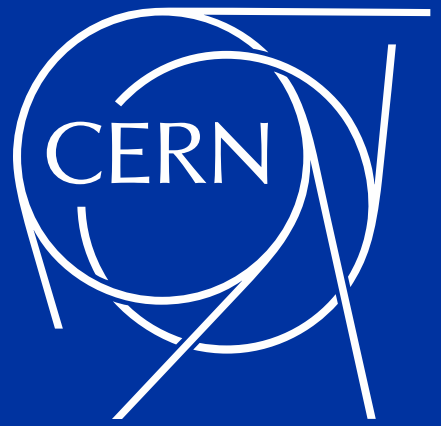


Migration between Kubernetes versions doesn't have to be error-prone

Adrian Karasinski
adrian.karasinski@cern.ch
3/16/2023

Service Mesh – static analysis of service dependencies

ORACLE

- We look at many different tools and things
- Part of the project was:
  - Investigation what we can use or do
  - Develop something together with Oracle
- And we concluded that Kubernetes (K8s) is good target

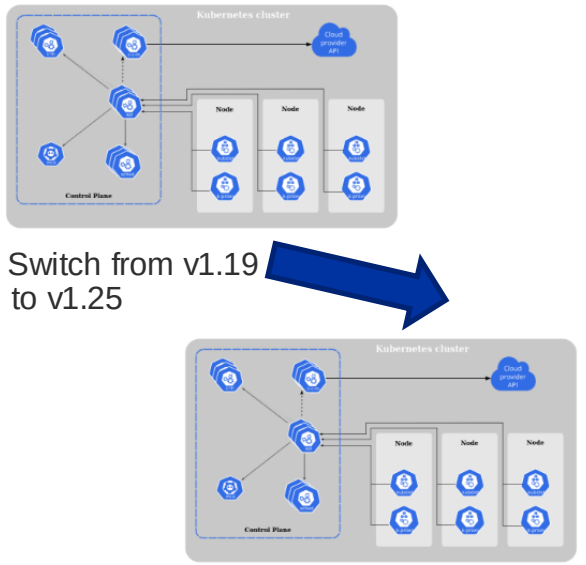
Specific case: Migration/Service Mesh in kubernetes

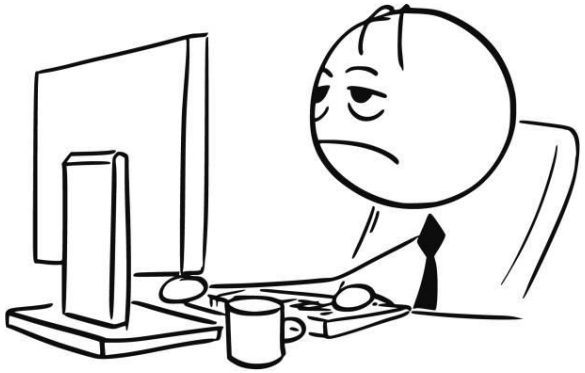
- Kubernetes is the main tool used in our CERN's group to application deployment
- Allows our applications (placed in containers) to talk and communicate with each other
- Popular worldwide and in various industries
- Available in public clouds, private clouds, hybrid solutions, on-premise, etc.

Migration in K8s – current process of work

Typical step-by-step scenario:

- Manually checking release notes and should, but not usually no one does every templates (possibly thousands of files)
- Trying to re-deploy (re-use) all existing definitions in new cluster version
- Getting feedback only after everything has happened
- Unmeasured amount of work and verifications to be done, unestimable plans due to lack of feedback





Typical annual migration – to err is human

At least once per year:

1. a new version of the cluster has been announced
2. a few thousand definitions? Let's check it 1:1
3. any omission or unforeseen behavior: a chance that some critical systems will not work temporarily, which means abandoning live or professional plans ones in favor of maintenance.
4. if we miss something in testing environment, then most critical apps like Access Control, impact.cern.ch, phonebook.cern.ch may work unstable or be unavailable

A way to improve

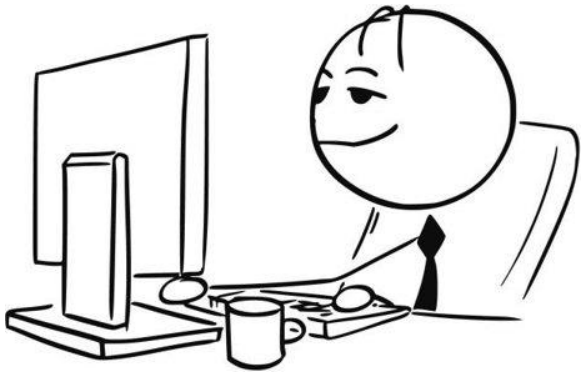
Let's automate our work! Why bother a man when a machine can do it for him?

