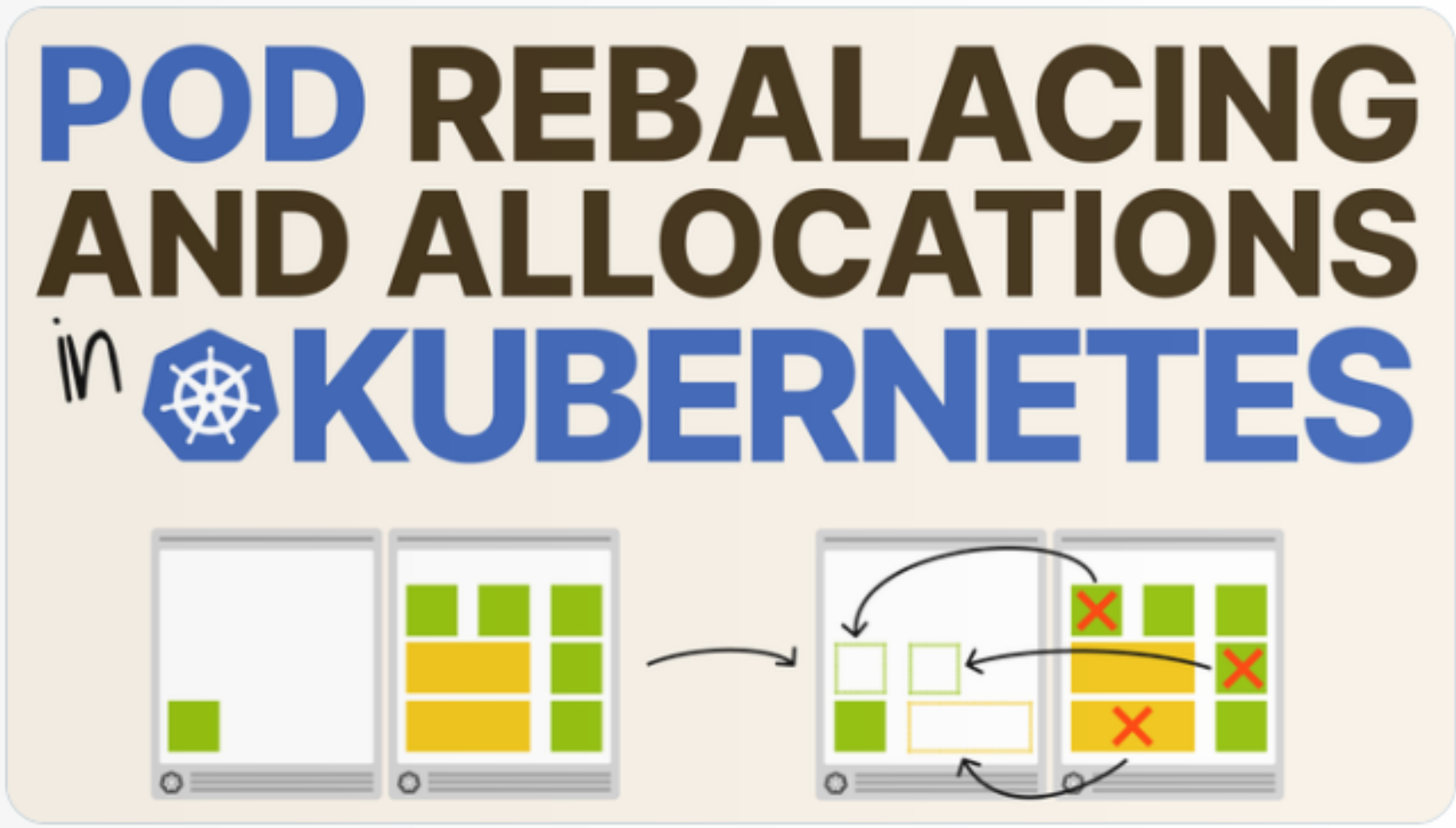
Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

Does Kubernetes rebalance your Pods?

If there's a node that has more space, does Kubernetes recompute and balance the workloads?

