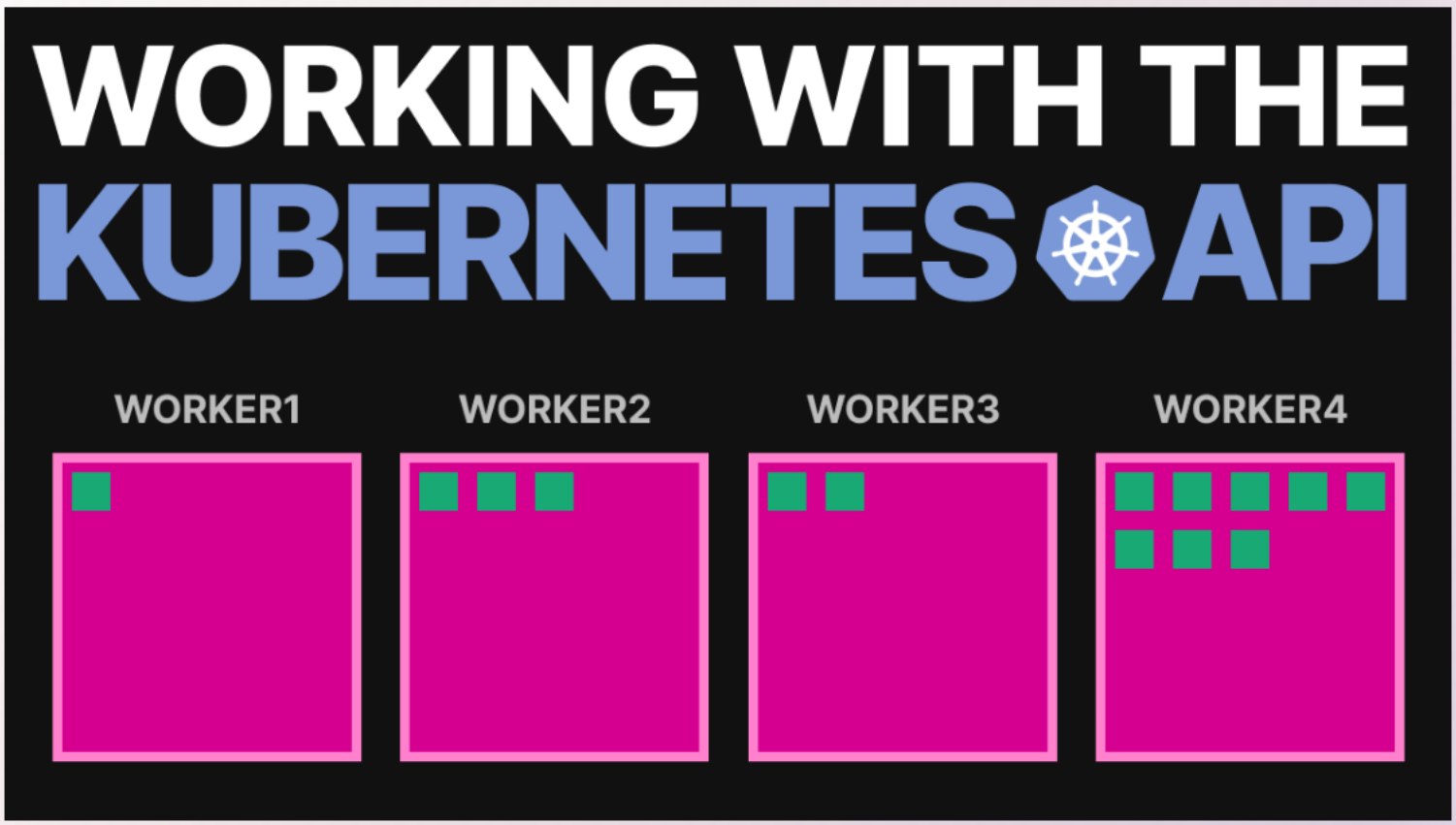
Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

Kubernetes exposes a robust API that lets you control every aspect of the cluster.

