

Fast Iteration on Knative Applications Using Tilt



Evan Anderson

23 Feb 2022

The Problem

Local development is *super fast*.

Just save your files and tools like nodemon and react start will automatically reload your application. This is super-convenient for debugging!

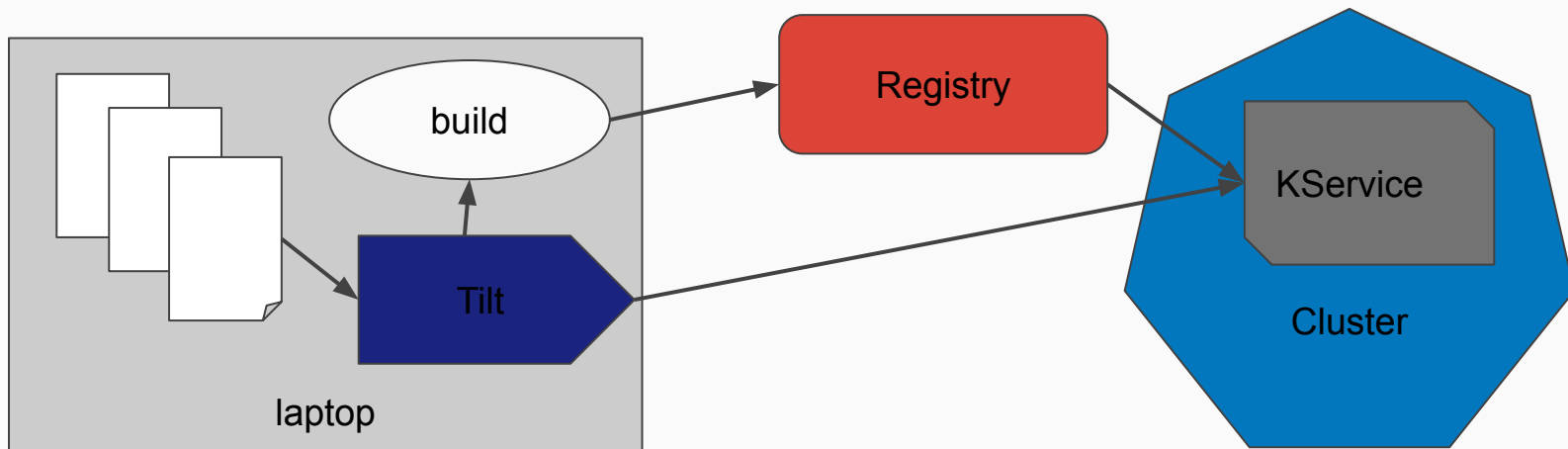
... **but**

Running your application locally is different than running in elsewhere. Especially if you're running locally, rather than in Docker.

But even if you're running in Docker, the rest of your environment is different (secrets, DNS names, etc)!

Tilt to the Rescue!

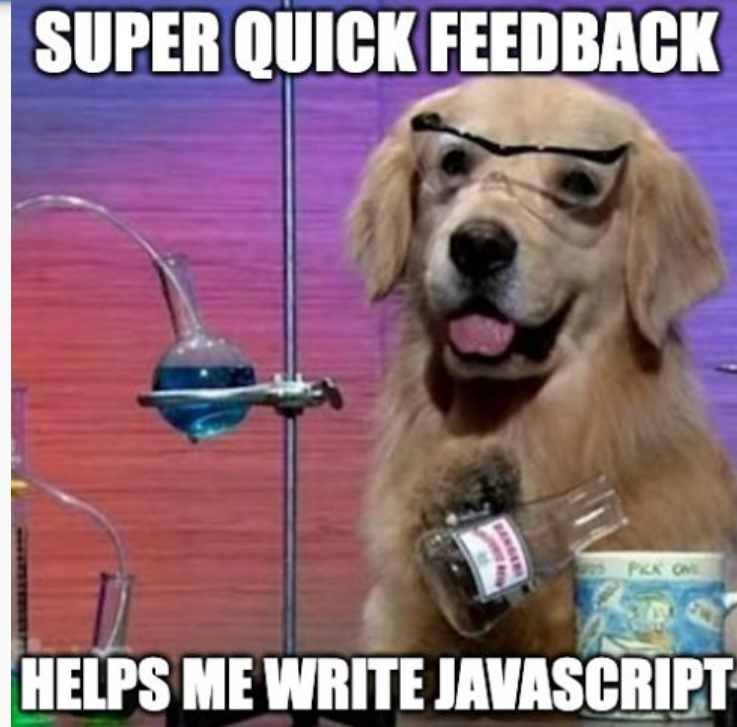
Tilt tracks changes in your local project directory, and enables rebuilding images and re-applying manifests.



