



# How CERN empowers its users with Kubernetes and OpenShift

Jack Henschel

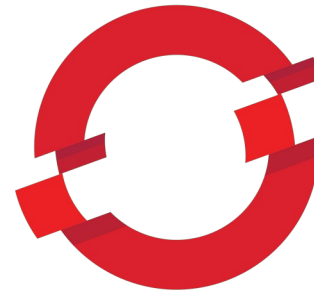
10<sup>th</sup> SIG-CISS meeting @ CS3 2023

# Cloud native @ CERN

**CERN IT offers users two ways of deploying cloud-native applications:**



**kubernetes**



**OPENSIFT**

# Benefits of unmanaged Kubernetes

- User is **Admin**
  - **full control** and **customizability** (e.g. CNI), but requires advanced DevOps skills and ongoing maintenance
- Users can **scale infrastructure** according to their **workload demands** (e.g. using specialized worker nodes)
- Lower infra complexity

# Benefits of managed Kubernetes

- **High-density, multi-tenant** clusters allow **efficient resource usage**
- Suitable for small and medium-sized workloads
- **Managed infra**  
→ user does not need to take care of maintenance, upgrades, etc.
- **User-friendly web UI** for all common operations

# Deploying Containers

Multitude of approaches for configuration management:

- Web Dashboard (OpenShift)
- custom YAML manifests (kubectl apply)
- Kustomize
- Helm charts
- ArgoCD / Flux (GitOps)

→ **no one size fits all solution**

# Resource management

All resources have a well-defined **owner** and are grouped into "projects":

- Kubernetes: Openstack project
- OpenShift: each namespace is tracked in Application Portal

**Lifecycle:** what happens when owner leaves CERN?

**Resource Quota**

# Kubernetes @ CERN

**Fully automated provisioning** with cluster templates based on **OpenStack Magnum**

**Feature toggles** for common addons and integrations: monitoring, logging, storage etc.

**Flexible** deployment options: critical area, technical network

**Continuous testing** with Argo Workflows

# Kubernetes @ CERN

```
$ openstack coe cluster template list
```

uuid	name
14638ec7-ccb6-41af-ba56-249e582c25ed	kubernetes-1.22.9-1
3b05fd04-f543-433c-aba1-320747dc29d0	kubernetes-1.24.7-6
1c9bf2d1-c5f5-4180-a07f-5ef3e0d52b5b	kubernetes-1.25.3-3

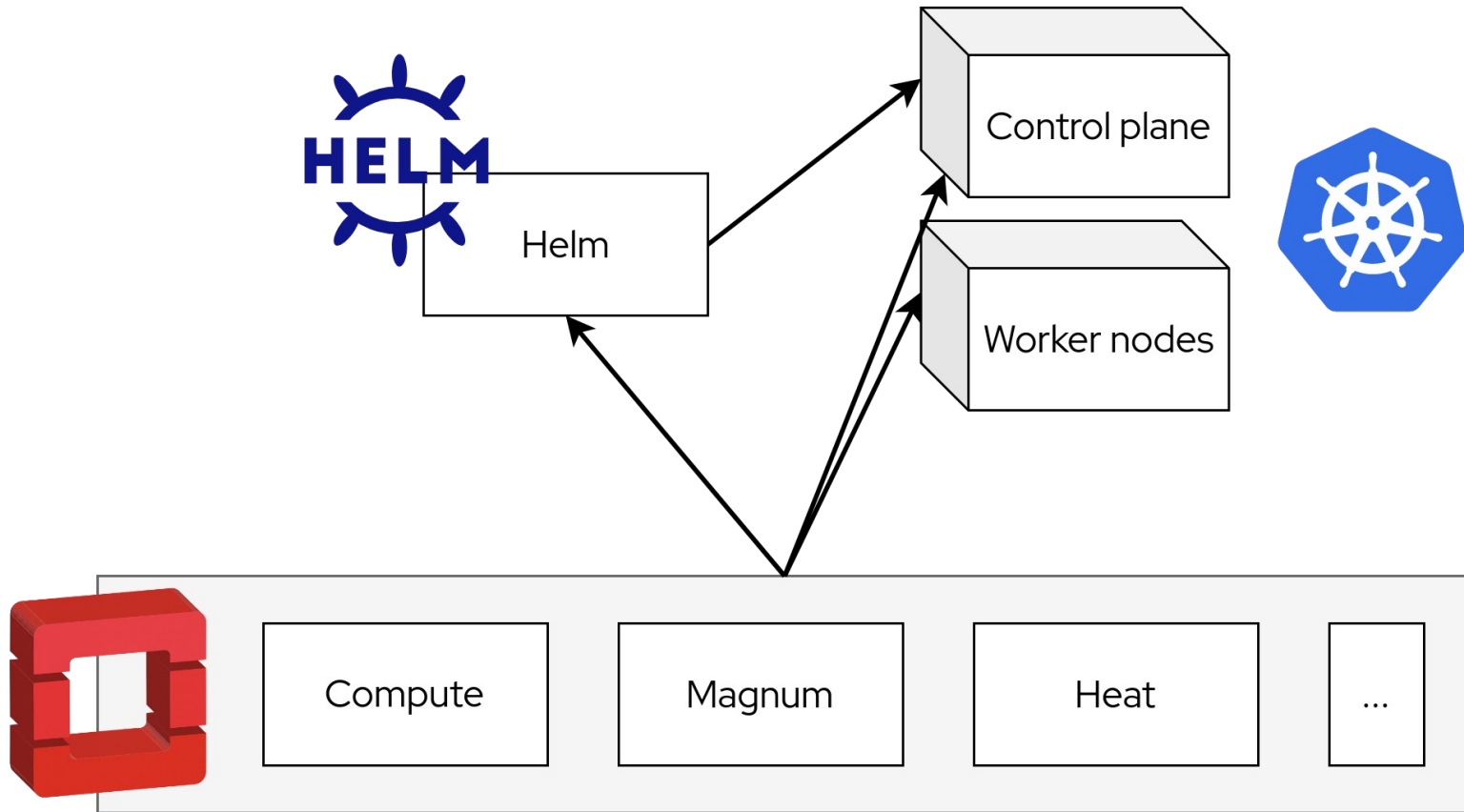
```
$ openstack coe cluster create jacks-cluster --keypair jacks-key \  
  --cluster-template kubernetes-1.25.3-3 \  
  --node-count 2 \  
  --labels monitoring_enabled=true
```

```
$ openstack coe cluster list
```

