**Centre de Calcul** de l'Institut National de Physique Nucléaire et de Physique des Particules

# WoK - Web on Kubernetes « What's cooking? »

*KEK @ CCIN2P3 – December 2nd, 2019*

Benjamin Guillon

CCIN2P3

cnrs

IN2P3

# What's cooking?

- The WoK Project
- The WoK Platform
  - Current status & lessons learned
- Next steps

- WoK -> Web on Kubernetes

- Complete rework of our web hosting service
  - Getting quite **old** and **hard** to maintain

- 300 web services
  - From the **basic** static website
  - Through **various** content management systems
  - To **complex** web based applications



# Why?

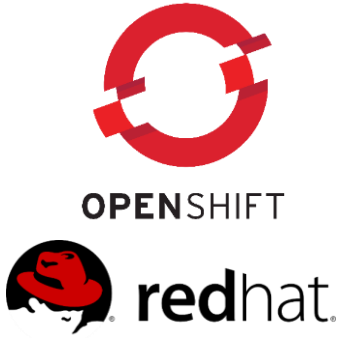
- Containers help us:
  - **Standardize** application deployment
  - Provide extensible **templates**
  - **Maintain** a complex technological ecosystem

- Container orchestration helps us:
  - **Automate processes** in this weird jungle
  - **Integrate** with external systems
    - CI/CD
    - Monitoring
    - Backups
    - Security
  - **Delegate** deployment responsibilities
    - Quotas
    - Role based access control

# The WoK Platform

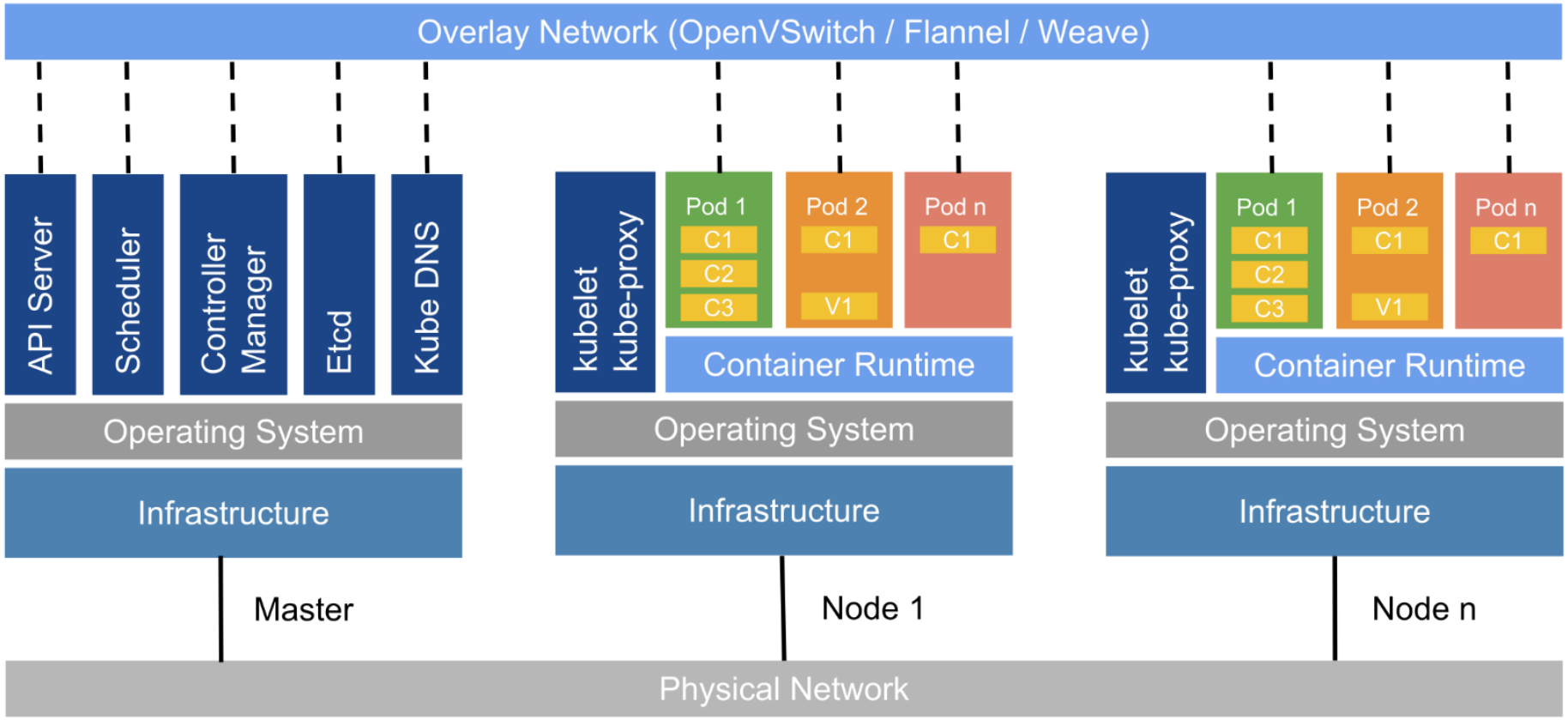
- Vanilla Kubernetes: endless possibilities, but …
  - We need a **reliable production ready** platform
  - We are still learning… ¯_(ツ)_/¯