- automatically validate things
- feedback before, not surprise after

- on-the-fly conversion
- catch potential changes and divergences

Our local world after the changes

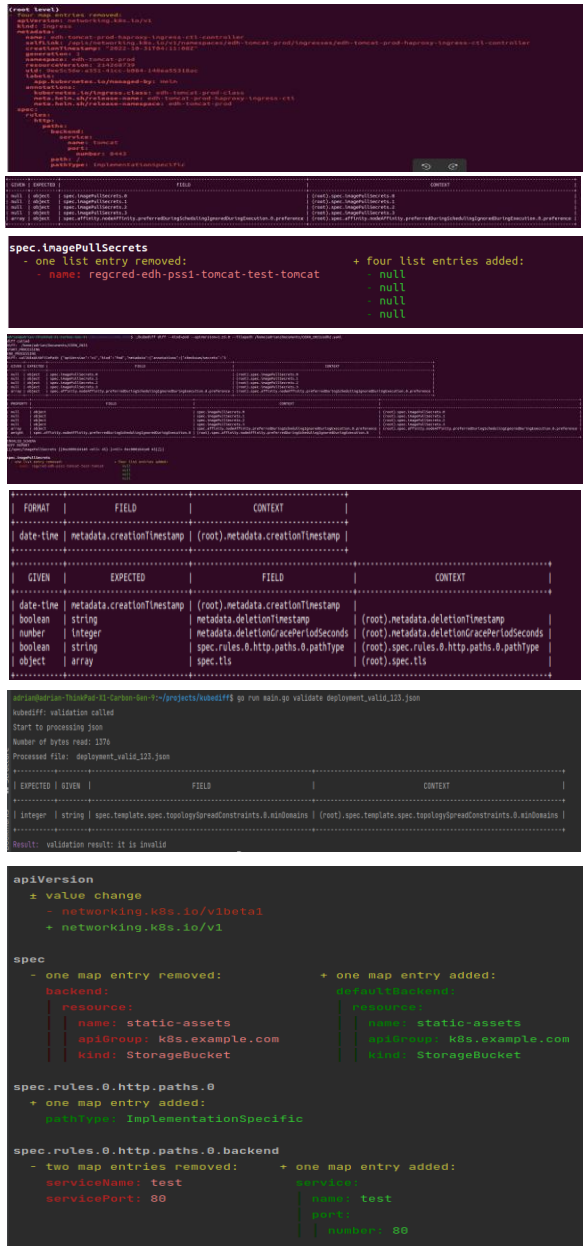
- Daily work planned without surprises
- Reduced errors - you can spend time safely with friends or family, not on the phone or at the laptop
- More time doing smart things at work and less time reading schematics that machines/computers should read;
- Let's leave creative work to people



Results

Technicalities:

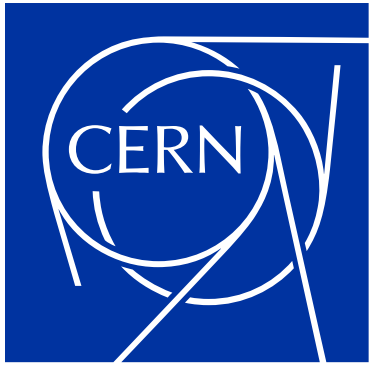
- 51+ supported K8s kinds
- Full integration with K8s cluster API (kubeconf settings file)
- Integration also local templates/resources storage
- Simple deprecated/removed API versions view



Plans for the future & features

Challanges:

- **Incoming support for CRDs (custom resource definitions)**
- **Possible integration with some kind of pipeline/s (like gitlab stages or jenkins jobs)**
- **Create plugin based application (e.g. as an extension for VS Code)**
- **Scrapping validation rules directly from K8s server codebase**
- **Licensing - spreading the idea to the open source world**
- **Your suggestions... ☺**


CERN