kubernetes for data scientists

an introduction
17 May 2018

louis taylor
jetstack

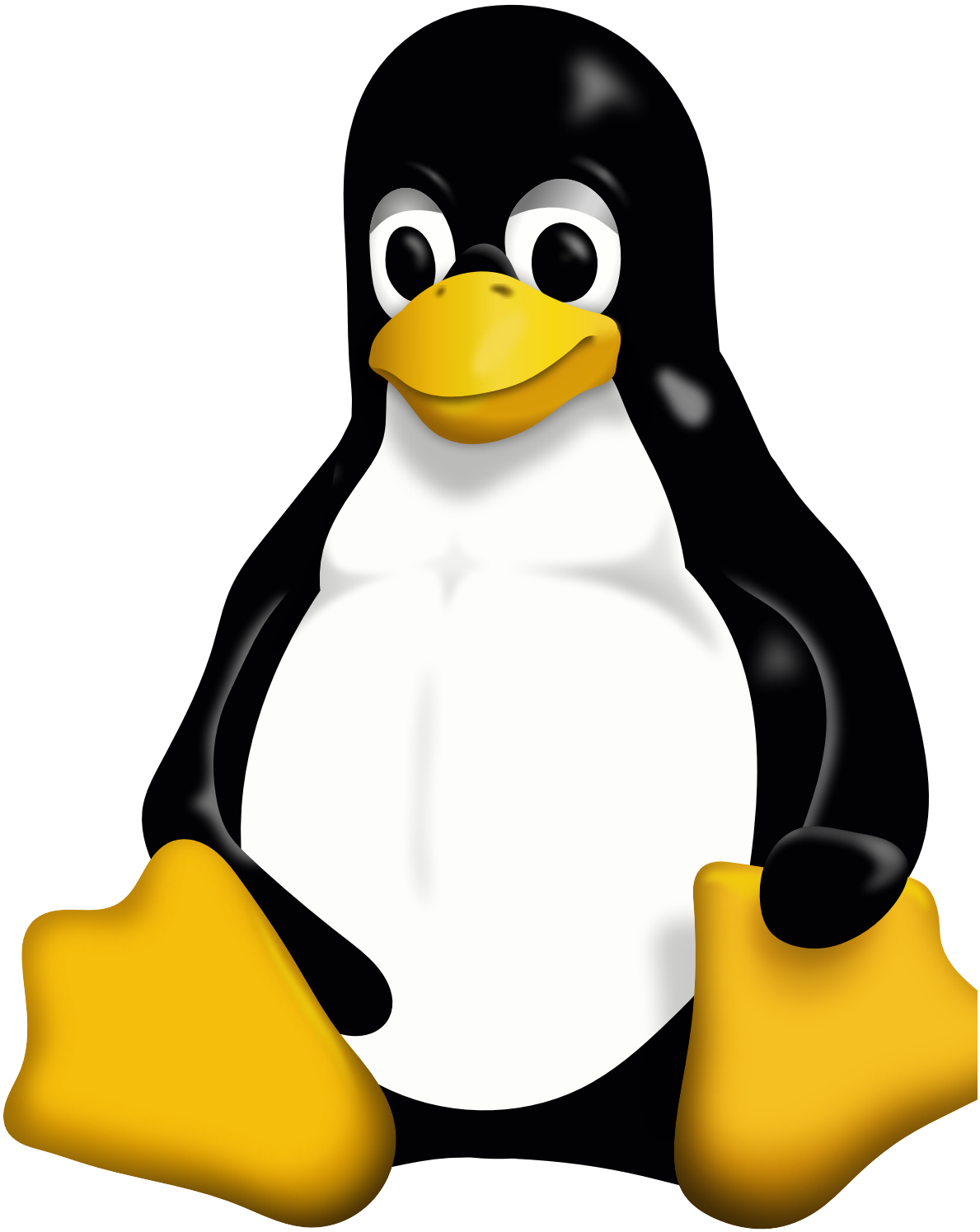
outline

- what are linux containers?
- what is kubernetes?
- how does one use kubernetes for data related tasks?
- summary?

myself

- work at jetstack
- does kubernetes stuff
- used to do robotic sailing stuff

linux containers



example

reproducibility

actually pretty simple

- https://github.com/kragniz/omochabako/
- pulls and runs docker images in ~500 lines of python

borg

Large-scale cluster management at G

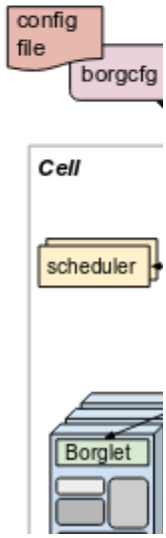
Abhishek Verma† Luis Pedrosa‡ Madhuka
David Oppenheimer Eric Tune John

Google Inc.

