Recap


Bryan
Serverless, Loki and more

Summary KubeCon 2019

Basic Facts

- May 20: Pre-Conference Co-Located Events + Lightning Talks
- May 21 – 23: Conference
- 7,700 Visitors
- More than 150 companies and exhibitors

Motivation

- Overview about the entire ekosystem
- Tools and Technologies for Kubernetes on prem

Serverless Summit

CNCF Serverless Landscape

2019-06-04T03:50:12Z 9ce6144

See the serverless interactive display at s.cncf.io

Greyed logos are not open source

Tools

CLOUDFLARE · dashbird · Epsagon · event gateway · HASURA · IO|pipe · Iron.io · python-λ · SCAR · SIGMA · STACKERY · THUNDRA

Security

intrinsic · Protego · PURESEC · threat stack

Framework

APEX · Architect · Chalice · AWS SAM · Claudia.js · FLOGO · gus.io · serverless · SPARTA · Spring Cloud Function

Hosted

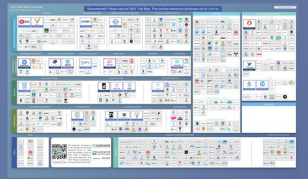
Installable

Platform

ALGORITHMIA · AWS Lambda · Azure Functions · BINARIS · cloudboost · CLOUDFLARE Workers · Google Cloud Functions · HUAWEI FunctionStage · IBM Cloud Functions · kitJune · netlify · nimbella · PubNub Functions · spotinst · stdlib · Tencent Cloud · twilio Functions · ZEIT

OpenWhisk · AppScale · DISPATCH · fission · fn · GALACTIC FOG · Knative · Kubeless · Kyma · LunchBadger · nuclio · OPENFAAS · PipelineAI · rff · Virtual Kubelet

Cloud Native Landscape



s.cncf.io

Serverless computing refers to a new model of cloud native computing, enabled by architectures that do not require server management to build and run applications. This landscape illustrates a finer-grained deployment model where applications, bundled as one or more functions, are uploaded to a platform and then executed, scaled, and billed in response to the exact demand needed at the moment.

CLOUD NATIVE Landscape

CLOUD NATIVE COMPUTING FOUNDATION

Redpoint

https://github.com/cncf/landscape

Serverless Summit

OpenFaaS

- Framework for building serverless functions based on docker
- started early 2017
- Goals: same developer experience for all platforms and providers
- Integrates with multiple providers and regions

  (e.g. AWS Lambda, Kubernetes, Docker swarm, ...)

- Features: function store, templating system, API gateway

Serverless Summit

Knative

- serverless on Kubernetes
- started early 2018
- 3 pillars: build, serving, events
- Knative build was moved to another project called Tekton

Knative

Serving

- Autoscaling based on requests
- Support for scale to zero
- Manages Code/Configuration revisions
- Integrates with service meshes
- Support for custom domains and certificate management
- Traffic splitting
- Pluggable

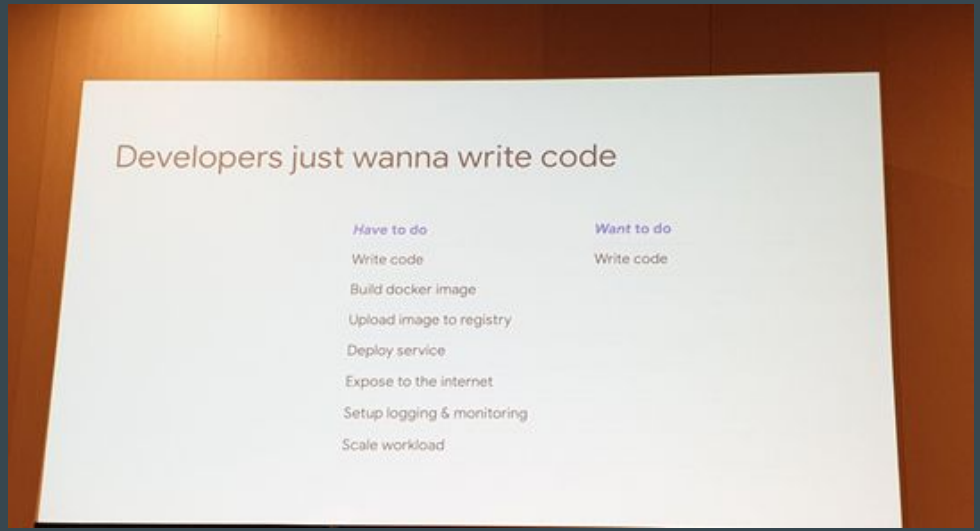
Eventing

- Orchestrates on/off cluster event sources
- Binds declaratively event sources, triggers and services
- Scalable
- Uses standard CloudEvents
- Routing
- Filtering
- Loosely coupled
- Pluggable

Serverless Summit

Takeaways

- Serverless is an abstraction layer above infrastructure and operation
- Devs don’t care about infrastructure
- Serverless != Functions
- Deeper look into Knative
- OpenFaaS could not convince me



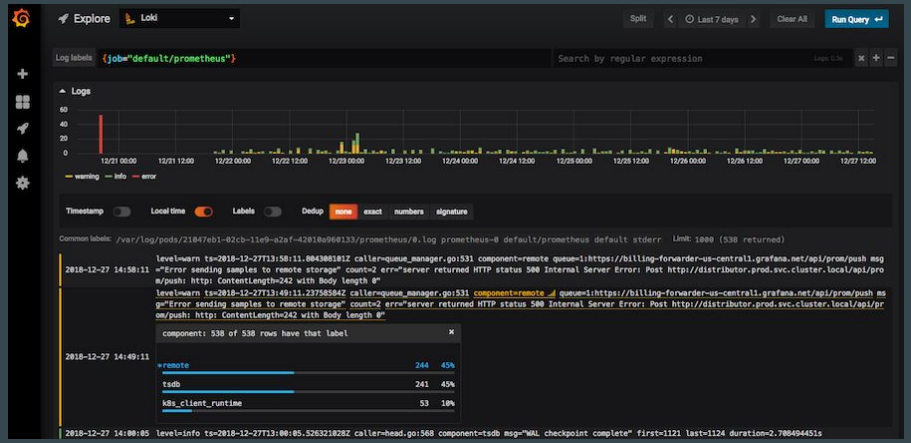
Grafana Loki

Grafana Loki

“Loki is a logging backend, optimized for users running Prometheus and Kubernetes.”

