

Intro to Telepresence:

Fast Development Workflows for Kubernetes



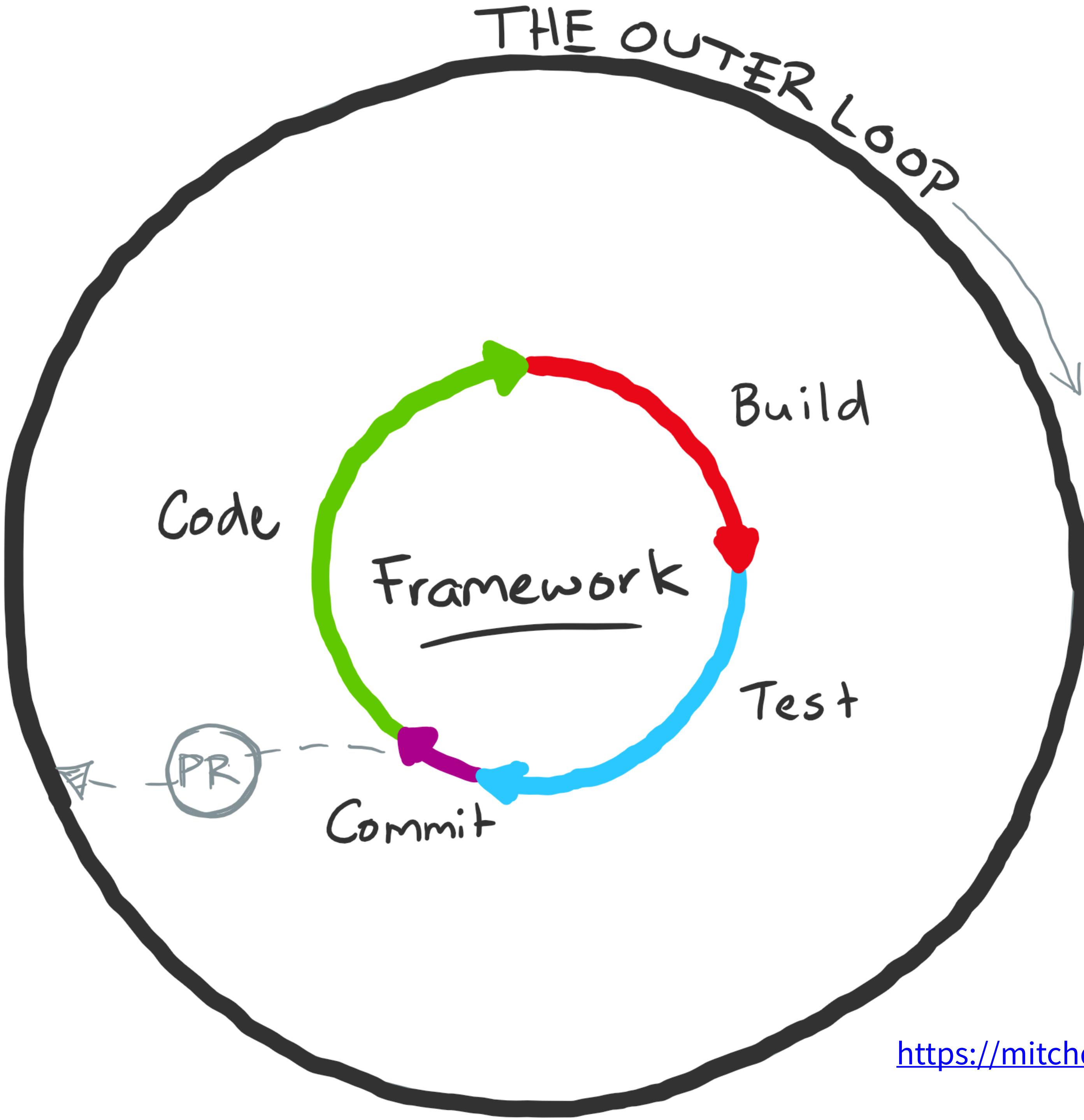
tl;dr

- The inner development loop can be painful with microservices + Kubernetes
- Telepresence proxies your dev machine into the cluster
- Telepresence supports several workflows, from small to large systems
- Ongoing work for splitting client and cluster components
- Please support this CNCF project – get involved!

Setting the Scene



DATAWIRE



<https://mitchdenny.com/the-inner-loop/>

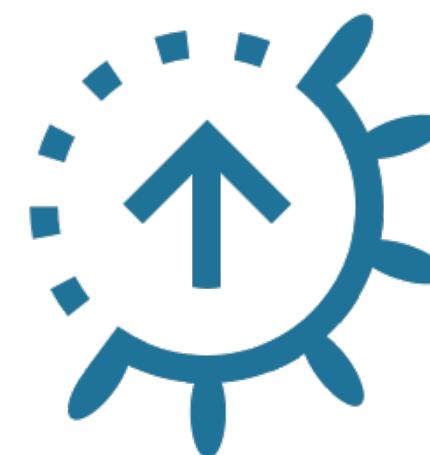
The Kubernetes inner & outer loop can be the same...