- Enters **Openshift**
  - RedHat flavoured Kubernetes
  - Production grade Kubernetes **distribution**



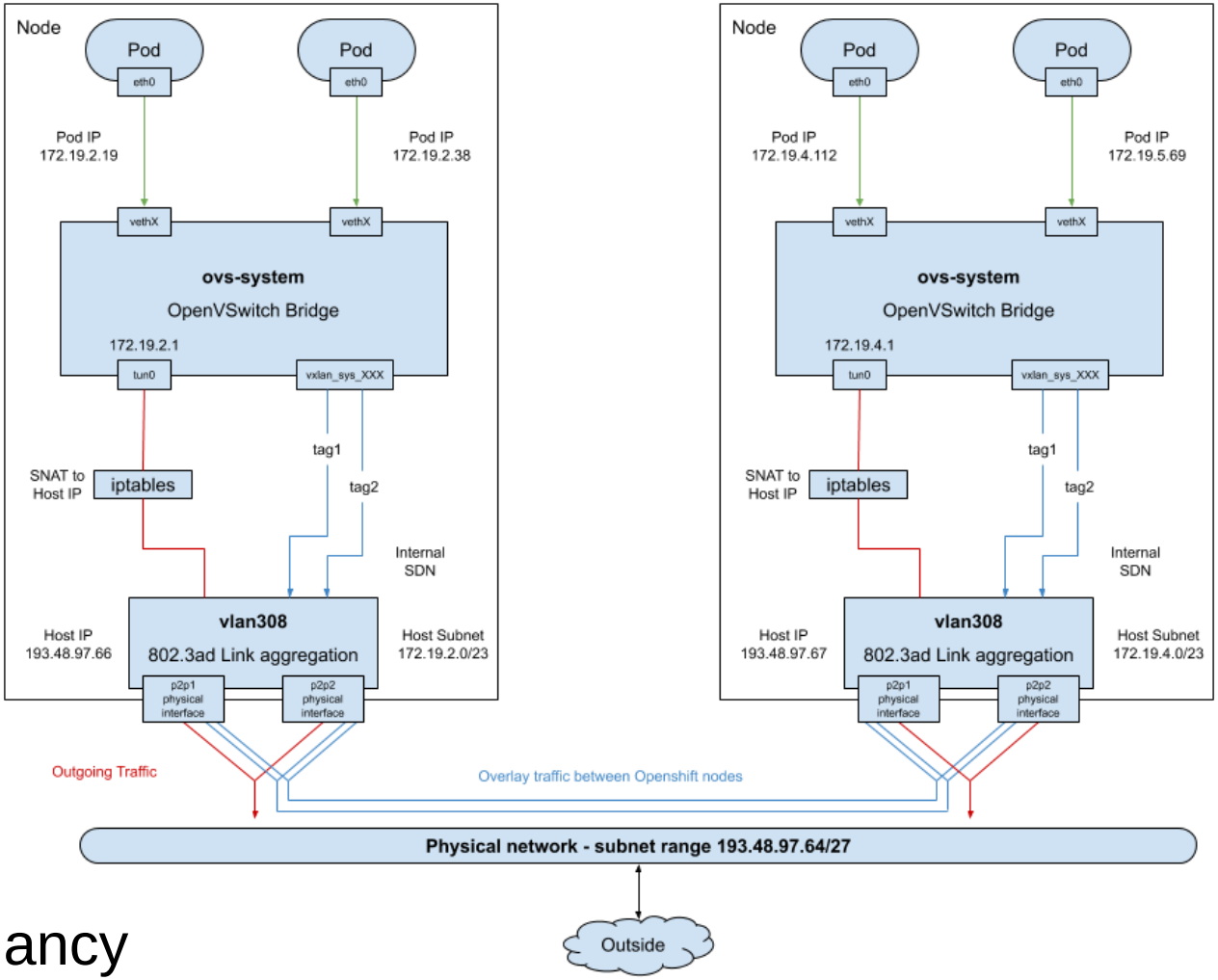

  - Cumbersome **choices** made for us:
    - Network stack, monitoring, platform management, UIs…
  - Application **development oriented**
    - CI/CD pipelines integrated
      - Image streams, Internal registries, Image builders
    - Service catalog/marketplace
      - Templates ready to go
  - **Security** enabled
    - Security Context Constraints (SCCs - SELinux inside™)
    - Role Based Acces Control – what can users do

# Platform architecture




*« Piece of cake! » said no one, ever.*

# Networking




- Multi-tenancy
  - **Seamless** connectivity between pods **wherever** they live
  - Network **isolation** at the **namespace** level
  - **Access rules** can be set

# Infrastructure Deployment

- Automation
  - Puppet for the base system / CCIN2P3 Ecosystem
  - Ansible for the Openshift platform



- Still takes a while to deploy
  - ~ One day (well known environment)
- Still *difficult* to make it work on our Openstack platform
  - OVS on top of OVS ?
- Should allow us to deploy multiple clusters in the end
- Will be revamped for Openshift v4
  - Immutable systems using the CoreOS technology

# Storage

- Mostly relying on Ceph RBD
  - Provision container volumes **dynamically**
  - Replica3 efficient storage
  - Snapshots available for backups


- Need « ReadWriteMany » volumes ?
  - Not provided by Ceph RBD
  - CephFS could be used but still experimental in Openshift
  - Relying on NFS, though lacking several features

# AAI

- Plugged to our new FreeIPA/Keycloak service
  - Provides user **identification** and **authentication**
- All the **authorization** mecanisms handled by K8s
  - RBAC: Role Based Access Control
- Still trying to make groups work
  - For now, still managed manually in the cluster


- What about service accounts?