Facts

- Project started in 03/18
- First Launch at KubeCon US in 12/18
- currently in Beta



https://grafana.com/blog/2019/01/02/closer-look-at-grafanas-user-interface-for-loki/

Grafana Loki

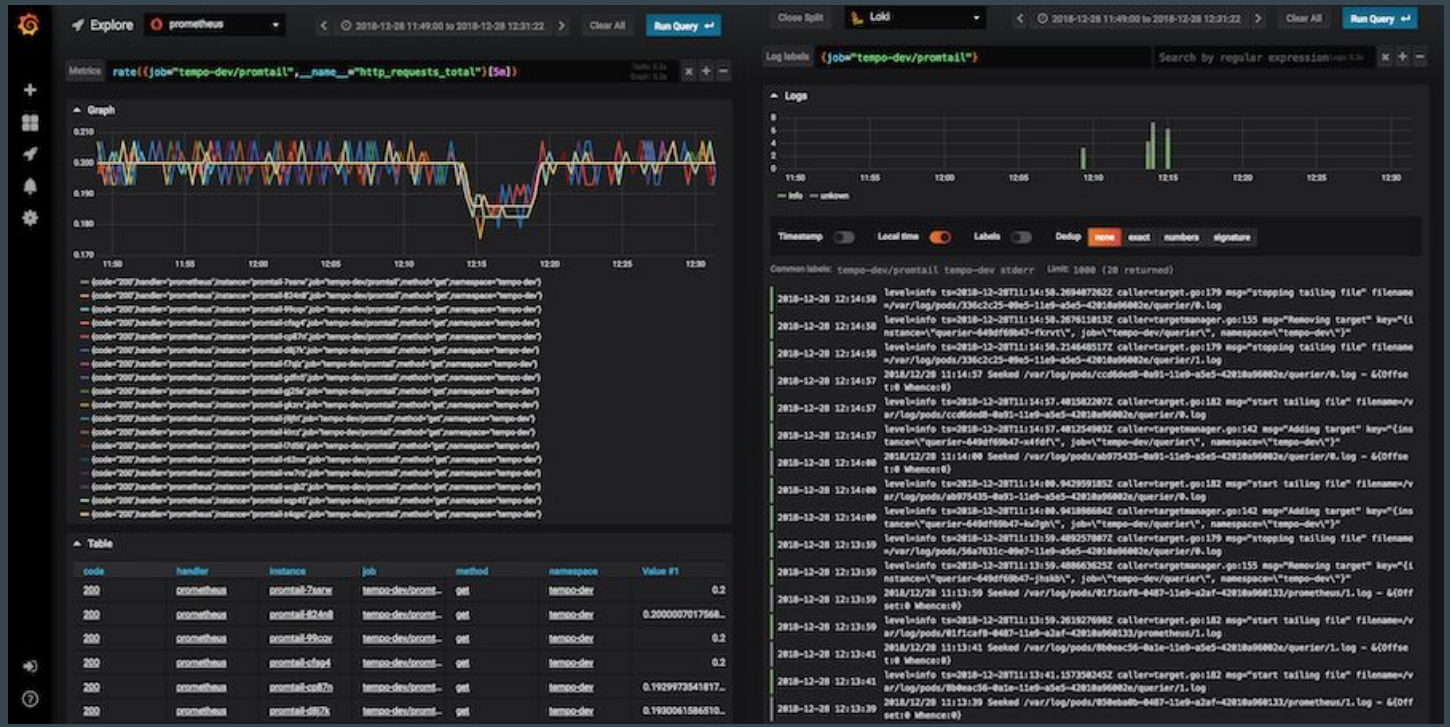
Features

- Easy to install
- No pagination
- Filtering
- Label Support
- Ad-hoc statistics
- LogQL Query Language: `{job="app"} |= "/foo" !~ "/foo/bar"`
- Context View
- Explore and Split view
- ....

No support for

- Indexing log patterns
- Multiline LOG Statements

Grafana Loki - Split Screen



https://grafana.com/blog/2019/01/02/closer-look-at-grafanas-user-interface-for-loki/