- Write code
- Build container
- Push to registry
- Deploy to cluster
- Test

And can be ... SLOW.

Automation helps speed it up ...



DRAFT



SKAFFOLD



garden



GITKUBE

But you want a *really* fast inner loop (no docker build/push) ...





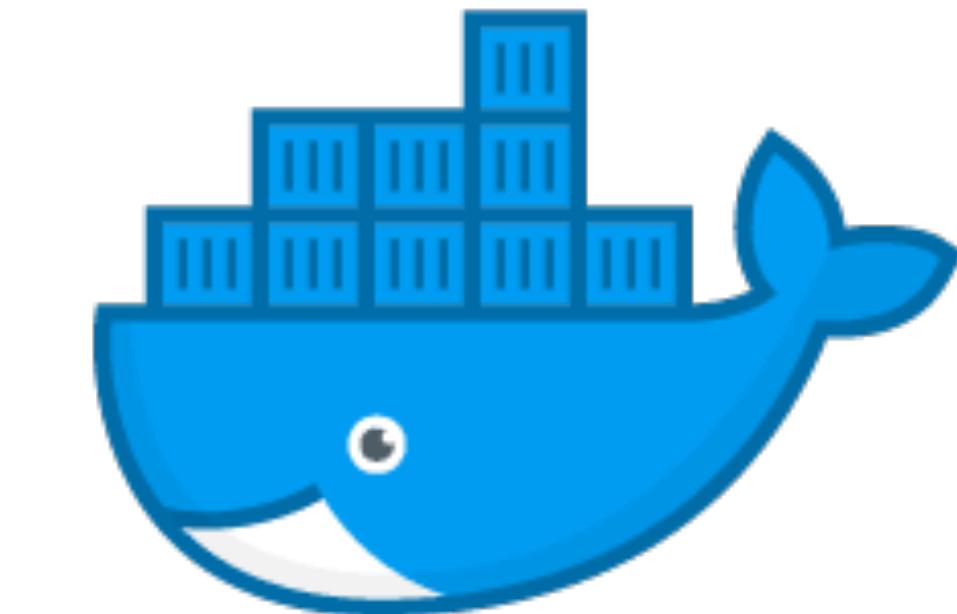
And you want to use your own tools.

Let's do everything locally...

...so it's really fast and I can use my tools!