Abstract

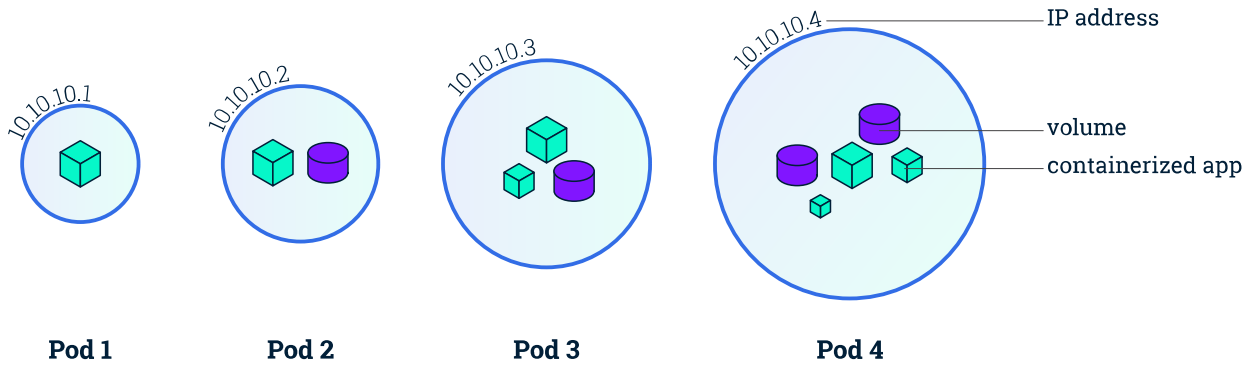
Google's Borg system is a cluster manager that runs hundreds of thousands of jobs, from many thousands of different applications, across a number of clusters each with up to tens of thousands of machines.

It achieves high utilization by combining admission control, efficient task-packing, over-commitment, and machine sharing with process-level performance isolation. It supports high-availability applications with runtime features that minimize fault-recovery time, and scheduling policies that reduce the probability of correlated failures. Borg simplifies life for its users by offering a declarative job specification language, name service integration, real-time job monitor-