Most of the time, it's hidden behind kubectl, but no one stops you from using it directly.

Let's dive into it.



8:09 PM · 13 Mar, 2023

3 replies **23** shares **97** likes

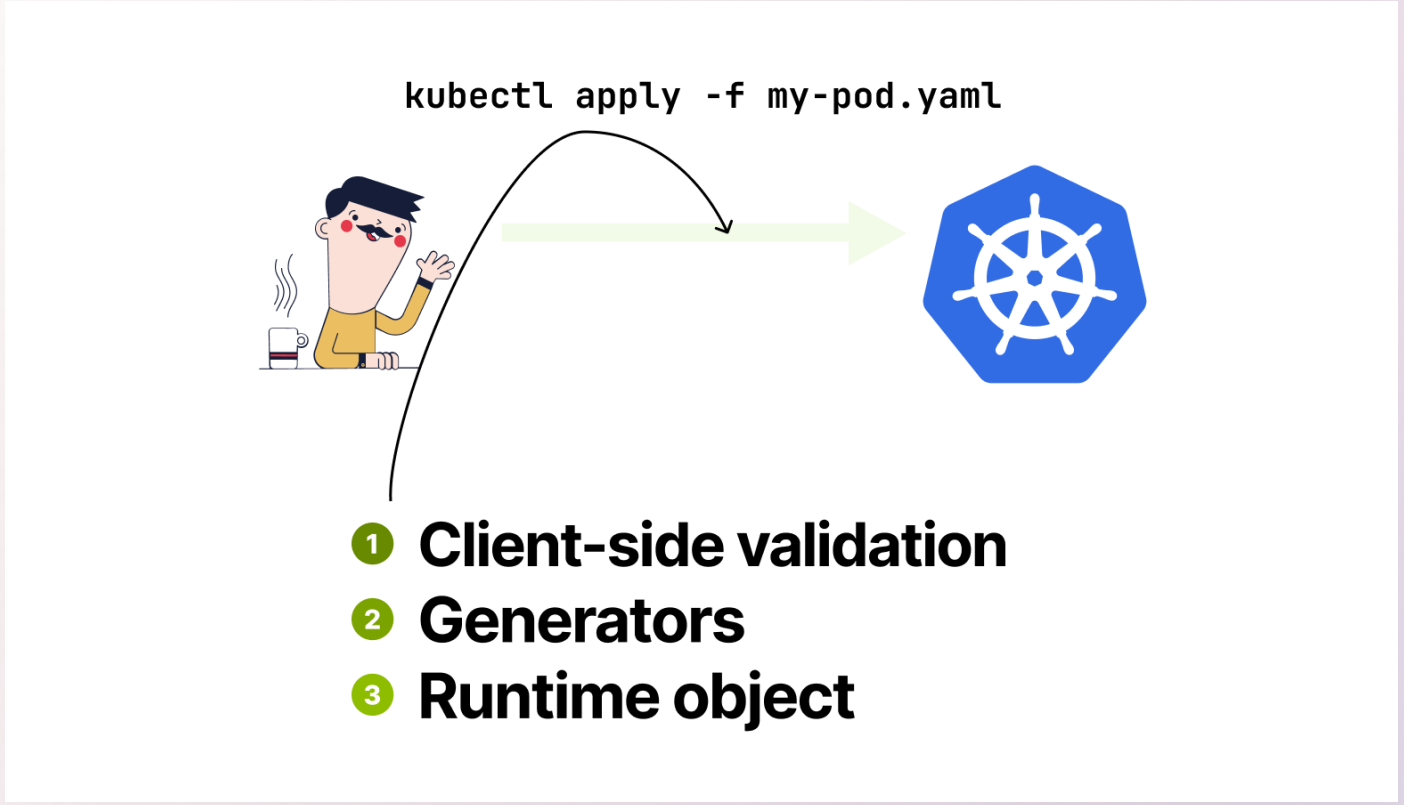
Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

1/

First, let's recap how the API works

When you type a command, kubectl:

