white duck

Containerized Build & Deployment Pipelines

Continuous Lifecyle / Container Conf, Mannheim 2019

Microsoft Partner

Microsoft

Gold DevOps
Gold Cloud Platform

Nico Meisenzahl

- Senior Cloud & DevOps Consultant at white duck
- Docker Community Leader & GitLab Hero
- loves Kubernetes, DevOps and Cloud

Phone:	+49 8031 230159 0
Email:	nico.meisenzahl@whiteduck.de
Twitter:	@nmeisenzahl
LinkedIn:	https://www.linkedin.com/in/nicomeisenzahl
Blog:	https://meisenzahl.org

Agenda

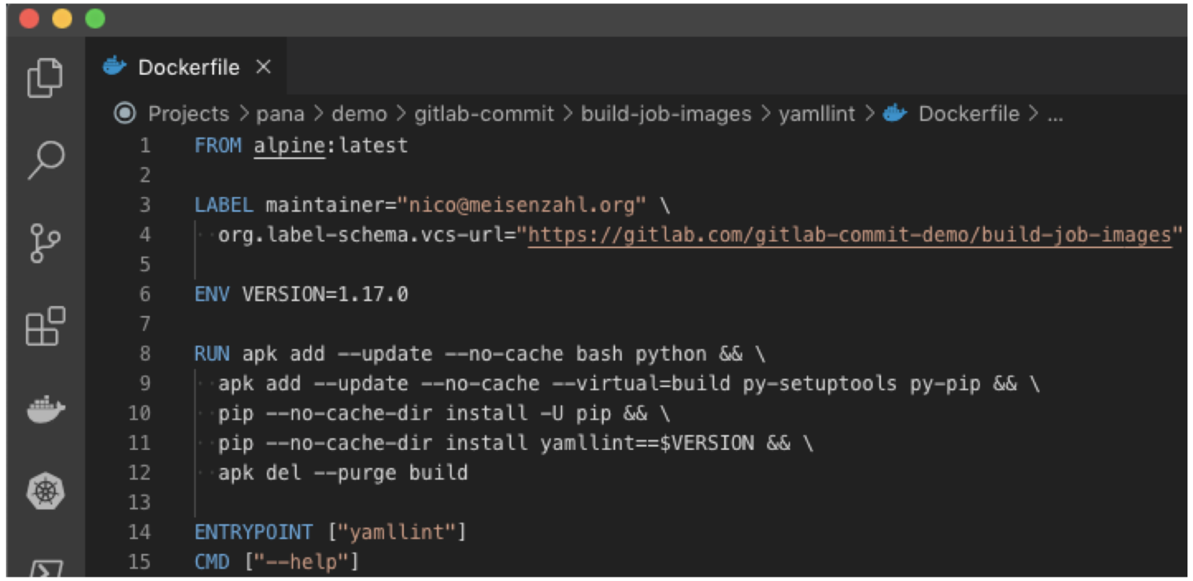
- why should you containerize your pipeline?
- pipeline workload on Kubernetes with GitLab CI/CD
- image builds on Kubernetes with Kaniko
- cloud-native pipelines with Tekton

Why should you containerize your pipeline?

- for the same reasons why you should use containers
  - isolation
  - dependencies
  - scalability
  - immutability
- example: your new project needs version X all others still require Y
  - you can insert any kind of build / deploy dependency
  - .NET Core, Go, Terraform, Ansible… you name it

Pipeline job image

- contains everything a single pipeline job needs
  - binaries, libraries, tools, ...
- use a pipeline to build/rebuild it periodically (security fixes!)
- you should define fix versions for your dependencies


Dockerfile
Projects > pana > demo > gitlab-commit > build-job-images > yamllint > Dockerfile > ...

```
FROM alpine:latest

LABEL maintainer="nico@meisenzahl.org" \
  org.label-schema.vcs-url="https://gitlab.com/gitlab-commit-demo/build-job-images"

ENV VERSION=1.17.0

RUN apk add --update --no-cache bash python && \
  apk add --update --no-cache --virtual=build py-setuptools py-pip && \
  pip --no-cache-dir install -U pip && \
  pip --no-cache-dir install yamllint==$VERSION && \
  apk del --purge build

ENTRYPOINT ["yamllint"]
CMD ["--help"]
```

Kubernetes vs. *docker run*

- every pipeline job runs in a container
  - based on an image with all requirements for this single job
- Build host with Docker daemon
  - or any other container solution
- GitLab Runner Kubernetes executor
  - integrates your CI/CD with Kubernetes
  - runs a pod per job
    - containing a container with the defined image along with some service containers
  - allows you to share your compute and scale your pipelines

