



# KUBERNETES ON OPENSTACK

Cluster creation, management and service deployment

12.06.2023 | TIM KREUZER

# WHAT TO EXPECT

## Why you're here and what you will learn

- Kubernetes in 3 minutes
- Create a kubernetes cluster on OpenStack
- Manage your cluster – Browser and CLI
- Deploy services – the easy way
- Deploy services – the right way

# KUBERNETES

## In (nearly) 3 minutes

- „Kubernetes, also known as K8s, is an open source system for managing containerized applications across multiple hosts. It provides basic mechanisms for deployment, maintenance, and scaling of applications.”

[github.com/kubernetes/kubernetes](https://github.com/kubernetes/kubernetes)

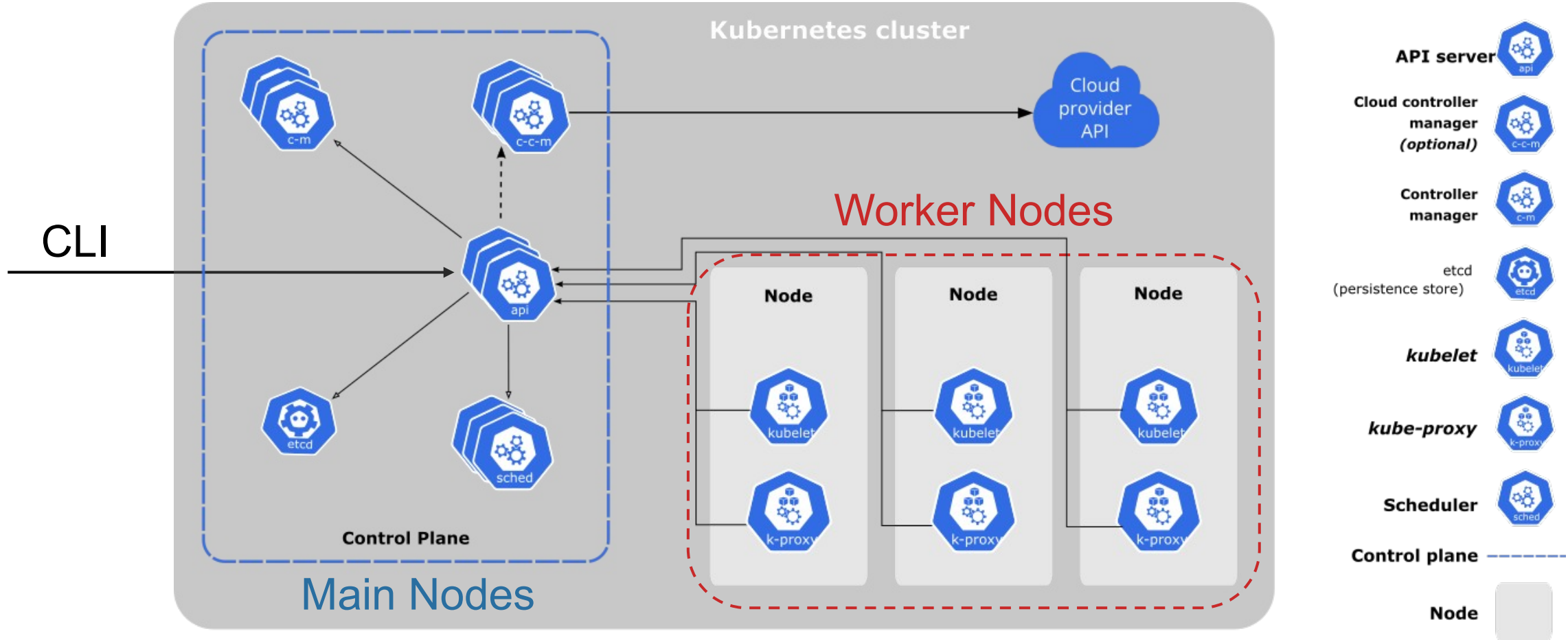
- Features:

- Load Balancing
- Storage orchestration
- Automated rollouts and rollbacks
- Automated bin packing
- Self-healing
- Secret and configuration management

[kubernetes.io/docs/concepts/overview/](https://kubernetes.io/docs/concepts/overview/)

