Exploring

Enhancements

Federico Hernandez

Principal Software Engineer
Meltwater

@recollir

30000+ customers

1500+ employees

55 global offices

Founded 2001 in Oslo

Meltwater
Outside Insight

@recollir

Meltwater Engineering

9+ engineering offices (7+ countries)

330+ engineers in 50+ teams

Kubernetes in Production since 2018

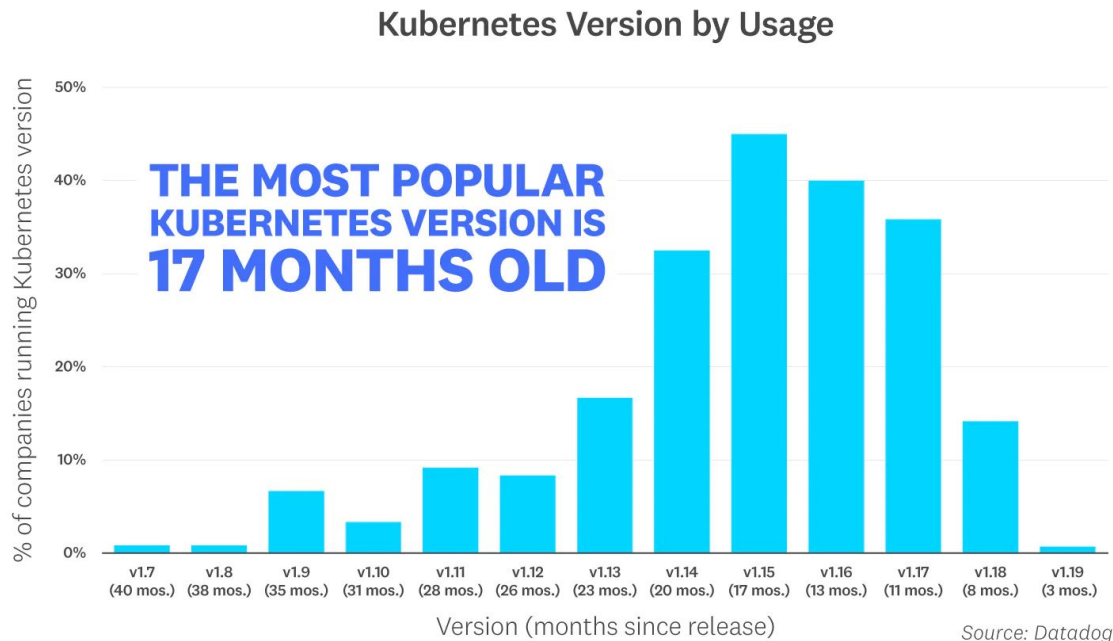2100 applications on Kubernetes
(50% production)

@recollr

Survey

https://sli.do

#37090

Datadog: 11 facts about real-world container use

Kubernetes Version by Usage

% of companies running Kubernetes version

THE MOST POPULAR KUBERNETES VERSION IS 17 MONTHS OLD

Version (months since release)

Source: Datadog

v1.7 (40 mos.) · v1.8 (38 mos.) · v1.9 (35 mos.) · v1.10 (31 mos.) · v1.11 (28 mos.) · v1.12 (26 mos.) · v1.13 (23 mos.) · v1.14 (20 mos.) · v1.15 (17 mos.) · v1.16 (13 mos.) · v1.17 (11 mos.) · v1.18 (8 mos.) · v1.19 (3 mos.)

https://www.datadoghq.com/container-report/

Enhancements from 1.15 to 1.19

1.16 has 31 enhancements
(September 2019)

- 8 graduating to stable
- 8 move to beta
- 15 move to alpha

1.17 has 22 enhancements
(December 2019)

- 14 graduating to stable
- 4 move to beta
- 4 move to alpha

1.18 has 38 enhancements
 (March 2020)

- 15 graduating to stable
- 11 move to beta
- 12 move to alpha

1.19 has 34 enhancements
 (August 2020)

- 10 graduating to stable
- 15 move to beta
- 9 move to alpha

Plus 100s of other changes, improvements, bug fixes

 https://k8s.devstats.cncf.io/

Enhancements from 1.15 to 1.19

1.16 has 31 enhancements
(September 2019)

- 8 graduating to stable
- 8 move to beta
- 15 move to alpha

1.17 has 22 enhancements
(December 2019)