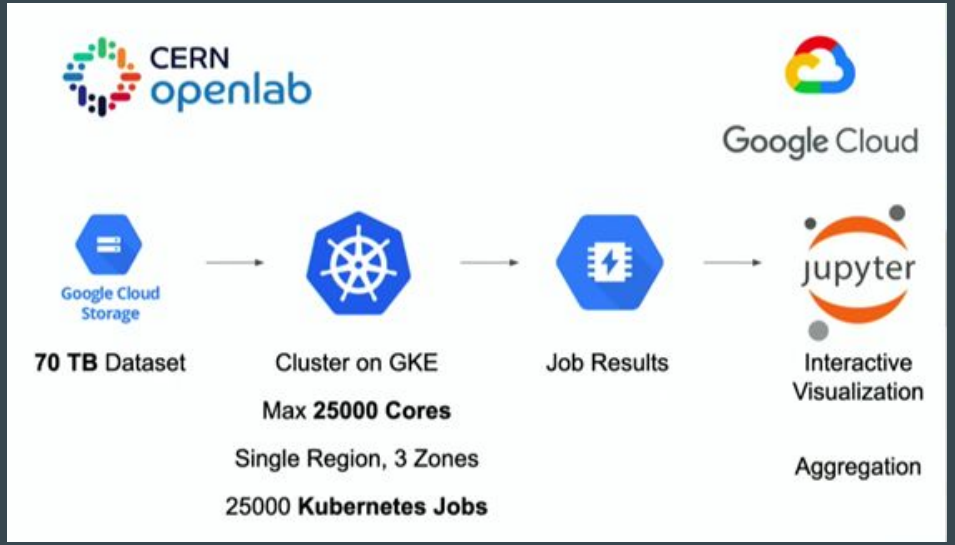
Other Sessions

Other Sessions

Reperforming a Nobel Prize Discovery on Kubernetes

by Ricardo Rocha & Lukas Heinrich

2 physicists performing a live demo to discover the higgs boson



https://www.youtube.com/watch?v=CTfp2woVEkA

Other Sessions

How Spotify Accidentally Deleted All its Kube Clusters with No User Impact

by David Xia

Infrastructure engineer speaks about failure and disaster recovery

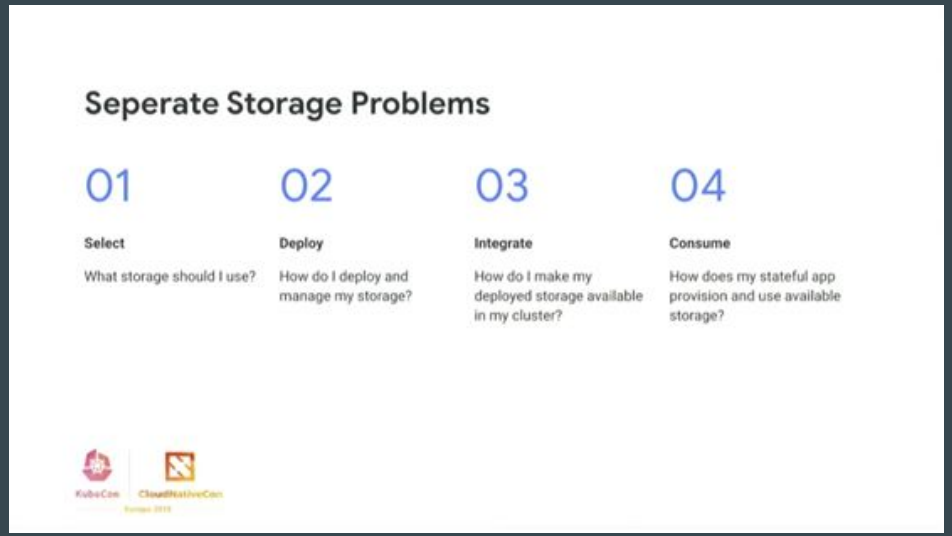
https://www.youtube.com/watch?v=ix0Tw8uinWs

Other Sessions

Debunking the Myth: Kubernetes Storage is Hard

by Saad Ali

Senior Software Engineer from Google talks about storage concepts in Kubernetes



https://www.youtube.com/watch?v=169w6QlWhmo

More information

- CNCF Serverless working group - https://github.com/cncf/wg-serverless
- #serverless Slack channel - https://slack.cncf.io/
- Knative - https://cloud.google.com/knative/
- openFaaS - https://github.com/openfaas/faas , https://www.openfaas.com/
- Grafana Loki - https://grafana.com/loki , https://github.com/grafana/loki
- CNCF Youtube Channel - https://www.youtube.com/channel/UCvqbFHwN-nwalWPjPUKpvTA


KubeCon
CloudNativeCon
Hall 8.1
Additional Meal Seating
Breakouts
Rooms G1-G4
CC8 Rooms
Quiet Room
Hall 7
Sponsor
Showcase

André

Helm 3,...

Helm 3

- bye bye Tiller
- some CLI changes
- Namespaced releases
- Chart dependencies are in chart.yaml now
- No autogenerated names by default
- OCI repos
- optional LUA templating
- Alpha 1 available now -> try it! :)
- Helm Summit 11-12 September 2019 in Amsterdam


3
Farewell, Tiller
- simpler & more flexible architecture, security, & upgrades
- now using Kubernetes API directly
- rendering Charts client-side; storing in release
- lowering the barrier of entry for contributors
@bridgetkromhout
#kubecon
@jldeen

Helm 3

- bye bye Tiller
- some CLI changes
- Namespaced releases
- Chart dependencies are in chart.yaml now
- No autogenerated names by default
- OCI repos
- optional LUA templating
- Alpha 1 available now -> try it! :)
- Helm Summit 11-12 September 2019 in Amsterdam


3
CLI changes
helm delete —-> helm uninstall
helm inspect —-> helm show
helm fetch —-> helm pull
--purge is now default (to override: helm uninstall --keep-history)
@bridgetkromhout
#kubecon
@jldeen

Helm 3

- bye bye Tiller
- some CLI changes
- Namespaced releases
- Chart dependencies are in chart.yaml now
- No autogenerated names by default
- OCI repos
- optional LUA templating
- Alpha 1 available now -> try it! :)
- Helm Summit 11-12 September 2019 in Amsterdam


3
Namespaces changes
- defaulting to a single namespace for a release and objects (no longer using the tiller namespace)
- creating resources in the same namespace as the release
- this is why you can't upgrade a helm 2 release with helm 3 (migration story is in progress!)
@bridgetkromhout
#kubecon
@jldeen

Helm 3

- bye bye Tiller
- some CLI changes
- Namespaced releases
- Chart dependencies are in chart.yaml now
- No autogenerated names by default
- OCI repos
- optional LUA templating
- Alpha 1 available now -> try it! :)
- Helm Summit 11-12 September 2019 in Amsterdam


3
Chart dependency management
- Old style: requirements.yaml and requirements.lock
- New style: Chart.yaml and Chart.lock
(breaking change if you use helm dependency subcommands)
@bridgetkromhout
#kubecon
@jldeen

Helm 3

- bye bye Tiller
- some CLI changes
- Namespaced releases
- Chart dependencies are in chart.yaml now
- No autogenerated names by default
- OCI repos
- optional LUA templating
- Alpha 1 available now -> try it! :)
- Helm Summit 11-12 September 2019 in Amsterdam


--generate-name
- Old style: auto-generated unless overridden
- New style: error unless auto-generation requested
@bridgetkromhout
#kubecon
@jldeen

Helm 3

- bye bye Tiller
- some CLI changes
- Namespaced releases
- Chart dependencies are in chart.yaml now
- No autogenerated names by default
- OCI repos
- optional LUA templating
- Alpha 1 available now -> try it! :)
- Helm Summit 11-12 September 2019 in Amsterdam


3
- iterating on Helm Chart Repository API
- working towards compat with OCI standard
(eventually: pluggable auth! novel artifact types!
host on ACR!)
@bridgetkromhout
#kubecon
@jldeen

Helm 3

- bye bye Tiller
- some CLI changes
- Namespaced releases
- Chart dependencies are in chart.yaml now
- No autogenerated names by default
- OCI repos
- optional LUA templating
- Alpha 1 available now -> try it! :)
- Helm Summit 11-12 September 2019 in Amsterdam