Let's have a look 👉



8:15 PM · Apr 3, 2023

440 Likes

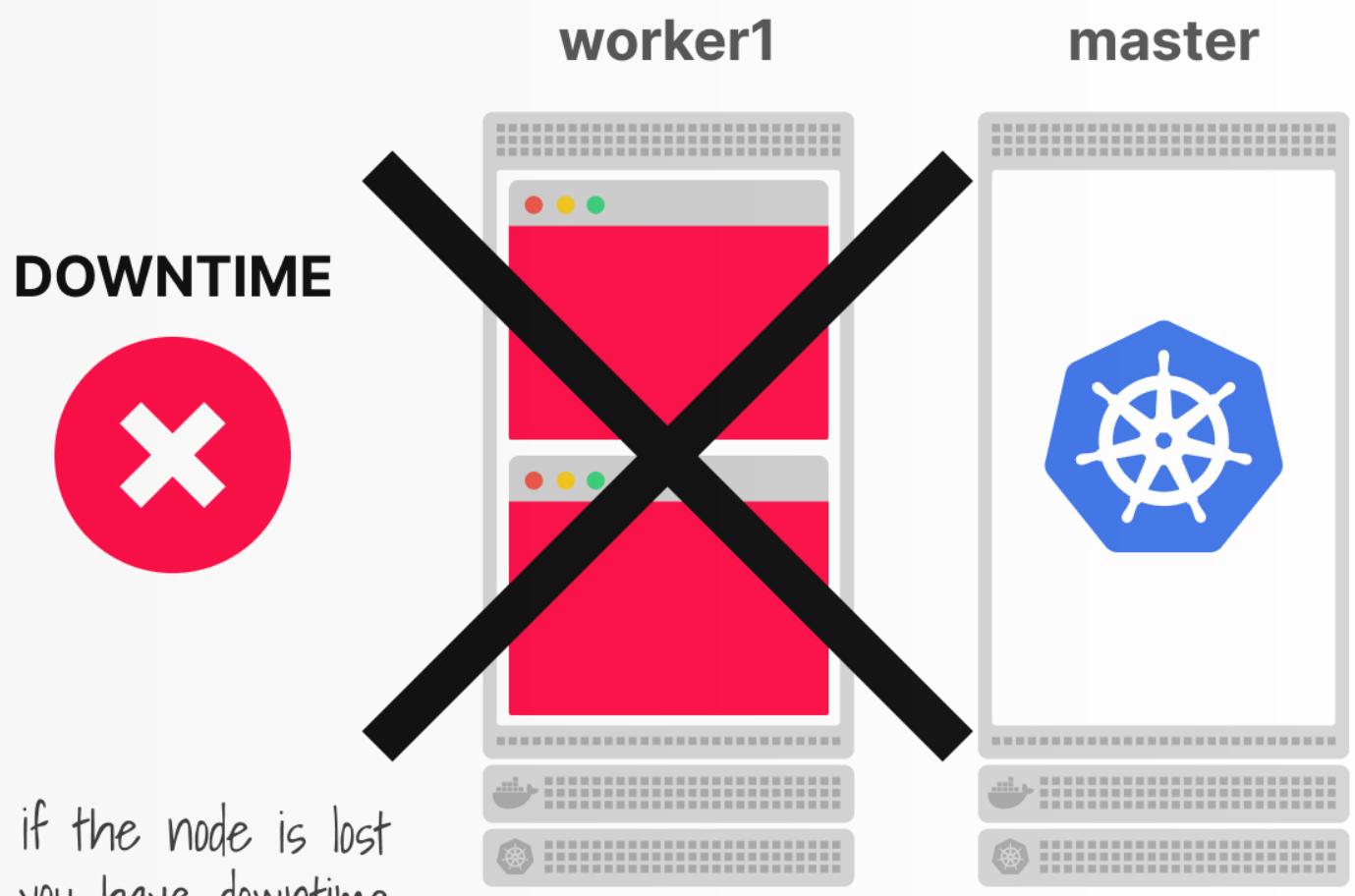
Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