- Validates the request client-side
- Generate YAML on the file (e.g. `kubectl run`)
- Construct the runtime object



8:09 PM · 13 Mar, 2023

1 reply

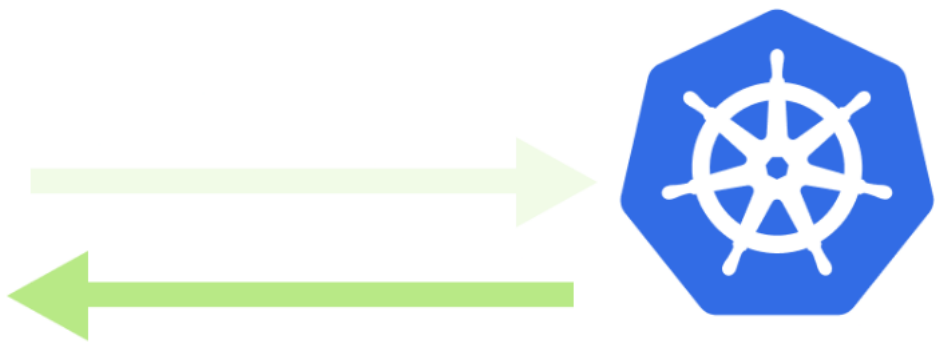
Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

2/

At this point, kubectl just acted locally without issuing any request to the cluster

As the next step, it queries the current API server and discovers all available API endpoints

kubectl apply -f my-pod.yaml



8:10 PM · 13 Mar, 2023

1 reply **1** likes

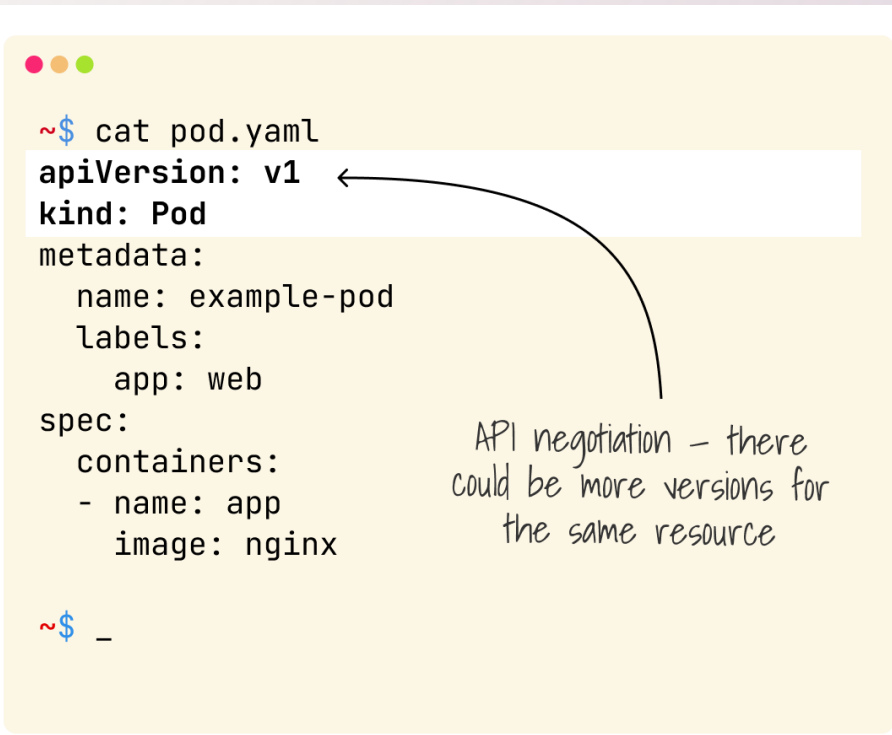
Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

3/

Finally, kubectl uses the runtime objected and endpoints to negotiate the correct API call

If your resource is a Pod, kubectl reads the `apiVersion` and `kind` fields and ensures those are available and supported in the cluster