# KUBERNETES

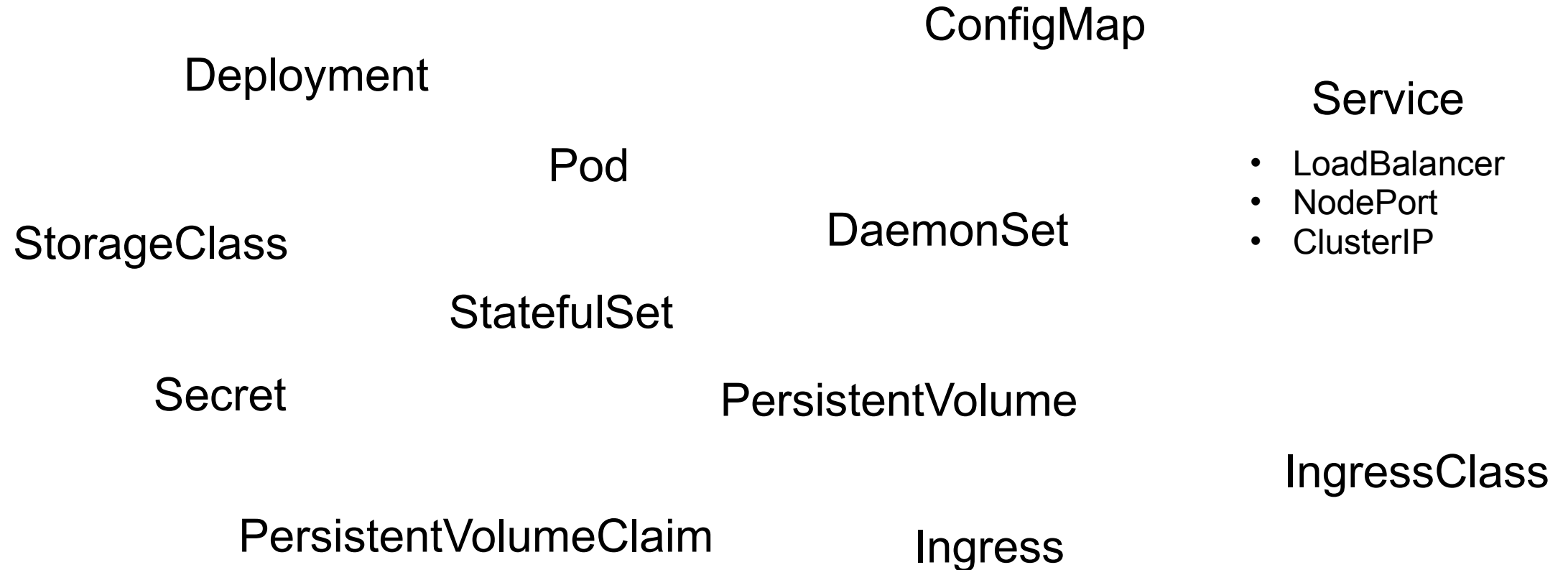
## Architecture



[kubernetes.io/docs/concepts/overview/components/](https://kubernetes.io/docs/concepts/overview/components/)





