



KubeCon



CloudNativeCon

North America 2019





KubeCon



CloudNativeCon

North America 2019

Handling Risky Business: Cluster Upgrades

Puneet Pruthi





KubeCon



CloudNativeCon

North America 2019



Puneet Pruthi

Engineering Manager, Orchestration



ppruthi@lyft.com

@iamppruthi

Handling Risky Business: Cluster Upgrades



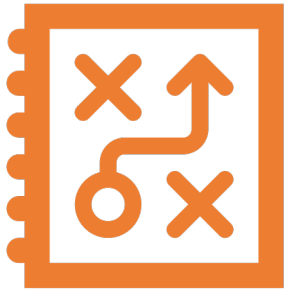
KubeCon



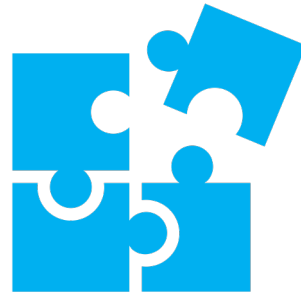
CloudNativeCon

North America 2019

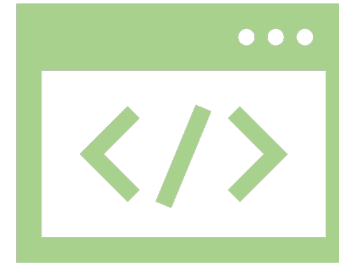
Agenda



How we got here ?



Solving the puzzle



What's in the box ?



State of things

Kubernetes @ Lyft: Overview



KubeCon



CloudNativeCon

North America 2019

- AWS
- Packer AMIs - Immutable Infrastructure
- Inbuilt cluster discovery and membership bootstrapping
- Complete control over Control & Data plane
- Scaling using Autoscaling Groups



Kubernetes @ Lyft: Upgrade Model



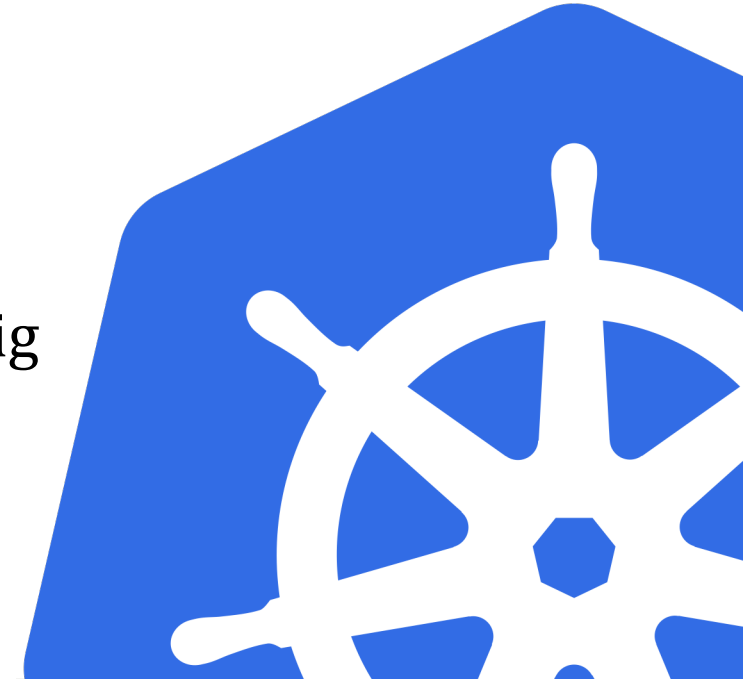
KubeCon



CloudNativeCon

North America 2019

- Build new AMIs
- Update the launch config for ASG
- Terminate old nodes -> ASG boot nodes with new launch config
- Reconcile cluster state



How did we get here ?



KubeCon



CloudNativeCon

North America 2019

Dec 2017, MVP K8s
clusters are born

Mid 2018, Multi-AZ clusters,
Migration for legacy services begins

May 2017,
Investigate
K8s on AWS

Mid 2018,
Instrumentation +
Testing workloads

2019, Hypergrowth:
Bulk migration



Define the problem



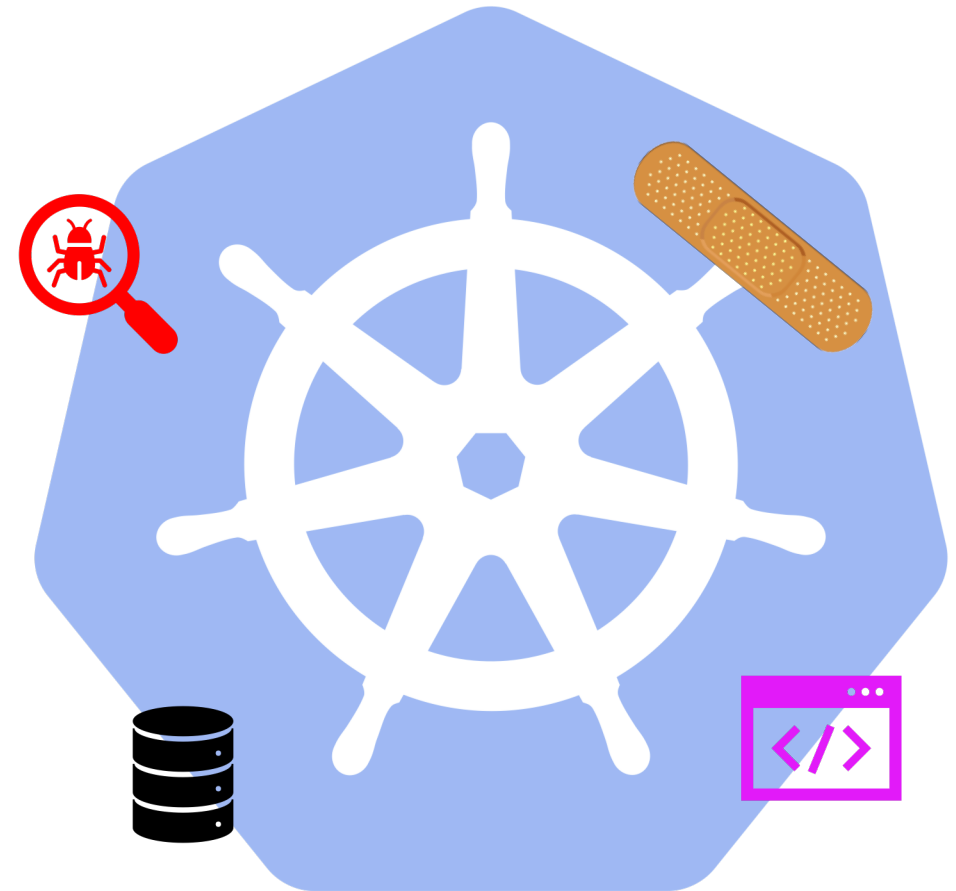
KubeCon



CloudNativeCon

North America 2019

- Why must we upgrade ?
 - Kubernetes version bump
 - OS Security Patch
 - Infrastructure changes/Features rollout
 - Bug fixes
 - Data retention guidelines
- What to upgrade ?
 - Entire cluster
 - Select nodes (function / placement / labels / taints)



Magic Wand



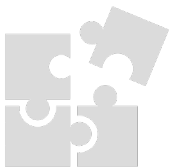
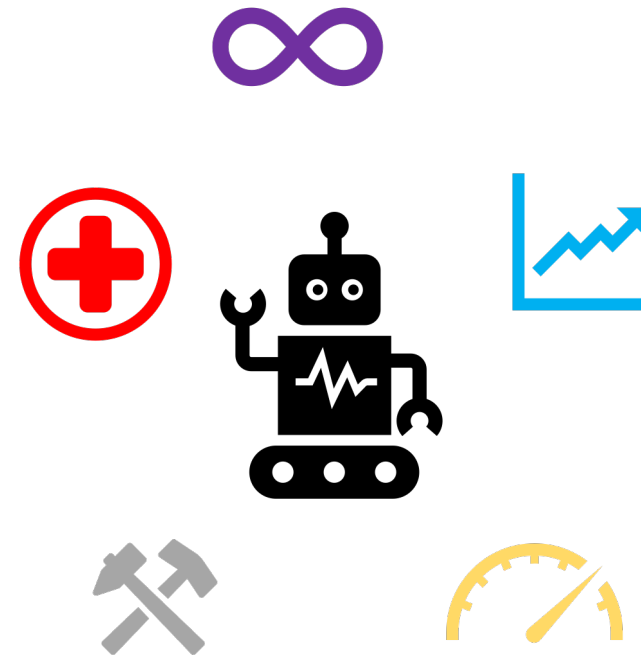
KubeCon



CloudNativeCon

North America 2019

- ~~Upgrade Window~~
- Maintain p95/p99 latencies
- Maintains Cluster Health
- Serverless tooling
- Automate + Override