3
Lua, in lieu of go templates (optionally)
Helm 2: late-binding error from k8s with a failed release.
Helm 3: “Can I actually create these resources?” could have a pre-install hook, see if “can I?” is true or “Do these CRDs exist? If yes, don’t create them.”
@bridgetkromhout
#kubecon
@jldeen

Helm 3

- bye bye Tiller
- some CLI changes
- Namespaced releases
- Chart dependencies are in chart.yaml now
- No autogenerated names by default
- OCI repos
- optional LUA templating
- Alpha 1 available now -> try it! :)
- Helm Summit 11-12 September 2019 in Amsterdam


Helm 3 alpha releases
alpha.1: tillerless, library charts, secrets storage backend, beginning OCI groundwork (chart commands)
alpha.2: Lua, XDG, start OCI integration (package, install, dependency update)
alpha.3: refactoring upgrade logic (maybe with server-side apply, 3-way diffs, etc), plugins written in Lua
@bridgetkromhout
#kubecon
@jldeen

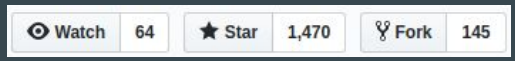
Helm 3

- Finally… bye bye Tiller!
- some CLI changes
- Namespaced releases
- Chart dependencies are in chart.yaml now
- No autogenerated names by default
- OCI repos
- optional LUA templating
- Alpha 1 available now -> try it! :)
- Helm Summit 11-12 September 2019 in Amsterdam


CFP till 31 May!
HELM
SUMMIT
11-12 September 2019, in Amsterdam
https://helm.sh/blog/helm-summit-eu-2019/
@bridgetkromhout
#kubecon
@jdeen

Alfred

buildkit, Pod Ready ++, Operators



buildkit - like `docker build` but fancy and complicated

Features stolen from https://github.com/moby/buildkit

- Automatic garbage collection
- Extendable frontend formats
- Concurrent dependency resolution
- Efficient instruction caching
- Build cache import/export
- Nested build job invocations
- Distributable workers
- Multiple output formats
- Pluggable architecture
- Execution without root privileges

Features (picked from the presentation slides)

- integration with Docker 18.06:

```
$ export DOCKER_BUILDKIT=1
$ docker build ...
```

- mount build secrets without baking them into the image:

```
RUN --mount=type=secret,id=aws,
target=/root/.aws/credentials \
aws s3 cp s3://... ...
```

- reuse caches, e.g. for package manager or go:

```
RUN --mount=type=cache,target=/root/.cache \
go build
```

buildkit - like `docker build` but fancy and complicated

Performance

- 2x faster for fresh build of https://github.com/moby/moby/blob/master/Dockerfile
- 7x faster for repeated build without changes
- 2x faster for repeated build with changes
- 9x faster for fresh build with remote cache

buildkit - like `docker build` but fancy and complicated

Of course you can run it on Kubernetes

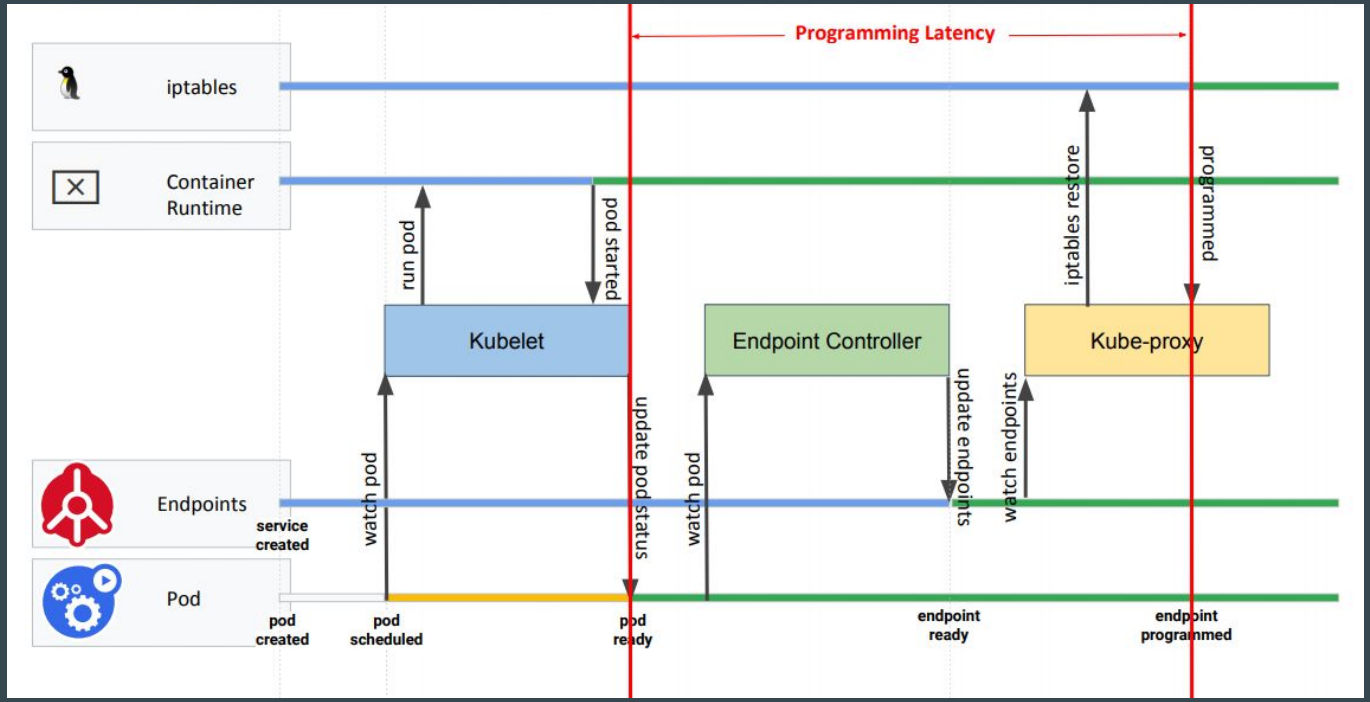
- needs to have seccomp and AppArmor disabled
- different strategies possible for deployment (e.g. StatefulSet)
  - use consistent hashing to make optimal use of local build caches
  - effectiveness depends on your type of workload
- works with Tekton for building images