# KUBERNETES

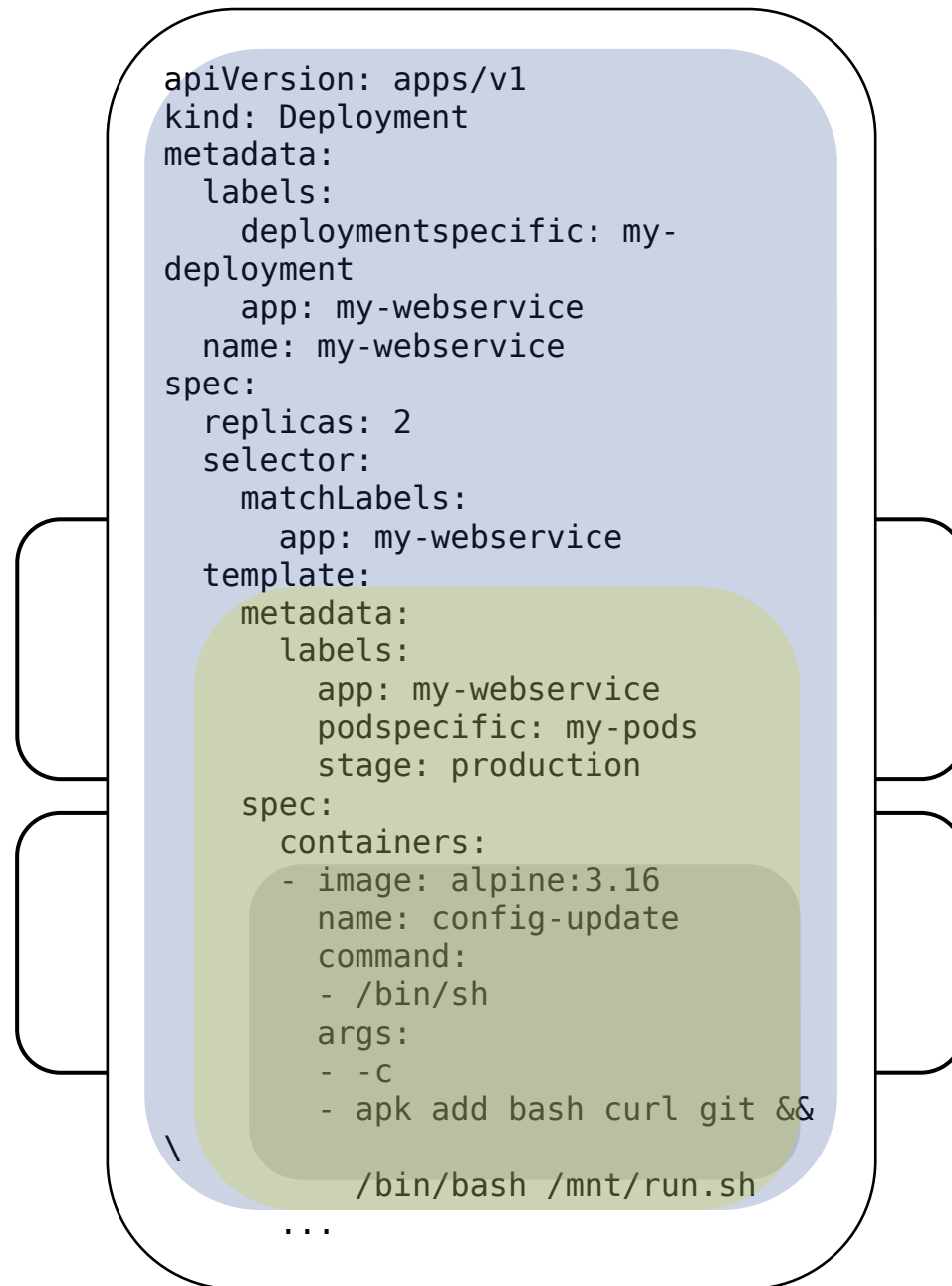
## Resources



# KUBERNETES

## Resources - Deployment





-  Node / VM
-  Deployment
-  Pod
-  Container

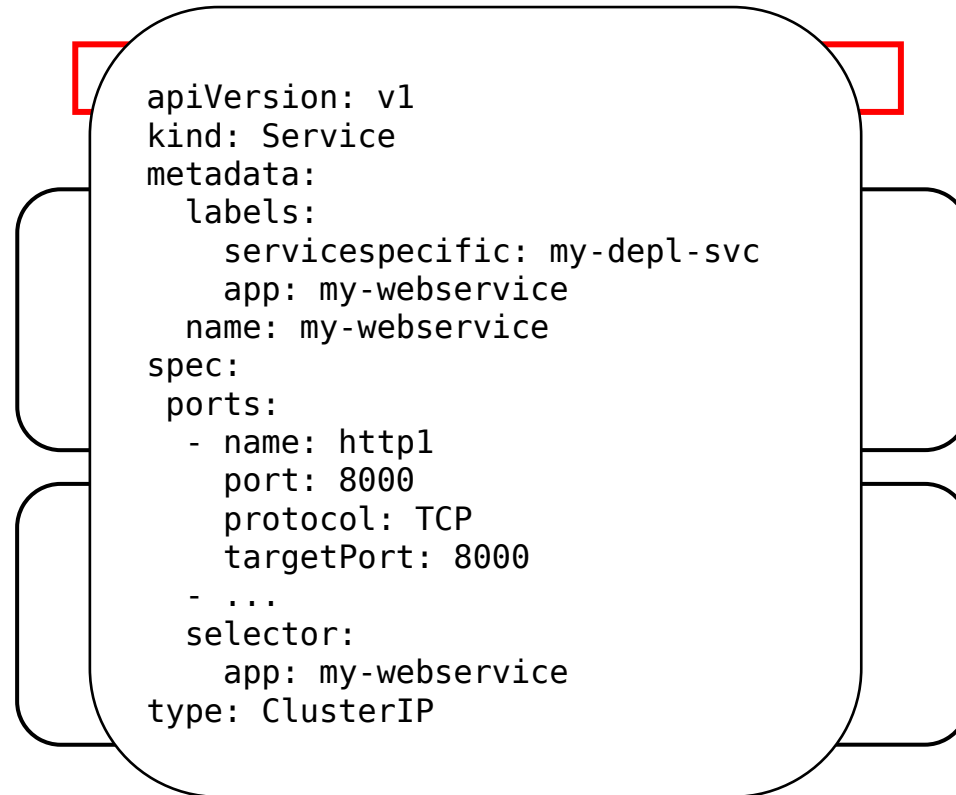


- C1: update config
- C2: webservice
- C3: serve static files

# KUBERNETES

## Resources - Service

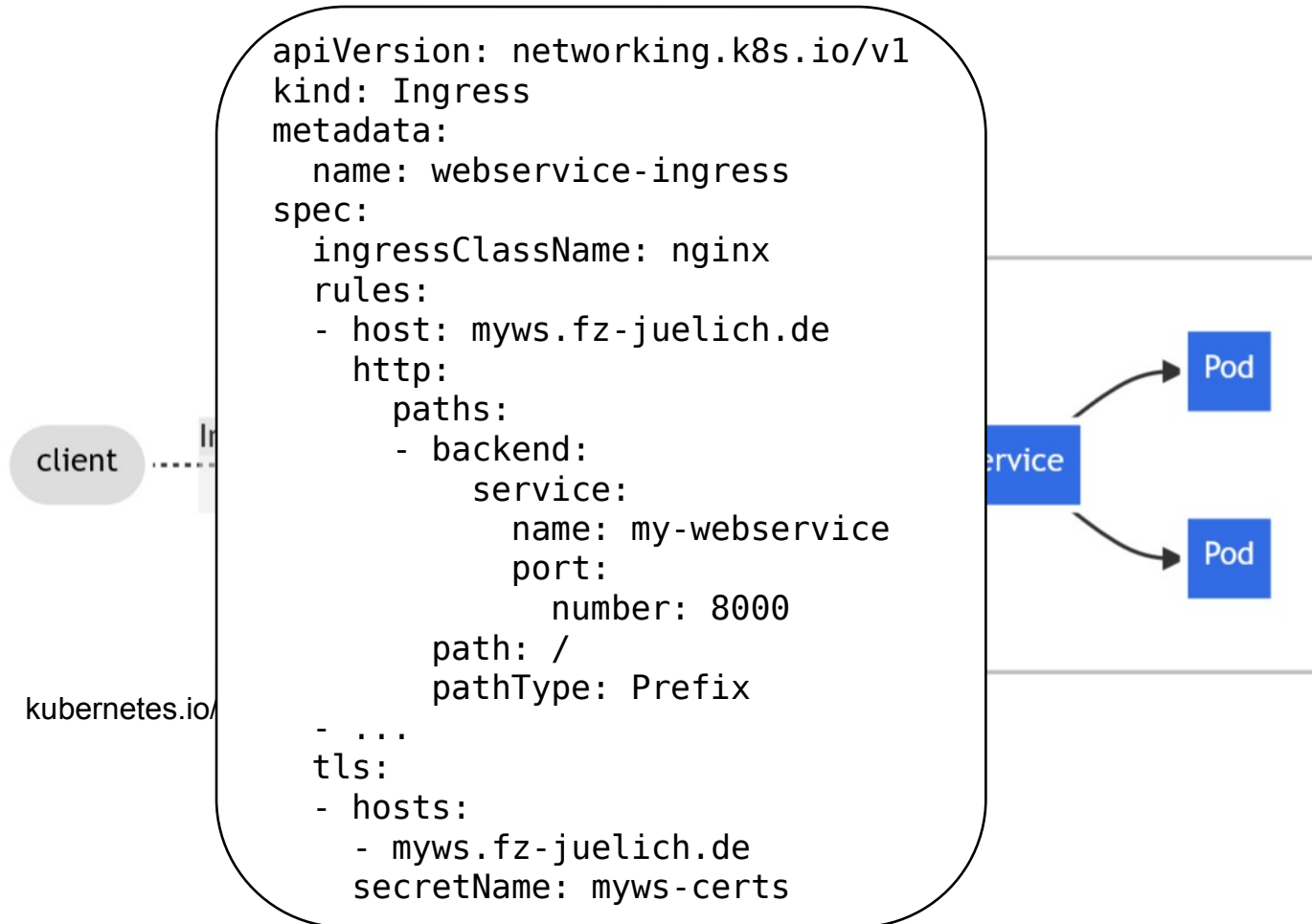
-  Node / VM
-  Deployment
-  Pod
-  Container



