SCALEOUT

Configuration Management Tools

Max Andersson

Kubernetes

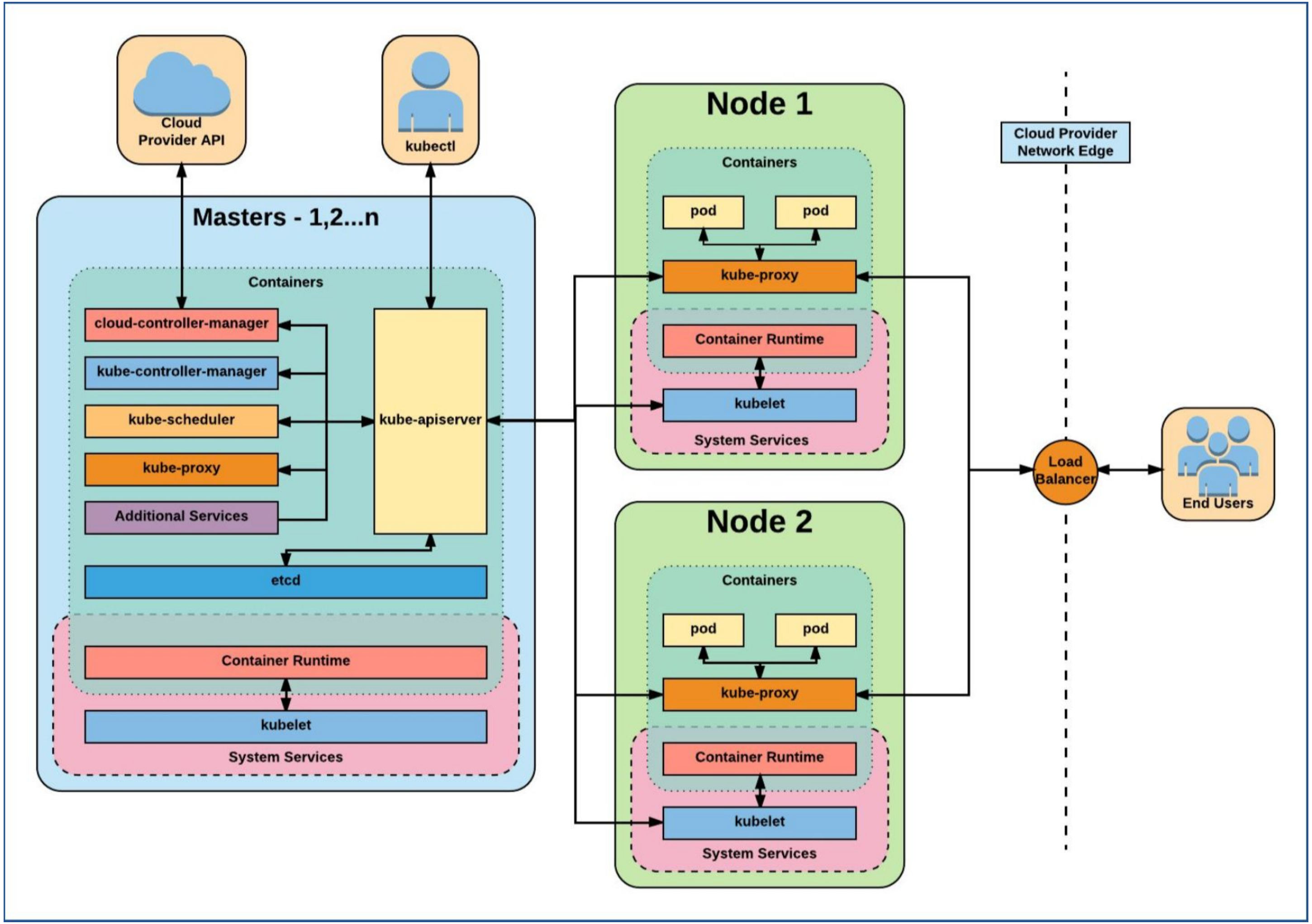
- What is kubernetes
    - Container Management environment
    - Microservices Platform
    - Provides Orchestration of compute, networking & storage on the behalf of the user
    - You can think of it as the kernel for distributesd systems