Landscape



KubeCon

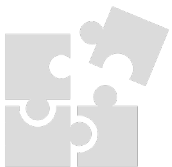
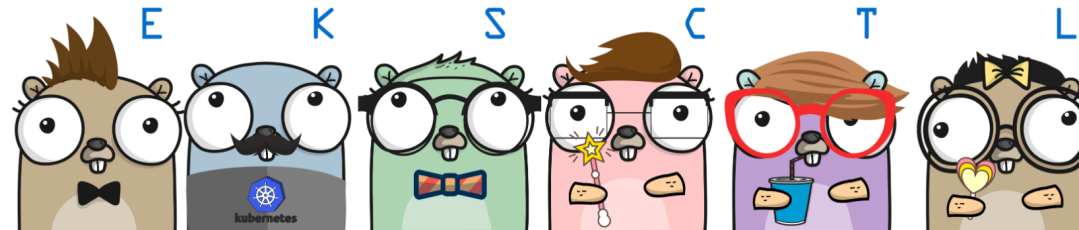


CloudNativeCon

North America 2019



gcloud





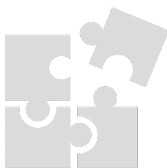
KubeCon



CloudNativeCon

North America 2019

Simplicity & Iteration





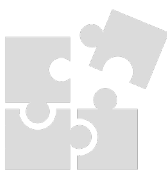
KubeCon



CloudNativeCon

North America 2019

```
do
while
  !is_cluster_updated() and
cluster_is_healthy() {
  NODE = get_old_node()
  kubectl cordons NODE
  kubectl drain NODE
  kubectl delete NODE
  terminate (NODE)
}
```





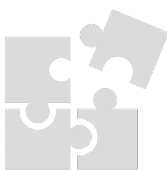
KubeCon



CloudNativeCon

North America 2019

```
do
while
    !is_cluster_updated() and
cluster_is_healthy() {
    NODE = get_old_node()
    kubectl cordons NODE
    kubectl drain NODE
    kubectl delete NODE
    terminate (NODE)
}
```



K8srotator: v0



KubeCon



CloudNativeCon

North America 2019

- Implements a custom controller in Golang
- Runs on Target Cluster as a singleton deployment
- Finite State Machine
- State transitions on simple triggers (ConfigMap object)
- Maintain cluster health (basic version)



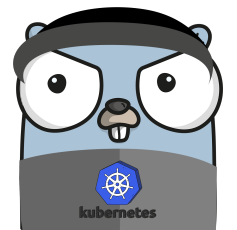
apiserver



etcd



stats



K8srotator: *FSM*



KubeCon



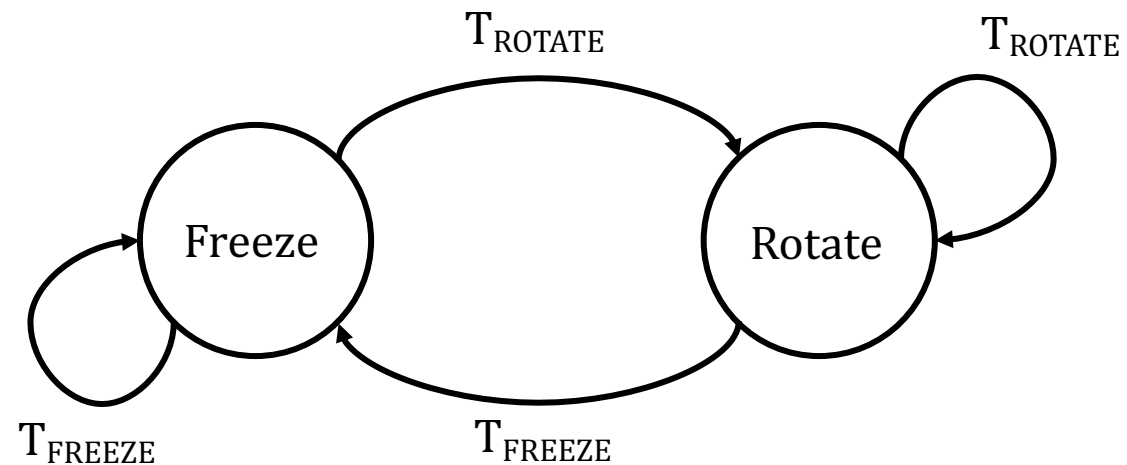
CloudNativeCon

North America 2019

- Finite State Machine

- Rotate
- Freeze

- State transitions - controlled by triggers / failure



K8srotator: *Triggers*



KubeCon



CloudNativeCon

North America 2019

- Trigger: Freeze
 - Default trigger on startup
 - Pauses / Halts ongoing cluster upgrade
 - Auto applied in case of failure / error

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: k8srotator-freeze
  namespace: lyft-system
  labels:
    app: k8srotator
    lyft.net/component: k8srotator
data:
  config: |
    command: FREEZE
    clusterName: REPLACE_CLUSTER_NAME
    metadata: REPLACE_METADATA
    ...
```



K8srotator: *Triggers*



KubeCon



CloudNativeCon

North America 2019

- Trigger: Rotate
 - Starts cluster rotation cycle
 - Specifies metadata about rotation
 - Target AMI info
 - maxSurge
 - maxDrainTime
 - etc.

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: k8srotator-rotate
  namespace: lyft-system
  labels:
    app: k8srotator
    lyft.net/component: k8srotator
data:
  config: |
    command: ROTATE
    clusterName: REPLACE_CLUSTER_NAME
    newAmiTag: REPLACE_AMI_TAG
    maxSurge: 1
    maxDrainTime: "REPLACE_MAX_DRAIN_TIME"
    dryRun: REPLACE_DRYRUN
    metadata: REPLACE_METADATA
    ...
```