- 14 graduating to stable
- 4 move to beta
- 4 move to alpha

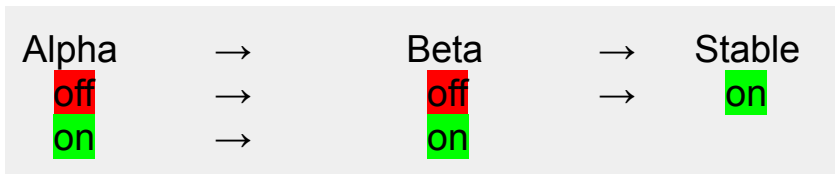1.18 has 38 enhancements
 (March 2020)

- 15 graduating to stable
- 11 move to beta
- 12 move to alpha

1.19 has 34 enhancements
 (August 2020)

- 10 graduating to stable
- 15 move to beta
- 9 move to alpha

Plus 100s of other changes, improvements, bug fixes

Alpha	→	Beta	→	Stable
off	→	off	→	on
on	→	on		

https://k8s.devstats.cncf.io/

@recollir

Feature gates for Kubernetes

```
--feature-gates="...,EphemeralContainers=true,ServiceTopology=true,..."
```

Kops example:

```
kubeProxy:
    metricsBindAddress: 0.0.0.0
    featureGates:
      EphemeralContainers: "true"
      HPAScaleToZero: "true"
      ImmutableEphemeralVolumes: "true"
      ServiceTopology: "true"
      EndpointSliceProxying: "true"
```

https://kubernetes.io/docs/reference/command-line-tools-reference/feature-gates/

Feature gates for Kubernetes

`EndpointSlice`	`false`	Alpha	1.16	1.16
`EndpointSlice`	`false`	Beta	1.17	
`EndpointSlice`	`true`	Beta	1.18	
`EndpointSliceProxying`	`false`	Alpha	1.18	1.18
`EndpointSliceProxying`	`true`	Beta	1.19	
`PodPriority`	`false`	Alpha	1.8	1.10
`PodPriority`	`true`	Beta	1.11	1.13
`PodPriority`	`true`	GA	1.14	-

https://kubernetes.io/docs/reference/command-line-tools-reference/feature-gates/

Exploring Enhancements

1. Ephemeral containers for kubectl debug
2. Configurable pod process namespace sharing
3. Even pod spreading across failure domains
4. Topology aware routing of services
5. A Startup Probe for slow starting pods
6. Configurable scale velocity for HPA
7. Changes in kubectl