 Service

# KUBERNETES

## Resources - Ingress





# WHAT TO EXPECT

## Why you're here and what you will learn

- Kubernetes in 3 minutes
- Create a kubernetes cluster on OpenStack
- Manage your cluster – Browser and CLI
- Deploy services – the easy way
- Deploy services – the right way

# OPENSTACK

Create a K8s cluster



[rancher.com](https://rancher.com)

# RANCHER

## Create a K8s cluster

- Multi-Cluster Management
- „Kubernetes-as-a-service“
- Integrated support for
  - Prometheus
  - Grafana
  - Fleet
- Including OpenStack driver

# RANCHER

## Create a K8s cluster

- OpenStack requirements:
  - Application credentials
  - SSH-Keypair
  - Network / Subnet
  - Security group for subnet
  - One „proxy“ VM

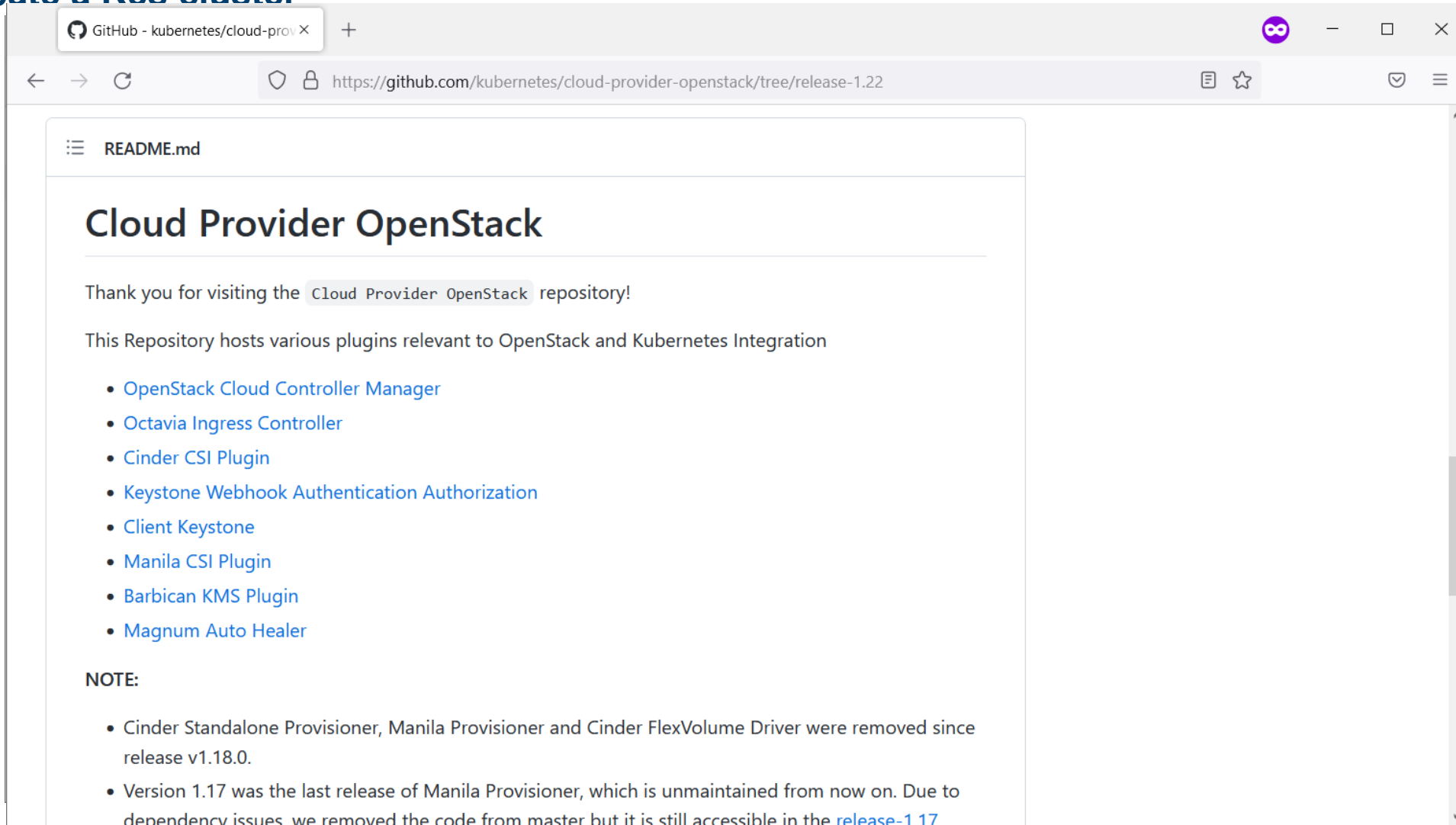
# RANCHER

## Create a K8s cluster

- „Proxy“ VM:
  - Add Floating IP
  - Accessible for administrators ( 134.94.0.0/16 )
  - Install docker
  - `docker run -d --restart=unless-stopped --net=host --privileged rancher/rancher:v2.6`
- Customization:
  - Persistent Rancher storage ( `-v ...:/var/lib/rancher` )
  - Certificate for rancher webservice ( `-v ...:/etc/rancher/ssl/` )
  - Install nfs server for your services