K8srotator: *node lifecycle*

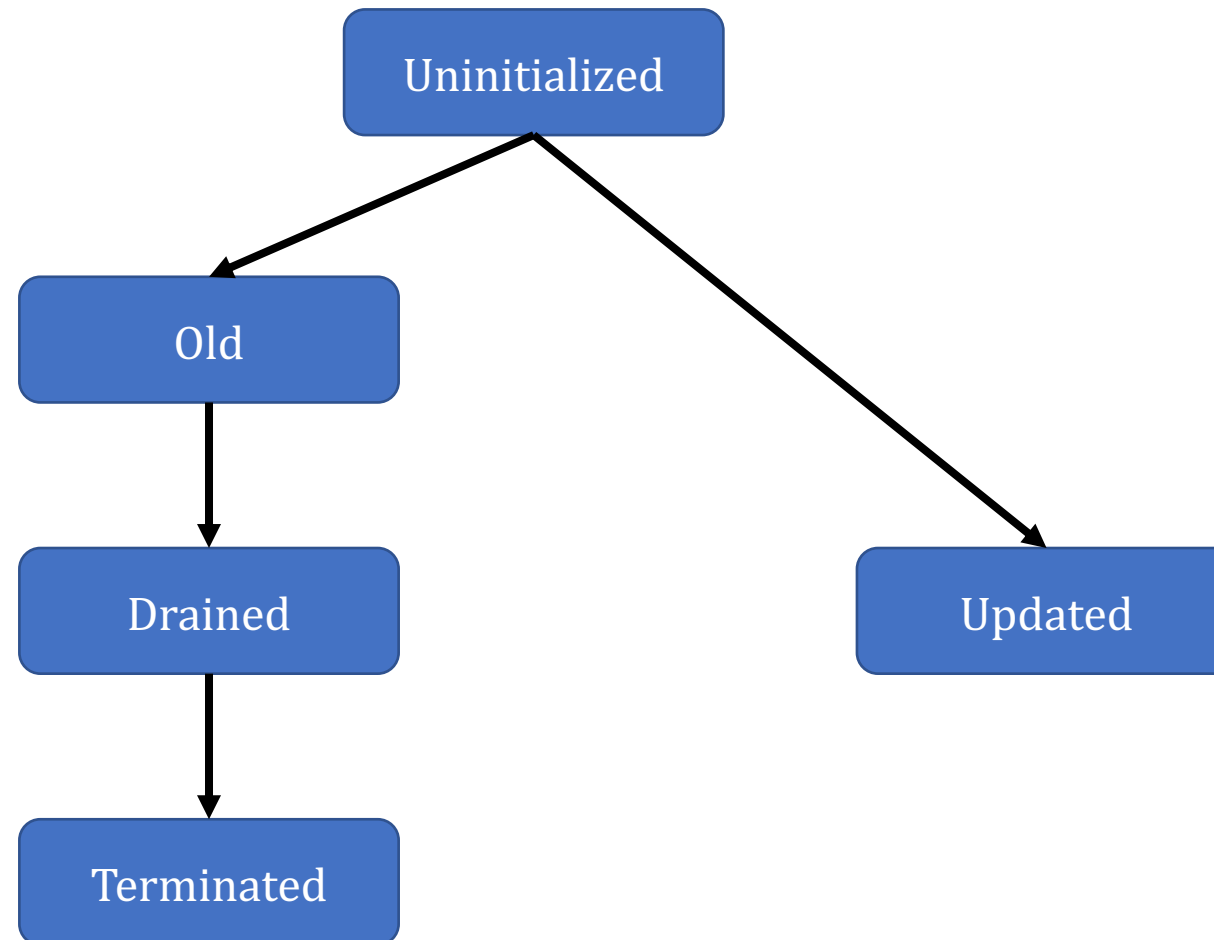


KubeCon



CloudNativeCon

North America 2019



K8srotator: Core functionality

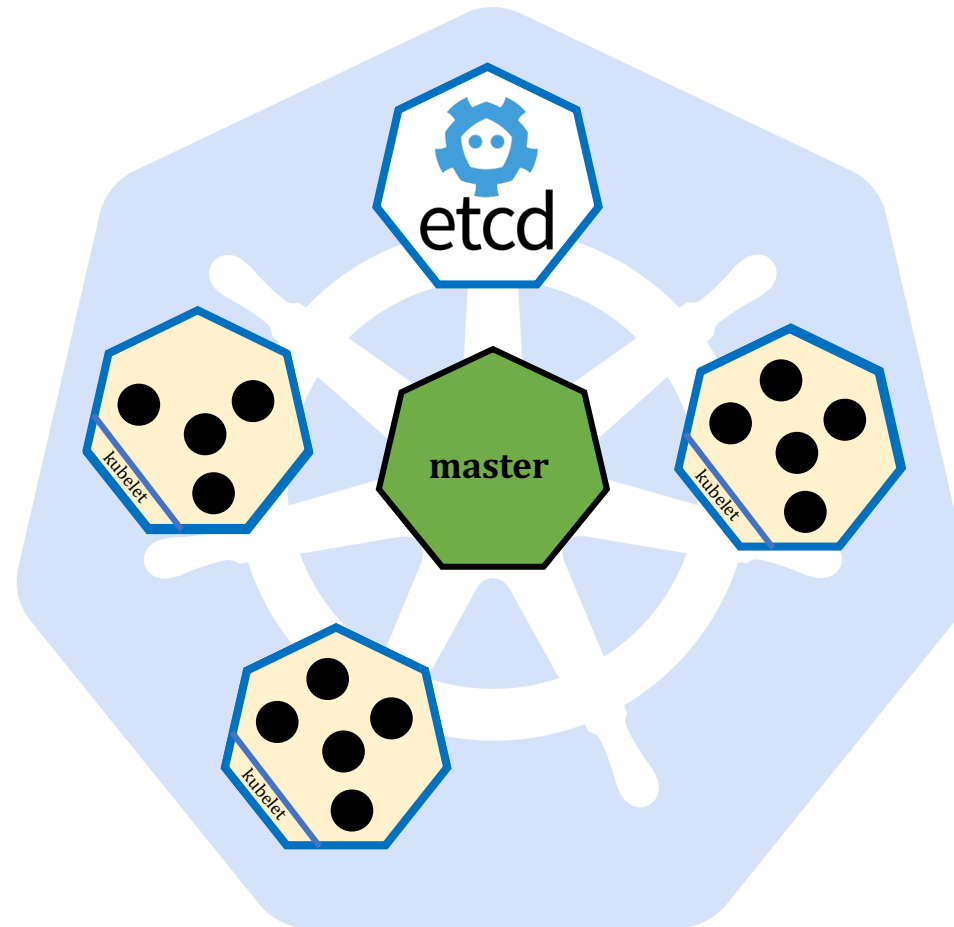


KubeCon



CloudNativeCon

North America 2019



K8srotator: Core functionality

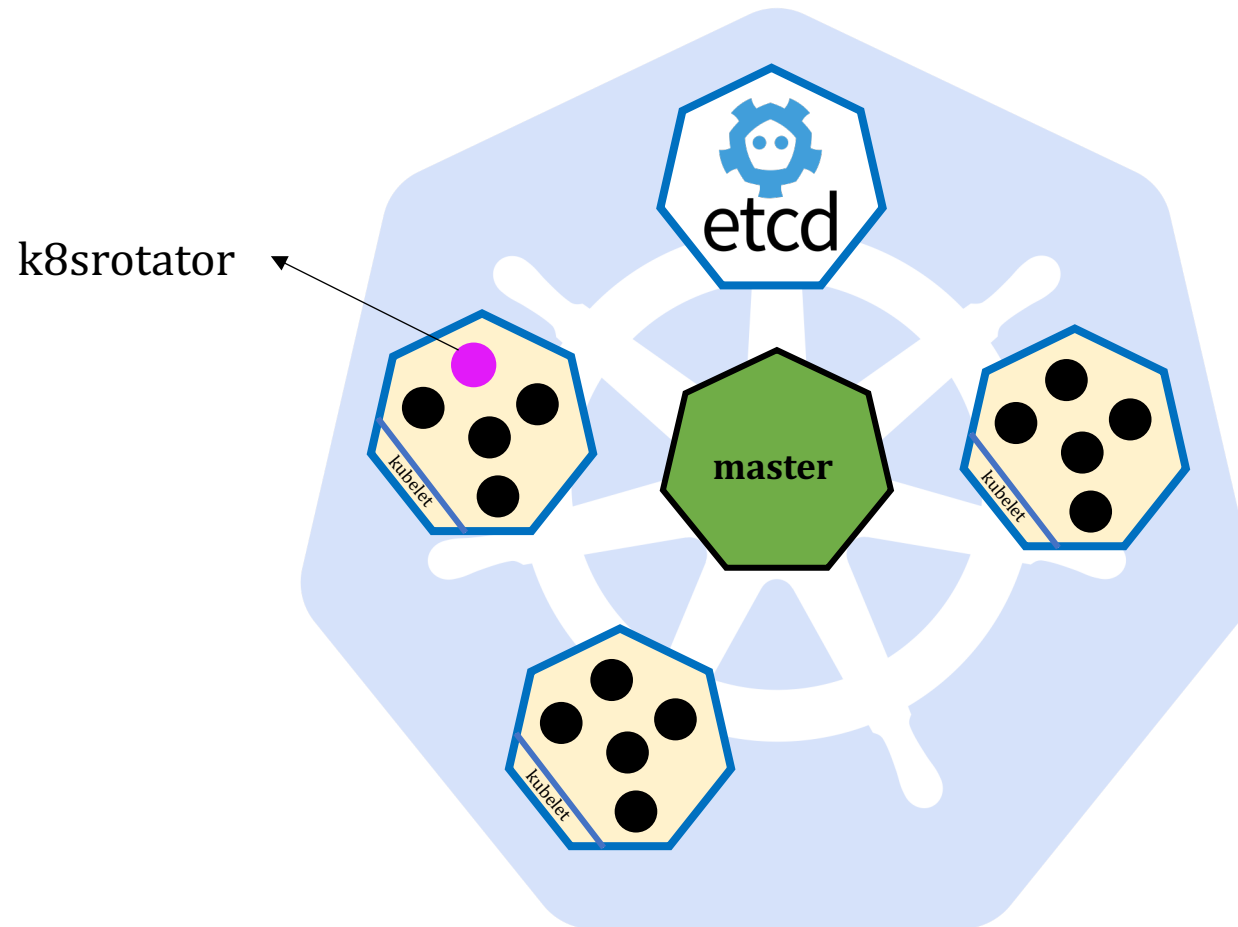






KubeCon



CloudNativeCon

North America 2019



-  uninitialized
-  old
-  drained
-  updated

K8srotator: Core functionality



KubeCon



CloudNativeCon

North America 2019

```
kubectl apply -f rotate.yaml
```

```
apiVersion: v1
kind: ConfigMap
metadata:
  name: k8srotator-rotate
  namespace: lyft-system
  labels:
    app: k8srotator
    lyft.net/component: k8srotator
data:
  config: |
    command: ROTATE
    clusterName: mycluster
    newAmiTag: bugfix-001
    maxSurge: 1
    maxDrainTime: "5m"
    nodeLabel: kubelet
```