uuid	name	nodes	masters	status
2582f192-480e-4329-ac05-32a8e5b1166b	jacks-cluster	2	1	CREATE_IN_PROGRESS



# Kubernetes Deployment Architecture



# OKD @ CERN

**OKD4** is the **Foundation of Webservices Infrastructure** at **CERN**

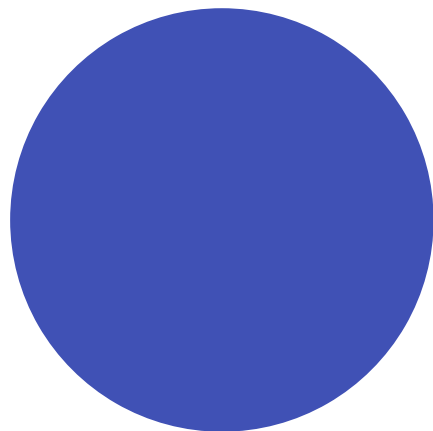
Provides a **multi-tenant, highly-available** and **secure base**

**Enhanced** by us with additional features/integrations for:

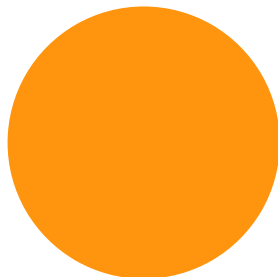
- Hostname registration, DNS setup, Certificates
- Storage: CephFS, EOS, CVMFS
- Ingress router sharding
- Lots of operators!

# OKD @ CERN

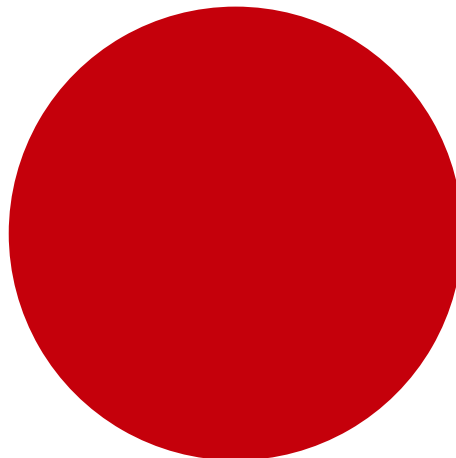
“Our” OKD provides **shared base** for different **cluster flavors**:



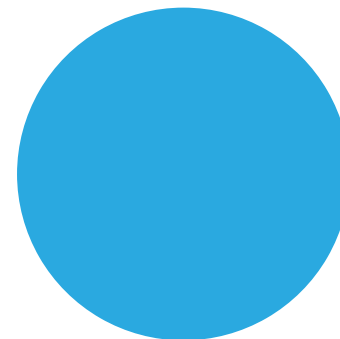
1400 projects, 96 nodes,  
1500 cores, 2.7 TiB memory



