

How to Build an Effective Container-Based Local Development Environment

Martino Fornasa - @mforNASA - <https://m.fornasa.it>

Local Development Workflow

- Fast developer onboarding
- Software dependency management
- Production-like environment
- Local testing
- Build knowledge

**Creating a local development
environment is hard**

.....

**Creating a local development
environment is hard and very
language-specific**

Node.js Dockerfile Tricks

```
# Dockerfile

...
ADD package.json yarn.lock /cache/
ADD .yarn-cache.tgz /

RUN cd /cache \
  && yarn \
  && cd /opt/app && ln -s /cache/node_modules node_modules \
  && tar czf /.yarn-cache.tgz /usr/local/share/.cache/yarn

COPY . /opt/app
```

So I rolled my own solution

```
# Dockerfile
FROM some-node-on-alpine
COPY . /opt/app
RUN yarn
CMD ["yarn", "run", "watch"]
```

```
# index.js
var express = require('express');
var app = express();
var server =
  app.listen(3000,function() {
    console.log("Listening on port 3000");
});
```

```
# docker-compose.yml
services:
  app:
    build: app/
    command: yarn run start-dev
    depends_on:
      - db
    ports:
      - '8080:8080'
    volumes:
      - ./app/:/opt/app
  db:
    image: postgres
    restart: always
    environment:
      ...
    ports:
      - '5432:5432'
```

What could have possibly gone wrong?

Speed

Sync Issues

Cleanup

@mfornasa

Docker: setup an easy way to cleanup old images/containers (local development) #3911

Closed

opened this issue on 23 Feb 2016 · 0 comments



commented on 23 Feb 2016 + ...

I'm still having issues with . After every change of the npm file, I have to go through the process of manually cleaning docker up or I can't start / `docker/run` it anymore.

the type of error that i get:

Recreating docker_app_1

ERROR: Cannot start container 80347486b3630977bfd5423bea4eb4373aaaf8373184bd292c992173829cd4

Assignees



Labels

raising the bar



Projects

None yet

Edit

New issue

Fix Docker local env #3807

Closed

mfornasa opened this issue on 11 Feb 2016 · 7 comments



mfornasa commented on 11 Feb 2016 + ...

The local env for our Boilerplate is not yet good enough.

Both @ and @ had issues, related to rsync not really working in Linux () and issues with npm nasty behaviour (thousands of connections) ().

We need to fix it.

Assignees



Labels

prt-normal

raising the bar



Docker runs out of disk space (too many images) #5266

Closed

opened this issue on 8 Sep 2016 · 1 comment



commented on 8 Sep 2016 + ...

This is happening for all of our developers every week or so. I believe it is due to new images being created and not removed afterwards.

A simple fix would be to have a quick script that can clean out the docker images every once and a while.. definitely don't want to add something that slows down the docker/run start process, but a script that could run once a week or something that can clear out images not being used.