1/

You have a cluster with a single node that can host 2 Pods

If the node crashes, you will experience downtime

You could have a second node with one Pod each to prevent this



8:15 PM · Apr 3, 2023

5 Likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

2/

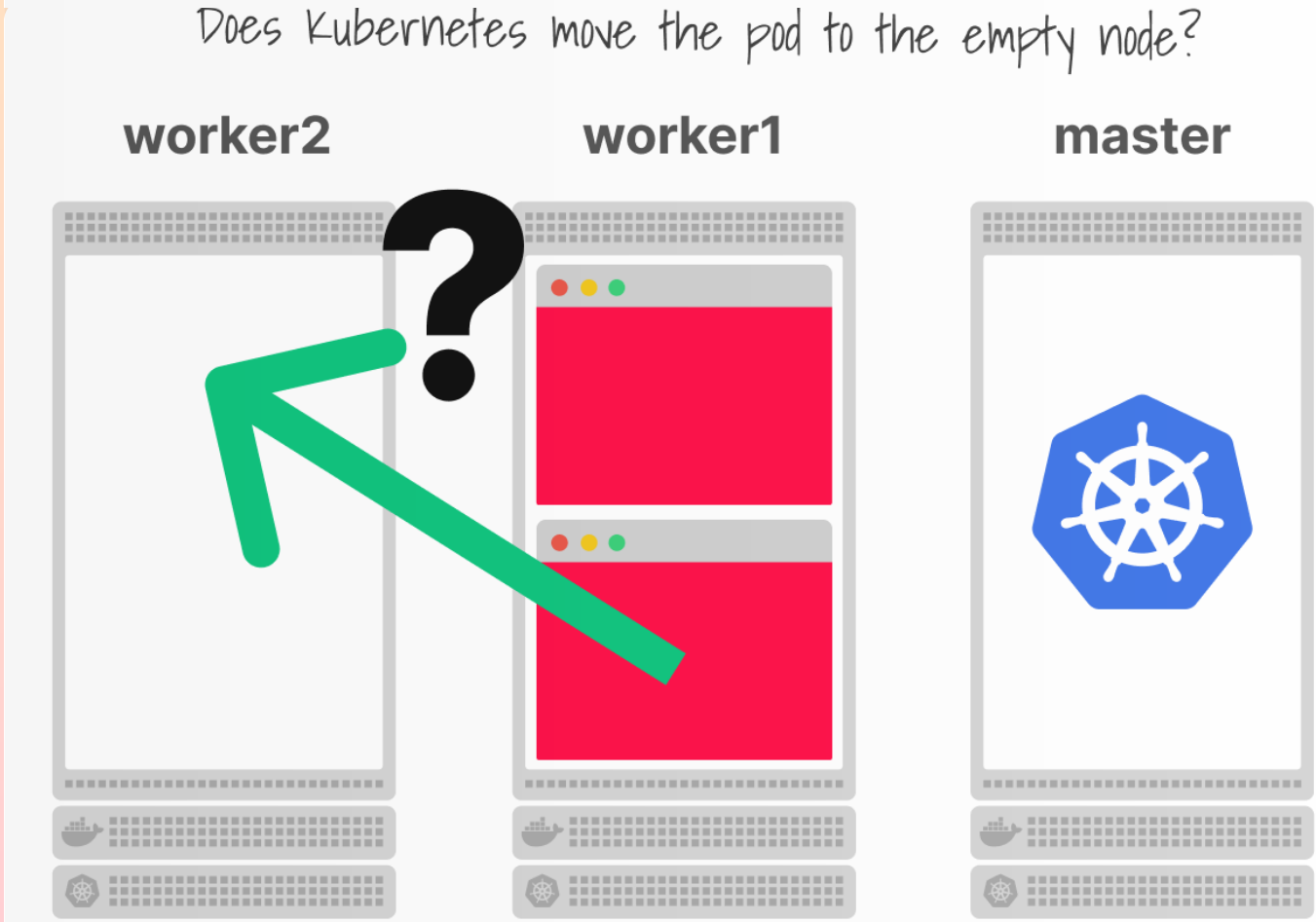
You provision a second node; what happens next?

Does Kubernetes notice that there's a space for your Pod?

Does it move the second Pod and rebalance the cluster?

Unfortunately, it does not

But why?



8:16 PM · Apr 3, 2023

5 Likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

3/

When you define a Deployment, you specify:

- The template for the Pod
- The number of copies (replicas)

```yaml
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:1.14.2
```

pod definition

8:16 PM · Apr 3, 2023

3 Likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

4/

But nowhere in that file you said you want one replica for each node!

