August 12 - 2026

Agenda

What happened? 👀

What's new in **v0.13.0**?

Future 🔮

What's planned in the near future?
Which Events will we attend.

Q&A 🗣️

We want to hear from you!
Do you have a story to tell?
Do you have questions?

Whats new?

New component

- Reloader

Catalogs

- Catalog Folder Structure
- Migration

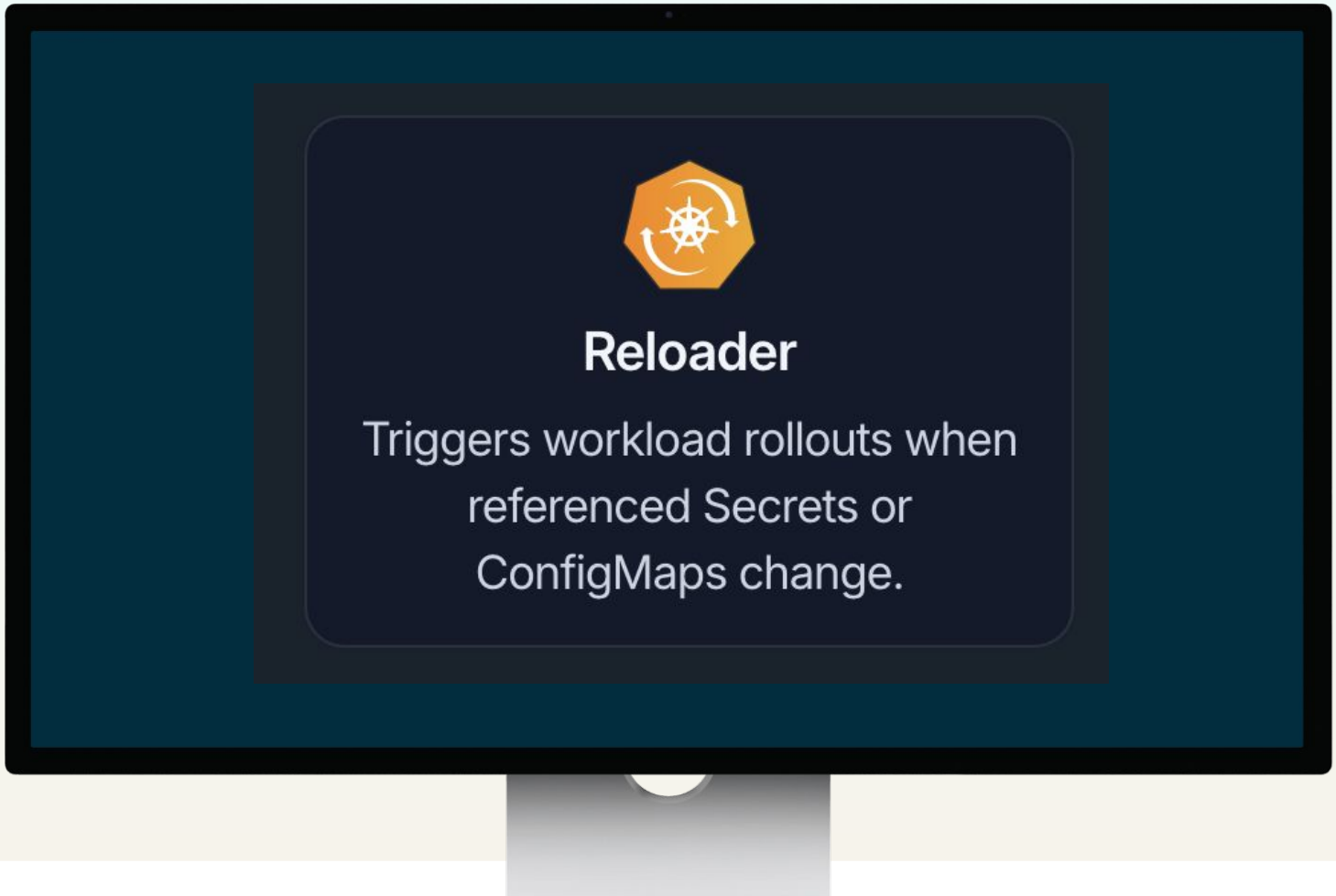
New Guides

- **HowTo:**
  Setup kubara on Google Cloud (GCP)

- **HowTo:**
  Scale and right-size

AI

- **Leverage kubara with AI**
  (Blog by Matthias Bregenzer)


Reloader
Triggers workload rollouts when referenced Secrets or ConfigMaps change.

Migration

This release introduces a refactoring of the directories of your GitOps repositories and catalogs to a more clear naming scheme and less nesting.

Your GitOps repository gets automatically migrated by running `kubara generate` once with the new version. Please be aware that you have to run `kubara bootstrap` once for each of your clusters with this new release. Otherwise the new templating will not be effective for Argo CD.