# RANCHER

## Create a K8s cluster



The screenshot shows a web browser window with the address bar displaying `https://github.com/kubernetes/cloud-provider-openstack/tree/release-1.22`. The page content includes a header for 'README.md', a main heading 'Cloud Provider OpenStack', a welcome message, a list of plugins, and a 'NOTE' section.

GitHub - kubernetes/cloud-prov X +

← → ↻ 🔒 `https://github.com/kubernetes/cloud-provider-openstack/tree/release-1.22` 📄 ☆ 🛡️ ☰

☰ README.md

## Cloud Provider OpenStack

Thank you for visiting the `Cloud Provider OpenStack` repository!

This Repository hosts various plugins relevant to OpenStack and Kubernetes Integration

- [OpenStack Cloud Controller Manager](#)
- [Octavia Ingress Controller](#)
- [Cinder CSI Plugin](#)
- [Keystone Webhook Authentication Authorization](#)
- [Client Keystone](#)
- [Manila CSI Plugin](#)
- [Barbican KMS Plugin](#)
- [Magnum Auto Healer](#)

**NOTE:**

- Cinder Standalone Provisioner, Manila Provisioner and Cinder FlexVolume Driver were removed since release v1.18.0.
- Version 1.17 was the last release of Manila Provisioner, which is unmaintained from now on. Due to dependency issues, we removed the code from master but it is still accessible in the [release-1.17](#)

# PLUGINS

## Connect Kubernetes and OpenStack

- Cloud Controller Manager
  - Manage load balancers
  
- Cinder CSI Plugin
  - Use cinder volumes as persistent storage

# WHAT TO EXPECT

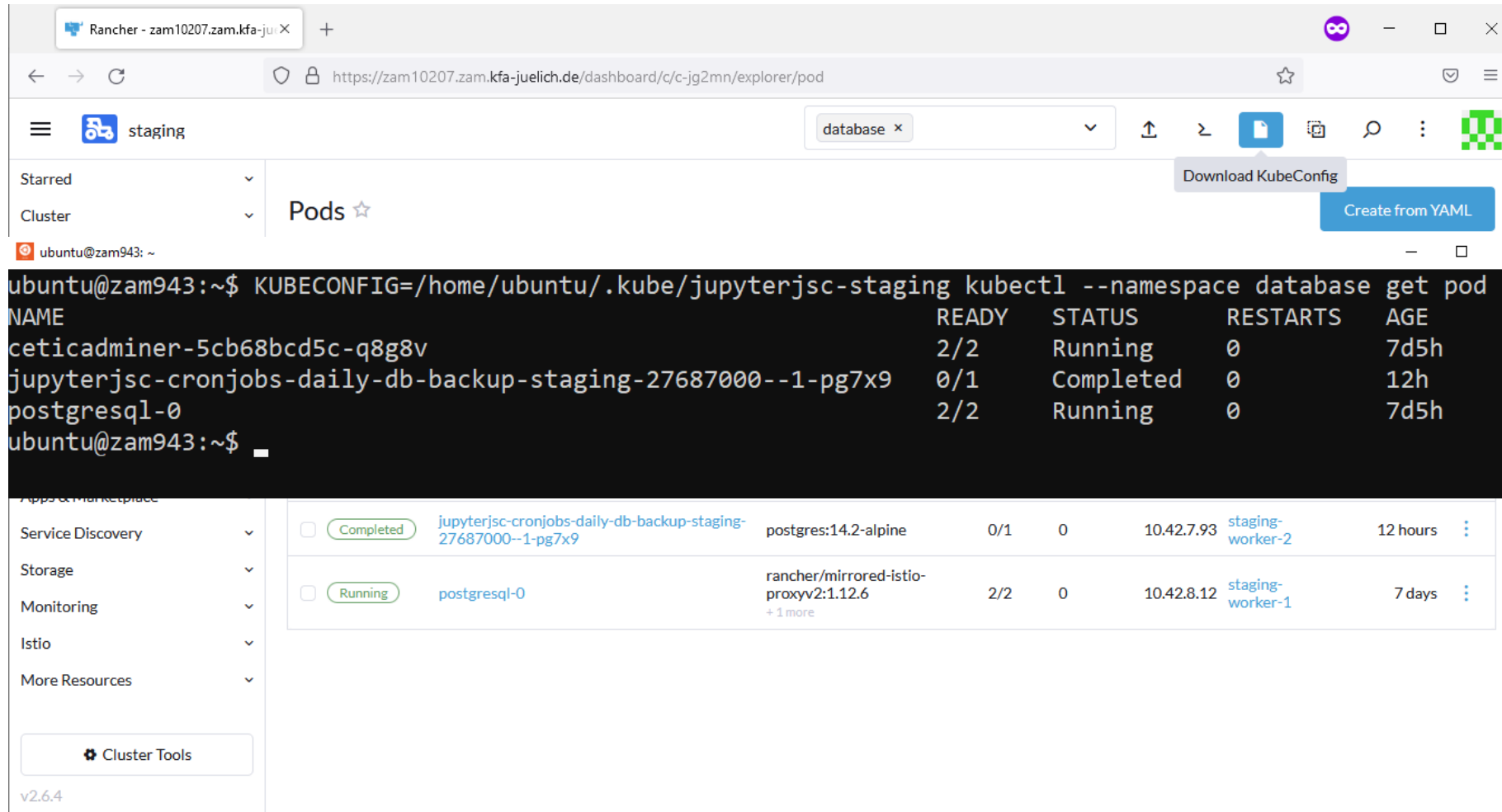
## Why you're here and what you will learn