300 projects, 50 nodes,  
400 cores, 770 GiB memory



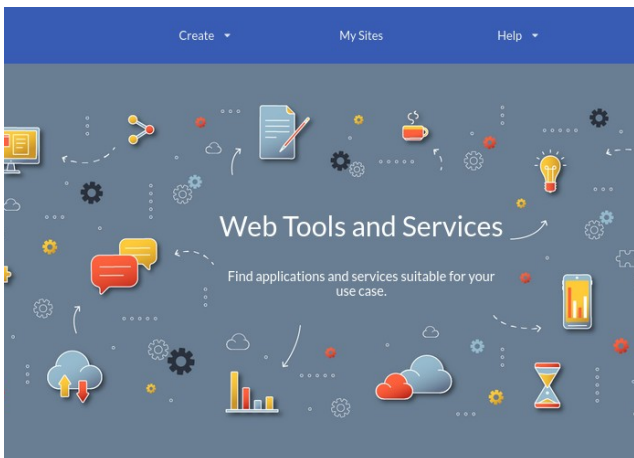
4000 projects, 20 nodes,  
270 cores, 600 GiB memory



800 projects, 60 nodes,  
900 cores, 1.7 TiB memory

# Web Services Portal

**Stateless** web UI for OKD clusters for easy usability for **non-technical users**



**Content Management**

Manage your fully-fledged projects and organizations with content management and WYSIWYG editor.

**Documentation**

Create structured documentation for a service, build your project's knowledge base, work on publications.

**Communication**

Allow your community to discuss topics and get their answers, surveys, send newsletters.

**Application & Site Hosting**

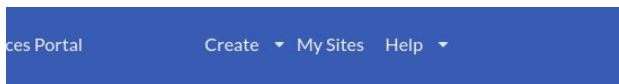
Host your self-made website and share it with others.

**Software development**

Get support for the whole software lifecycle: issue tracking, version control, continuous integration and deployment, repository management, and others.

**Monitoring Solutions**

Add analytics to your web application performance, operational problems with monitoring.



## Creating a WebEOS site

You're minutes away from getting your site up and running.

1. WebEOS sites have their content stored on EOS. Share the EOS location of your choice with user a : `wwwEOS` via the cernbox web interface.  
See how: [for personal sites](#) , [for project sites](#)
2. Create `index.html` file in the EOS location. [See example](#)
3. Fill in the form below and let us do the magic

### Our recommendations for choosing a site name and category

Personal - deleted when owner leaves

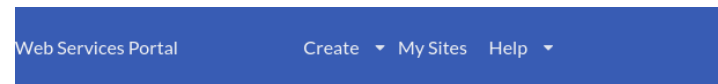
my-site .docs.cern.ch

Documentation for my-site

/eos/user/j/jack/www

- I have read and agreed to the [CERN Computing Rules](#) and taken into account the [design guidelines](#) for websites and the [website lifecycle policy](#).

Create



## Creating a Drupal site

You're minutes away from getting your site up and running.

### Our recommendations for choosing a site name and category

Official - reassigned when owner leaves

my-site .web.cern.ch

My Website Description

Profile - CERN

webservices-containers

- I have read and agreed to the [CERN Computing Rules](#) and taken into account the [design guidelines](#) for websites and the [website lifecycle policy](#).