Presentation slides:

https://kccnceu19.sched.com/event/MPX5/building-images-efficiently-and-securely-on-kubernetes-with-buildkit-akihiro-suda-ntt-corporation

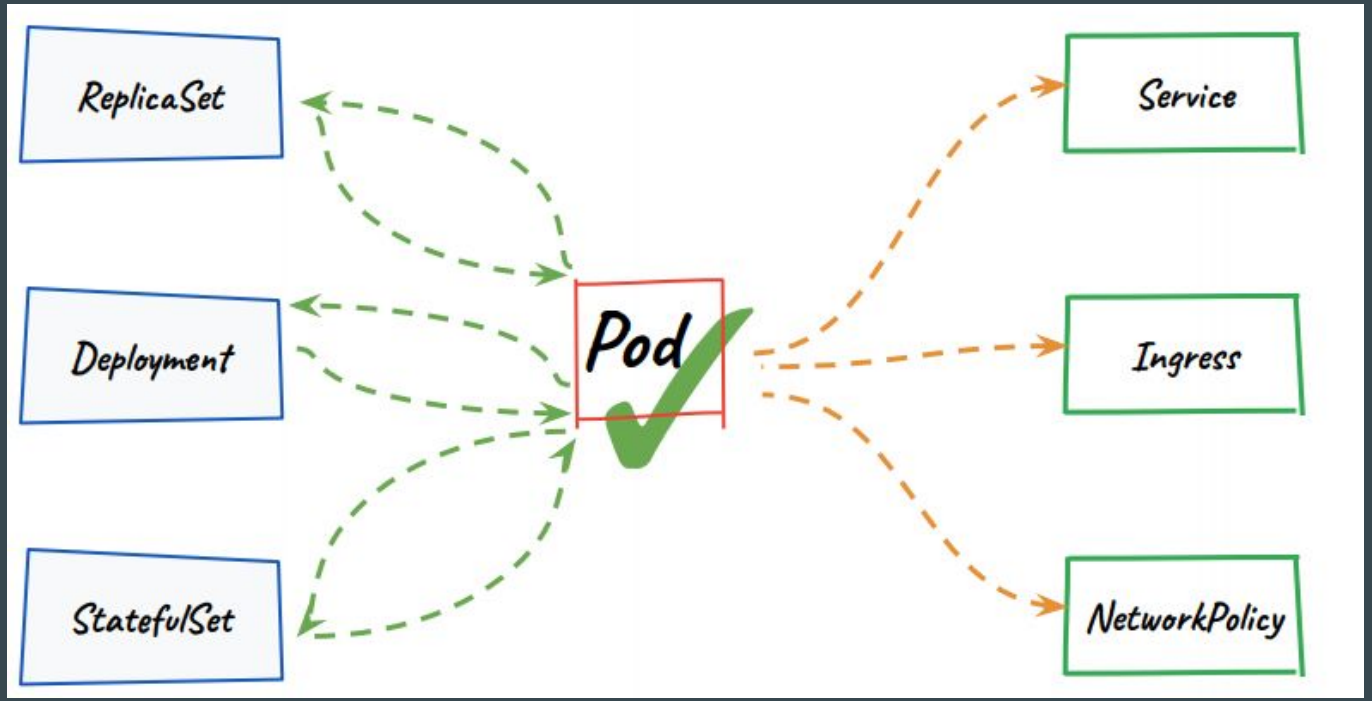
Pod Ready++ or Pod Readiness Gates

Problem statement



Pod Ready++ or Pod Readiness Gates

Problem statement



Pod Ready++ or Pod Readiness Gates

Usage

```
Kind: Pod
...
spec:
  readinessGates:
  - conditionType: readiness-gate-a
  - conditionType: readiness-gate-b
...
status:
  conditions:
  - lastTransitionTime: 2018-01-01T00:00:00Z
    status: "False"
    type: Ready
  - lastTransitionTime: 2018-01-01T00:00:00Z
    status: "False"
    type: readiness-gate-a
  - lastTransitionTime: 2018-01-01T00:00:00Z
    status: "True"
    type: readiness-gate-b
...
```

```
$ kubectl get pod -o wide

NAME      READY   STATUS    RESTARTS   AGE    IP            NODE   NOMINATED NODE   READINESS GATES
pod1      1/1     Running   0          11d    10.64.1.96    node   <none>           1/1
pod2      2/2     Running   0          11d    10.64.1.95    node   <none>           2/2
pod3      2/2     Running   0          175m   10.64.2.64    node   <none>           <none>
pod4      3/3     Running   0          175m   10.64.3.85    node   <none>           <none>
```

Containers

Readiness Gates

- transparent integration: have your controllers automatically:
  - inject `readinessGates` into pods with `MutatingAdmissionWebhook`
  - patch condition status in pods

Presentation:

https://kccnceu19.sched.com/event/MPaQ/ready-a-deep-dive-into-pod-readiness-gates-for-service-health-management-minhan-xia-google-ping-zou-intuit

Operators

What's an operator?

- Operator = `CustomResourceDefinition` + Controller
- custom resource definition:
  - OpenAPI spec of arbitrary custom resources (currently v3, very powerful; will be restricted going forward)
  - can be "installed" in your Kubernetes cluster by `kubectl apply`
  - don't do anything on their own
  - starting Kubernetes 1.14: OpenAPI specs for CRDs will be published by API server
- controller:
  - watches (custom) resources
  - evaluates actual and desired state
  - acts to make both states match

Operators

How can you write your own?

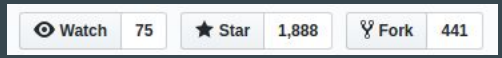
- from scratch (have fun)
- attended a Google-hosted workshop for kubebuilder

  https://github.com/kubernetes-sigs/kubebuilder

  - currently being worked on for v2
  - Documentation is in a kind-of-meh state right now
- had a look at operator-sdk afterwards

  https://github.com/operator-framework/operator-sdk

- both very similar, basically CLI tools for scaffolding project structure, generating:
  - Go type definitions for your APIs → later generate CRDs (YAML files) out of these with CLI
  - controller prototype implementation
- both use `controller-runtime` (official Go library from Kubernetes project)

Operators

How can you write your own?