- Kubernetes in 3 minutes
- Create a kubernetes cluster on OpenStack
- Manage your cluster – Browser and CLI
- Deploy services – the easy way
- Deploy services – the right way



# RANCHER

## Manage your Cluster



The screenshot displays the Rancher dashboard interface for a cluster named 'staging'. The user is viewing the 'Pods' page for the 'database' namespace. A terminal window shows the execution of the command `kubectl --namespace database get pod`, resulting in the following output:

```
ubuntu@zam943:~$ KUBECONFIG=/home/ubuntu/.kube/jupyterjsc-staging kubectl --namespace database get pod
NAME                                READY   STATUS    RESTARTS   AGE
ceticadminer-5cb68bcd5c-q8g8v       2/2     Running   0           7d5h
jupyterjsc-cronjobs-daily-db-backup-staging-27687000--1-pg7x9  0/1     Completed 0           12h
postgresql-0                         2/2     Running   0           7d5h
ubuntu@zam943:~$
```

Below the terminal, the dashboard's pod list table is visible, showing the following entries:

| Pod Name  | Image                                 | Ready | Status    | Restarts | IP         | Node             | Age      |
|---|---------------------------------------|-------|-----------|----------|------------|------------------|----------|
| jupyterjsc-cronjobs-daily-db-backup-staging-27687000--1-pg7x9 | postgres:14.2-alpine                  | 0/1   | Completed | 0        | 10.42.7.93 | staging-worker-2 | 12 hours |
| postgresql-0  | rancher/mirrored-istio-proxyv2:1.12.6 | 2/2   | Running   | 0        | 10.42.8.12 | staging-worker-1 | 7 days   |

# WHAT TO EXPECT

## Why you're here and what you will learn

- Kubernetes in 3 minutes
- Create a kubernetes cluster on OpenStack
- Manage your cluster – Browser and CLI
- Deploy services – the easy way
- Deploy services – the right way

# INSTALL SERVICES

## Helm Charts

- Debian: apt – Fedora: yum – Kubernetes: helm
- Public repositories hosting thousands of charts
  - Listed at [artifacthub.io](https://artifacthub.io)
  - Used directly as git repository
- Versioning, rollouts, rollbacks
- Templates / parameters to fit your needs
- Create your own Helm charts

# INSTALL SERVICES

## Helm Charts

The screenshot shows the Rancher Helm chart installation interface for PostgreSQL 11.8.1. The browser address bar shows the URL: `https://zam10207.zam.kfa-juelich.de/dashboard/c/c-jg2mn/apps/charts/install?repo-type=cluster&repo=bitnami&chart=postgresql`. The interface is in the "staging" environment and displays the "Install: Step 2" configuration screen. A notification banner at the top states: "All charts have at least one version that is installable on clusters with Linux and Windows nodes unless otherwise indicated." The chart is labeled "Linux only". The configuration is shown in a code editor with the following content:

```
210   periodSeconds: 10
211   successThreshold: 1
212   timeoutSeconds: 5
213   name: primary
214   nodeAffinityPreset:
215     key: ""
216     type: ""
217     values: []
218   nodeSelector: {}
219   persistence:
220     accessModes:
221       - ReadWriteOnce
222     annotations: {}
223     dataSource: {}
224     enabled: true
225     existingClaim: ""
226     mountPath: /bitnami/postgresql
227     selector: {}
228     size: 8Gi
229     storageClass: ""
230     subPath: ""
231     pvcHbaConfiguration: ""
```

At the bottom of the interface, there are buttons for "Cluster Tools", "Cancel", "Previous", and "Install". The version number "v2.6.4" is visible in the bottom left corner.

# WHAT TO EXPECT

## Why you're here and what you will learn

- Kubernetes in 3 minutes
- Create a kubernetes cluster on OpenStack
- Manage your cluster – Browser and CLI
- Deploy services – the easy way
- **Deploy services – the right way**