The ReplicaSet counts 2 Pods, and that matches the desired state

Kubernetes won't take any further action

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nginx-deployment
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
      - name: nginx
        image: nginx:1.14.2
```

There's no field for "rebalance"!

8:17 PM · Apr 3, 2023

7 Likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

5/

In other words, Kubernetes does not rebalance your pods automatically

But you can fix this

There are three popular options:

① Pod (anti-)affinity
② Pod topology spread constraints
③ The Descheduler

8:17 PM · Apr 3, 2023

10 Likes

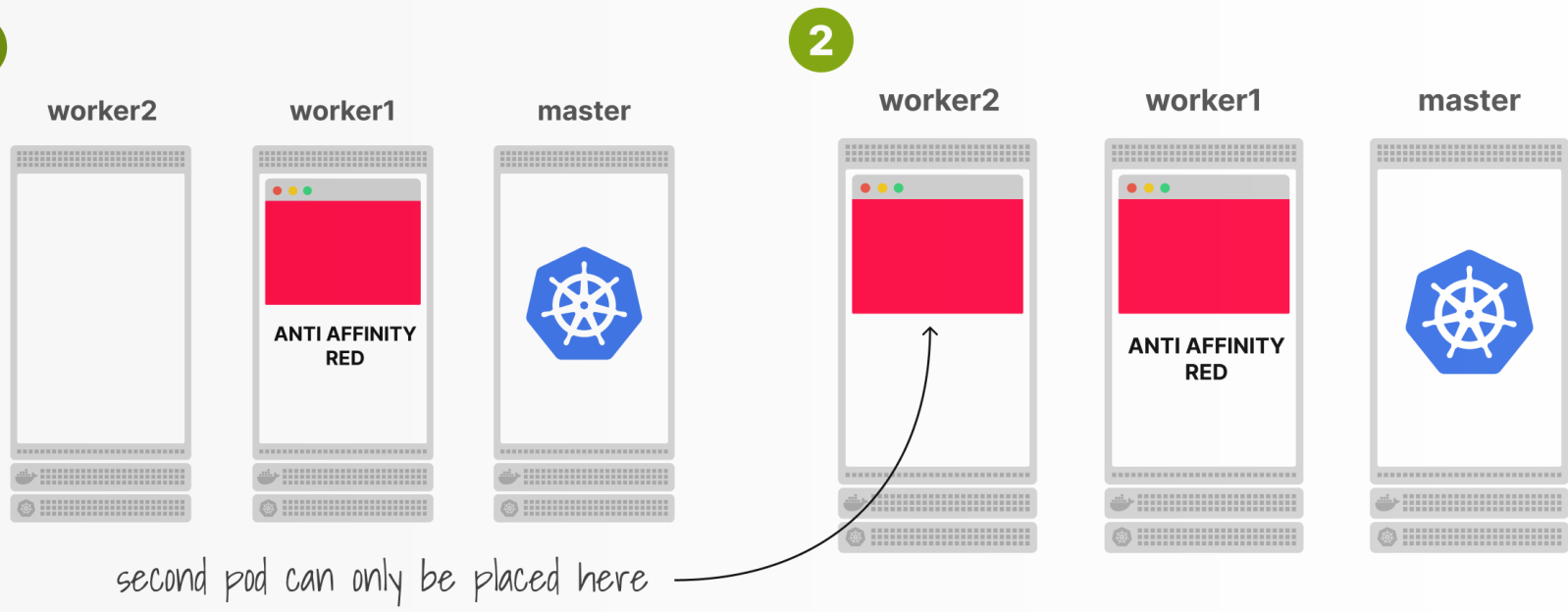
Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

6/

The first option is to use pod anti-affinity

With pod anti-affinity, your Pods repel other pods with the same label, forcing them to be on different nodes

You can read more about pod anti-affinity here:
kubernetes.io/docs/concepts/...