Kubernetes

- Goals
  - Portable
  - General-purpose
  - Flexible
  - Extensible
  - Automatable
  - Advance the state of art

Kubernetes

- What kubernetes provides
    - Master : A Control Plane (with an api, for custom interactions)
        - kube-apiserver
        - Kube-controller-manager
        - Kube-scheduler
    - Nodes - Workers of the cluster
        - Kubelet
        - Kube-proxy


Cloud Provider API
kubectl
Masters - 1,2...n
Containers
cloud-controller-manager
kube-controller-manager
kube-scheduler
kube-proxy
Additional Services
kube-apiserver
etcd
Container Runtime
kubelet
System Services
Node 1
Containers
pod
pod
kube-proxy
Container Runtime
kubelet
System Services
Node 2
Containers
pod
pod
kube-proxy
Container Runtime
kubelet
System Services
Cloud Provider Network Edge
Load Balancer
End Users

Kubernetes

- Cluster control plane (or master)
  - Supports
    - Self-healing
    - Scaling
    - Updates
    - Termination
  - Orchestration of workloads
    - ReplicaSet
    - Deployment
    - Job
    - CronJob
    - DaemonSet
    - StatefulSet

Kubernetes

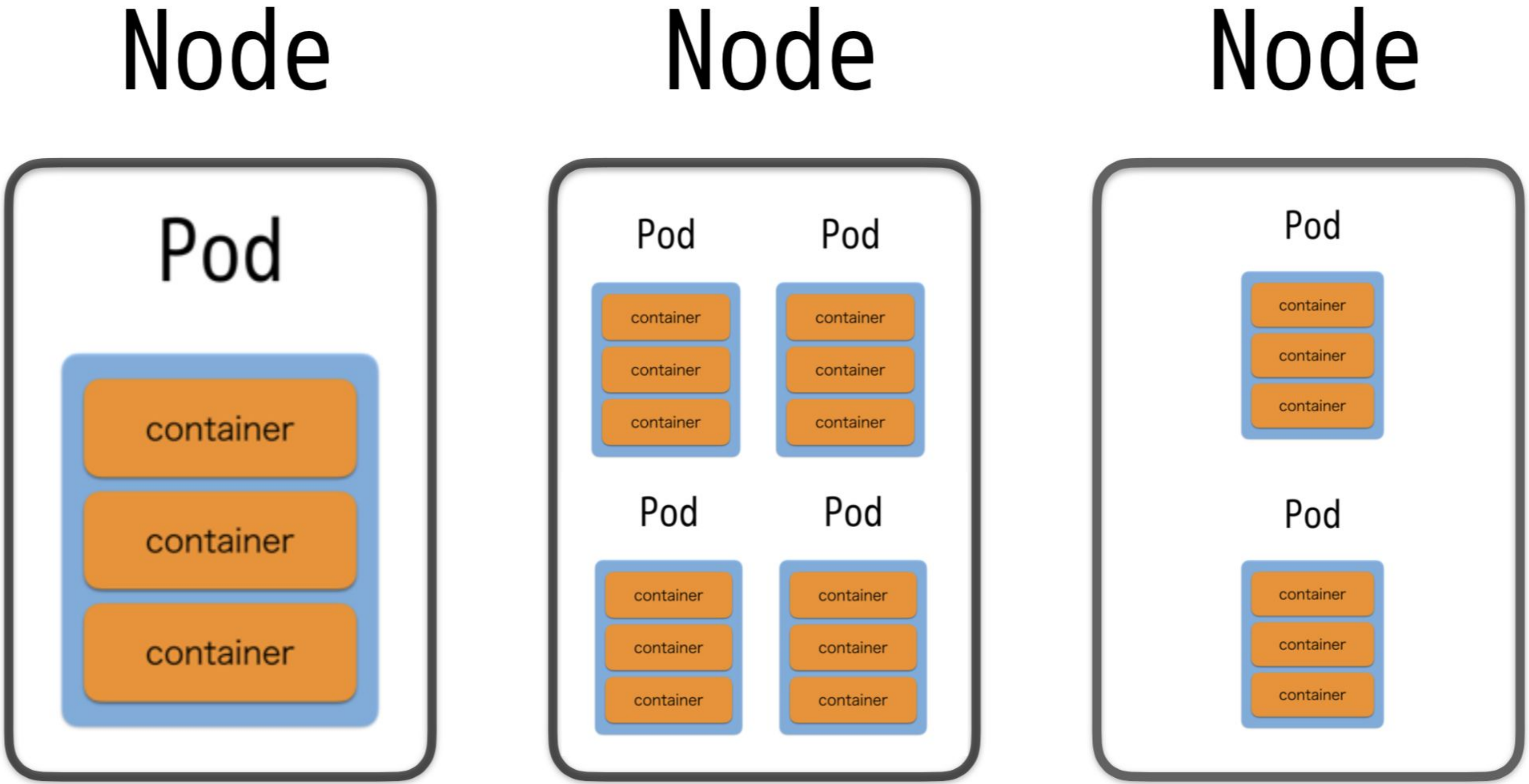
- Kubernetes Objects
  - Pods
  - Services
  - Volumes
  - Namespace
- Kubetnetes Controllers (also objects)
  - ReplicaSet
  - Deployment
  - StatefulSet
  - DaemonSet
  - Job
- **Object Metadata**
  - **Label**
  - **Annotations**

```
"metadata": {
  "annotations": {
    "key1" : "value1",
    "key2" : "value2"
  }
  "labels": {
    "key1" : "value1",
    "key2" : "value2"

    }


  }
```