A. Graduation of Ingress to v1
B. Removed and Deprecated APIs

1. Ephemeral containers for kubectl debug

FEATURE STATE: v1.18 [alpha]
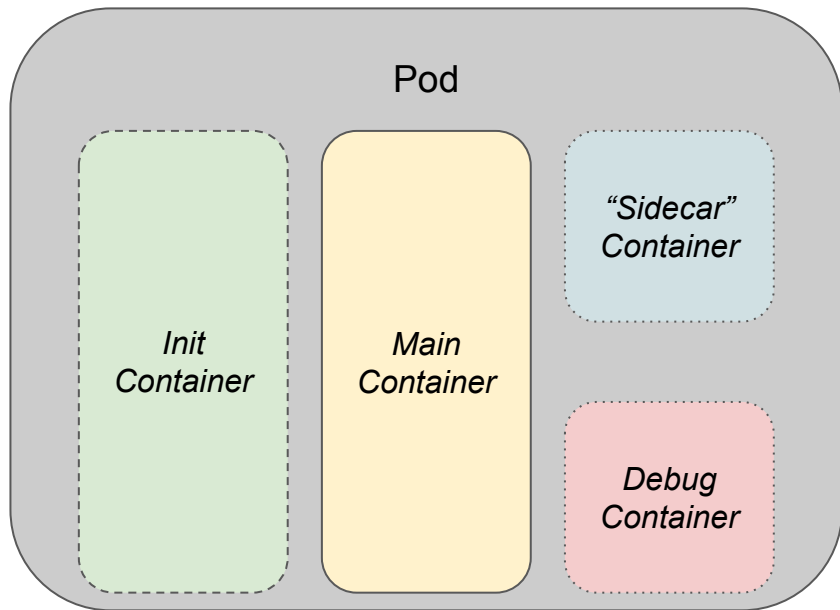

Pod

Init Container

Main Container

"Sidecar" Container
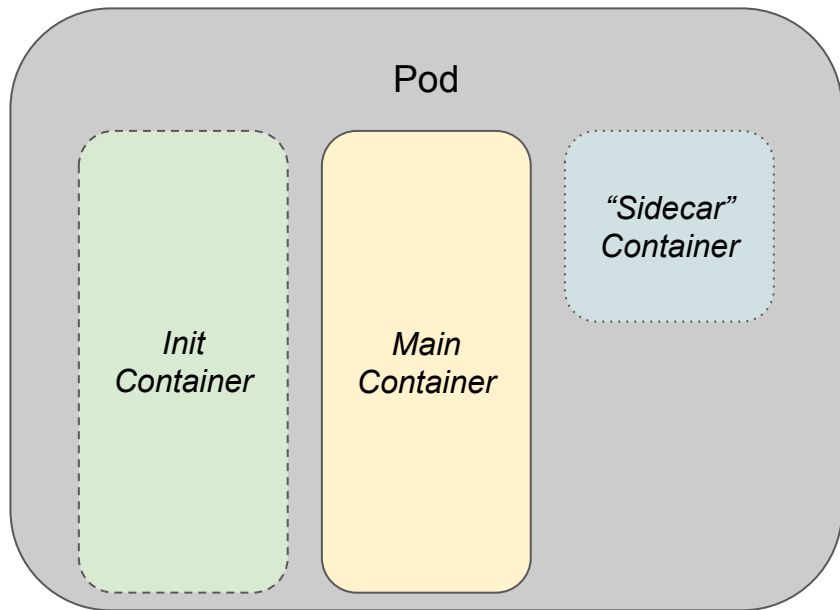
```
$ kubectl exec -ti foo -- sh

OCI runtime exec failed: exec failed:
container_linux.go:349: starting container
process caused "exec: \"sh\": executable
file not found in $PATH": unknown command
terminated with exit code 126
```

1. Ephemeral containers for kubectl debug

FEATURE STATE: v1.18 [alpha]

Pod

Init Container

Main Container

"Sidecar" Container

Debug Container

```
$ kubectl alpha debug
  -it pod-name
  --image=busybox
  --target=container-name
```

https://kubernetes.io/docs/tasks/debug-application-cluster/debug-running-pod/#ephemeral-container
https://kubernetes.io/docs/concepts/workloads/pods/ephemeral-containers/

@recollir

2. Pod process namespace sharing

FEATURE STATE: v1.17 [stable]

```
apiVersion: v1
kind: Pod
metadata:
  name: nginx
spec:
  shareProcessNamespace: true
  containers:
  - name: nginx
    image: nginx
  - name: shell
    image: busybox
```
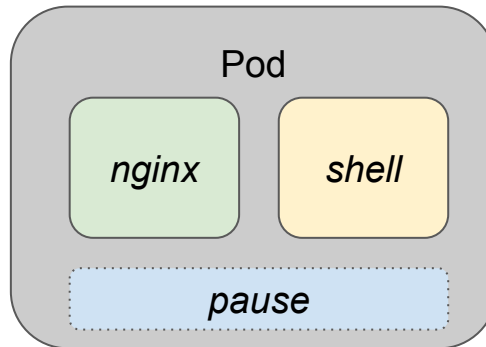


```
$ kubectl attach -ti nginx -c shell
# ps aux
PID   USER      TIME  COMMAND
   1  root      0:00  /pause
   6  root      0:00  nginx: master process
  33  101       0:00  nginx: worker process
  34  root      0:00  sh
  39  root      0:00  ps aux
```

https://kubernetes.io/docs/tasks/configure-pod-container/share-process-namespace/

@recollir

3. PodTopologySpread

FEATURE STATE: v1.17 [alpha]

Affinity
Anti-Affinity
NodeSelector

Zone A

Zone B

Zone C
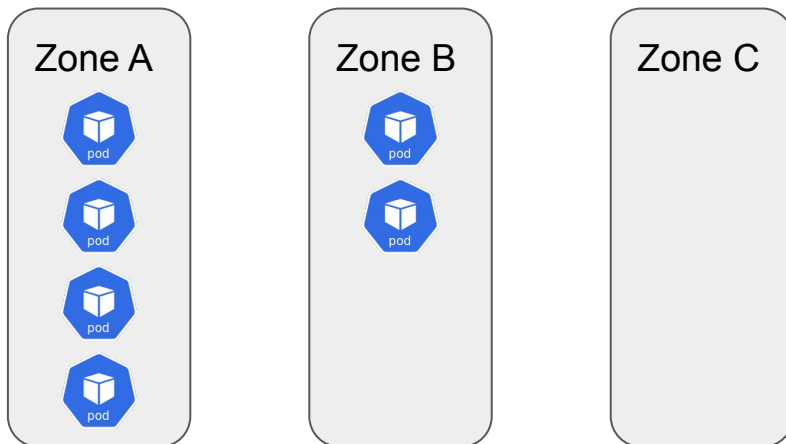
3. PodTopologySpread

FEATURE STATE: v1.17 [alpha]

```
spec:
  topologySpreadConstraints:
  - maxSkew: 1
    topologyKey: zone
    whenUnsatisfiable: ScheduleAnyway
    labelSelector:
      matchLabels:
        app: bar
```