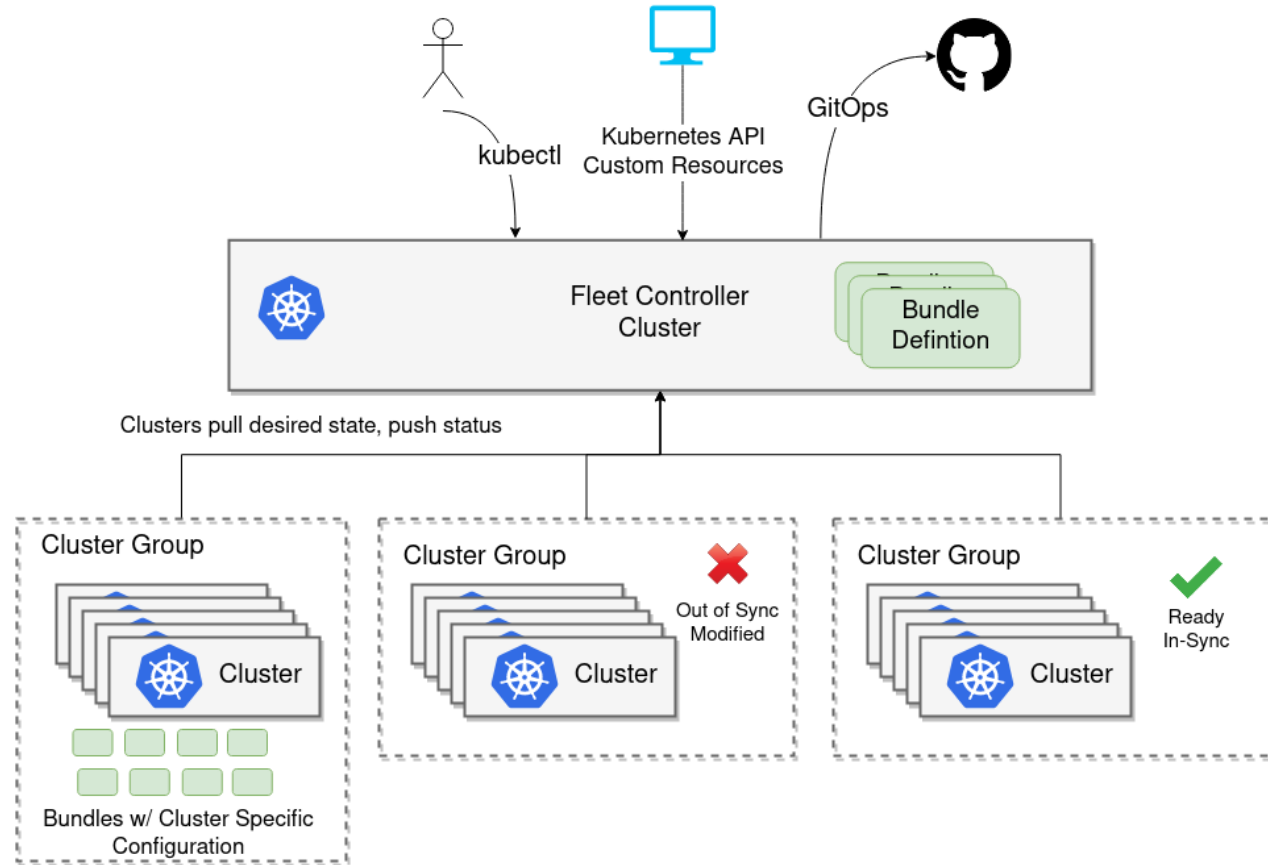
# FLEET

## Continuous Delivery

- Container management and deployment engine
- A Rancher project
- Integrated UI in Rancher
- GitOps – get desired cluster state from git
- Scalability – Manage up to a million clusters - or just one
- Cluster specific configuration possible

# FLEET

Continuous Delivery



fleet.rancher.io

# FLEET

## Continuous Delivery

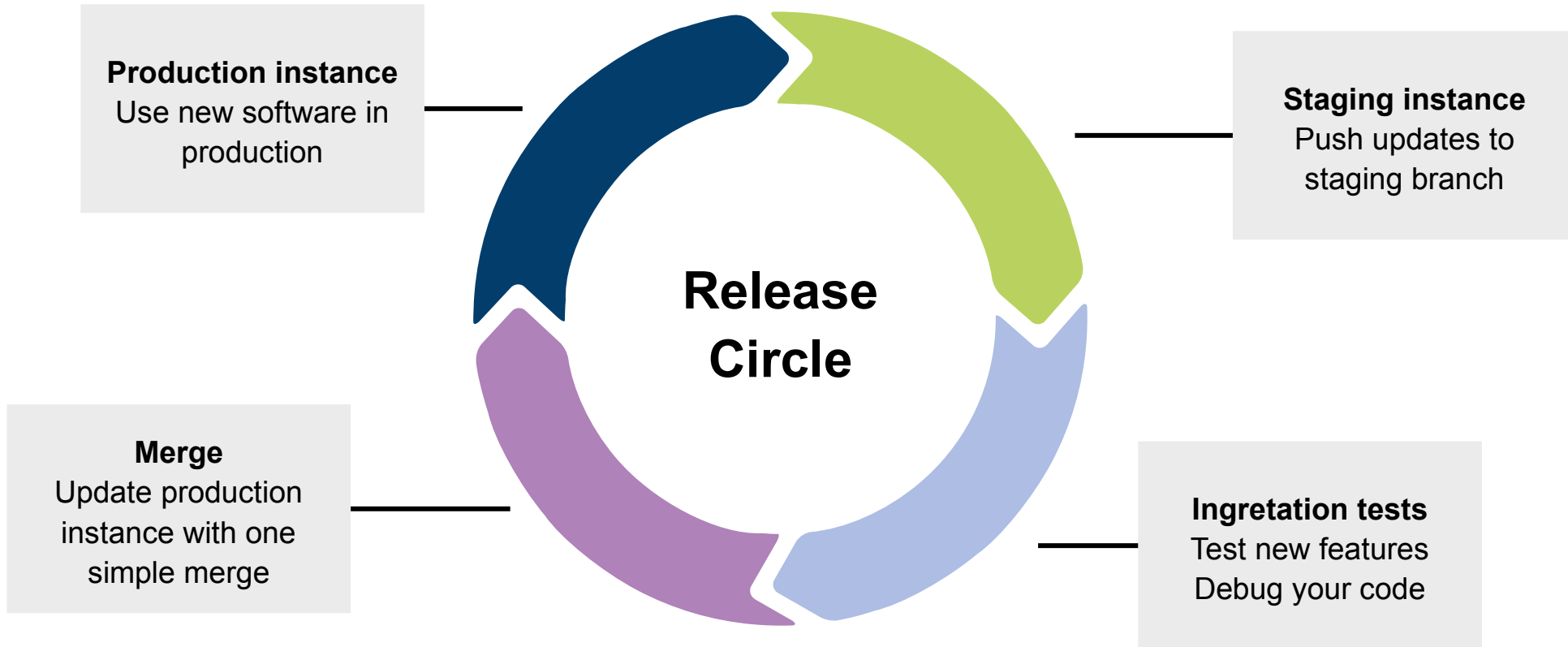
The screenshot shows the Rancher Continuous Delivery interface. On the left, a sidebar lists navigation options: Dashboard, Git Repos (2), Clusters (1), Cluster Groups (0), and Advanced. The main area displays a table of services, each with a 'State' column (all 'Active') and a 'Name' column. A callout box highlights the Helm chart values for a service, including the repository URL, chart name, version, and target customizations.

```
defaultNamespace: database
helm:
  releaseName: postgresql
  repo:
    https://charts.bitnami.com/bitnami
  chart: postgresql
  version: 11.1.28
  values:
    ...
targetCustomizations:
  - name: staging
    clusterSelector:
      matchLabels:
        stage: staging
    helm:
      values:
        ...
dependsOn:
  - basics-postgresql-storage
  - basics-istio
```