Assignees

No one—assign yourself



Labels

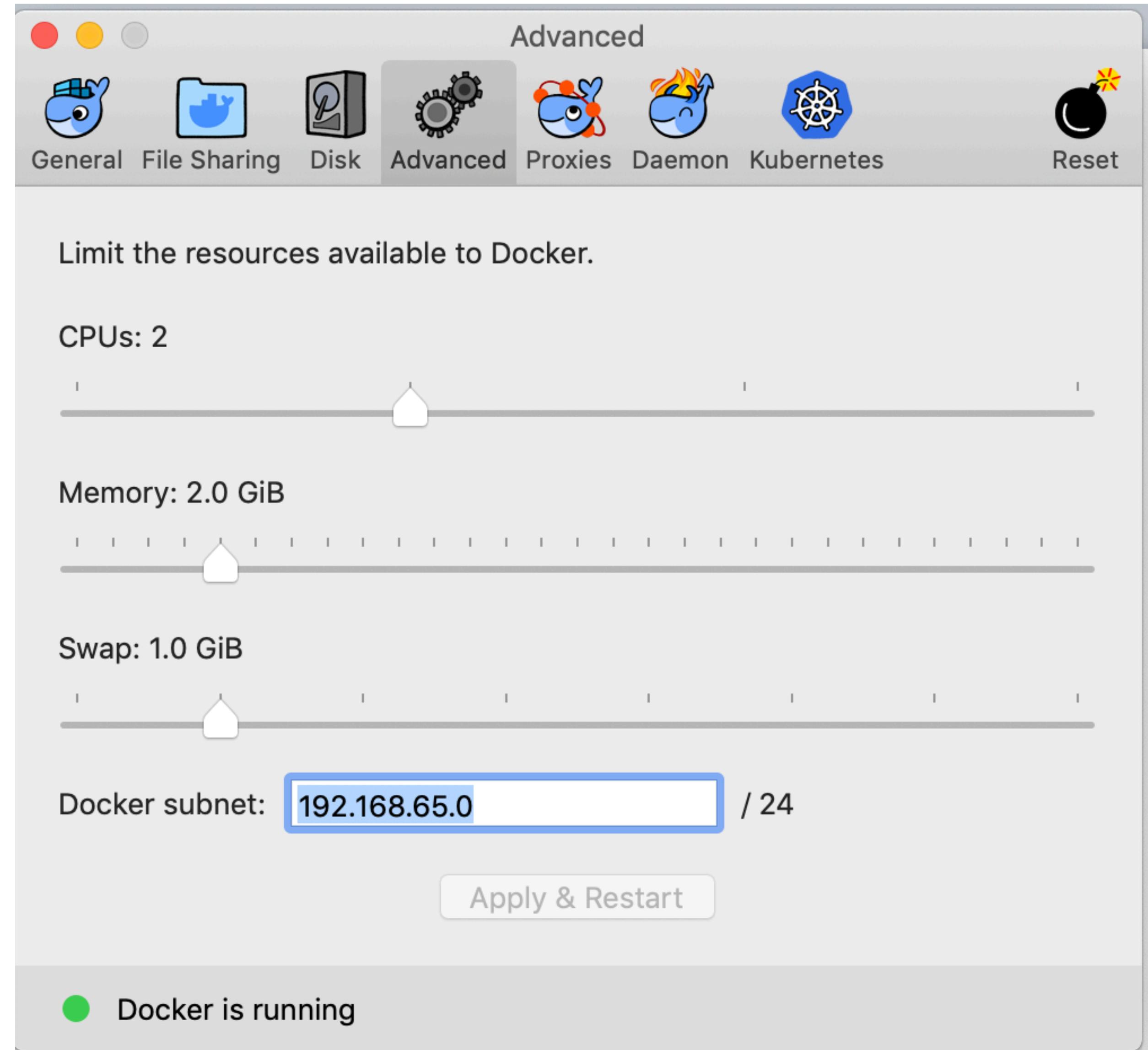
prt-normal



Projects



Resources



People hated it...

Implementation goals

- Reduce friction
- Make it easy to do the right thing
- Allow people to use their preferred tools (IDE, linters, debuggers, etc.)
- Low latency!

A development workflow for Docker Kubernetes applications

Benefits of a Kubernetes workflow

- Shift left the environmental parity
- Make it easier to setup complex environments (databases, cache engines, search engines)
- Same (or similar) tools for dev and deploy
- Microservices
- Each developer being able to run the full system (maybe also connecting to external systems)
- **CI tool builds the final image and enforce policies**

Three approaches

- 1. Auto rebuild and deploy**
- 2. Run on cluster + Sync filesystem**
- 3. Run locally + Bridge network**

1.

Auto rebuild and deploy

- Higher guarantee of correctness
- Easy to do on a remote cluster
- Better choice for compiled languages
- Speed issues