Create

Service privacy notice

# Behind the scenes

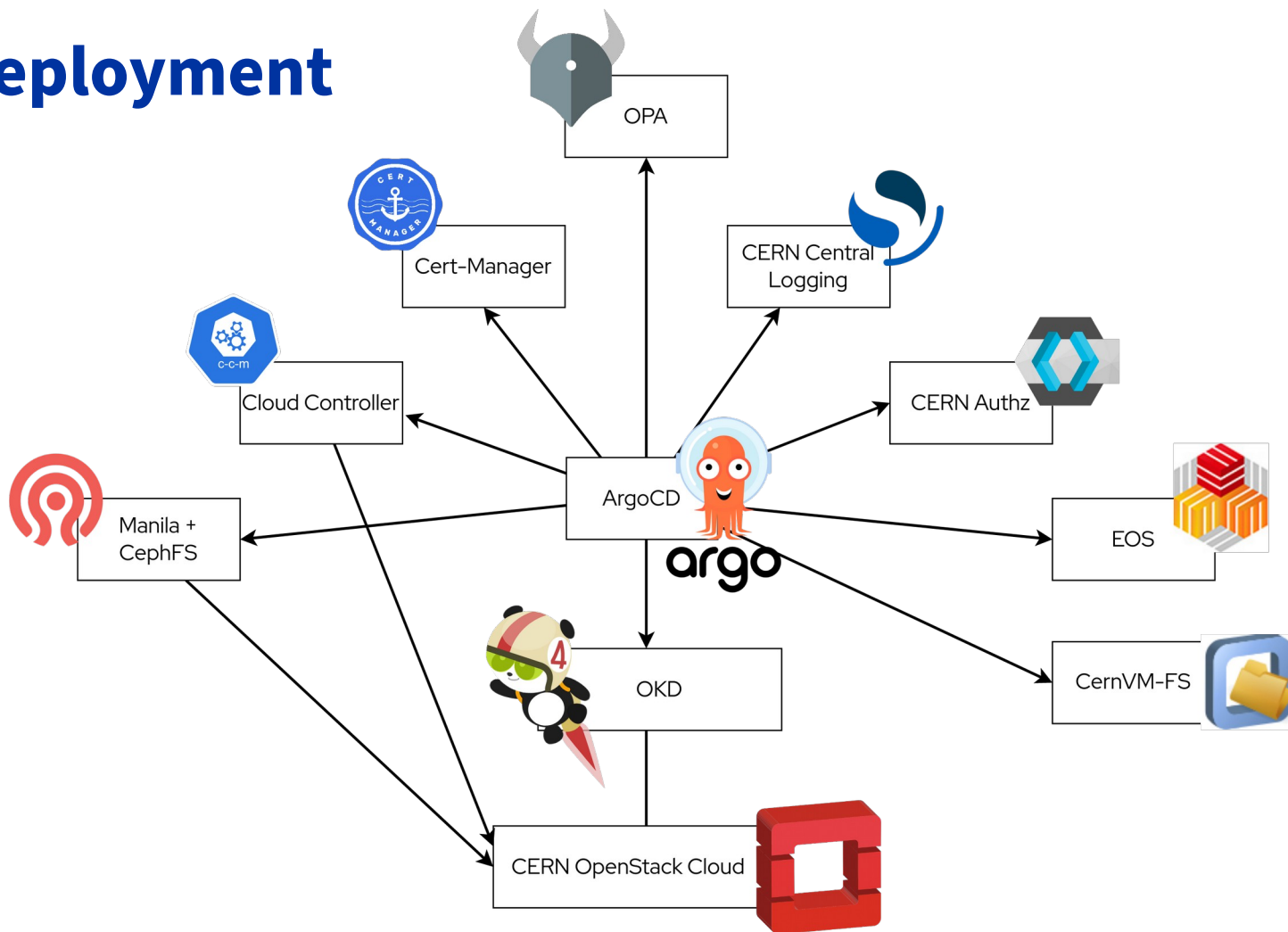
```
apiVersion: drupal.webservices.cern.ch/v1alpha1
kind: DrupalSite
metadata:
  name: drupal-tools
spec:
  configuration:
    databaseClass: standard
    diskSize: 1G
    qosClass: standard
    scheduledBackups: enabled
  siteUrl:
    - drupal-tools.web.cern.ch
  version:
    name: v9.4-2
    releaseSpec: RELEASE-2023.02.13T13-47-51Z
  status:
    availableBackups: [...]
    dbUpdatesLastCheckTimestamp: 'Feb 14, 2023 at 7:38am (UTC)'
    expectedDeploymentReplicas: 1
```

# OKD cluster management

- Clusters are **pets**: production clusters are **stateful** since they run and store **user workload**
- Each cluster is completely **self-sufficient** and **isolated**
- OKD4 **in-place cluster upgrades** are completely **automated** and **seamless**
- All “custom” **infra workloads** are **managed by ArgoCD**
- Developed internal ***okdctl* tool** to facilitate common operations (creating/deleting clusters, replacing nodes)



# OKD deployment



# GitOps with ArgoCD



- **Natural extension** of Kubernetes' **continuous reconciliation** model
- Ensures all resources converge to the desired state
  - despite manual actions in the cluster (troubleshooting, debugging etc.)
  - automatic alerts if this is not the case
- Fits the **operator-driven cluster management** of OKD
- **CLI & Web UI** are useful **for understanding** which resources are deployed and **what their state is**



# Spotlight: OpenPolicyAgent



**OPA** is used for a wide range of use cases (to **help admins & users**):

- **Unique hostnames** across all clusters
- **Ingress** sharding and publishing **DNS** records
- **Volume** labels & annotations (used for backups and mount permissions)
- **Network** visibility (Internet/Intranet/Technical Network)
- **Automation** of EOS mounts (initContainer + sidecar injection)

# Lessons learned

**Splitting** the “Kubernetes-as-a-Service” offering between “power users” and “casual users” **benefits both services**

Both services can **share common components, expertise** and **experiences**

Users are very **happy about internal documentation**

**Operators** are a great way to provide automation **for users and admins**  
- but they are also **very sharp tools** → **use soft deletion where possible!**

**Not every manual operation has to be automated**

# Thank you!



**Email: [jack.henschel@cern.de](mailto:jack.henschel@cern.de)**

**Website: <https://blog.cubieserver.de>**



[home.cern](http://home.cern)