Then it sends the request



8:10 PM · 13 Mar, 2023

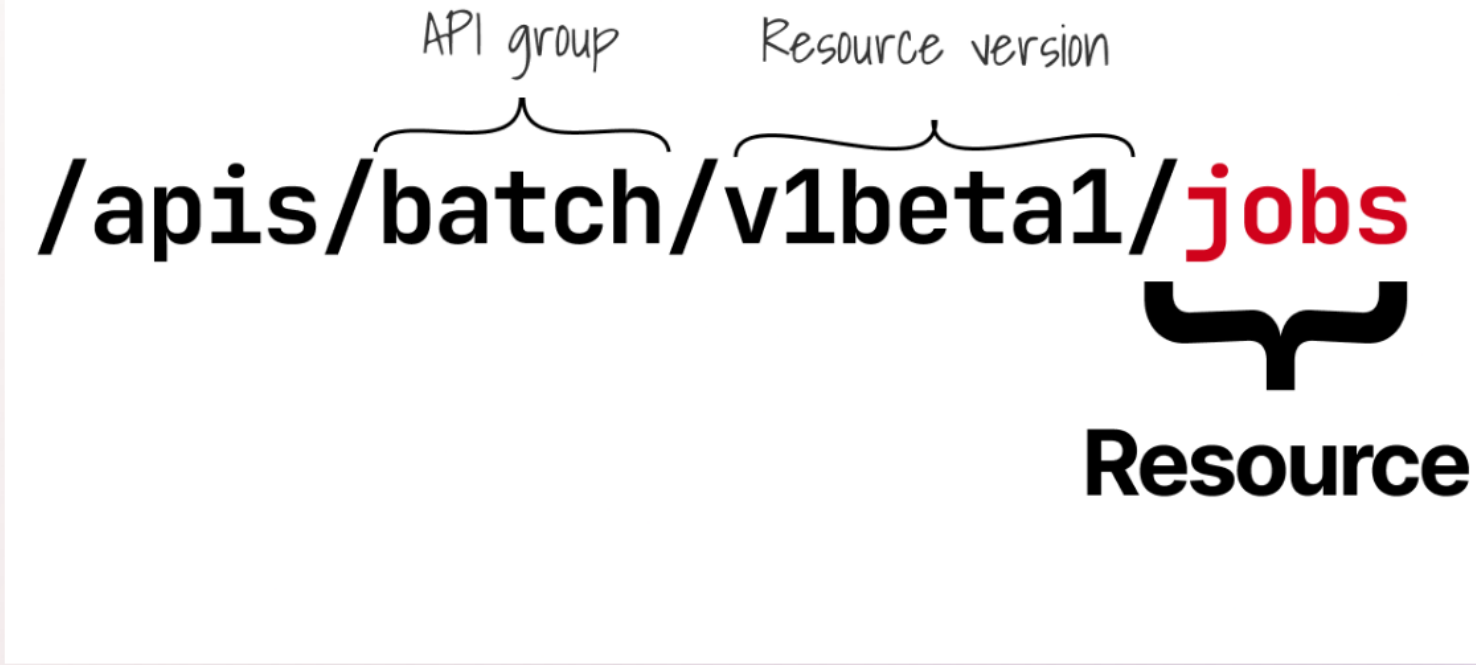
1 reply

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

4/

It's important to understand that APIs in Kubernetes are grouped

To further segregate multiple versions, resources are versioned



8:10 PM · 13 Mar, 2023

1 reply **1** shares **1** likes

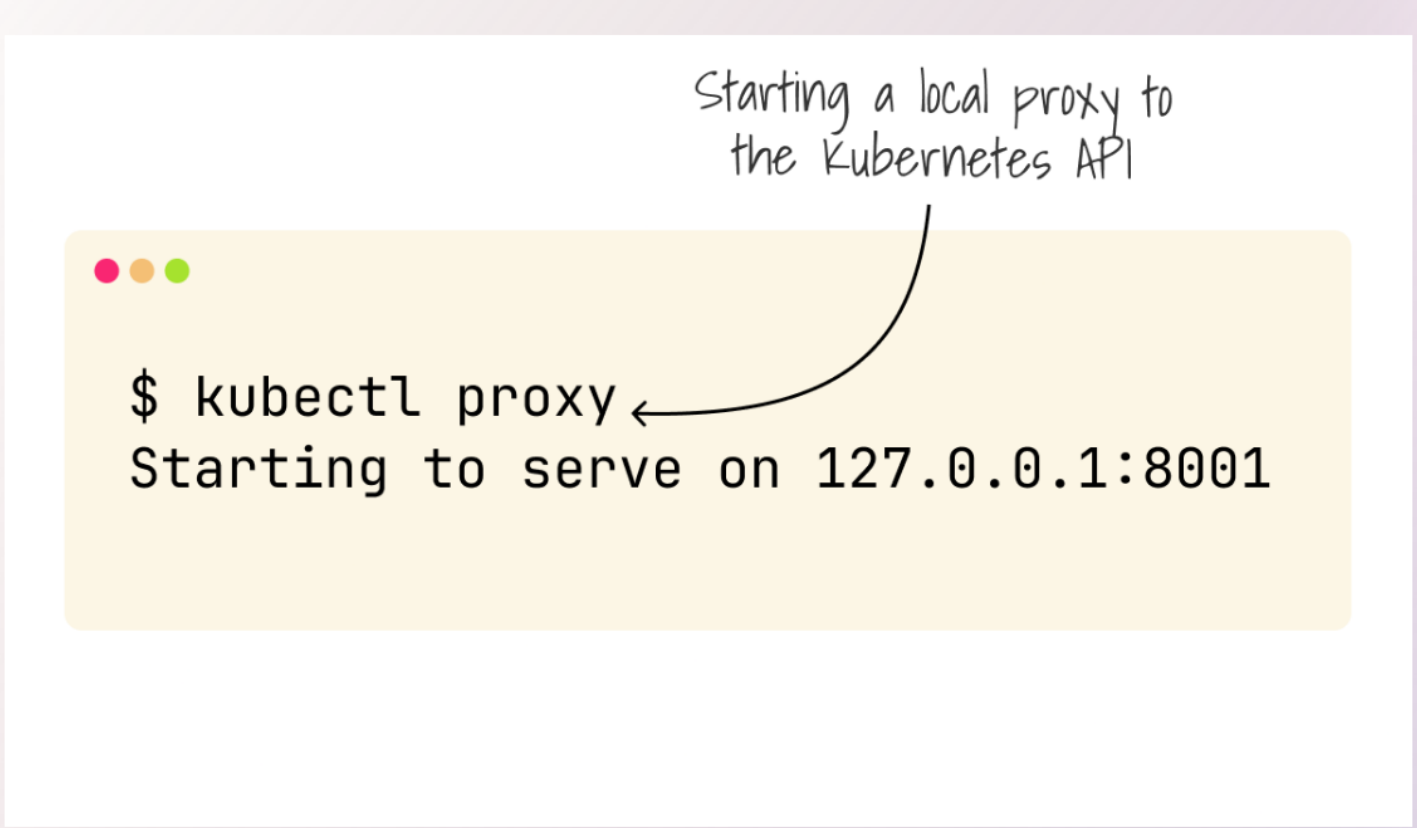
Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

5/

Now that you mastered the basics, let's look at an example

You can start a local tunnel to the API server with `kubectl proxy`

But how can you retrieve all deployments?



8:11 PM · 13 Mar, 2023

1 reply

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

6/

Deployments belong to the "app" group and have a `v1` version

You can list them with:

`curl localhost:8001/apis/apps/v1/namespaces/{namespace}/deployments`

```
$ curl localhost:8001/apis/apps/v1/namespaces/default/deployments

{
  "kind": "DeploymentList",
  "apiVersion": "apps/v1",
  "metadata": {
    "resourceVersion": "468"
  },
  "items": [
    {
      "metadata": {
        "name": "hello-world",
        "namespace": "default",
        "uid": "d5eaedc5-77f9-4b1d-a69e-e65b61a4fdb2",
        "resourceVersion": "465",
        "generation": 1,
```

Response is a list of Deployments

the first deployment

8:11 PM · 13 Mar, 2023

1 reply **1** shares

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

7/

What about listing all running pods?

