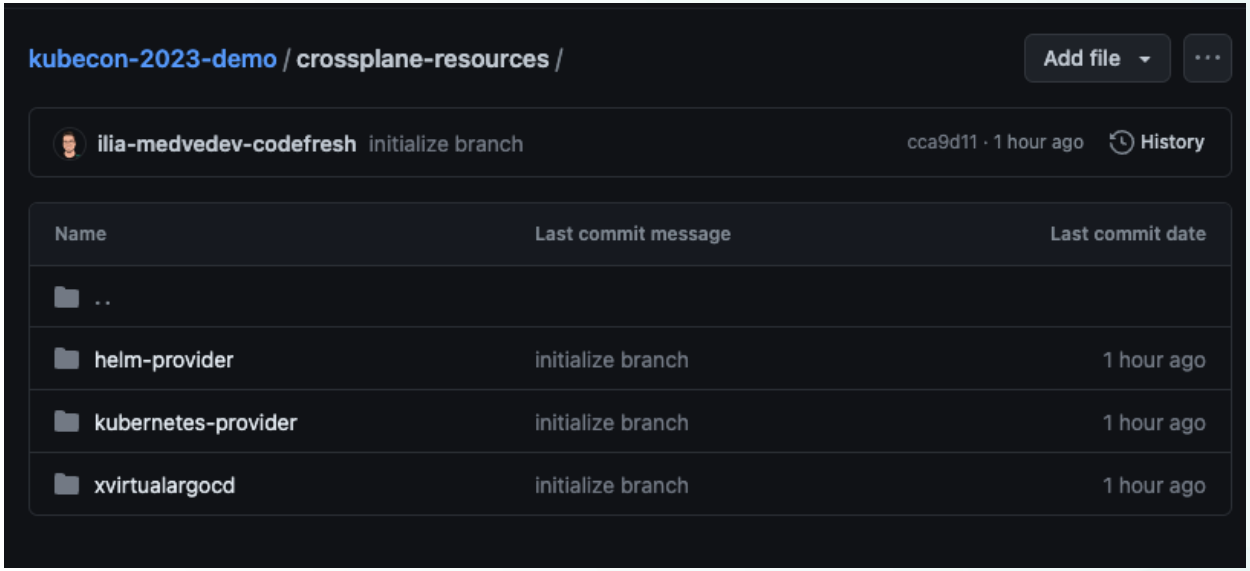
# Demo time!

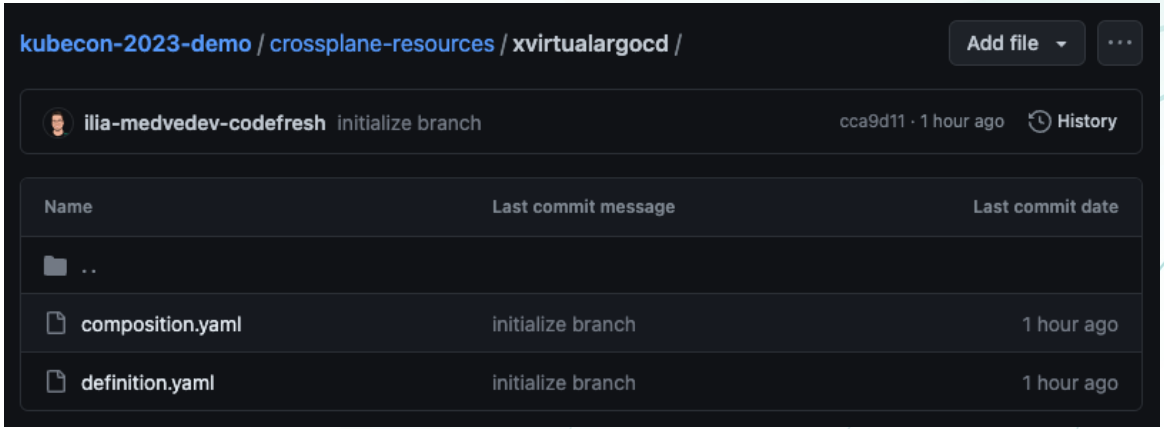
Provisioning and de-provisioning of hosted ArgoCD




https://github.com/codefresh-contrib/kubecon-eu-2023-demo-crossplane-vcluster.git