- general structure

  `func (r *ReconcileMyCustomResource) Reconcile(request reconcile.Request) (reconcile.Result, error) {…}`
  - will be invoked by controller runtime whenever it deems it necessary
- be careful to properly handle:
  - requeuing of reconciling requests (e.g. when waiting for some external action to complete or in case of error)
  - deletion / cleanup of resources created by your operator
    - internal resources (set `ownerReference` on your created resources to enforce cascaded deletion)
    - external resources (add `finalizers` to your custom resource before acting upon it)

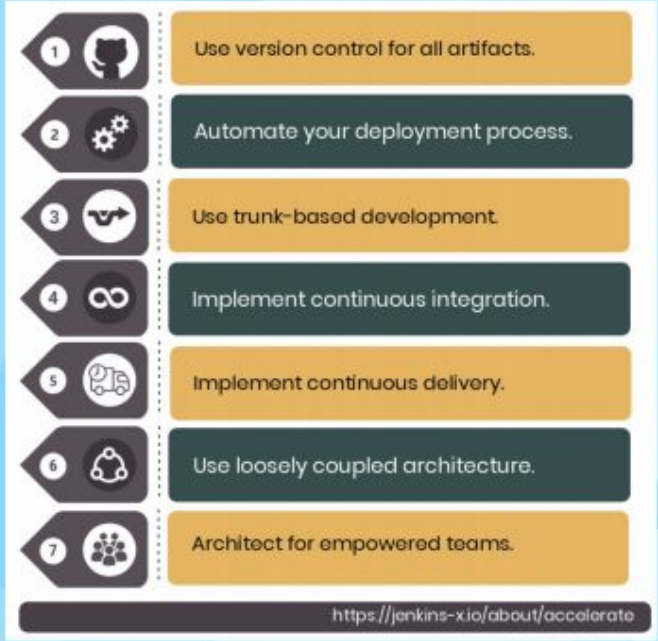
Presentations:

https://kccnceu19.sched.com/event/MPZz/openapi-specs-towards-native-user-experience-of-crds-stefan-schimanski-red-hat

https://kccnceu19.sched.com/event/MPaN/keep-the-space-shuttle-flying-writing-robust-operators-illya-chekrygin-upbound (funny guy, watch the recording)


Peter
CI/CD - Jenkins X, Tekton, ...
MultiCluster network with WireGuard
KubeCon
CloudNativeCon
Europe 2019
THE HUB
The Hub
Play
Station
Fast Food

Jenkins X

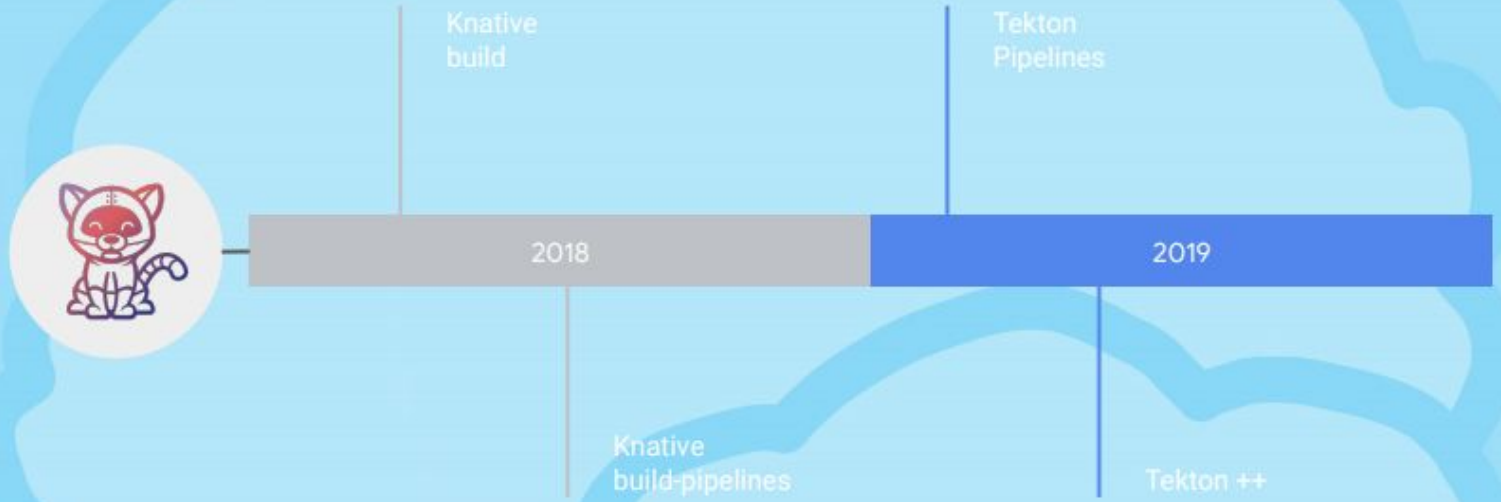


- Developer experience for Kubernetes
- Build traditional and modern cloud native workloads
- Create new or import existing applications onto Kubernetes
- Automated CI/CD
- Where possible use capabilities recommended from Accelerate
- ...