Pods belong to the "" (empty) group and have a `v1` version

You can list them with:

`curl localhost:8001/api/v1/namespaces/{namespace}/pods`

```
$ curl localhost:8001/api/v1/namespaces/default/pods

{
  "kind": "PodList",
  "apiVersion": "v1",
  "metadata": {
    "resourceVersion": "555"
  },
  "items": [
    {
      "metadata": {
        "name": "hello-world-54f9544b48-fktn9",
        "generateName": "hello-world-54f9544b48-",
        "namespace": "default",
        "uid": "79985d81-355f-4986-89ce-366cb8917ddf",
        "resourceVersion": "477",
```

Response is a list of pods

the first pod

8:11 PM · 13 Mar, 2023

1 reply

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

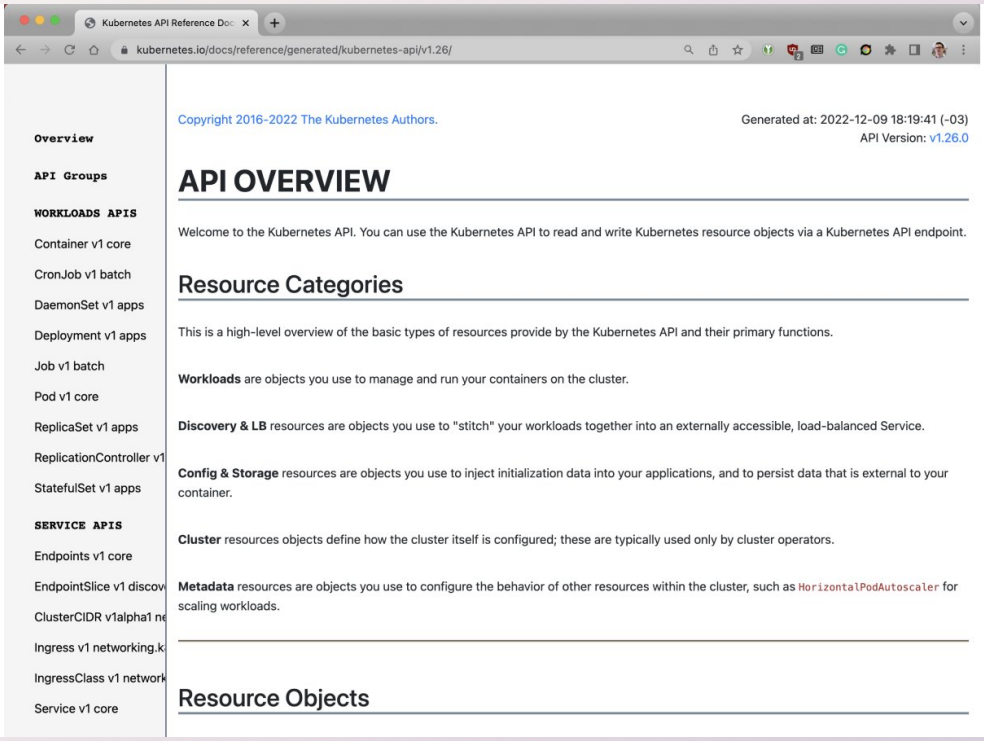
8/

The empty group is... weird — are there more exceptions?

Well, the reality is that there's an easier way to construct URLs

I usually use the Kubernetes API reference, as the paths are all neatly listed

kubernetes.io/docs/reference...



8:11 PM · 13 Mar, 2023

1 reply **1** shares

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

9/

Let's look at another example but this time with the help of the API reference

What if you want to be notified of changes to the pod?