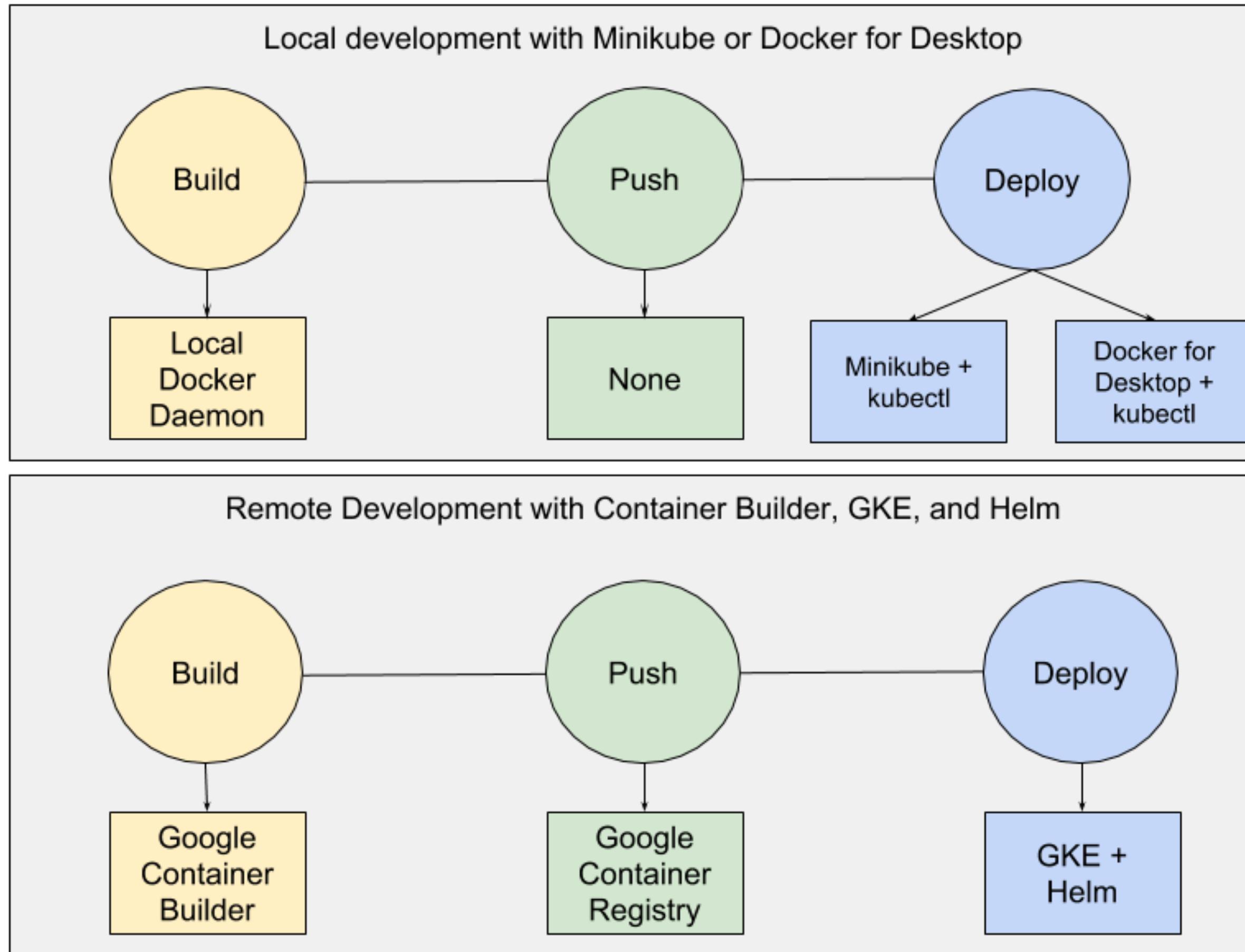


draft init
draft create
draft up
draft connect

- Build, push and deploy
- Detect language, generates Dockerfile + Helm Chart
- Supports out of the box: .NET, Go, Node, PHP, Java, Python, Ruby
- In progress:
 - Full-fledged remote debugging support
 - Improved support for multiple components



SKAFFOLD



- **Watches your source code and the dependencies of your Docker images for changes and runs a build and deploy when changes are detected**
- **Streams logs from deployed containers**
- **Continuous build-deploy loop, only warn on errors**



SKAFFOLD

```
apiVersion: skaffold/v1alpha4
kind: Config
build:
  artifacts:
  - image: gcr.io/k8s-skaffold/node-example
    context: backend
    sync:
      '*.js': .
deploy:
  kubectl:
    manifests:
    - "k8s/**"
```

2.