codefresh


kubecon-2023-demo / crossplane-resources /
Add file
ilia-medvedev-codefresh initialize branch
cca9d11 · 1 hour ago
History
Name
Last commit message
Last commit date
..
helm-provider
initialize branch
1 hour ago
kubernetes-provider
initialize branch
1 hour ago
xvirtualargocd
initialize branch
1 hour ago


kubecon-2023-demo / crossplane-resources / xvirtualargocd /
Add file
ilia-medvedev-codefresh initialize branch
cca9d11 · 1 hour ago
History
Name
Last commit message
Last commit date
..
composition.yaml
initialize branch
1 hour ago
definition.yaml
initialize branch
1 hour ago

codefresh

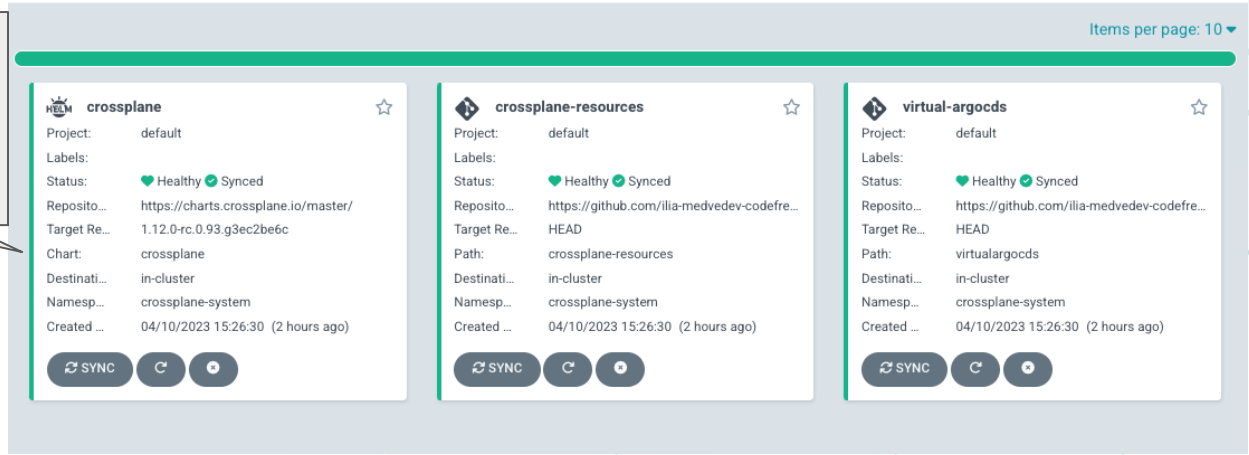
kubecon-2023-demo / argocd-applications /

Add file ···

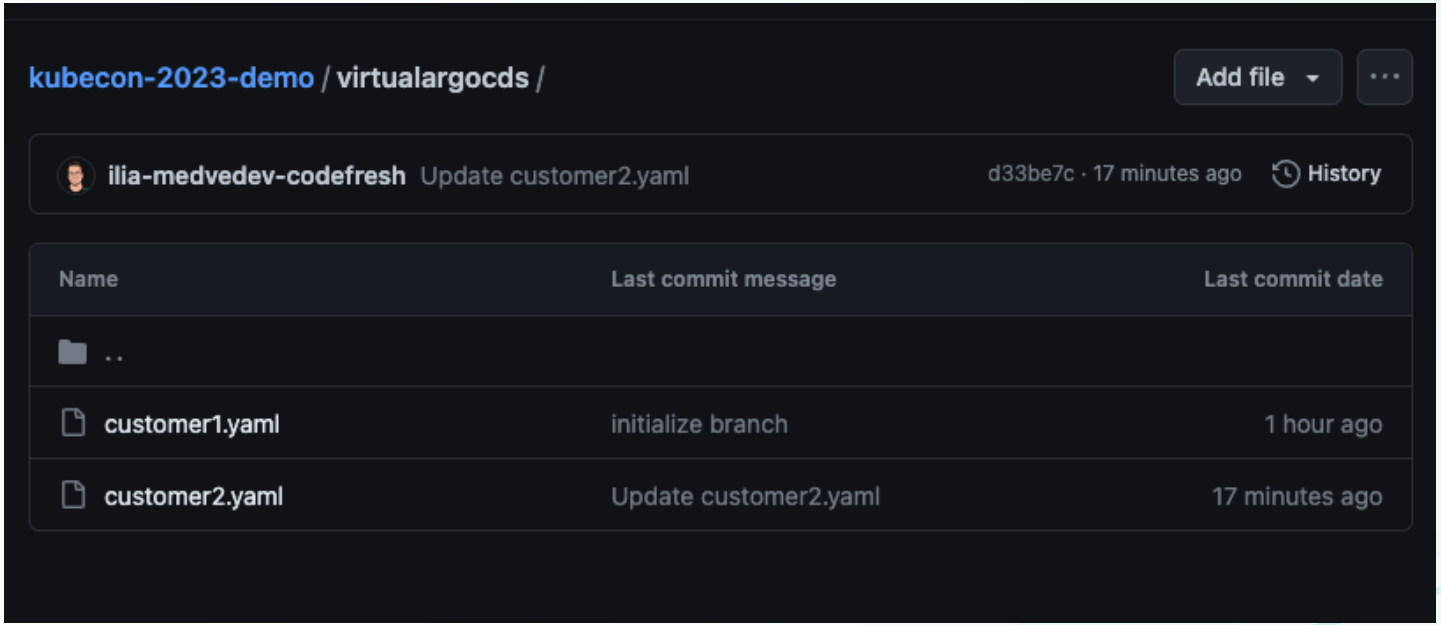
ilia-medvedev-codefresh initialize branch cca9d11 · 1 hour ago History