GitOps repo structure will be changed automatically by `kubara generate`:

```
# rename
managed-service-catalog/ -> platform-components/

# order from <tooling>/<cluster> to <cluster>/<tooling>
customer-service-catalog/{helm,terraform,...}/{cluster-1,cluster-xxx} -> platform-configs/{cluster-1,clus

# rename
additional-values.yaml -> values-additional.yaml
```

Catalogs ARE NOT automatically migrated to the new structure:

Catalog structure changes:

```
managed-service-catalog/helm/ -> platform-components/helm/
managed-service-catalog/terraform/providers/{stackit,xxx} -> platform-components/terraform/{stackit,xxx
customer-service-catalog/{helm,terraform,...}/example -> platform-configs/{helm,terraform,...}
```

Catalog

Catalog Folder Structure

```
my-catalog/
├── catalog.yaml              # optional: name, version, description
├── services/                 # ServiceDefinition files
│   ├── argocd.yaml
│   └── cert-manager.yaml
├── platform-components/      # previously: managed-service-catalog
│   └── helm/
│       ├── argocd/
│       └── cert-manager/
└── platform-configs/           # previously:  customer-service-catalog
    └── cluster/                # previously:  helm/cluster
        └── helm/               # now:         cluster/helm
            └── argocd/
            └── cert-manager/
```

ServiceDefinition Example

```yaml
apiVersion: kubara.io/v1alpha1
kind: ServiceDefinition
metadata
  name: argocd
  annotations:
    kubara.io/category: app-management
spec:
  chartPath: argo-cd
  status: enabled
  clusterTypes:
    - hub
```

Migration

Catalogs ARE NOT automatically migrated to the new structure:

Catalog structure changes:

```
managed-service-catalog/helm/ -> platform-components/helm/
managed-service-catalog/terraform/providers/{stackit,xxx} -> platform-components/terraform/{stackit,xxx}
customer-service-catalog/{helm,terraform,...}/example -> platform-configs/{helm,terraform,...}
```

Therefore we provide the following migration script for your existing catalogs:

See script in the Release Notes of v0.12.0
https://github.com/kubara-io/kubara/releases/tag/v0.12.0

https://docs.kubara.io/latest-stable/3_infrastructure/google-cloud/

Kubara x Argo CD Sizing & Limits

All object counts = live objects in the Argo CD controller cache · tested with applications distributed per cluster

S · Starter

Start small, cost efficiency first, no HA needed

~6k

controller cache objects — limit of the default configuration

Tested configuration

Clusters	~50
Apps per cluster	6
Applications total	~300
Argo CD HA	No

Required action

None — covered by the default kubara catalog.

M · Standard

Growing apps & clusters, HA not yet required

~10k

controller cache objects — tested configuration

Tested configuration

Clusters	~100
Apps per cluster	6
Applications total	~600
Argo CD HA	No

Required action

Increase limits, requests & storage.

L · Scale

Many teams, clusters & repos — HA and high performance

~16k

controller cache objects — tested configuration

Tested configuration

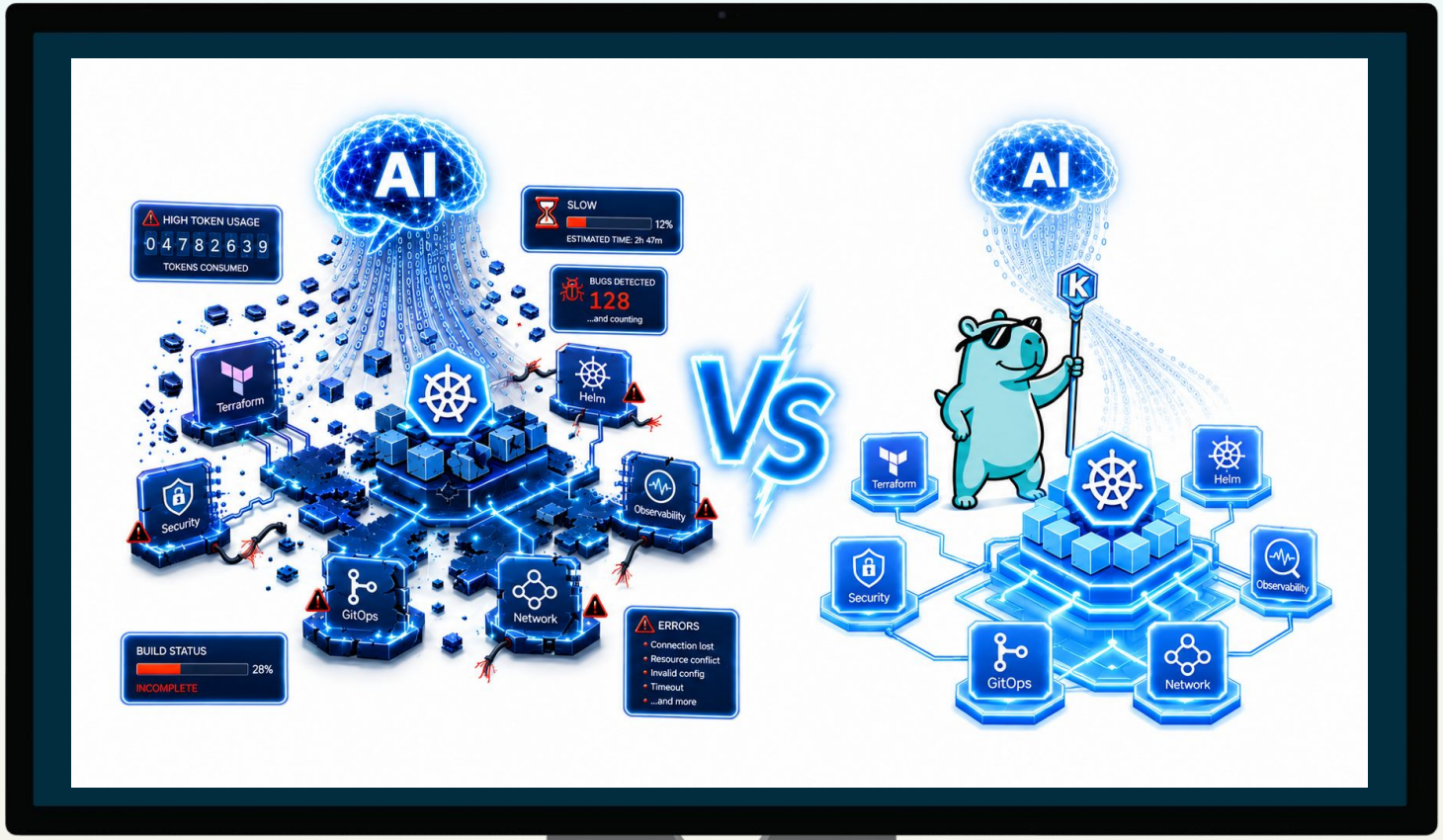
Clusters	~150
Apps per cluster	6
Applications total	~900
Argo CD HA	Yes — required