kubernetes

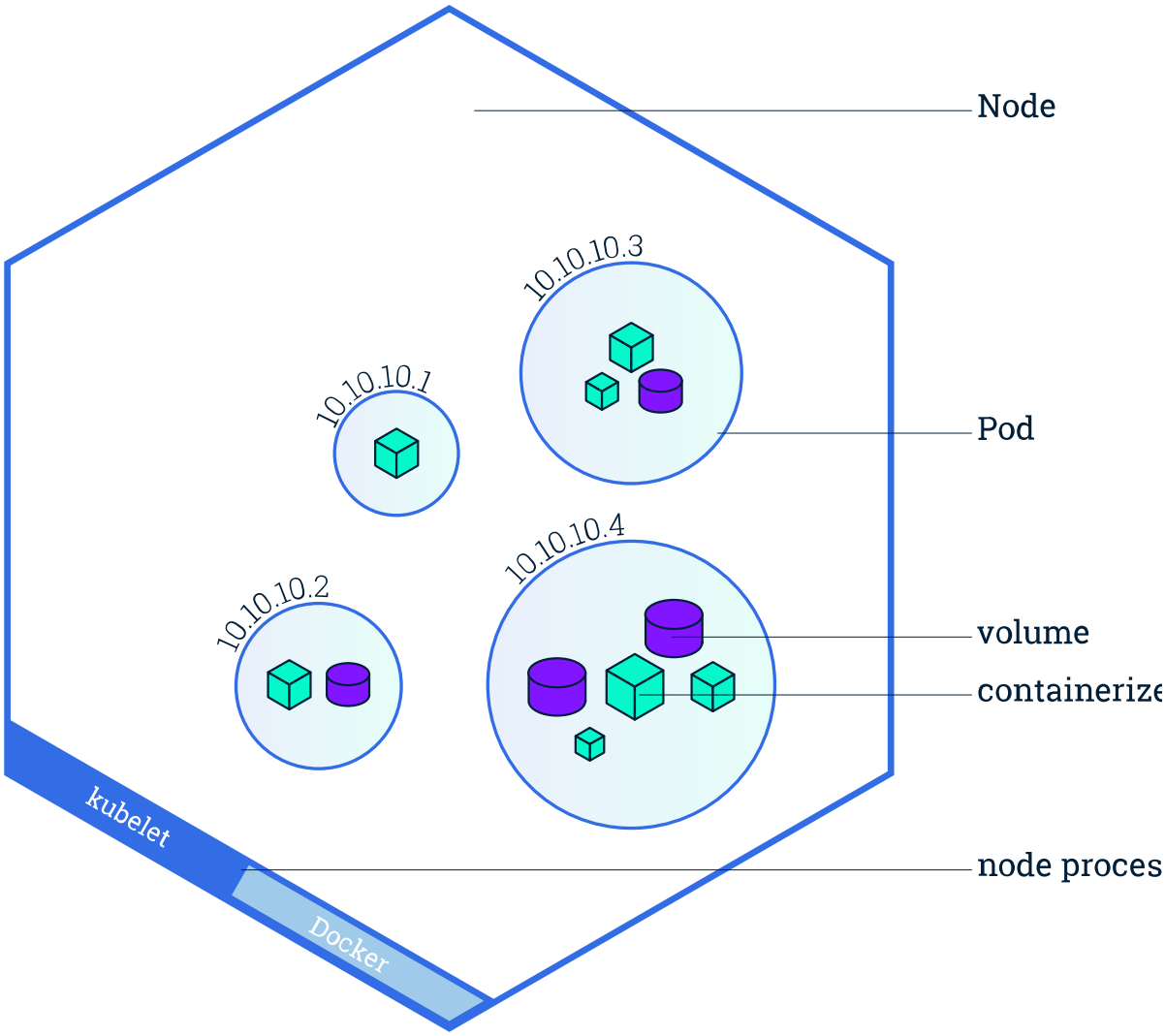
pods


10.10.10.1
10.10.10.2
10.10.10.3
10.10.10.4
IP address
volume
containerized app
Pod 1
Pod 2
Pod 3
Pod 4

pods

```
apiVersion: v1
kind: Pod
metadata:
  name: cuda-vector-add
spec:
  restartPolicy: OnFailure
  containers:
    - name: cuda-vector-add
      image: "k8s.gcr.io/cuda-vector-add:v0.1"
      resources:
        requests:
          memory: 2Gi
          cpu: "2"
          nvidia.com/gpu: 1
        limits:
          memory: 2Gi
          cpu: "2"
          nvidia.com/gpu: 1
```

nodes


Node
10.10.10.3
10.10.10.1
Pod
10.10.10.4
10.10.10.2
volume
containerize
kubelet
Docker
node proces

not just pods

controllers

```
1  get current state of thing
2  get desired state of thing
3  make changes to thing to move towards desired state
```

deployments

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: cuda-vector-add-deployment
  labels:
    app: cuda-vector-add
spec:
  replicas: 3
  selector:
    matchLabels:
      app: cuda-vector-add
  template:
    metadata:
      labels:
        app: cuda-vector-add
    spec:
      containers:
        - name: cuda-vector-add
          image: "k8s.gcr.io/cuda-vector-add:v0.1"
          resources:
            requests:
              memory: 2Gi
              cpu: "2"
              nvidia.com/gpu: 1
            limits:
              memory: 2Gi
              cpu: "2"
              nvidia.com/gpu: 1
```

not just for web developers

kubeflow

kubeflow

- https://github.com/kubeflow/kubeflow
- jupyterhub
- tensorflow

kubeflow

```
apiVersion: "kubeflow.org/v1alpha1"
kind: "TFJob"
metadata:
  name: "example-job"
spec:
  replicaSpecs:
    - replicas: 1
      tfReplicaType: MASTER
      template:
        spec:
          containers:
            - image: gcr.io/tf-on-k8s-dogfood/tf_sample:dc944ff
              name: tensorflow
    - replicas: 1
      tfReplicaType: WORKER
      template:
        spec:
          containers:
            - image: gcr.io/tf-on-k8s-dogfood/tf_sample:dc944ff
              name: tensorflow
    - replicas: 2
      tfReplicaType: PS
```

kubeflow

- https://www.katacoda.com/kubeflow

dask

- https://dask.pydata.org/
- https://github.com/dask/dask-kubernetes

```
from dask_kubernetes import KubeCluster

cluster = KubeCluster.from_yaml('worker-spec.yml')
cluster.scale_up(10)  # specify number of nodes explicitly

cluster.adapt(minimum=1, maximum=100)  # or dynamically scale ba
```

lots of others

- polyaxon https://github.com/polyaxon/polyaxon
- riseml https://riseml.com/

summary

- kubernetes is a useful building block
- more tools will build on top of it in the future
- maybe consider building your tooling around it

Thank you

louis taylor
jetstack

💩@kragniz.eu (mailto:%f0%9f%92%a9@kragniz.eu)
@kragniz (http://twitter.com/kragniz)
https://kragniz.eu (https://kragniz.eu)
https://github.com/kragniz (https://github.com/kragniz)