8:18 PM · Apr 3, 2023

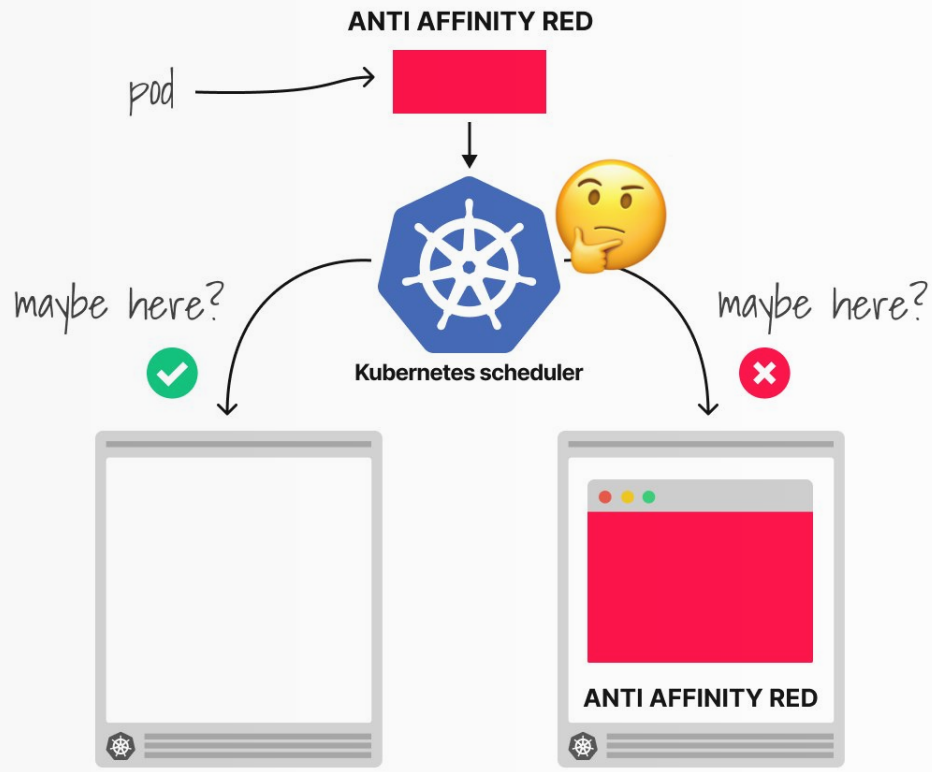
8 Likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

7/

Notice how pod affinity is evaluated when the scheduler allocates the pods

It is not applied retroactively, so you might need to delete a few pods to force the scheduler to recompute the allocations



8:18 PM · Apr 3, 2023

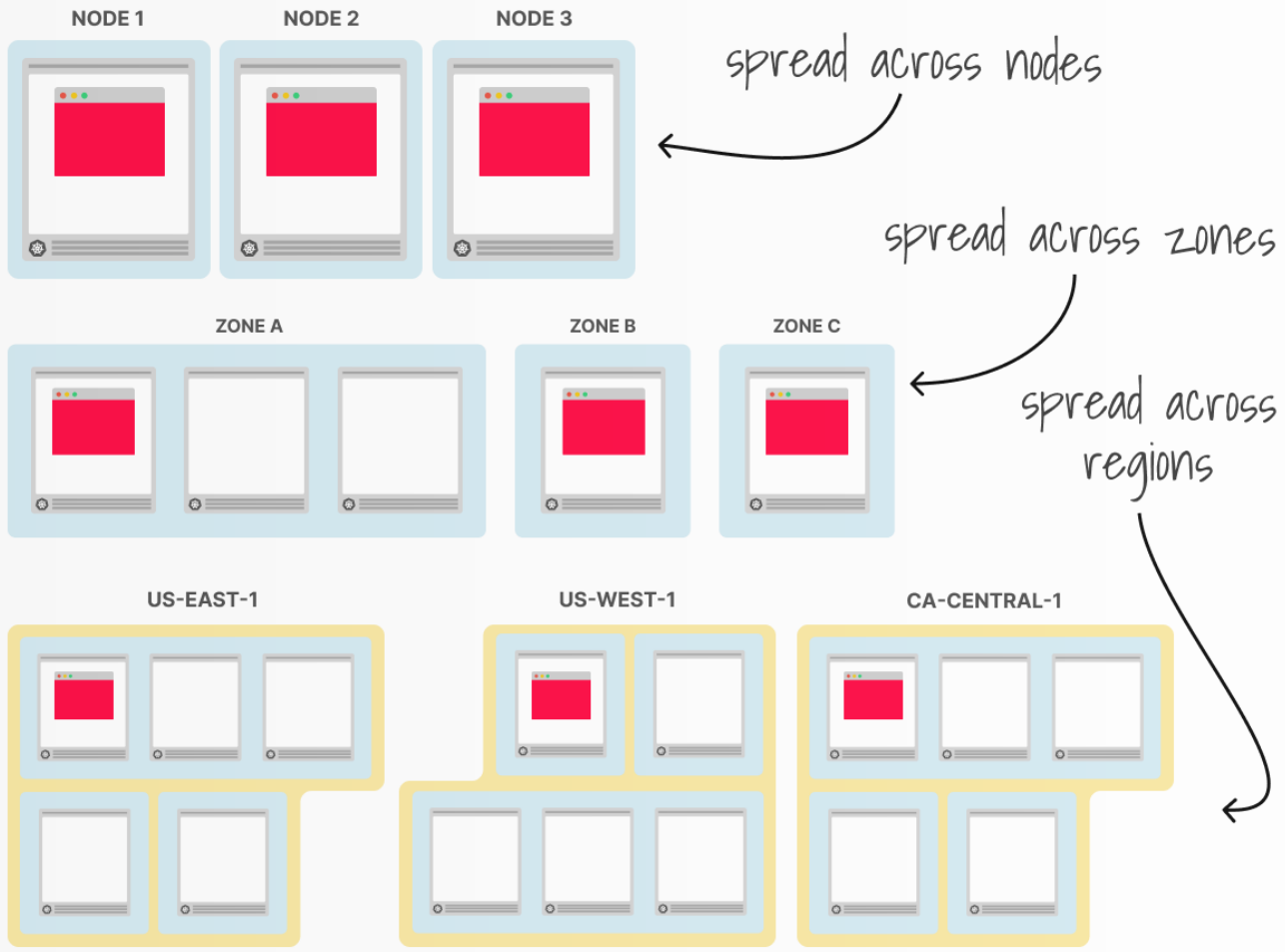
2 Likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

8/

Alternatively, you can use topology spread constraints to control how Pods are spread across your cluster among failure domains such as regions, zones, nodes, etc

This is similar to pod affinity but more powerful



8:19 PM · Apr 3, 2023

5 Likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

9/

With topology spread constraints, you can pick the topology and choose the pod distribution (skew), what happens when the constraint is unfulfillable (schedule anyway vs don't) and the interaction with pod affinity and taints