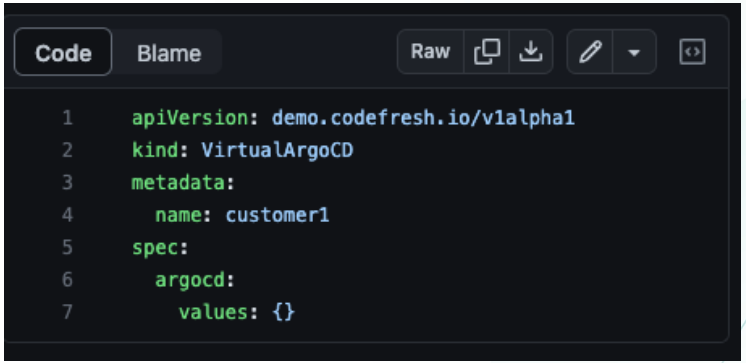
| Name | Last commit message | Last commit date |
|---|---|---|
| .. | | |
| crosspalne-resources.yaml | initialize branch | 1 hour ago |
| crossplane.yaml | initialize branch | 1 hour ago |
| virtual-argocds.yaml | initialize branch | 1 hour ago |

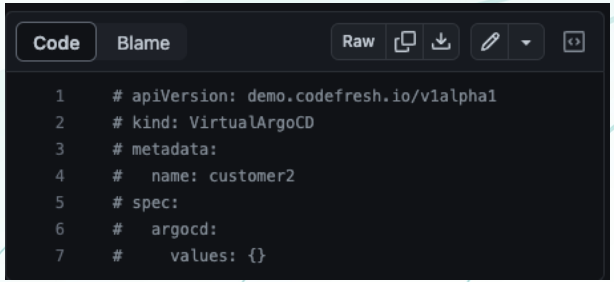
Crossplane official Helm chart



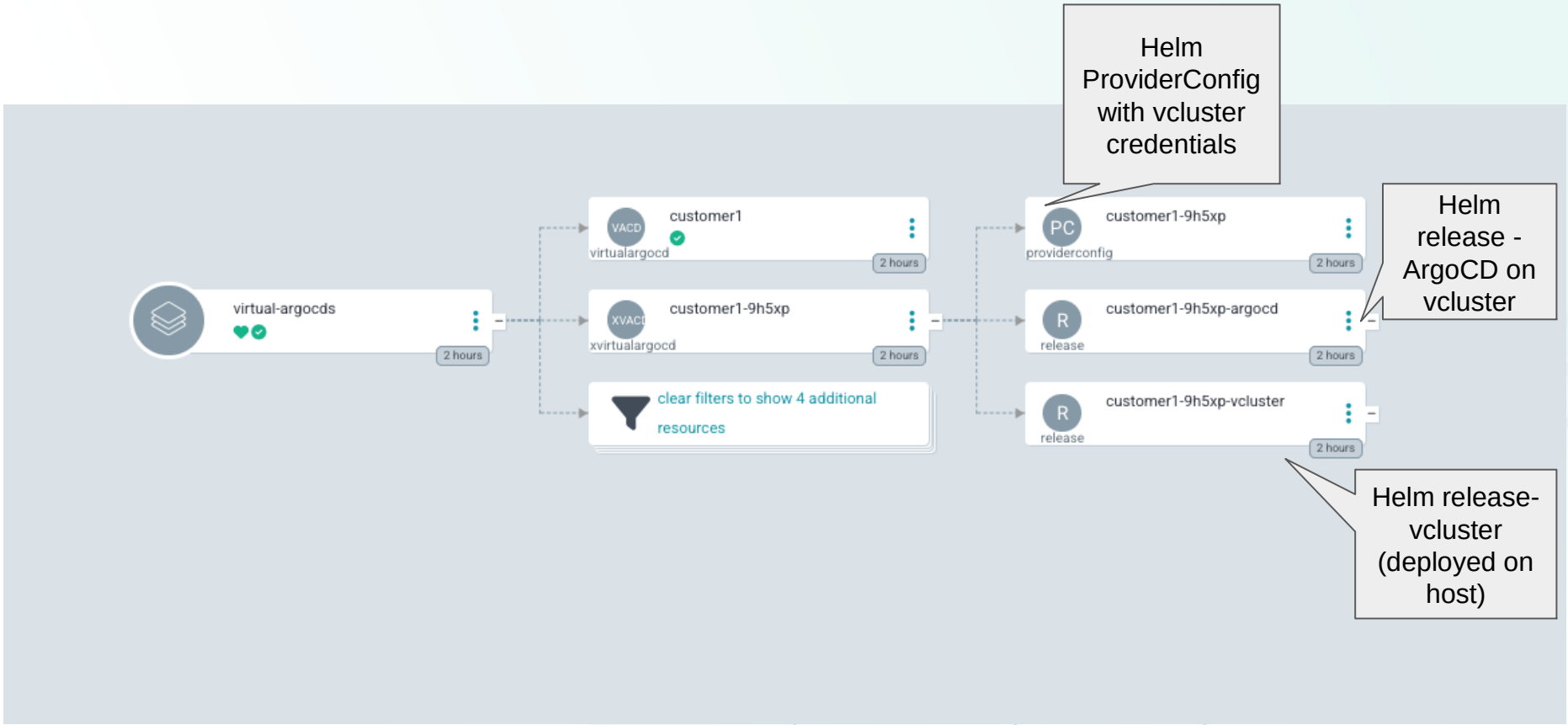

codefresh


kubecon-2023-demo / virtualargocds /
Add file
ilia-medvedev-codefresh Update customer2.yaml
d33be7c · 17 minutes ago
History
Name
Last commit message
Last commit date
..
customer1.yaml
initialize branch
1 hour ago
customer2.yaml
Update customer2.yaml
17 minutes ago


Code
Blame
Raw
apiVersion: demo.codefresh.io/v1alpha1
kind: VirtualArgoCD
metadata:
name: customer1
spec:
argocd:
values: {}


Code
Blame
Raw
# apiVersion: demo.codefresh.io/v1alpha1
# kind: VirtualArgoCD
# metadata:
# name: customer2
# spec:
# argocd:
# values: {}

codefresh


Helm ProviderConfig with vcluster credentials
Helm release - ArgoCD on vcluster
Helm release-vcluster (deployed on host)
virtual-argocds
2 hours
VACD
customer1
virtualargocd
2 hours
XVACD
customer1-9h5xp
xvirtualargocd
2 hours
clear filters to show 4 additional resources
PC
customer1-9h5xp
providerconfig
2 hours
R
customer1-9h5xp-argocd
release
2 hours
R
customer1-9h5xp-vcluster
release
2 hours

codefresh

codefresh

## Resources

- vcluster.com (also loft.sh)
- crossplane.io (also upbound.io)
- codefresh.io
- learning.codefresh.io (Argo CD certification)

# Questions?

Scan QR and give us feedback please!