Zone A
Zone B
Zone C

https://kubernetes.io/blog/2020/05/introducing-podtopologyspread/
https://kubernetes.io/docs/concepts/workloads/pods/pod-topology-spread-constraints/
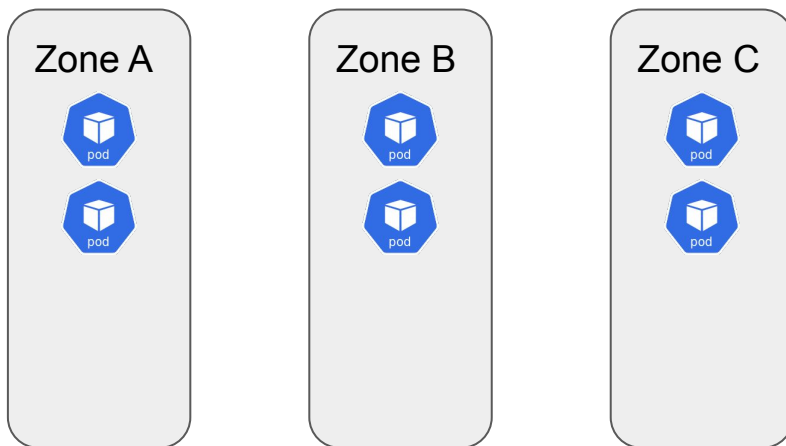
3. PodTopologySpread
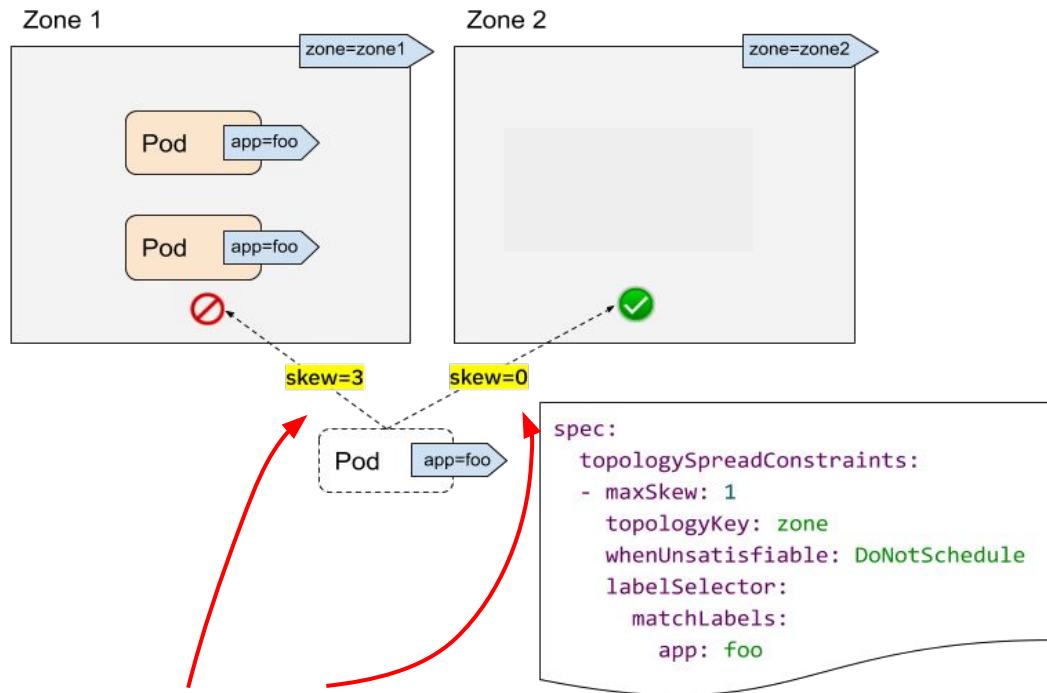
skew = Pods **number** matched in **current** topology - **min** Pods matches in a topology

https://kubernetes.io/blog/2020/05/introducing-podtopologyspread/
https://kubernetes.io/docs/concepts/workloads/pods/pod-topology-spread-constraints/