```
kind: Pod
apiVersion: v1
metadata:
  name: mypod
  labels:
    foo: bar
spec:
  topologySpreadConstraints:
  - maxSkew: 1 (1)
    topologyKey: zone (2)
    whenUnsatisfiable: DoNotSchedule (3)
    labelSelector:
      matchLabels:
        foo: bar
  containers:
  - name: pause
    image: registry.k8s.io/pause:3.1
```

the degree to which Pods may be unevenly distributed

this could be node, region, zone, rack, etc.

what happens when the scheduler can't fulfil the request?

8:20 PM · Apr 3, 2023

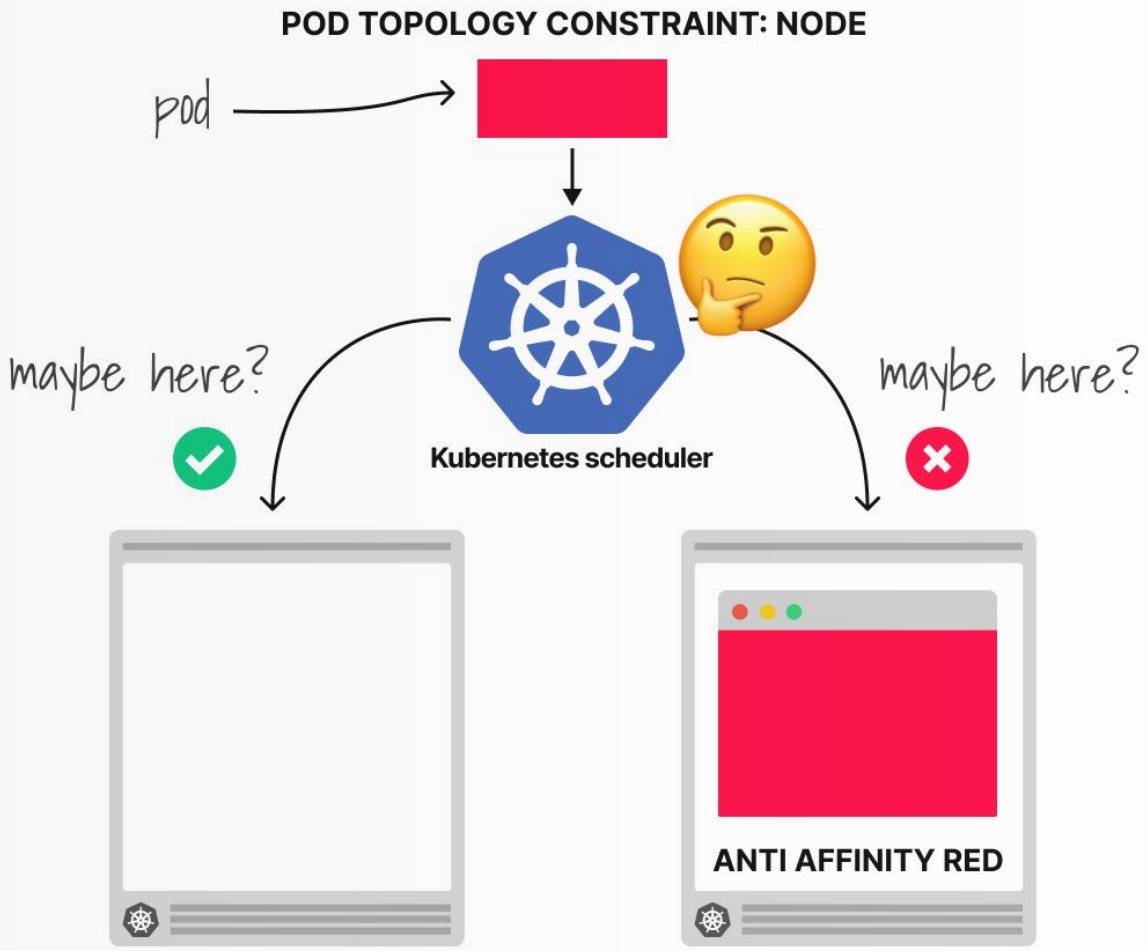
9 Likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

10/

However, even in this case, the scheduler evaluates topology spread constraints when the pod is allocated

It does not apply retroactively — you can still delete the pods and force the scheduler to reallocate them



8:20 PM · Apr 3, 2023

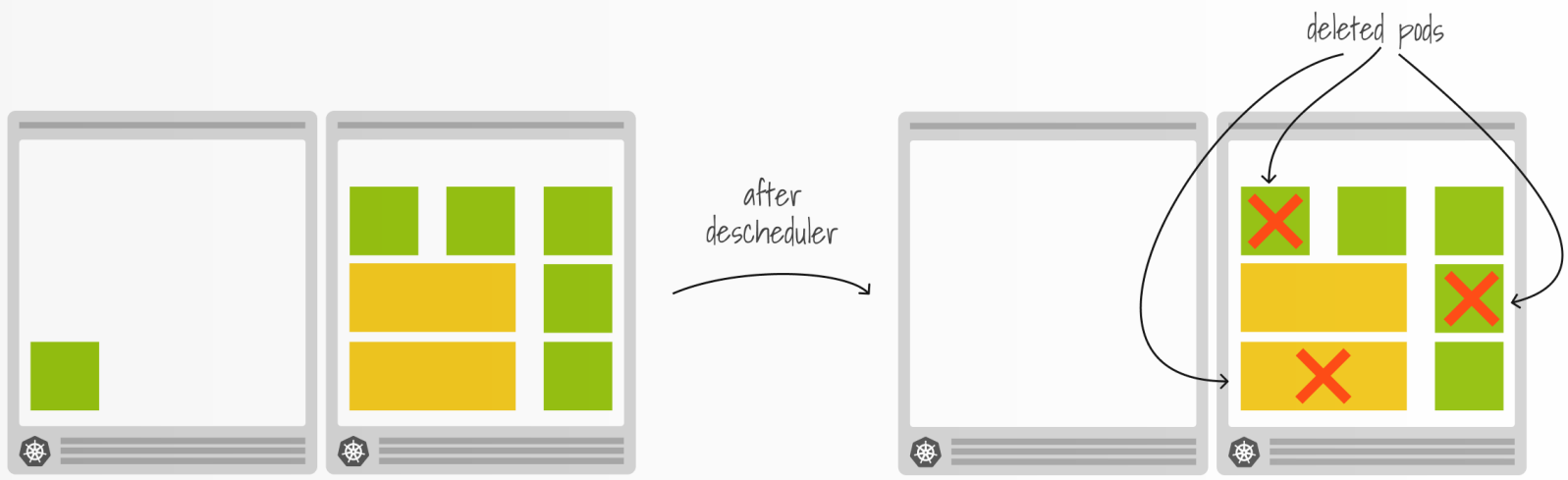
2 Likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

11/

If you want to rebalance your pods dynamically (not just when the scheduler allocates them), you should check out the Descheduler

The Descheduler scans your cluster at regular intervals, and if it finds a node that is more utilized than others, it deletes a pod in that node



8:20 PM · Apr 3, 2023

7 Likes

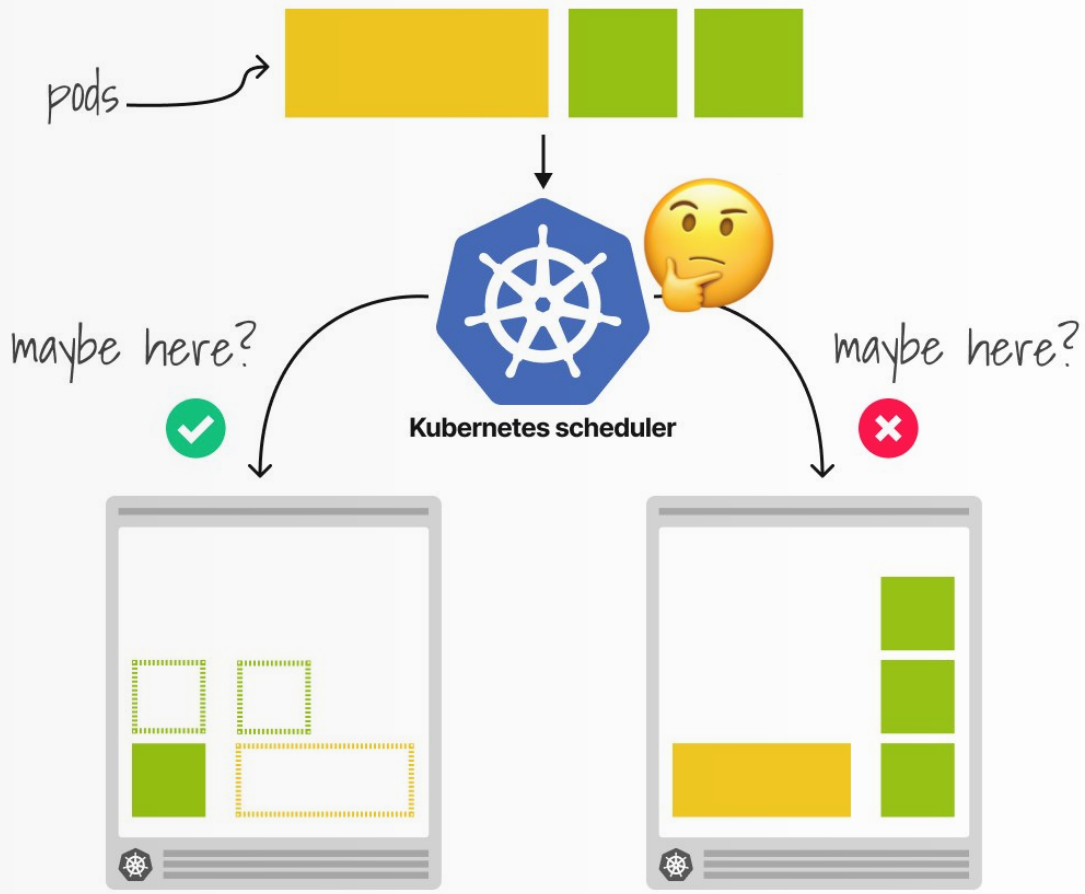
Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

12/

What happens when a Pod is deleted?