Required action

Enable Argo CD HA + increase limits, requests & storage.

How to read the object counts: How to read the object counts: 1 application ≈ 10–12 desired objects from Git. The controller cache holds more: live objects (Pods, ReplicaSets) plus the baseline resources of every targeted cluster — roughly 1.5–2× the desired count, i.e. ~17–20 cached objects per application. Heavier applications reach the object thresholds at proportionally lower application counts, which is why the cached-object count, not the application count, is the metric to watch. In tier L, object counts are totals across all controller shards; memory values are per replica.

See:
https://docs.kubara.io/latest-stable/2_concepts/scale_and_ha/


AI
HIGH TOKEN USAGE
0 4 7 8 2 6 3 9
TOKENS CONSUMED
SLOW
12%
ESTIMATED TIME: 2h 47m
BUGS DETECTED
128
...and counting
Terraform
Helm
Security
Observability
GitOps
Network
BUILD STATUS
28%
INCOMPLETE
ERRORS
Connection lost
Resource conflict
Invalid config
Timeout
...and more
Vs
AI
Terraform
Helm
Security
Observability
GitOps
Network

Future

TUI

Interactive TUI for setting up and upgrading kubara

Agnostic GitOps Support

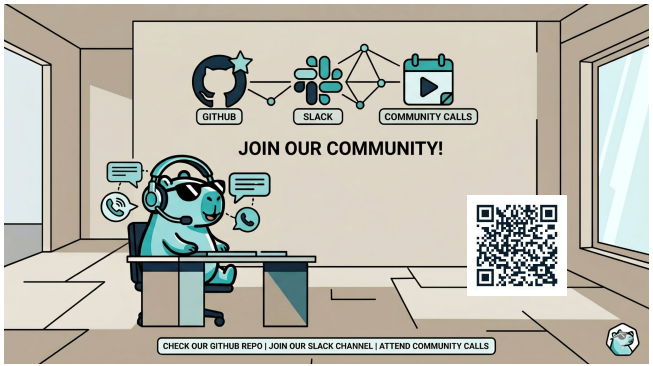
Abstract away ArgoCD in kubara to make space for support of other tools like:

- FluxCD
- Sveltos

Stay tuned!

... and a lot more to come!

Further Infos



#kubara Slack Channel in official K8s Slack Space

Next Community Call

Sept 9, 2pm CET

Next Event

See next slide

Meet us!

ContainerDays
Hamburg 🇩🇪 - 2nd-4th September

CloudX Summit
(Wiesbaden 🇩🇪 - Registration closed, Invite only)

EdgeCase ‘26
Hilversum 🇳🇱 - 24th September

OpenInfraDays
Saarbrücken 🇩🇪 - 24th - 25th October

Q&A - Questions and Story Time

Lets chat!

Do you have a story to tell?
Do you have questions?

Changelogs

https://github.com/kubara-io/kubara/releases/tag/v0.11.0

https://github.com/kubara-io/kubara/releases/tag/v0.12.0

https://github.com/kubara-io/kubara/releases/tag/v0.13.0