@recollir

4. ServiceTopology

FEATURE STATE: v1.17 [alpha]

```
apiVersion: v1
kind: Service
metadata:
  name: my-service
spec:
  selector:
    app: my-app
  ports:
    - protocol: TCP
      port: 80
      targetPort: 9376
  topologyKeys:
    - "kubernetes.io/hostname"
    - "topology.kubernetes.io/zone"
    - "*"
```


https://kubernetes.io/docs/concepts/services-networking/service-topology/

@recollir

4. ServiceTopology

FEATURE STATE: v1.17 [alpha]

https://kubernetes.io/docs/concepts/services-networking/service-topology/
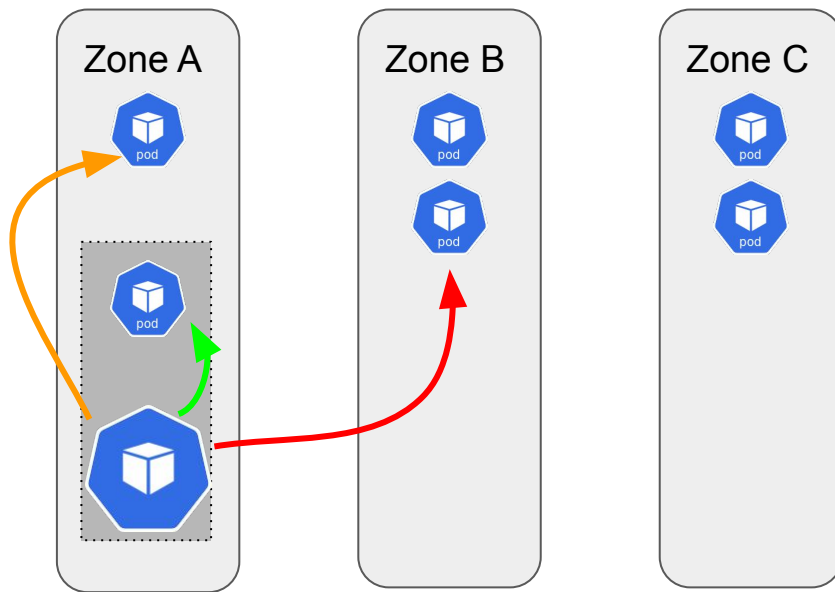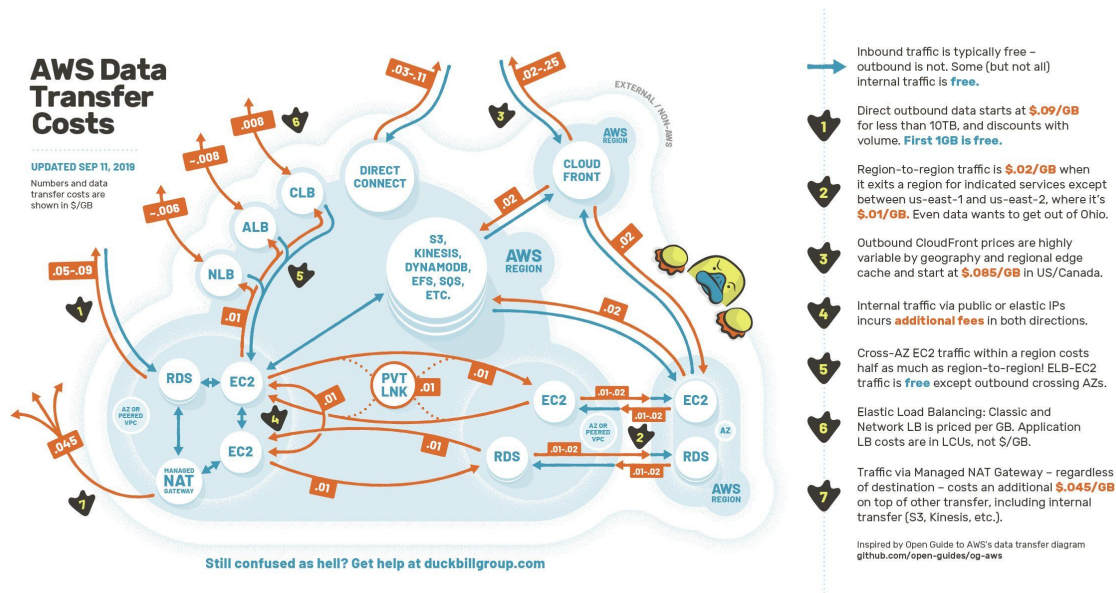
4. ServiceTopology