K8srotator: Core functionality

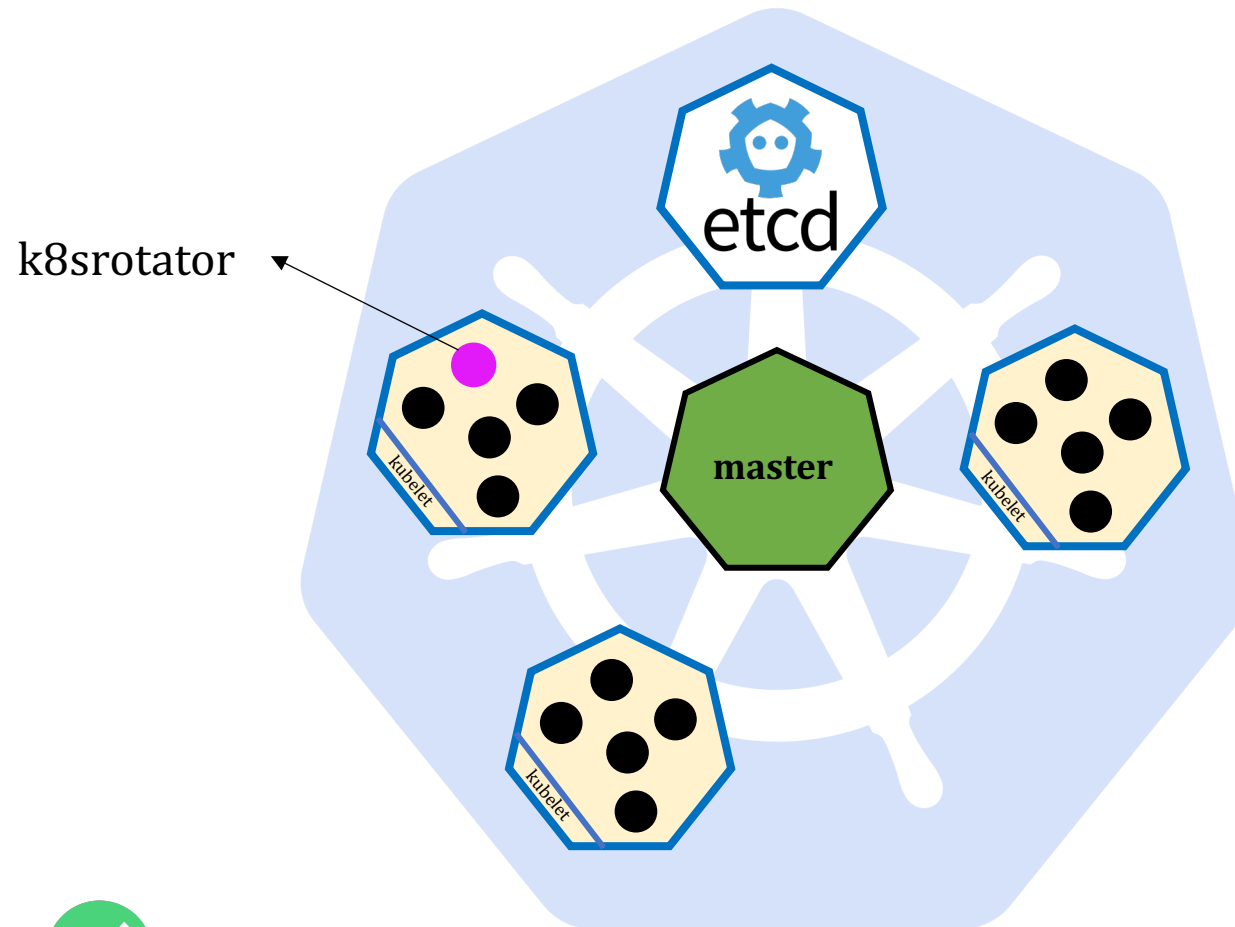


KubeCon



CloudNativeCon

North America 2019




apiserver



etcd



stats

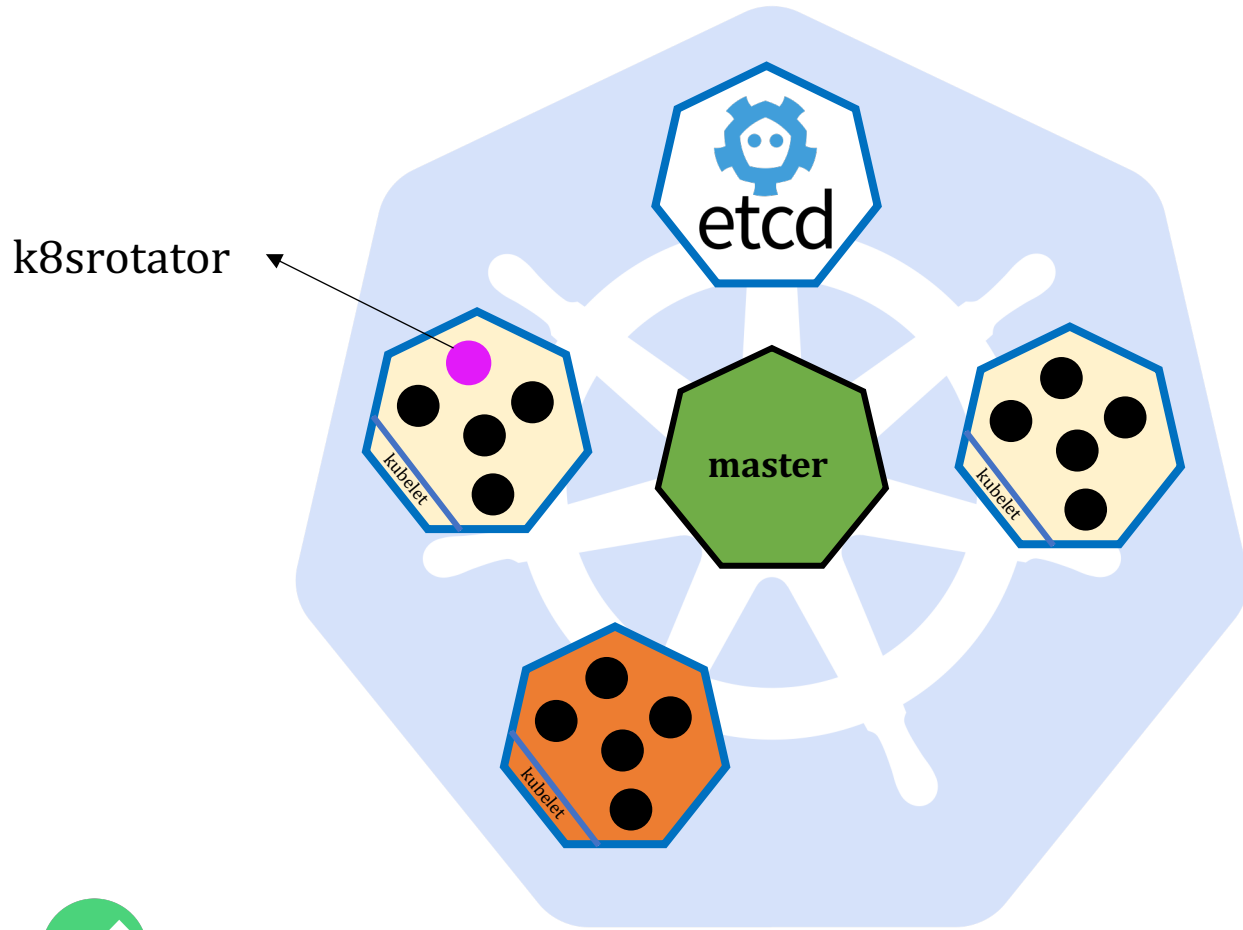
 uninitialized





 old

 drained

 updated

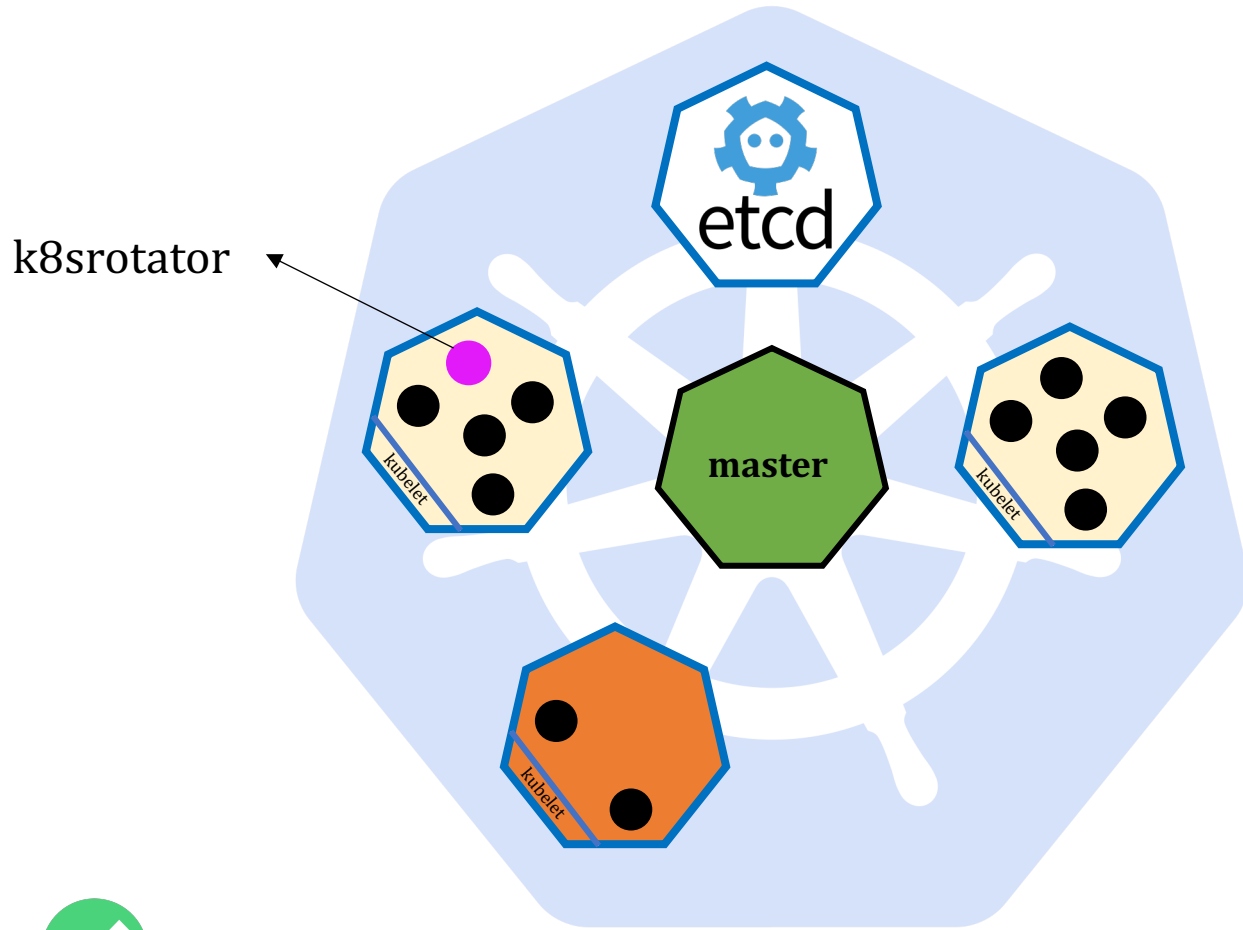
K8srotator: Core functionality



-  uninitialized
-  old
-  drained
-  updated

-  apiserver
-  etcd
-  stats

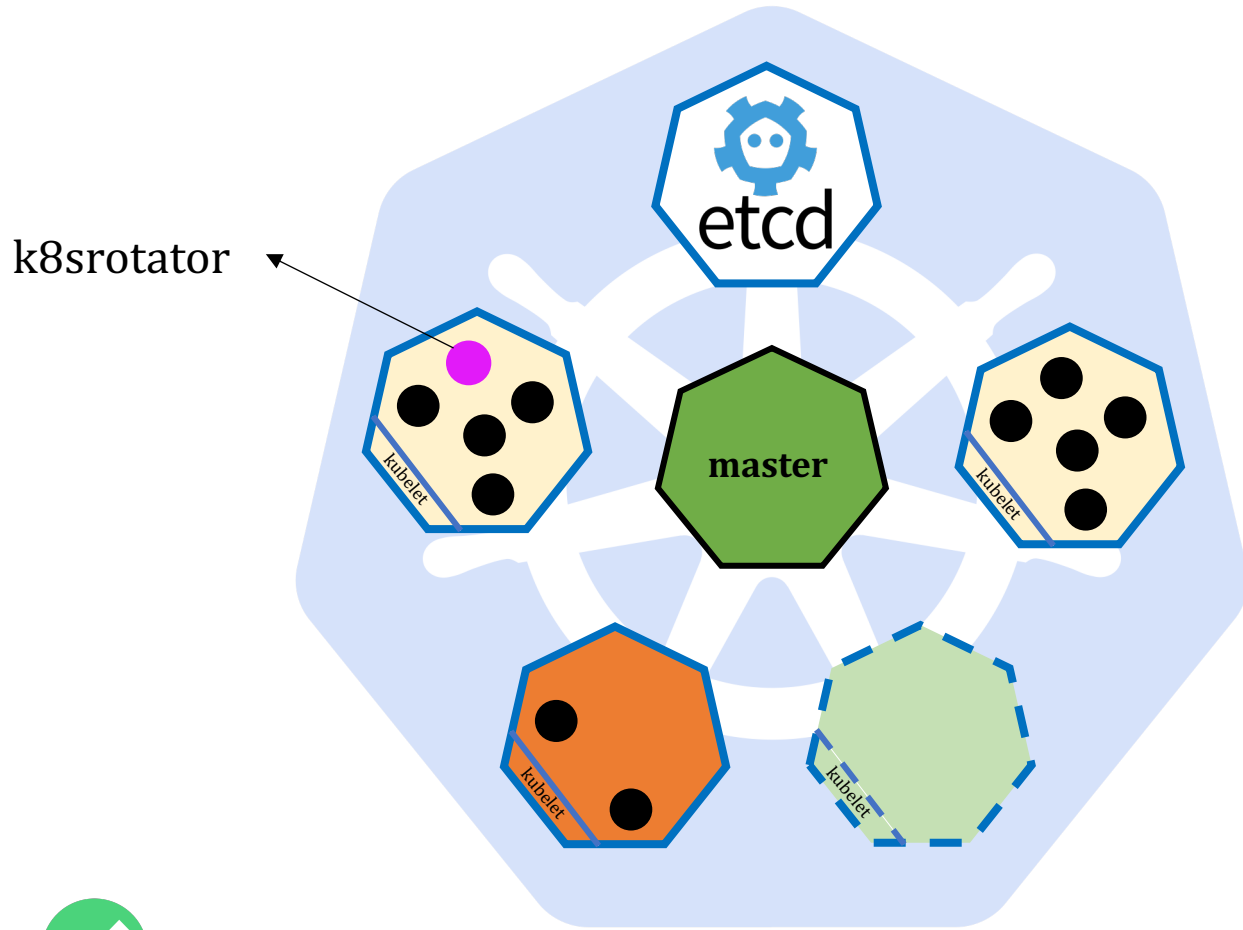
K8srotator: Core functionality



- uninitialized
- old
- drained
- updated