Kubernetes - Pod

- Kubernetes Objects
  - **Pods**
  - Services
  - Volumes
  - Namespace
- Kubetnetes Controllers (also objects)
  - ReplicaSet
  - Deployment
  - StatefulSet
  - DaemonSet
  - Job
- Object Metadata
  - Label
  - Annotations

```
apiVersion: v1
kind: Pod
metadata:
  name: cuda-test
spec:
  containers:
    - name: cuda-test
      image:
"k8s.gcr.io/cuda-vector-add:v0.1"
      resources:
        limits:
          nvidia.com/gpu: 1
  nodeSelector:

      accelerator:
  nvidia-tesla-p100
```



Node

Pod

container

container

container

Node

Pod

container

container

container

Pod

container

container

container

Pod

container

container

container

Pod

container

container

container

Node

Pod

container

container

container

Pod

container

container

container

Kubernetes - Services

- Kubernetes Objects
  - Pods
  - **Services**
  - Volumes
  - Namespace
- Kubetnetes Controllers (also objects)
  - ReplicaSet
  - Deployment
  - StatefulSet
  - DaemonSet
  - Job
- Object Metadata
  - Label
  - Annotations

```
kind: Service
apiVersion: v1
metadata:
  name: my-service
spec:
  selector:
    app: MyApp
  ports:
  - name: http
    protocol: TCP
    port: 80
    targetPort: 9376
  - name: https
    protocol: TCP
    port: 443

      targetPort: 9377
```

Kubernetes - Volumes

- Kubernetes Objects
  - Pods
  - Services
  - **Volumes**
  - Namespace
- Kubetnetes Controllers (also objects)
  - ReplicaSet
  - Deployment
  - StatefulSet
  - DaemonSet
  - Job
- Object Metadata
  - Label
  - Annotations

```
apiVersion: v1
kind: Pod
metadata:
  name: test-cinder
spec:
  containers:
  - image:
k8s.gcr.io/test-webserver
    name: test-cinder-container
    volumeMounts:
    - mountPath: /test-cinder
      name: test-volume
  volumes:
  - name: test-volume
    # This OpenStack volume must
already exist.
    cinder:
      volumeID: <volume-id>

      fsType: ext4
```