1 deployment + 1 HPA
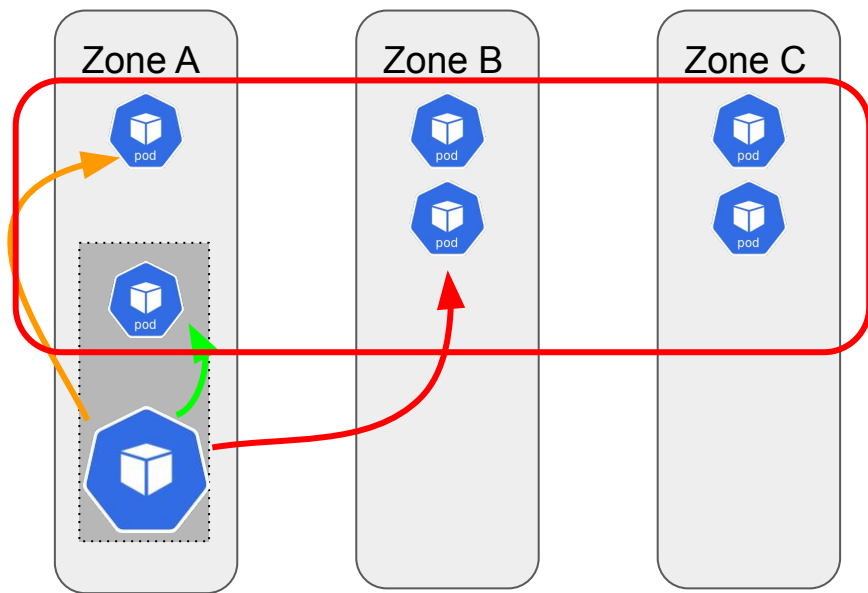
3 deployment + 3 HPA

Zone A · Zone B · Zone C

Zone A · Zone B · Zone C

https://kubernetes.io/docs/concepts/services-networking/service-topology/
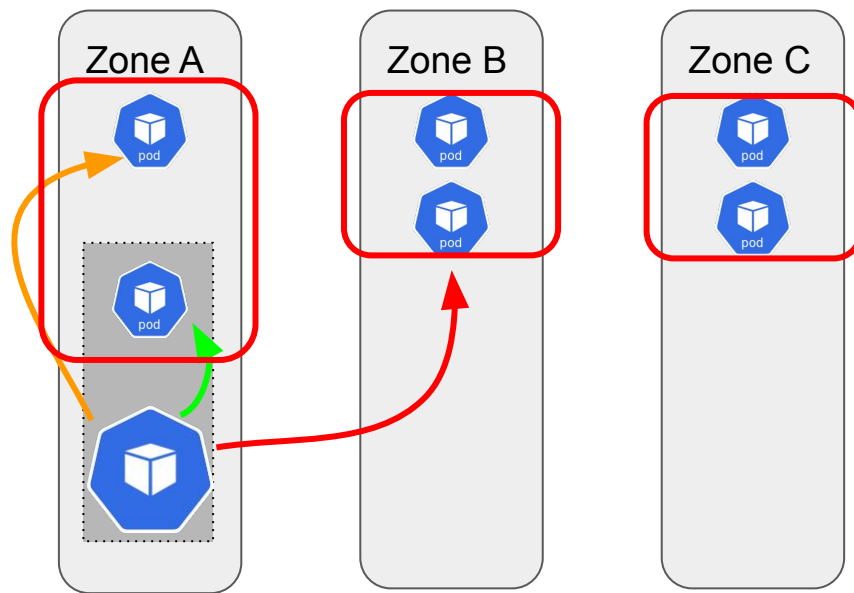
5. A Startup Probe for slow starting pods

FEATURE STATE: v1.16 [alpha], v1.18 [beta]

```
startupProbe:
  httpGet:
    path: /healthz
    port: liveness-port
  failureThreshold: 30
  periodSeconds: 10
```

startupProbe

livenessProbe
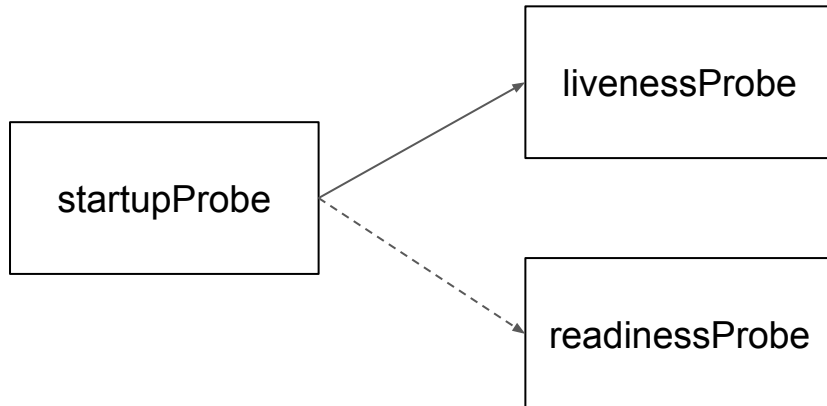
readinessProbe

```
livenessProbe:
  httpGet:
    path: /healthz
    port: liveness-port
  failureThreshold: 2
  periodSeconds: 10

readinessProbe:
  httpGet:
    path: /readyz
    port: readiness-port
  failureThreshold: 1
  periodSeconds: 10
```

https://kubernetes.io/docs/tasks/configure-pod-container/configure-liveness-readiness-startup-probes/#define-startup-probes
https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle/#container-probes