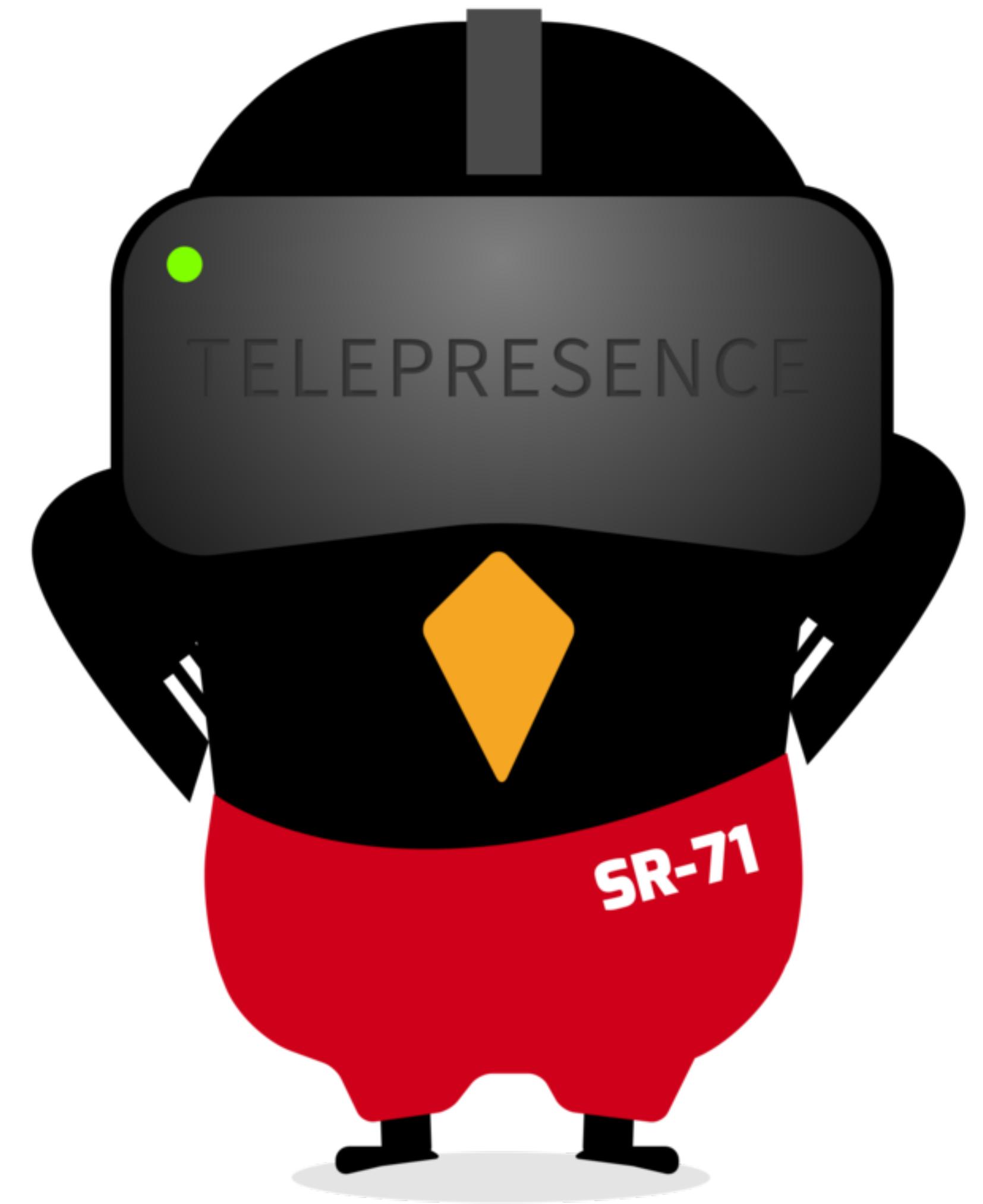
minikube



docker

**“My laptop only
has 16GB RAM and
you’re asking it to
run Kubernetes
and a JVM and a
database???”**





TELEPRESENCE



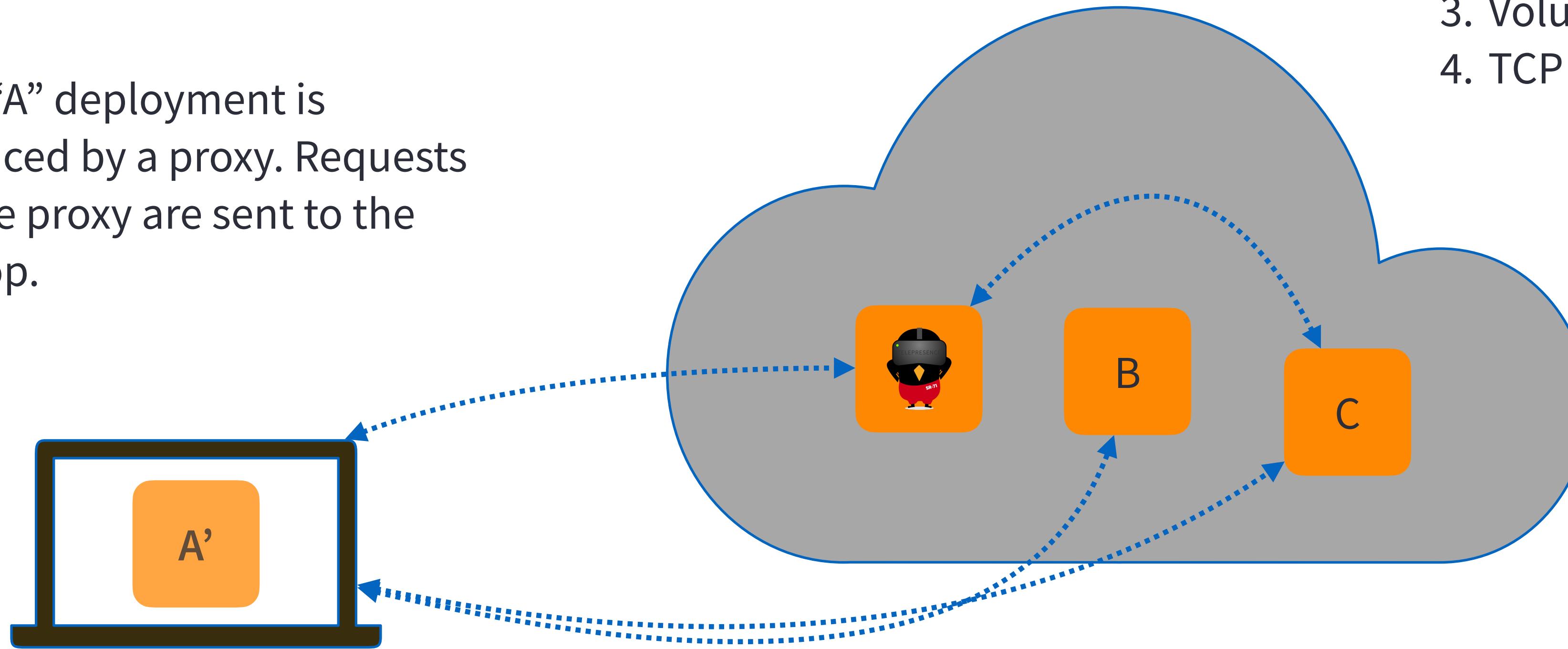
DATAWIRE

What is Telepresence?