GitLab Runner Kubernetes executor

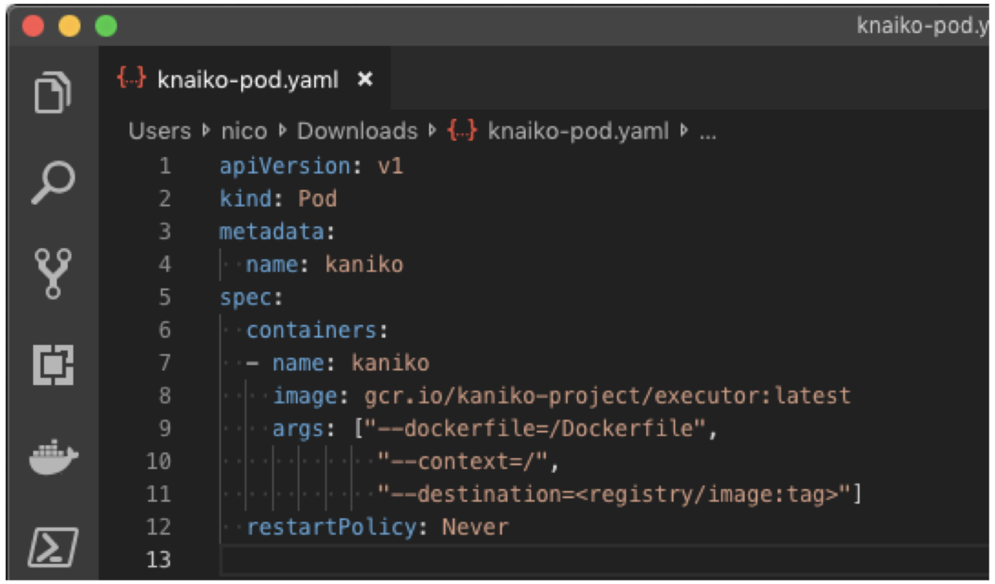
- runs itself in a pod
- needs to be installed in your Kubernetes Cluster
  - automatable Helm deployment
- schedules job pods
- build steps of a pipeline job
  - prepare → creates pod with build and service containers
  - pre-build → clones repo, restore cache, download artifacts
  - build → user build steps
  - post-build → creates caches and upload artifacts

Demo

- containerized pipelines on Kubernetes with GitLab Runner

Image builds on Kubernetes with Kaniko

- any Docker-in-Docker solution has issues
  - exposing Docker socket
  - mounting /var/lib/docker
  - privileged mode
- image builds without the need of any privileges or dependencies
- runs in a container
  - http://gcr.io/kaniko-project/executor
- use build caching to speed up your pipeline



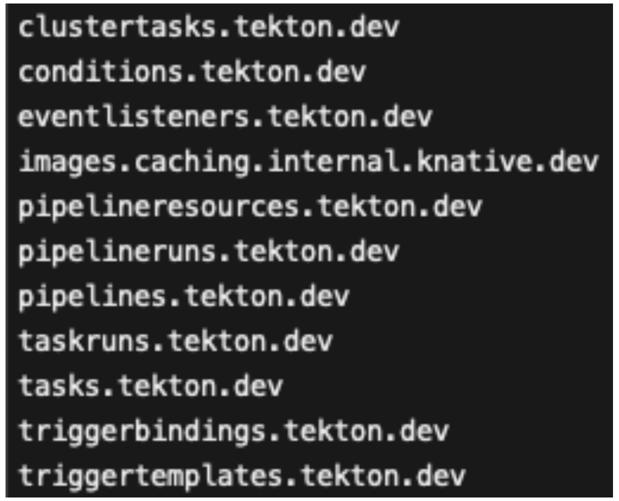
```yaml
apiVersion: v1
kind: Pod
metadata:
  name: kaniko
spec:
  containers:
  - name: kaniko
    image: gcr.io/kaniko-project/executor:latest
    args: ["--dockerfile=/Dockerfile",
            "--context=/",
            "--destination=<registry/image:tag>"]
  restartPolicy: Never
```

Demo

- containerized image builds on Kubernetes with Kaniko

Cloud-native pipelines with Tekton

- Tekton Pipelines emerged out of the Knative build project
- Jenkins X pipelines are based on Tekton
- moves your whole CI/CD into Kubernetes
- based on
  - CRDs (Custom Resource Definitions)
  - Controllers
- Tekton Triggers can be used to call a pipeline
- project also provides a CLI and Dashboard



```
clustertasks.tekton.dev
conditions.tekton.dev
eventlisteners.tekton.dev
images.caching.internal.knative.dev
pipelineresources.tekton.dev
pipelineruns.tekton.dev
pipelines.tekton.dev
taskruns.tekton.dev
tasks.tekton.dev
triggerbindings.tekton.dev
triggertemplates.tekton.dev
```

Demo

- cloud-native pipelines with Tekton Pipelines

Questions?

Slides: https://www.slideshare.net/nmeisenzahl
Demo: https://gitlab.com/nmeisenzahl/conconf-conli

Nico Meisenzahl (Senior Cloud & DevOps Consultant)

Phone:	+49 8031 230159 0
Email:	nico.meisenzahl@whiteduck.de
Twitter:	@nmeisenzahl
LinkedIn:	https://www.linkedin.com/in/nicomeisenzahl
Blog:	https://meisenzahl.org