The ReplicaSet will create a new Pod, and the scheduler will likely place it in a less utilized node

If your pod has topology spread constraints or pod affinity, it will be allocated accordingly



8:21 PM · Apr 3, 2023

4 Likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

13/

The Descheduler can evict pods based on policies such as:

- Node utilization
- Pod age

- Failed pods
- Duplicates
- Affinity or taints violations

Evictor Plugin configuration

Name	type	Default Value	Description
`nodeSelector`	`string`	`nil`	limiting the nodes which are processed
`evictLocalStoragePods`	`bool`	`false`	allows eviction of pods with local storage
`evictSystemCriticalPods`	`bool`	`false`	[Warning: Will evict Kubernetes system pods] allows eviction of pods with any priority, including system pods like kube-dns
`ignorePvcPods`	`bool`	`false`	set whether PVC pods should be evicted or ignored
`evictFailedBarePods`	`bool`	`false`	allow eviction of pods without owner references and in failed phase
`labelSelector`	`metav1.LabelSelector`		(see label filtering)
`priorityThreshold`	`priorityThreshold`		(see priority filtering)
`nodeFit`	`bool`	`false`	(see node fit filtering)

8:21 PM · Apr 3, 2023

9 Likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

14/

If your cluster has been running long, the resource utilization is not very balanced

The following two strategies can be used to rebalance your cluster based on CPU, memory or number of pods