- ✓ apiserver
- ✓ etcd
- ✓ stats

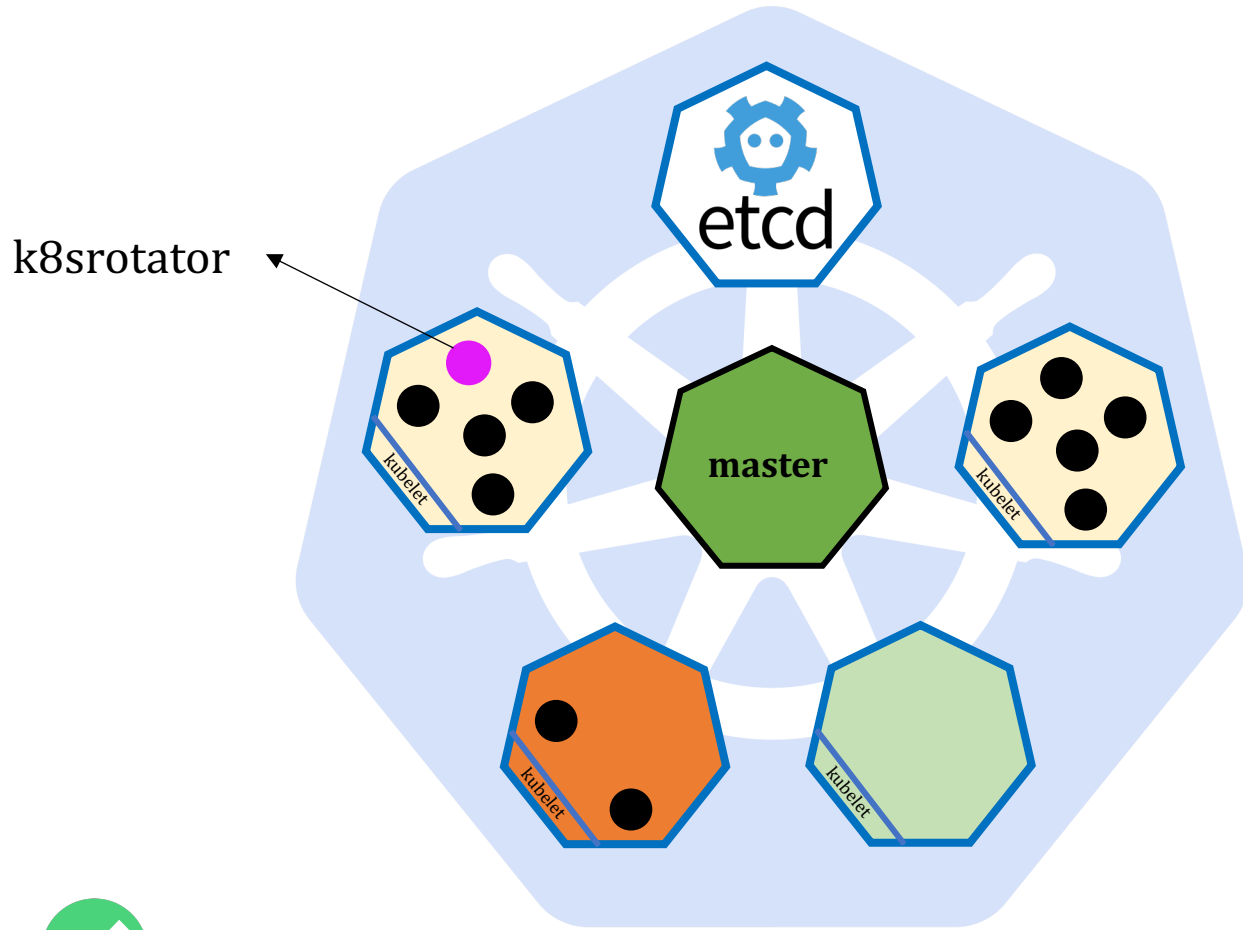
K8srotator: Core functionality



- uninitialized
- old
- drained
- updated

- ✓ apiserver
- ✓ etcd
- ✓ stats

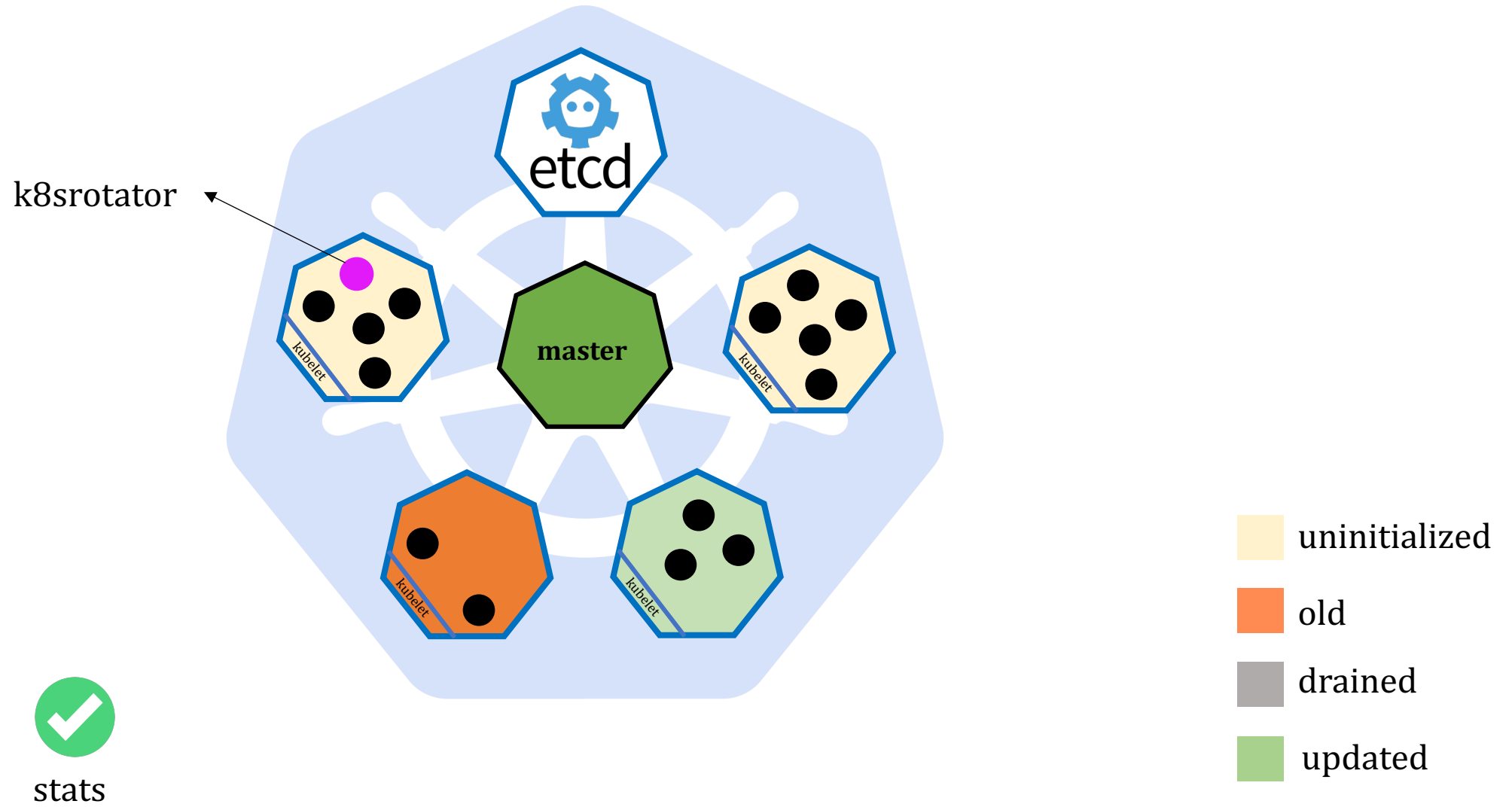
K8srotator: Core functionality



- uninitialized
- old
- drained
- updated

- ✓ apiserver
- ✓ etcd
- ✓ stats

K8srotator: Core functionality

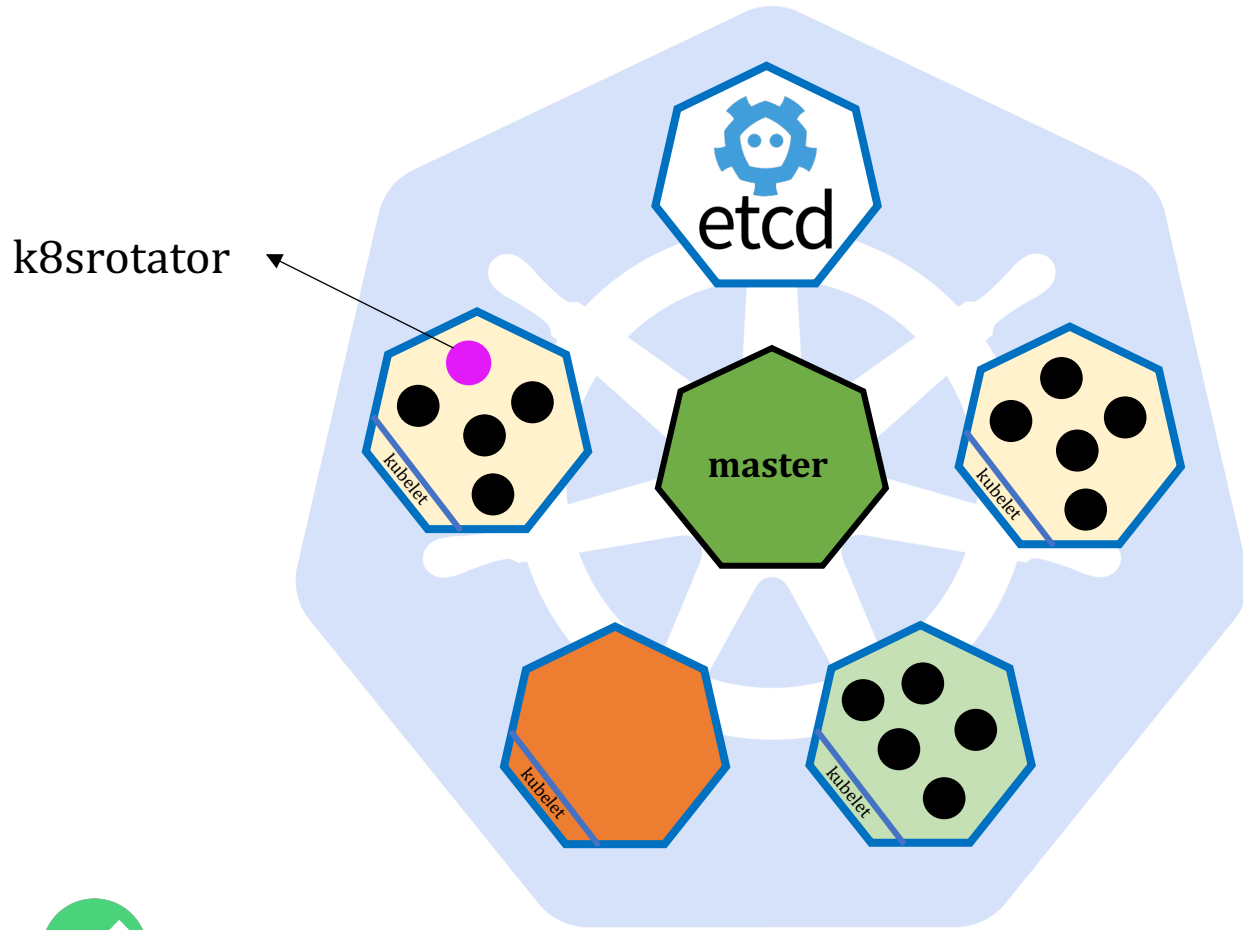



apiserver


etcd


stats

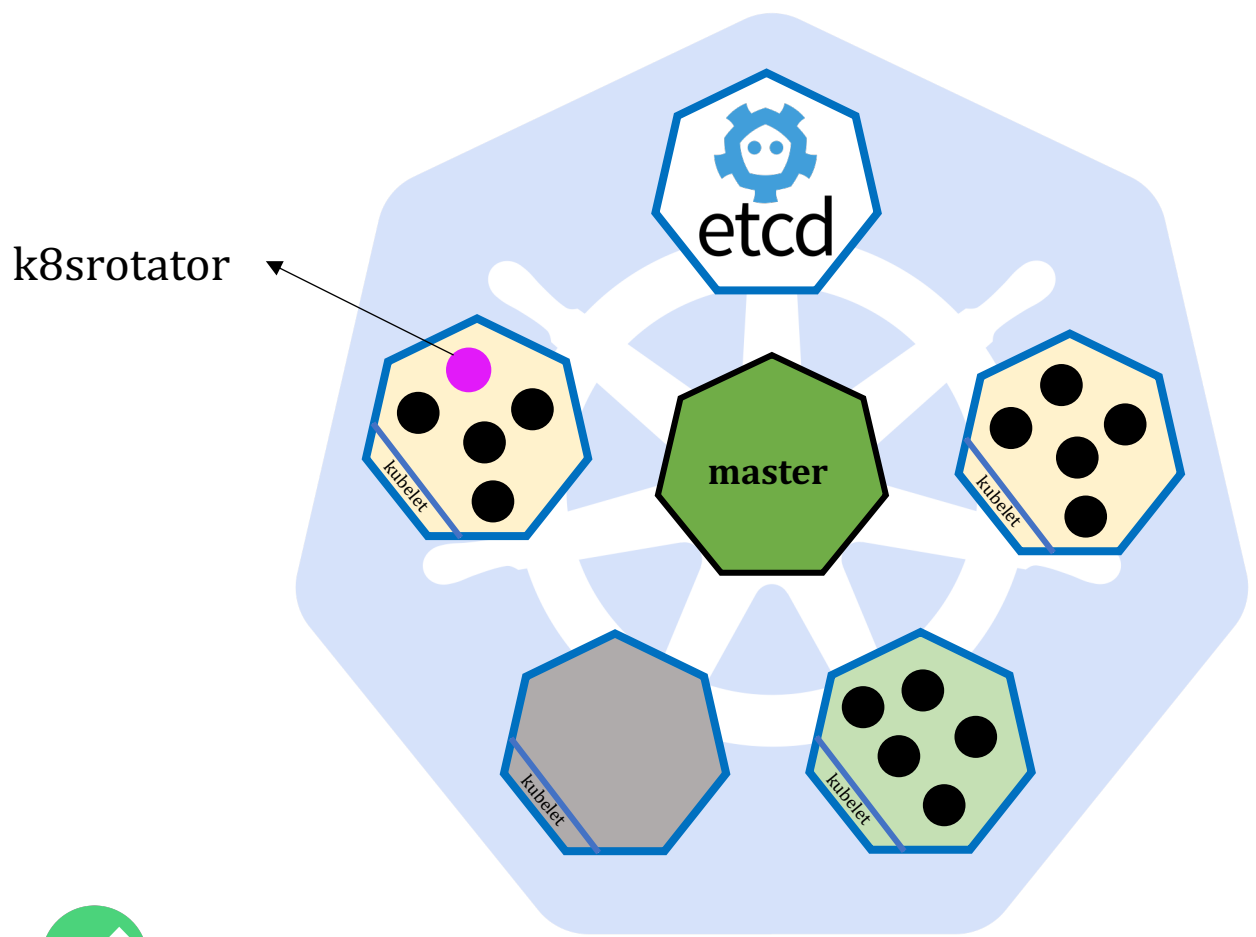
K8srotator: Core functionality