... so what?

This is super quick and convenient for experimenting with!

And it can be a big confidence booster when learning something new.



Demo!

What Happened?

I added two small files to my existing React + Express typescript app:

20 lines, Skylark (python subset)

Tiltfile

18 lines, Kubernetes yaml

Knative manifest

Tiltfile

Build with buildpacks

Define Knative Services, and where the “image” is within them.

```
load('ext://pack', 'pack')
k8s_kind('Service', api_version='serving.knative.dev/v1',
        image_json_path='{.spec.template.spec.containers[*].image}')
```

```
allow_k8s_contexts('tilt-demo-admin@tilt-demo')
```

Okay to stomp resources in this context.

```
pack(
    'registry.kn-tilt.majordemo.com/tilt-demo-2022',
    ignore=[ 'Tiltfile', 'service.yaml', '.*' ],
    live_update = [
        fall_back_on(['package.json', 'build/package.json']),
        sync('.', '/workspace'),
    ],
    deps=["."],
)
k8s_yaml('service.yaml')
k8s_resource('tilt-demo', extra_pod_selectors={'serving.knative.dev/service': 'tilt-demo'})
```

Buildpack to build the Javascript

Deploy this

How to find the pods for the Knative service

Knative Service

```
apiVersion: serving.knative.dev/v1
kind: Service
metadata:
  name: tilt-demo
spec:
  template:
    spec:
      containers:
        - image: registry.kn-tilt.majordemo.com/tilt-demo-2022:first
          command: ['no-reload']
          env:
            - name: DANGEROUSLY_DISABLE_HOST_CHECK
              value: "true"
          ports:
            - containerPort: 3000
              protocol: TCP
```

Is overwritten by the Tiltfile

Need this for react start to proxy requests to the express server ("proxy" directive).

The port for
react-scripts start

How is it so fast?

It cheats!

Tilt supports a concept called `live_update`, which can sync files directly to the running container(s), and then restart the processes if needed.

In our case, we're running `nodemon`, so we only need to sync the files that changed.

How Is It So Fast? Part 2

Tilt also tracks which items have changed, and only performs updates for components which have changed.

No `k8s_yaml` change → no k8s resource update

You can also manually trigger steps if needed, and add non-k8s steps (like clearing a database).

Stuff That's Changed / Improved

Since 0.23, Tilt `live_update` supports multiple pods. Prior to this, you needed to play with `min-scale` and `max-scale` and revision labels to avoid rollouts.

Tilt defaults to working with Docker, but the `custom_build` extension makes it easy to plug in other tools, especially with tilt extensions. I used this to avoid needing to mess with a Dockerfile.

Some other cool things in the demo

I'm using an [on-cluster registry](#) for storing the images. This may not be faster, but it makes it easy to clean up when I'm done.

I'm using [buildpacks](#), rather than needing to write my own Docker file. When I do need to do a buildpack build, it takes about 3-4 minutes running locally, but it doesn't happen often.

VSCode has a built-in browser, so you can have your app on one side and your code on the other.

The Tilt extension quacks when it encounters an error.

Demo Source

<https://github.com/evankanderson/knative-tilt-demo-2022>

@evankanderson