# FLEET

## Continuous Delivery





# KUBERNETES ON OPENSTACK

Cluster creation, management and service deployment

12.06.2023 | TIM KREUZER

# APPENDIX

## Make life easier

- Monitoring with Prometheus & Grafana
- CLI – Manage multiple clusters in one terminal
- Local cluster with K3s
- How To Kubernetes – starting point

# APPENDIX

## Monitoring

- Install Rancher monitoring app
- Integrated in Rancher UI
- More than 40 dashboards by default
- Add mail configuration to monitor chart
- Add Grafana alerts to monitor your services
- Default username/password: admin/prom-operator

```
values.yaml:  
  
grafana.ini:  
  smtp:  
    enabled: true  
    from_address: your-svc@fz-  
juelich.de  
    from_name: Your Service  
    host: mail.fz-juelich.de:25
```

# APPENDIX

## Monitoring

The screenshot shows a web browser window with two tabs: 'Rancher - zam10207.zam.kfa-juelich.de' and 'Alerting: Notification channels - x'. The address bar shows the URL: `https://zam10207.zam.kfa-juelich.de/k8s/clusters/c-jg2mn/api/v1/namespaces/cattle-monitoring-system/services/http.rancher.io/`. The page content is dark-themed and displays the 'New notification channel' configuration form. The form includes the following fields:

- Name:** `all_admins`
- Type:** `Email` (selected from a dropdown menu)
- Addresses:** A text input field with a placeholder '...' and a note: 'You can enter multiple email addresses using a ";" separator'. This field is highlighted with a blue border.
- Optional Email settings:** A link with a right-pointing chevron.
- Notification settings:** A link with a right-pointing chevron.

At the bottom of the form are three buttons: 'Save' (highlighted in blue), 'Test', and 'Back'. A left sidebar contains navigation icons for search, home, alerting, settings, and help.

# APPENDIX

## Make life easier

- Monitoring with Prometheus & Grafana
- CLI – Manage multiple clusters in one terminal
- Local cluster with K3s
- How To Kubernetes – starting point

# APPENDIX

## CLI

- Install kubectl: <https://kubernetes.io/docs/tasks/tools/install-kubectl-linux/>
- Set env variable KUBECONFIG to your current cluster configuration
- Bash completion: <https://kubernetes.io/docs/tasks/tools/included/optional-kubectl-configs-bash-linux/>
- Aliases are your friend (but enemies of bash completion)

```
ubuntu@zam943:~$ alias skj
alias skj='KUBECONFIG=/home/ubuntu/.kube/jupyterjsc-staging kubectl -n jupyterjsc'
ubuntu@zam943:~$ alias pkj
alias pkj='KUBECONFIG=/home/ubuntu/.kube/jupyterjsc-production kubectl -n jupyterjsc'
ubuntu@zam943:~$ alias jkcms
alias jkcms='KUBECONFIG=/home/ubuntu/.kube/jupyterjsc-jusufcloud kubectl -n cattle-monitoring-system'
```

- Another way: Everything in one config file. `kubectl config` to switch between clusters

# APPENDIX

## Make life easier

- Monitoring with Prometheus & Grafana
- CLI – Manage multiple clusters in one terminal
- Local cluster with K3s
- How To Kubernetes – starting point



# K3S

## Lightweight Kubernetes

- Built for IoT, CI & Edge computing
- <50Mb binary
- Great for development & first local tests
- K3d
  - Another Rancher project
  - Lightweight wrapper to run k3s
  - One Node in K8s == One Container in K3s
  - K3s cluster with n main and m workers -> n+m+1 container on your machine
  - Cluster started and ready within 3 minutes

# APPENDIX

## Make life easier

- Monitoring with Prometheus & Grafana
- CLI – Manage multiple clusters in one terminal
- Local cluster with K3s
- How To Kubernetes – starting point

# KUBERNETES

## How To

- <https://kubernetes.io/docs/tutorials/>
- [https://www.youtube.com/playlist?list=PLy7NrYWoggjwPggqtFsl\\_zMAwwG0SqYCb](https://www.youtube.com/playlist?list=PLy7NrYWoggjwPggqtFsl_zMAwwG0SqYCb)
- Short videos (10-15 min) for multiple Kubernetes topics