Run on cluster

Sync filesystem

- Higher chance of drifting from the image built by CI
- Better choice for interpreted languages
- Usually faster





```
ksync init  
ksync watch &  
kubectl apply -f config.yaml  
ksync create --selector=app=app $  
  (pwd)/ksync /code  
ksync get
```

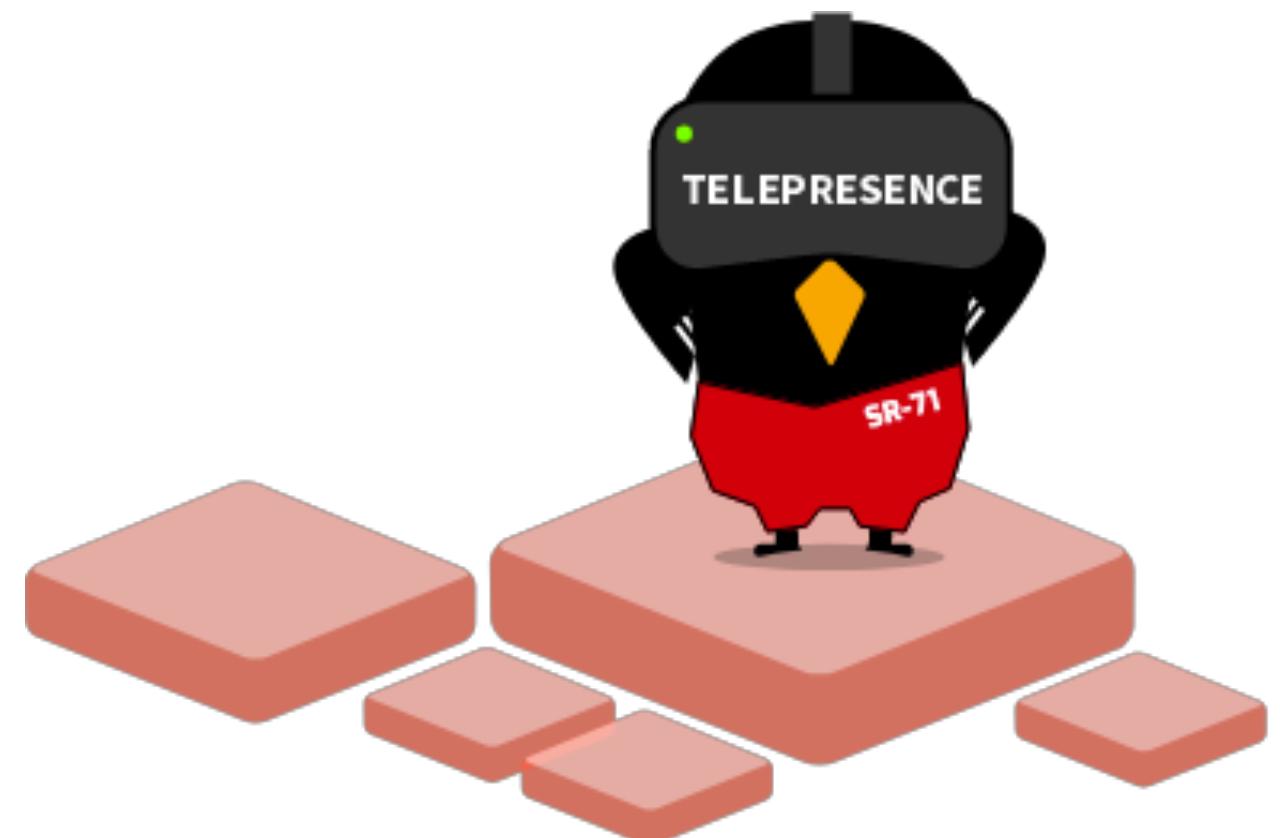
- Sync files with the container
- Server component (DaemonSet)
- In progress: remote command execution