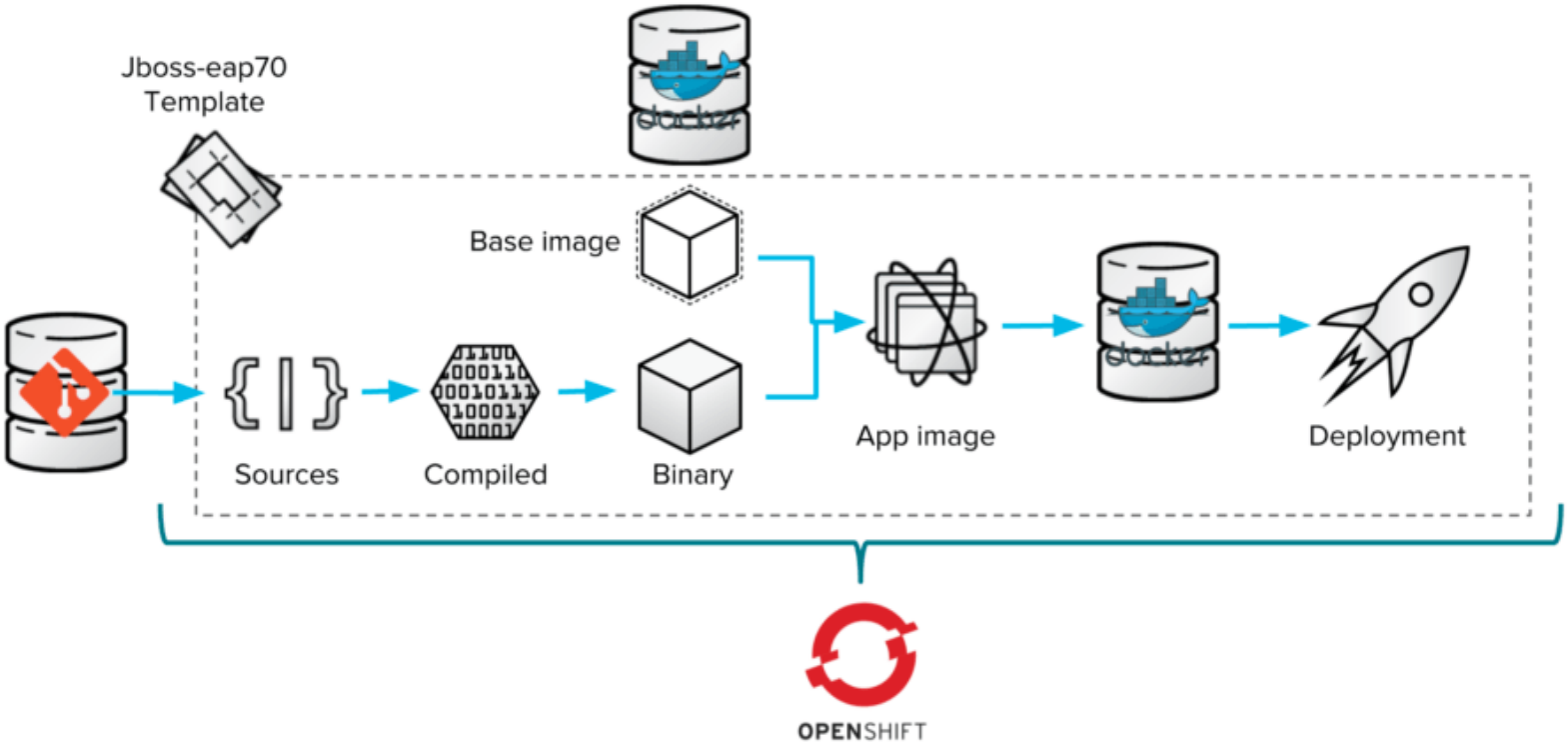
# Monitoring

- Metrics, logs & alerts
  - Application level: users
  - Infrastructure level: admins

- Two monitoring stacks are provided by Openshift
  - Metrics: Prometheus & AlertManager w/ Grafana
  - Logs: Fluentd, ElasticSearch w/ Kibana


- Legal **log retention** prerequisite
  - One year in France
  - Exported to Colossus, CCIN2P3 central monitoring service