@recollir

5. A Startup Probe for slow starting pods

FEATURE STATE: v1.16 [alpha], v1.18 [beta]

```
@@ -30,18 +30,23 @@ spec:
30          httpGet:                          30          httpGet:
31            path: /_status                  31            path: /_status
32            port: 8080                      32            port: 8080
33 -          initialDelaySeconds: 180         33 +          periodSeconds: 10
34 -          periodSeconds: 100
35          failureThreshold: 3              34          failureThreshold: 3
36          successThreshold: 1              35          successThreshold: 1
37        livenessProbe:                     36        livenessProbe:
38          httpGet:                         37          httpGet:
39            path: /_status                 38            path: /_status
40            port: 8080                     39            port: 8080
41 -          initialDelaySeconds: 180        40 +          periodSeconds: 10
42 -          periodSeconds: 100
43          failureThreshold: 3              41          failureThreshold: 3
44          successThreshold: 1              42          successThreshold: 1
                                             43 +        startupProbe:
                                             44 +          httpGet:
                                             45 +            path: /_status
                                             46 +            port: 8080
                                             47 +          periodSeconds: 5
                                             48 +          failureThreshold: 60
                                             49 +          successThreshold: 1
45        resources:                         50        resources:
46          requests:                        51          requests:
47            cpu: 1500m                     52            cpu: 1500m
```

https://kubernetes.io/docs/tasks/configure-pod-container/configure-liveness-readiness-startup-probes/#define-startup-probes
https://kubernetes.io/docs/concepts/workloads/pods/pod-lifecycle/#container-probes

6. Configurable scale velocity for HPA

FEATURE STATE: v1.18 [v2beta2 of HPA]

```
minReplicas: 1
  maxReplicas: 10
  metrics:
  - type: Resource
    resource:
      name: cpu
      target:
        type: Utilization
        averageUtilization: 50
```

```
desiredReplicas = currentReplicas * ( currentMetricValue / desiredMetricValue )
```

7. Configurable scale velocity for HPA

FEATURE STATE: v1.18 [v2beta2 of HPA]

```
behavior:
  scaleDown:
    stabilizationWindowSeconds: 300
    policies:
    - type: Pods
      value: 4
      periodSeconds: 60
    - type: Percent
      value: 10
      periodSeconds: 60
```

```
behavior:
  scaleUp:
    policies:
    - type: percent
      value: 200%
```

1 → 2 → 4 → 8

https://kubernetes.io/docs/tasks/run-application/horizontal-pod-autoscale/#support-for-configurable-scaling-behavior

7. Changes in kubectl

SINCE VERSION: v1.18

Remove kubectl run generators #87077

⑂ Merged **k8s-ci-robot** merged 1 commit into `kubernetes:master` from `soltysh:remove_run_generators`

- **kubectl run** now only creates pods. Use **kubectl create** for other objects instead. But… :-/
- **kubectl diff** graduated to stable
- **kubectl get deployment bar -o yaml** contains now managed fields… :-/
  https://github.com/kubernetes/kubernetes/issues/90066
- **kubectl get --export** query parameter removed but there is a plugin for it… :-/
- **kubectl apply --dry-run=client|server|none** is no longer boolean
- **kubectl logs --prefix** prefix each log line with pod and container name :-)
- **kubectl alpha debug** for ephemeral debug containers
- … … …
- … … …

Appendix A: Graduation of Ingress to v1

FEATURE STATE: v.18 [v1beta1], v1.19 [stable]

Since these features are new in Kubernetes 1.18, each Ingress controller implementation will need some time to develop support for these new features.

```
apiVersion: "networking.k8s.io/v1beta1"
kind: "Ingress"
metadata:
  name: "example-ingress"
spec:
  ingressClassName: "external-lb"
  rules:
  - host: "*.example.com"
    http:
      paths:
      - path: "/example"
        pathType: "Prefix"
        backend:
          serviceName: "example-service"
          servicePort: 80
```

@recollir

Appendix B: Deprecated and Removed APIs

SINCE VERSION: v1.16

Deployment in the **extensions/v1beta1**, **apps/v1beta1**, and **apps/v1beta2** API versions is no longer served