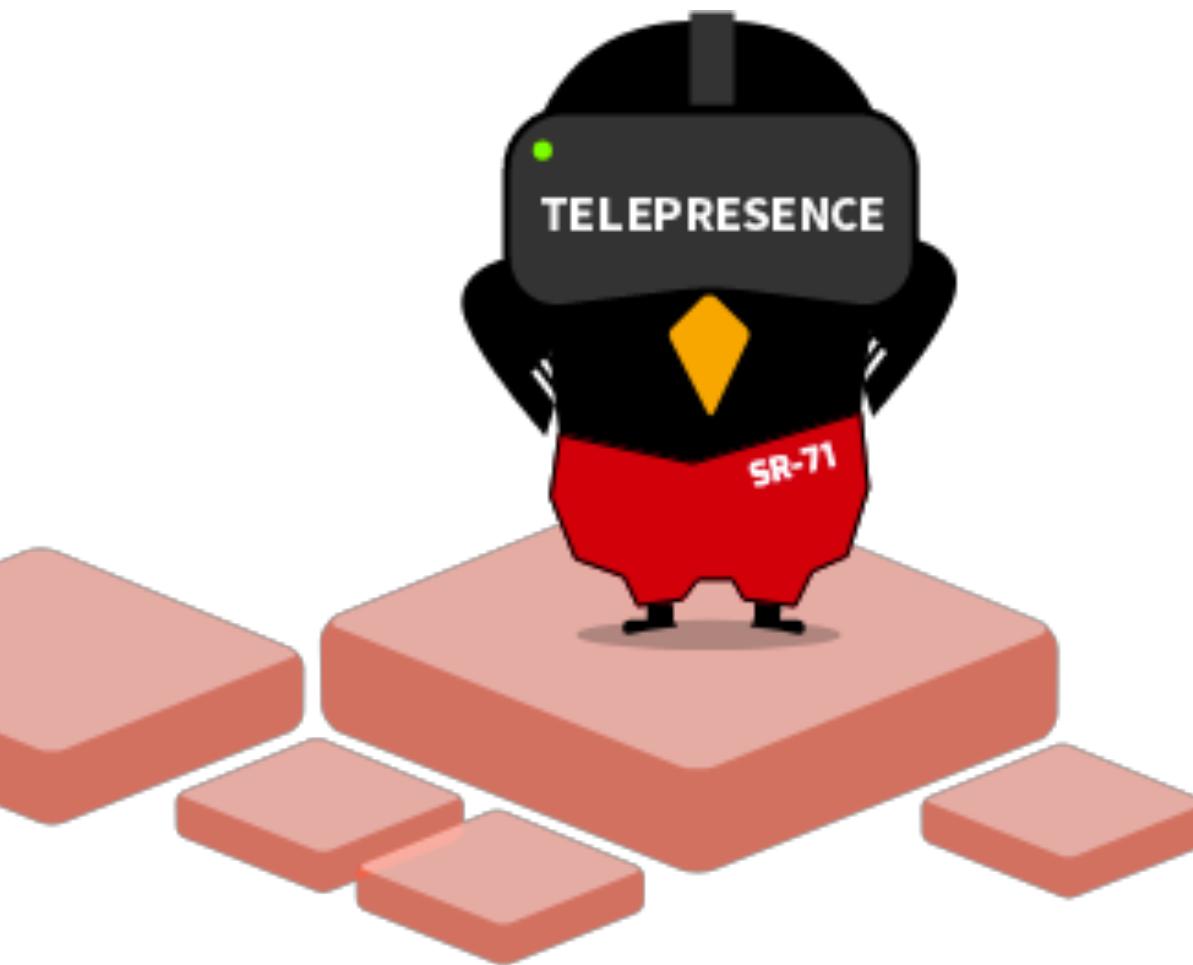
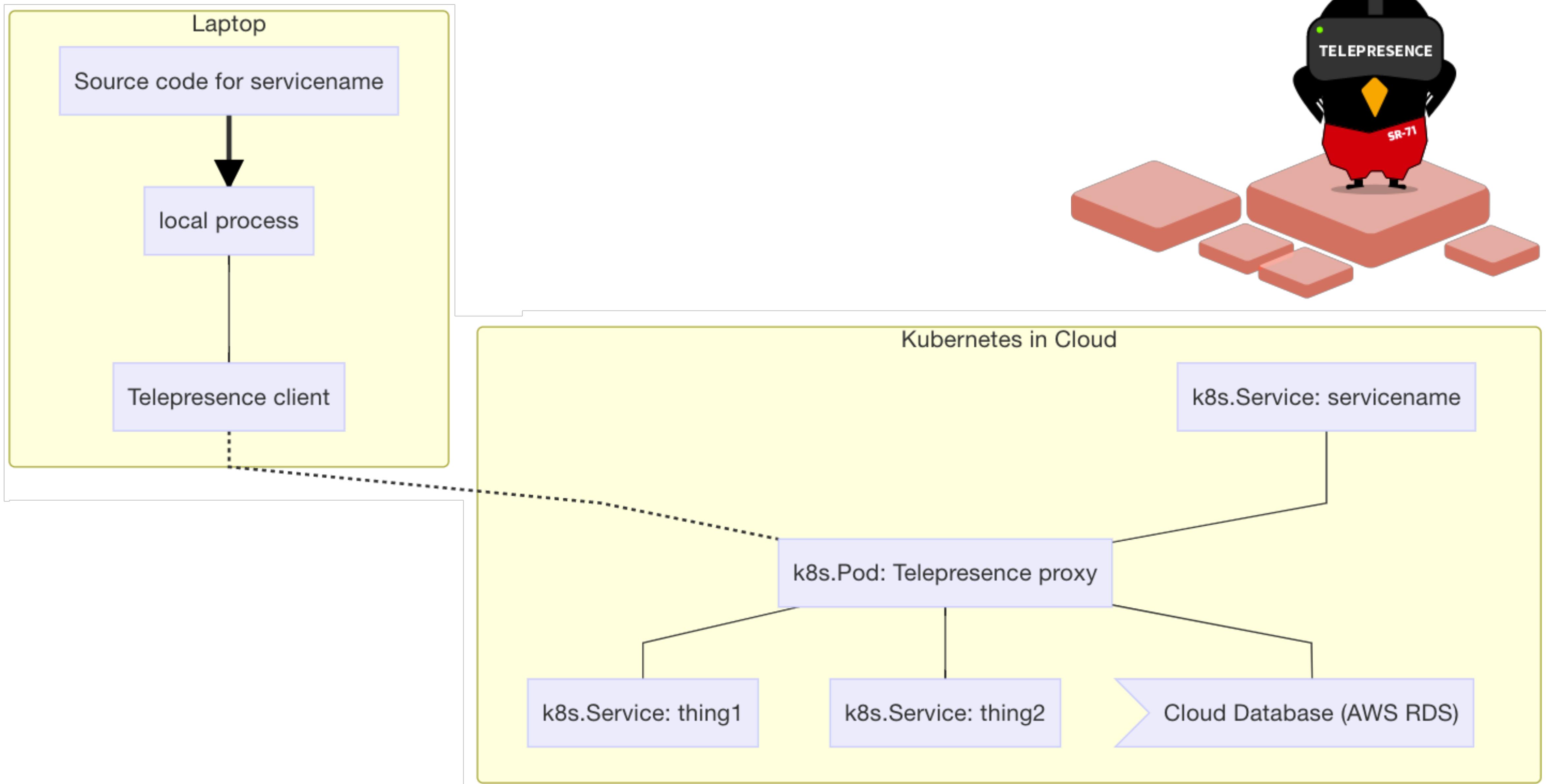
3.

Run locally

Bridge network

- Running code locally
- Developing against remote resources





Thank You All Day DevOps Sponsors

Platinum Sponsors



Gold Sponsors



GENERAL DYNAMICS
Information Technology



Carnegie
Mellon
University
Software
Engineering
Institute



SCALED AGILE

Media Sponsors



Solutions
Review

Thank You All Day DevOps Supporters



Meet Me in the Slack Channel for Q&A

bit.ly/addo-slack

References

- Jez Humble, David Farley, Continuous Delivery, ch. 3.
- Joe Beda, TGI Kubernetes 039
- Daniel Bryant, Creating an Effective Developer Experience on Kubernetes