@jdrawlings

@bobcatwilson

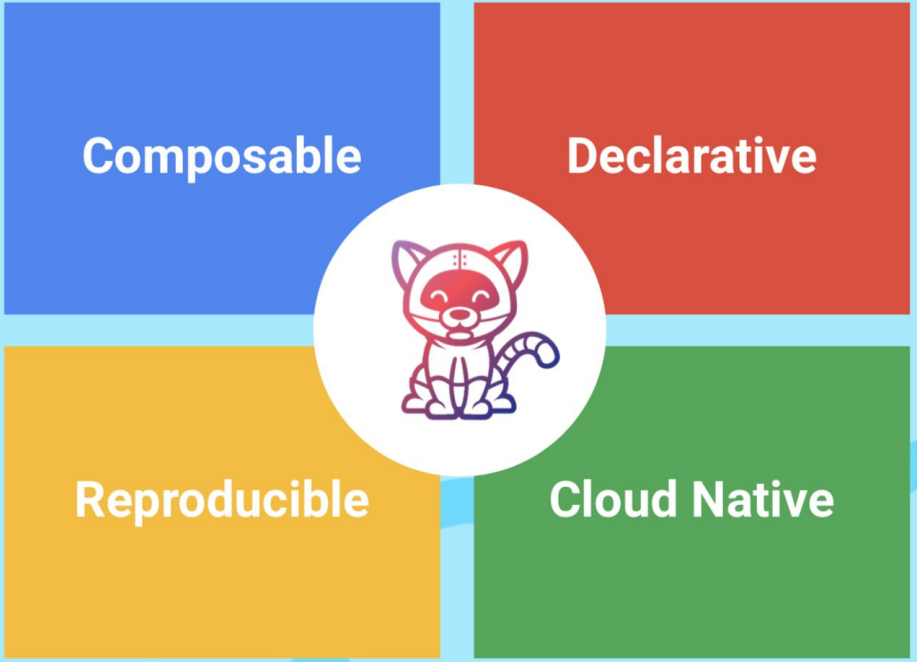
The Tekton Story



@jdrawlings

@bobcatwilson

Tekton Goals



@jdrawlings

@bobcatwilson

https://static.sched.com/hosted_files/kccnceu19/98/Modern%20CI_CD%20with%20Tekton%20and%20Prow%20Automated%20via%20Jenkins%20X.pdf

Jenkins X ❤ Tekton Pipelines

- Leverages Prow to trigger Tekton Pipelines
- Next Gen Pipeline jenkins-x.yml
- Build Packs to maintain common pipelines but with extensions
- Dogfooding with Jenkins X has dramatically improved
  - Stability
  - Performance
  - Reduced cloud costs

jenkins-x.yml

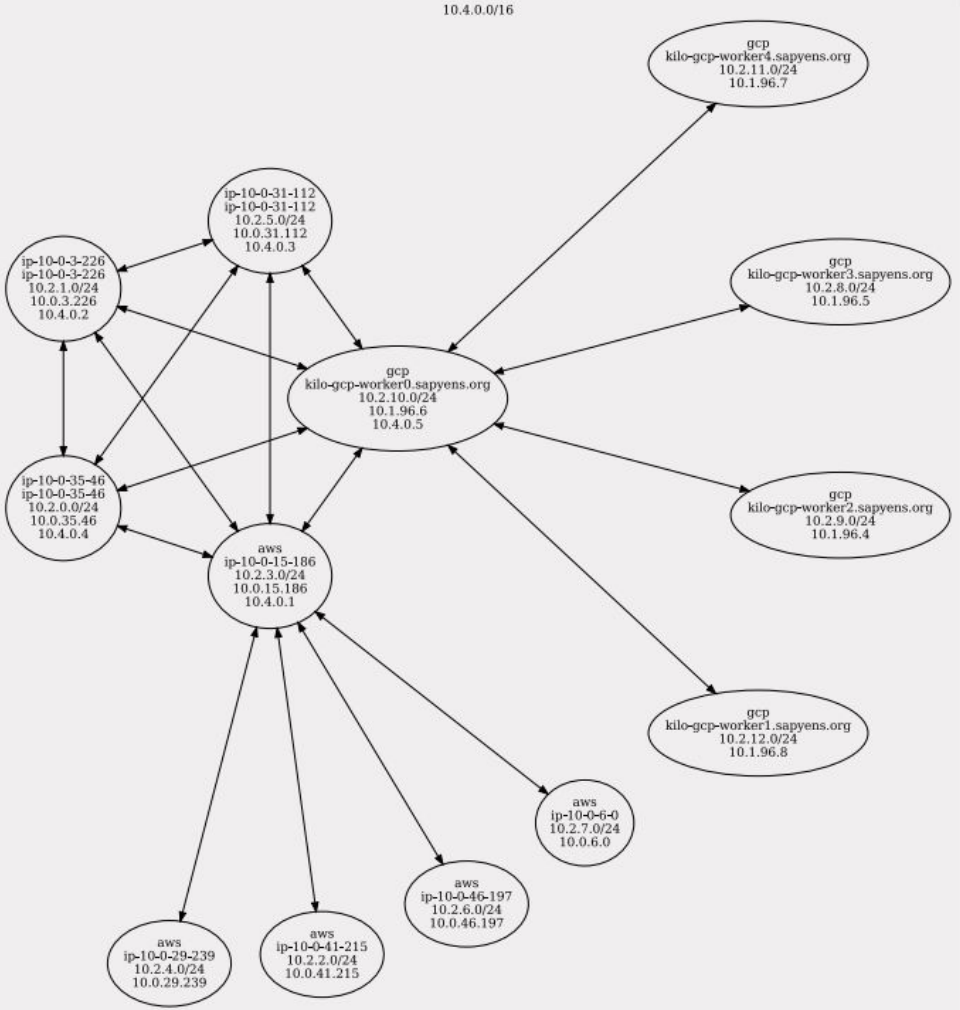
```
buildPack: none
pipelineConfig:
  pipelines:
    pullRequest:
      pipeline:
        stages:
          - name: ci
            environment:
              - name: GIT_COMMITTER_EMAIL
                value: jenkins-x@googlegroups.com
              - name: GIT_COMMITTER_NAME
                value: jenkins-x-bot
            steps:
            - name: build-binary
              image: docker.io/golang:1.11.5
              command: make
              args: ['linux']

            - name: build-and-push-image
              image: gcr.io/kaniko-project/executor
              command: /kaniko/executor
              args:
              - --dockerfile=/workspace/source/Dockerfile
              - --destination=docker.io/jenkinsxio/jx:${inputs
```