```
apiVersion: "descheduler/v1alpha2"
kind: "DeschedulerPolicy"
profiles:
  - name: ProfileName
    pluginConfig:
    - name: "DefaultEvictor"
    - name: "LowNodeUtilization"
      args:
        thresholds:
          "memory": 20
        targetThresholds:
          "memory": 70
    plugins:
      evict:
        enabled:
          - "DefaultEvictor"
      balance:
        enabled:
          - "LowNodeUtilization"
```

Balance high utilization nodes

```
if (nodes memory > 70%) {
  move pods to nodes with 20%< memory
}
```

```
apiVersion: "descheduler/v1alpha2"
kind: "DeschedulerPolicy"
profiles:
  - name: ProfileName
    pluginConfig:
    - name: "DefaultEvictor"
    - name: "HighNodeUtilization"
      args:
        thresholds:
          "memory": 20
    plugins:
      evict:
        enabled:
          - "DefaultEvictor"
      balance:
        enabled:
          - "HighNodeUtilization"
```

Balance low utilization nodes

8:21 PM · Apr 3, 2023

8 Likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

15/

Another practical policy is preventing developers and operators from treating pods like virtual machines

You can use the descheduler to ensure pods only run for a fixed time (e.g. 7 days)

```
apiVersion: "descheduler/v1alpha2"
kind: "DeschedulerPolicy"
profiles:
  - name: ProfileName
    pluginConfig:
    - name: "DefaultEvictor"
    - name: "PodLifeTime"
      args:
        maxPodLifeTimeSeconds: 604800
    plugins:
      evict:
        enabled:
          - "DefaultEvictor"
      deschedule:
        enabled:
          - "PodLifeTime"
```

pods can live up to 7 days

8:22 PM · Apr 3, 2023

6 Likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

16/

And lastly, you can combine the Descheduler with Node Problem Detector and Cluster Autoscaler to automatically remove Nodes with problems

The Descheduler can be used to deschedule workloads from those Nodes

8:22 PM · Apr 3, 2023 · **1,393** Views

4 Likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

17/

The Descheduler is an excellent choice to keep your cluster efficiency in check, but it isn't installed by default

It can be deployed as a Job, CronJob or Deployment

More info: github.com/kubernetes-sig...

8:22 PM · Apr 3, 2023 · **1,543** Views

15 Likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

18/

Also, if you want to dig in more a few relevant links:

- kubernetes.io/docs/concepts/...
- kubernetes.io/docs/concepts/...
- github.com/kubernetes-sig...

8:22 PM · Apr 3, 2023

5 Likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

19/

And finally, if you've enjoyed this thread, you might also like:

- The Kubernetes workshops that we run at Learnk8s learnk8s.io/training
- This collection of past threads twitter.com/danielepolenci...
- The Kubernetes newsletter I publish every week learnk8s.io/learn-kubernet...

8:22 PM · Apr 3, 2023

7 Likes