That's called "watch" in the API, and the command is:

`GET /api/v1/watch/namespaces/{namespace}/pods/{name}`

```
# start the API locally
$ kubectl proxy &
Starting to serve on 127.0.0.1:8001

# retrieves incremental Pod updates from the API
$ curl localhost:8001/api/v1/pods?watch=1

# the response is a stream of JSON separated by new lines
{"type":"ADDED","object":{"kind":"Pod","apiVersion":"v1",/* more json */}}
{"type":"MODIFIED","object":{"kind":"Pod","apiVersion":"v1",/* more json */}}
{"type":"DELETED","object":{"kind":"Pod","apiVersion":"v1",/* more json */}}
```

8:12 PM · 13 Mar, 2023

1 reply **1** likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

10/

Great, but what's the point of all of this?

Accessing the API directly allows you to build scripts to automate tasks

Or you can build your own kubernetes extensions

Let me show you

8:12 PM · 13 Mar, 2023

1 reply **1** shares

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

11/

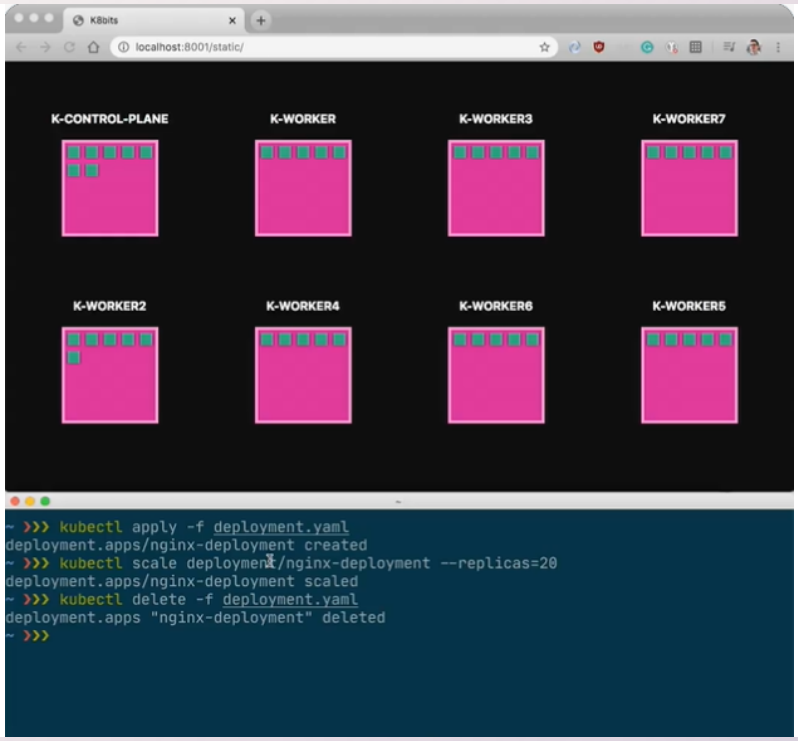
This is a small kubernetes dashboard in ~130 lines of Javascript

It uses 2 API calls:

① List all pods

② Watch for changes to pods

The remaining code is used to group and display the nodes



8:13 PM · 13 Mar, 2023

1 reply **2** shares **2** likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

12/

In Kubernetes, combining the listing and updating resources is so common that it's a pattern called shared informer

The JS/Typescript API has an excellent example of the shared informer in action

But it's just a fancy name for 2 GET requests (and some caching)

8:13 PM · 13 Mar, 2023

1 reply **1** shares

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

13/

But the API doesn't stop at reading resources

You can also create new resources and amend existing ones

For example, you can amend the replicas for a deployment with:

`PATCH /apis/apps/v1/namespaces/{namespace}/deployments/{name}`

```
$ curl -X PATCH \
  127.0.0.1:8001/apis/apps/v1/namespaces/default/deployments/hello-world \
  -d '{"spec": {"replicas":2}}' \
  -H 'Content-Type: application/strategic-merge-patch+json'

{
  "kind": "Deployment",
  "apiVersion": "apps/v1",
  "metadata": {
    "name": "hello-world",
    "namespace": "default",
    "uid": "d5eaedc5-77f9-4b1d-a69e-e65b61a4fdb2",
    "resourceVersion": "928",
    "generation": 2
  },
  "spec": {
    "replicas": 2,
    "selector": {
      "matchLabels": {
        "name": "app"
      }
    },
```

patched to 2 replicas!