- Migrate to use the **apps/v1** API version, available since v1.9. Existing persisted data can be retrieved/updated via the new version.
- Notable changes:
  - `spec.rollbackTo` is removed
  - `spec.selector` is now required and immutable after creation; use the existing template labels as the selector for seamless upgrades
  - `spec.progressDeadlineSeconds` now defaults to `600` seconds (the default in `extensions/v1beta1` was no deadline)
  - `spec.revisionHistoryLimit` now defaults to `10` (the default in `apps/v1beta1` was `2`, the default in `extensions/v1beta1` was to retain all)
  - `maxSurge` and `maxUnavailable` now default to `25%` (the default in `extensions/v1beta1` was 1)

The **v1.22** release will stop serving the following deprecated API versions in favor of newer and more stable API versions:

- Ingress in the **extensions/v1beta1** API version will no longer be served
  - Migrate to use the **networking.k8s.io/v1beta1** API version, available since v1.14. Existing persisted data can be retrieved/updated via the new version.

https://kubernetes.io/blog/2019/07/18/api-deprecations-in-1-16/

@recollir

Don't Panic: Kubernetes and Docker

Wednesday, December 02, 2020

Authors: Jorge Castro, Duffie Cooley, Kat Cosgrove, Justin Garrison, Noah Kantrowitz, Bob Killen, Rey Lejano, Dan "POP" Papandrea, Jeffrey Sica, Davanum "Dims" Srinivas

Kubernetes is deprecating Docker as a container runtime after v1.20.

You do not need to panic. It's not as dramatic as it sounds.

tl;dr Docker as an underlying runtime is being deprecated in favor
Interface(CRI) created for Kubernetes. Docker-produced images
runtimes, as they always have.

s is no longer served

data can be retrieved/updated via the new

sting template labels as the selector for

The **v1.22** release will stop serving the following
- Ingress in the **extensions/v1beta1** API ve
  - Migrate to use the **networking.k8s.**
    retrieved/updated via the new ver

Can I still use Docker in Kubernetes 1.20?

Yes, the only thing changing in 1.20 is a single warning log printed at kubelet startup if using Docker as the runtime.

When will dockershim be removed?

Given the impact of this change, we are using an extended deprecation timeline. It will not be removed before Kubernetes 1.22, meaning the earliest release without dockershim would be 1.23 in late 2021. We will be working closely with vendors and other ecosystem groups to ensure a smooth transition and will evaluate things as the situation evolves.

https://kubernetes.io/blog/2019/07/18/api-deprecations-in-1-16/

@recollir

Further information

Kubernetes release announcements:
- 1.16: https://kubernetes.io/blog/2019/09/18/kubernetes-1-16-release-announcement/
- 1.17: https://kubernetes.io/blog/2019/12/09/kubernetes-1-17-release-announcement/
- 1.18: https://kubernetes.io/blog/2020/03/25/kubernetes-1-18-release-announcement/
- 1.19: https://kubernetes.io/blog/2020/08/26/kubernetes-release-1.19-accentuate-the-paw-sitive/

Sysdig - What's new in Kubernetes:
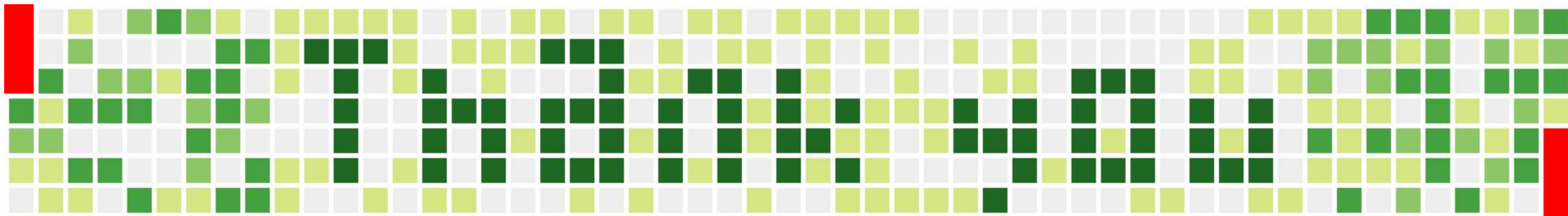- 1.16: https://sysdig.com/blog/whats-new-kubernetes-1-16
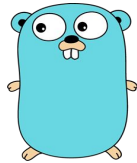- 1.17: https://sysdig.com/blog/whats-new-kubernetes-1-17
- 1.18: https://sysdig.com/blog/whats-new-kubernetes-1-18
- 1.19: https://sysdig.com/blog/whats-new-kubernetes-1-19

(as well as the *lengthy* CHANGELOG like
https://github.com/kubernetes/kubernetes/blob/master/CHANGELOG/CHANGELOG-1.19.md)

Federico Hernandez

@recollir

We are looking for a
Junior Platform Engineer (GOT)

https://emp.jobylon.com/jobs/69272