Meal Seating

BUILDING MULTI-CLOUD CLUSTERS WITH WIREGUARD

Lucas Servén Marín



github.com/squat/kilo

README.md

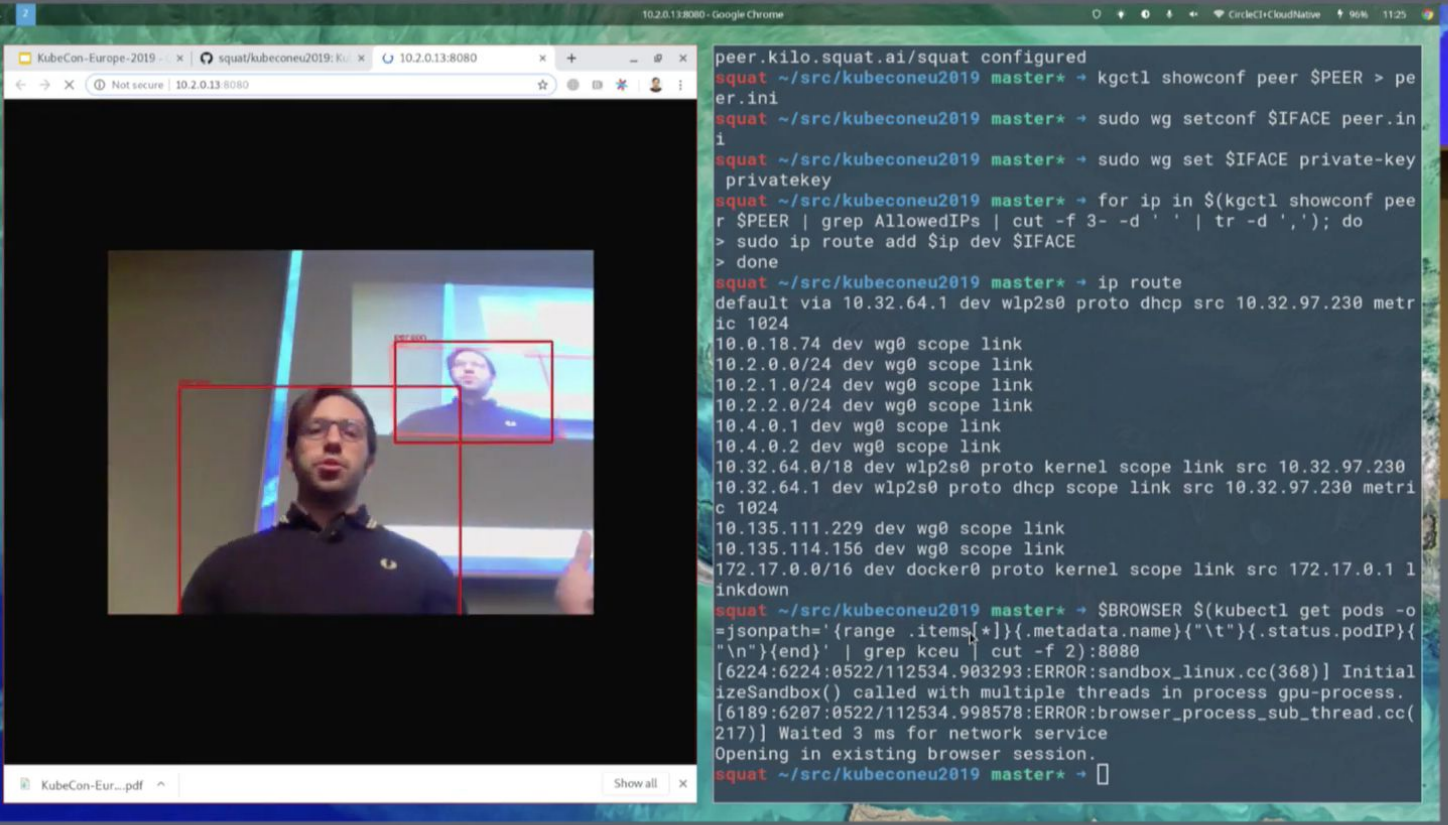
KubeCon EU 2019

This repository contains the demo code for my KubeCon EU 2019 talk about building multi-cloud clusters using WireGuard. In this demo we will imagine we are a company like Nest that is running object detection processes on video captured by IoT devices. We will run a web-app in the cloud connected to a GPU-powered image detection and labeling service in a different public cloud provider. The web-app will stream video from the IoT device over a WireGuard connection to keep the data safe.

Specifically we will:

- create a multi-cloud cluster that spans between DigitalOcean and AWS
- create some GPU workers in AWS
- run the workload that captures video in a device on the edge, e.g. your host capturing video from the webcam
- peer the workload with the cluster in the cloud
- run a computer vision process on the video captured by the edge workload
- accelerate the computer vision using GPUs in AWS.

Demo Setup: https://github.com/squat/kubeconeu2019



Video: https://www.youtube.com/watch?v=iPz_DAOOCKA&list=PLj6h78yzYM2PpmMAnvpvsnR4c27wJePh3&index=148&t=0s


Christian
Storage, Auth, Service Meshes

Storage

Keynote: Debunking the Myth: Kubernetes Storage is Hard - Saad Ali, Senior Software Engineer, Google

https://www.youtube.com/watch?v=169w6QlWhmo

Build a Kubernetes Based Cloud Native Storage Solution From Scratch (Longhorn)
https://kccnceu19.sched.com/event/MPXK/build-a-kubernetes-based-cloud-native-storage-solution-from-scratch-sheng-yang-rancher-labs

Auth

Smarter Kubernetes Access Control: A Simpler Approach to Auth

https://kccnceu19.sched.com/event/MPdi/smarter-kubernetes-access-control-a-simpler-approach-to-auth-rob-scott-reactiveops

Sharing is Caring: Your Kubernetes Cluster, Namespaces, and You

https://kccnceu19.sched.com/event/MPXl/sharing-is-caring-your-kubernetes-cluster-namespaces-and-you-amy-chen-eryn-muetzel-vmware

Service Meshes

Sponsored Keynote: Democratizing Service Mesh on Kubernetes (Service mesh interface)

https://kccnceu19.sched.com/event/MRz7/sponsored-keynote-democratizing-service-mesh-on-kubernetes-gabe-monroy-lead-product-manager-microsoft-azure-container-compute

Istio, We Have a Problem! Understanding and Fixing Bugs with a Service-Mesh

https://kccnceu19.sched.com/event/MPfG/istio-we-have-a-problem-understanding-and-fixing-bugs-with-a-service-mesh-david-gageot-google

JustFootball's Journey to gRPC + Linkerd in Production

https://kccnceu19.sched.com/event/MPf4/justfootballs-journey-to-grpc-linkerd-in-production-ben-lambert-justfootball-kevin-lingerfelt-buoyant

Danilo

Harbor

Links

Flickr Pictures
https://www.flickr.com/photos/143247548@N03/albums/72157707188120301

YouTube Playlist
https://www.youtube.com/playlist?list=PLj6h78yzYM2PpmMAnvpvsnR4c27wJePh3

Schedule
https://events.linuxfoundation.org/events/kubecon-cloudnativecon-europe-2019/schedule/