8:13 PM · 13 Mar, 2023

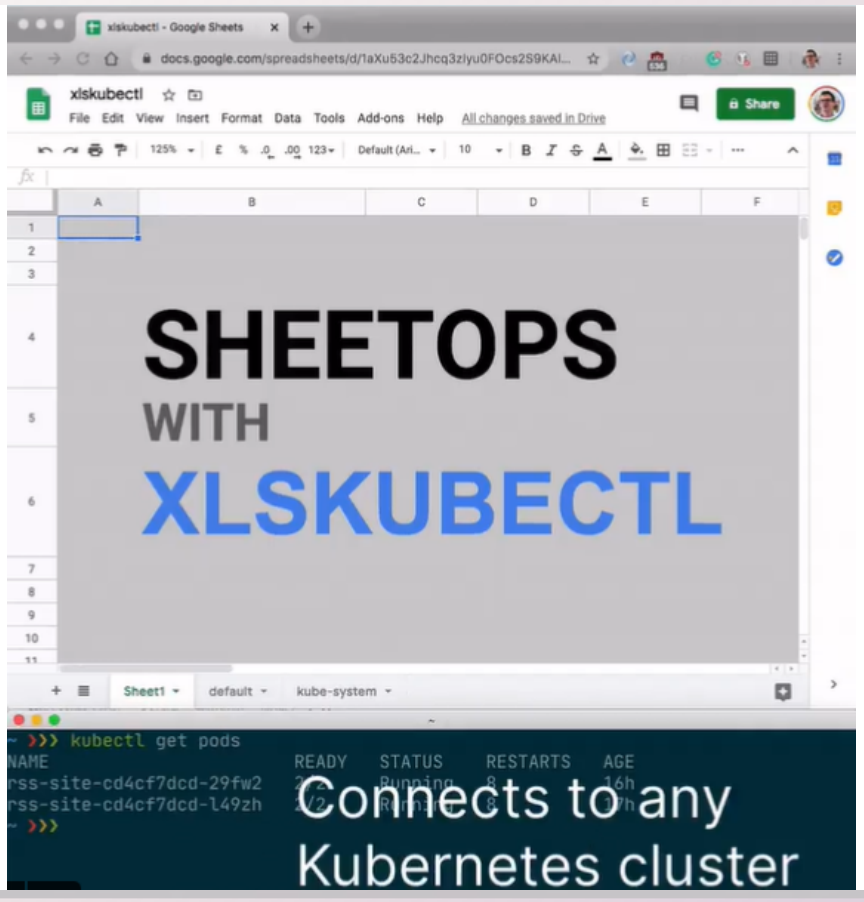
1 reply

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

14/

To experiment, I built something... unconventional

xlskubectl is my attempt at controlling a kubernetes cluster using Excel/Google Sheets



8:14 PM · 13 Mar, 2023

1 reply **2** shares **3** likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

15/

The code is remarkably similar to the Javascript dashboard:

① It uses the shared informer (almost)

② It polls for updates from google sheets

③ It renders everything as cells

8:14 PM · 13 Mar, 2023

1 reply

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

16/

Is it a good idea? Probably not

Hopefully, it helps you realize the potential of using the Kubernetes API directly

Also, none of this code was written in Go — you can use any programming language

8:14 PM · 13 Mar, 2023

1 reply **1** shares

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

17/

Relevant links:

- github.com/learnk8s/xlsku...
- github.com/learnk8s/k8bit
- iximiuz.com/en/series/work...

8:14 PM · 13 Mar, 2023

1 reply **2** shares **3** likes

Daniele Polencic — @danielepolencic@hachyderm.io
@danielepolencic

18/

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

- The Kubernetes newsletter I publish every week
learnk8s.io/learn-kubernet...

8:15 PM · 13 Mar, 2023

1 shares **2** likes