- uninitialized
- old
- drained
- updated

- ✓ apiserver
- ✓ etcd
- ✓ stats

K8srotator: Core functionality



-  uninitialized
-  old
-  drained
-  updated

-  apiserver
-  etcd
-  stats

K8srotator: Core functionality

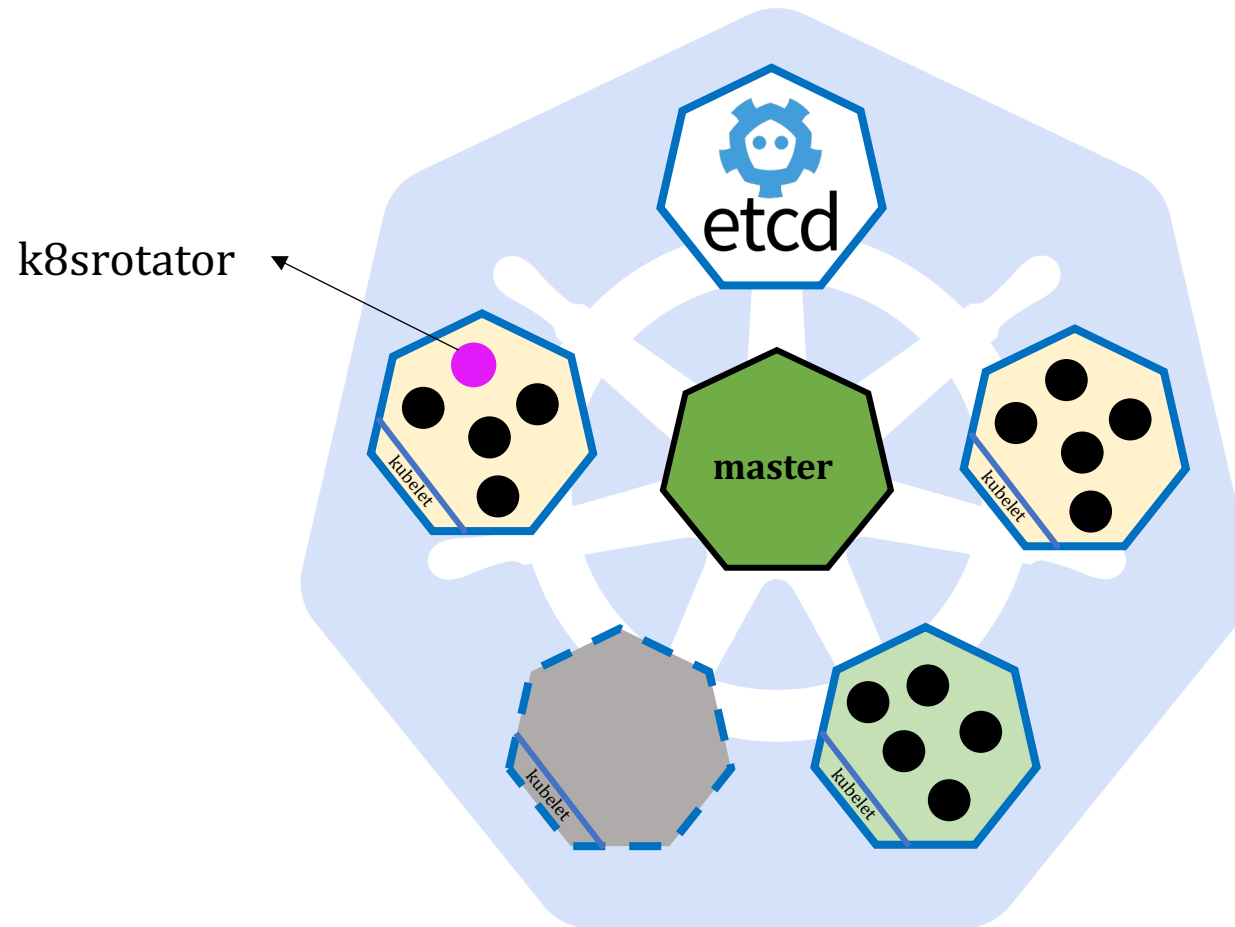


KubeCon



CloudNativeCon

North America 2019



uninitialized

old

drained

updated

K8srotator: Core functionality

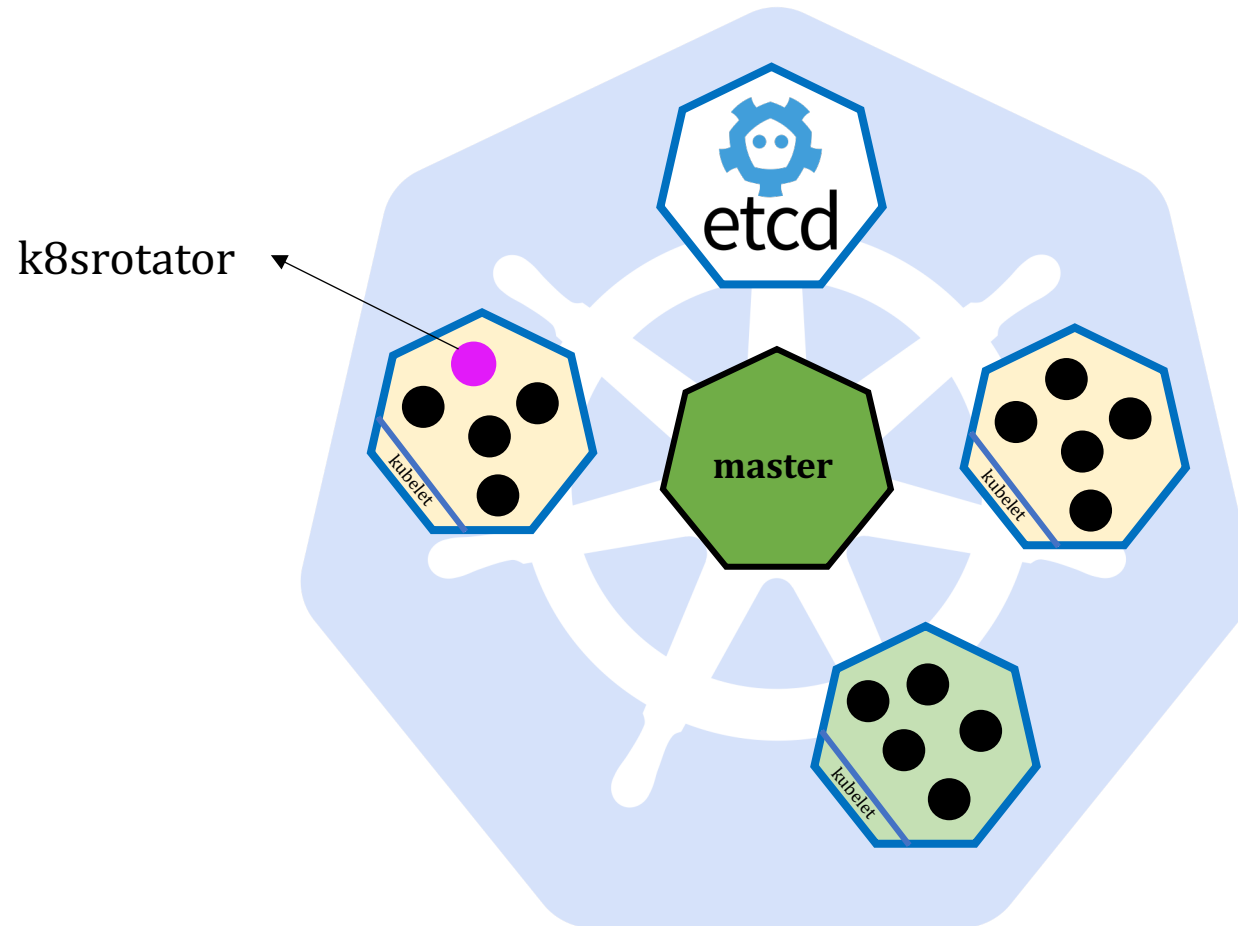



KubeCon



CloudNativeCon

North America 2019