Kubernetes - Namespace

- Kubernetes Objects
  - Pods
  - Services
  - Volumes
  - **Namespace**
- Kubetnetes Controllers (also objects)
  - ReplicaSet
  - Deployment
  - StatefulSet
  - DaemonSet
  - Job
- Object Metadata
  - Label
  - Annotations

```
$ kubectl get namespace
```

```
NAME          STATUS    AGE
default       Active    1d
kube-system   Active    1d
kube-public   Active    1d
```

Kubernetes - ReplicaSet

- Kubernetes Objects
  - Pods
  - Services
  - Volumes
  - Namespace
- Kubetnetes Controllers (also objects)
  - **ReplicaSet**
  - Deployment
  - StatefulSet
  - DaemonSet
  - Job
- Object Metadata
  - Label
  - Annotations

```
apiVersion: apps/v1
kind: ReplicaSet
metadata:
  name: frontend
  labels:
    app: guestbook
    tier: frontend
spec:
  # modify replicas according to
your case
  replicas: 3
  selector:
    matchLabels:
      tier: frontend
  template:
    metadata:
      labels:
        tier: frontend
    spec:
      containers:
      - name: php-redis
          image:
 gcr.io/google_samples/gb-fronte
 nd:v3
```

Kubernetes - Deployment

- Kubernetes Objects
  - Pods
  - Services
  - Volumes
  - Namespace
- Kubetnetes Controllers (also objects)
  - ReplicaSet
  - **Deployment**
  - StatefulSet
  - DaemonSet
  - Job
- Object Metadata
  - Label
  - Annotations

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
  labels:
    app: nginx
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:1.7.9
        ports:
          - containerPort: 80
```

Kubernetes - StatefulSet

- Kubernetes Objects
  - Pods
  - Services
  - Volumes
  - Namespace
- Kubetnetes Controllers (also objects)
  - ReplicaSet
  - Deployment
  - **StatefulSet**
  - DaemonSet
  - Job
- Object Metadata
  - Label
  - Annotations

```
apiVersion: apps/v1
kind: StatefulSet
metadata:
  name: web
spec:
  selector:
    matchLabels:
      app: nginx # has to match
.spec.template.metadata.labels
  serviceName: "nginx"
  replicas: 3 # by default is 1
  template:
    metadata:
      labels:
        app: nginx # has to match
.spec.selector.matchLabels
    spec:
      containers:
      - name: nginx
        image:
k8s.gcr.io/nginx-slim:0.8
        ports:
        - containerPort: 80
          name: web
        volumeMounts:
        - name: www
          mountPath:
/usr/share/nginx/html
  volumeClaimTemplates:
  - metadata:
      name: www
```

Kubernetes - DaemonSet

- Kubernetes Objects
  - Pods
  - Services
  - Volumes
  - Namespace
- Kubetnetes Controllers (also objects)
  - ReplicaSet
  - Deployment
  - StatefulSet
  - **DaemonSet**
  - Job
- Object Metadata
  - Label
  - Annotations

```
apiVersion: apps/v1
kind: DaemonSet
metadata:
  name: fluentd-elasticsearch
  namespace: kube-system
  labels:
    k8s-app: fluentd-logging
  spec:
...
```

Kubernetes - Job

- Kubernetes Objects
  - Pods
  - Services
  - Volumes
  - Namespace
- Kubetnetes Controllers (also objects)
  - ReplicaSet
  - Deployment
  - StatefulSet
  - DaemonSet
  - **Job**
- Object Metadata
  - Label
  - Annotations

```
apiVersion: batch/v1
kind: Job
metadata:
  name: pi
spec:
  template:
    spec:
      containers:
      - name: pi
        image: perl
        command: ["perl",
"-Mbignum=bpi", "-wle", "print
bpi(2000)"]
      restartPolicy: Never
  backoffLimit: 4
```