# Deploying apps



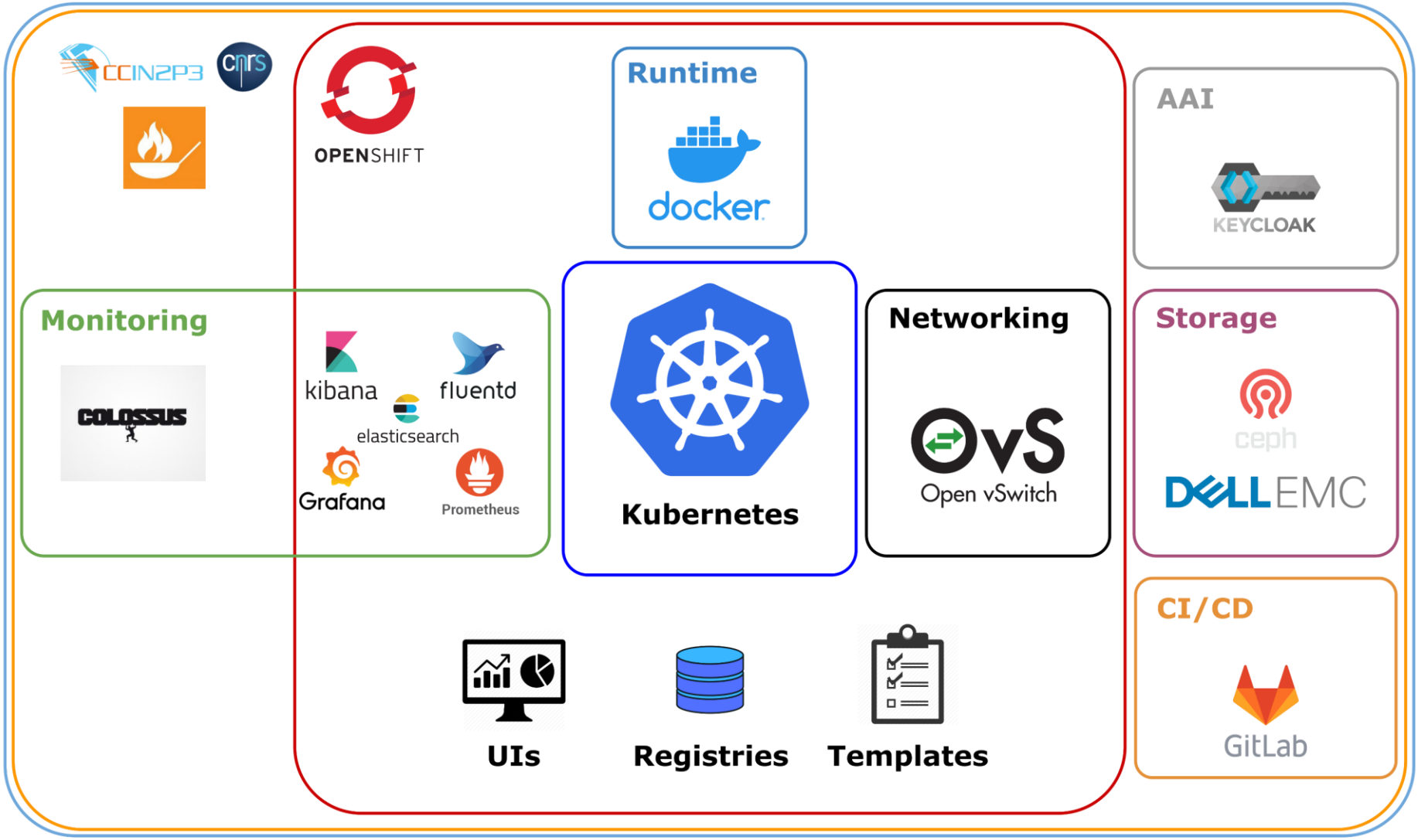
- Leveraging Kubernetes Operators for SSL certification
  - Let's Encrypt certificate **generation** and **renewal**
  - Fully automated: fire and forget

- A simple **metadata** associated to a https endpoint

```
apiVersion: route.openshift.io/v1
kind: Route
metadata:
  annotations:
     kubernetes.io/tls-acme: 'true'
[…]
```

- Picked up by the letsencrypt operator on the fly

# The big picture



# Next steps

- **Polishing** all this clockwork

- **Migration** of the web services on WoK

  ~ 250 sites to go

- **Investigating** other uses
  - HTC/HPC Computing ?
    - Getting rid of the workflow managers and batch schedulers??
  - Storage on demand?
  - Functions as a Service?
  - Anything on demand??

# I talked way too much, sorry.

That’s all folks, thanks!