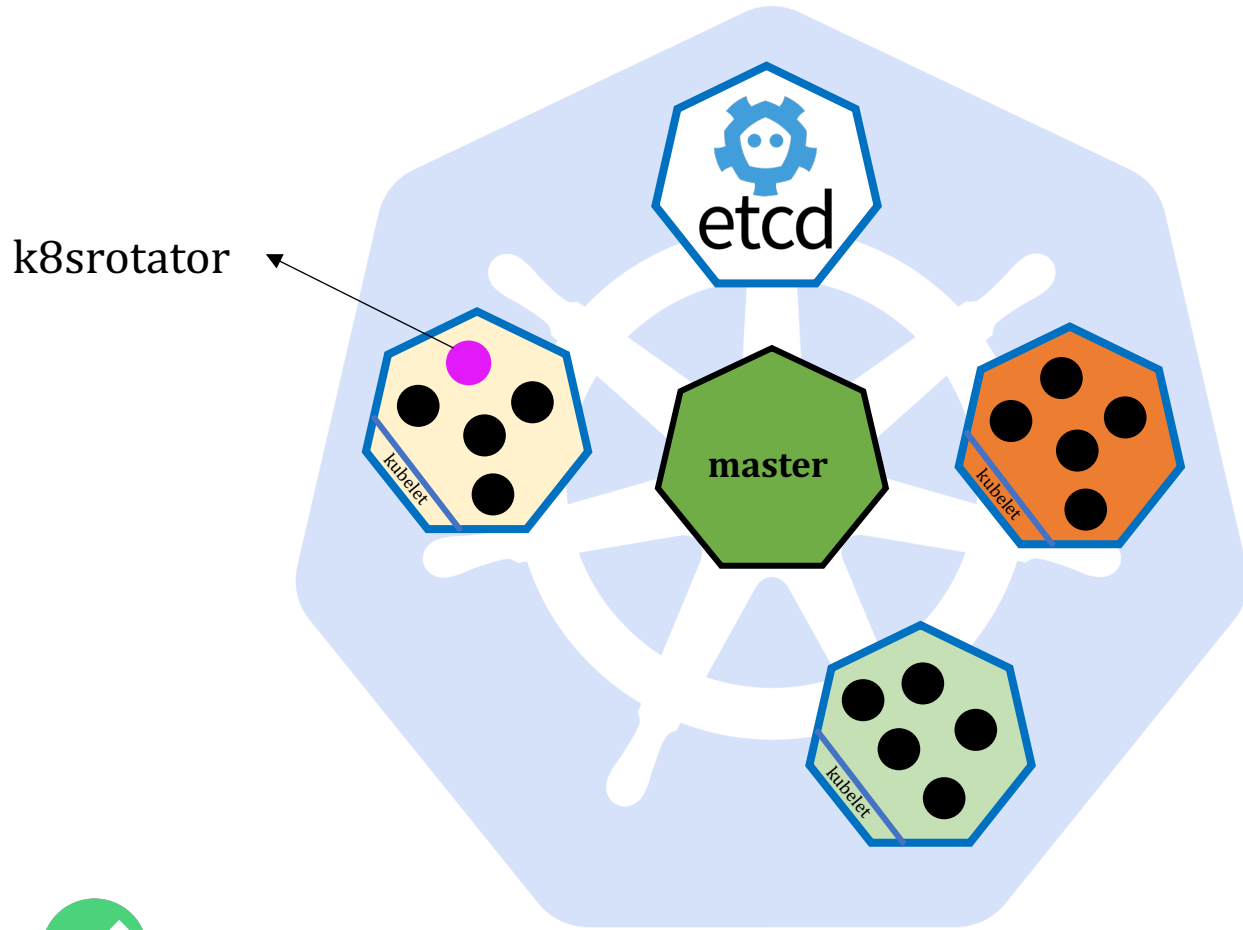
 uninitialized





 old

 drained

 updated

K8srotator: Core functionality



-  uninitialized
-  old
-  drained
-  updated

-  apiserver
-  etcd
-  stats

K8srotator: v1



KubeCon



CloudNativeCon

North America 2019

- Extended cluster health
 - Stricter Control Plane liveness
 - Log aggregator
 - Infra Stack Health
- Filter specific nodes
 - Choose type of nodes: etcd, apiserver, worker node
 - Select AZs, region
 - Using Node Labels
- Custom cluster configuration



Failure Scenarios to tackle



KubeCon



CloudNativeCon

North America 2019

- etcds loose quorum
- Apiserver/ingress is overloaded
- AWS AZ Capacity Limits
- Incompatible “in-place” Version upgrade
- AWS API Rate limits



Recovering from failures



KubeCon



CloudNativeCon

North America 2019

- Soft failures
- Hard failures
- Damage control:
 - Apply T_{Freeze}
 - Switch to FREEZE state
 - Alert the user / admin / team
- First Aid:
 - Logs / Stats
 - Remove T_{Freeze}
 - Switch back to ROTATE state