- “Fancy Kubernetes VPN for development”
- “kubectl port-forward on steroids”
- A network bridge between your laptop and the Kubernetes cluster

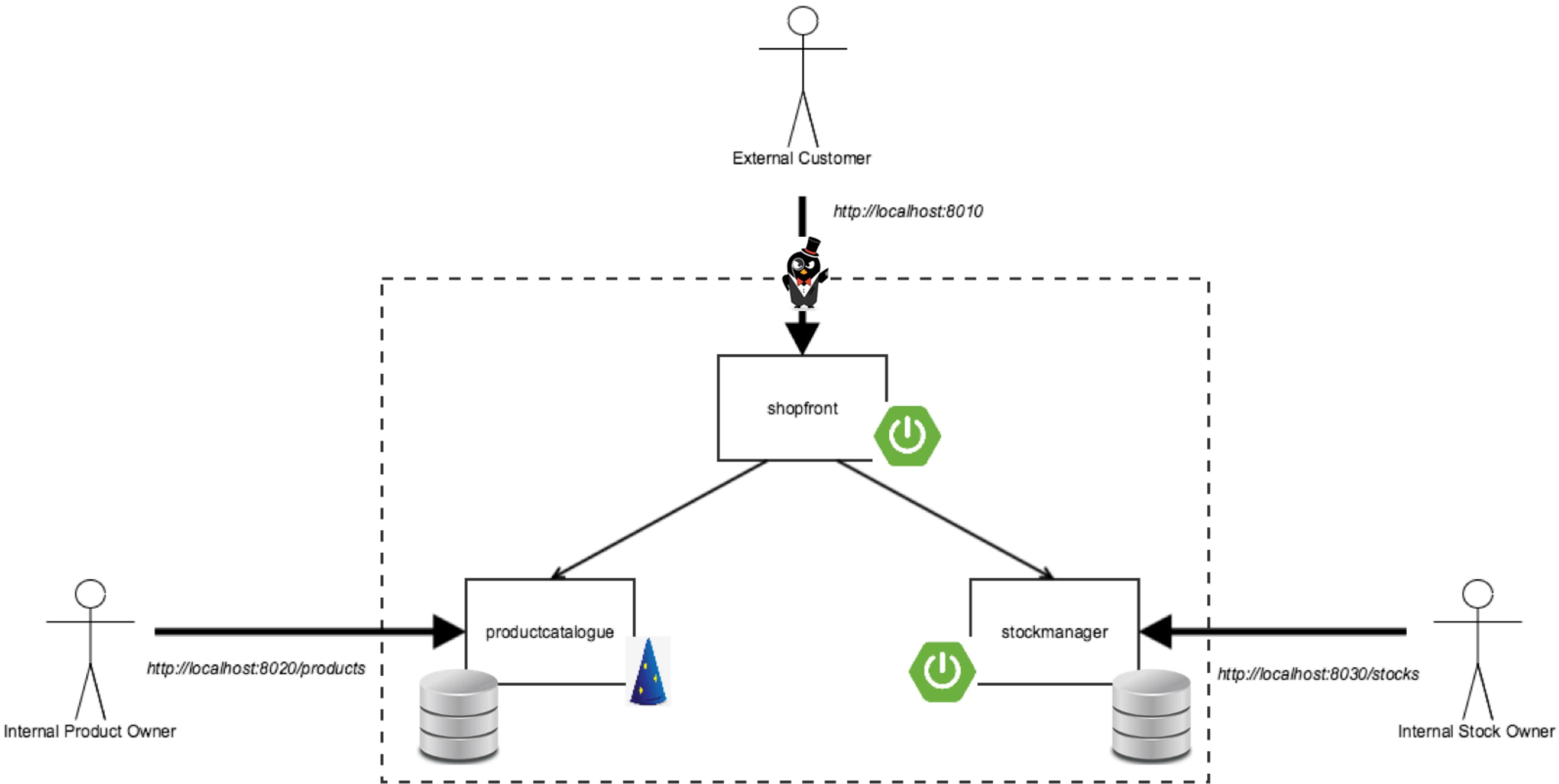
How it works

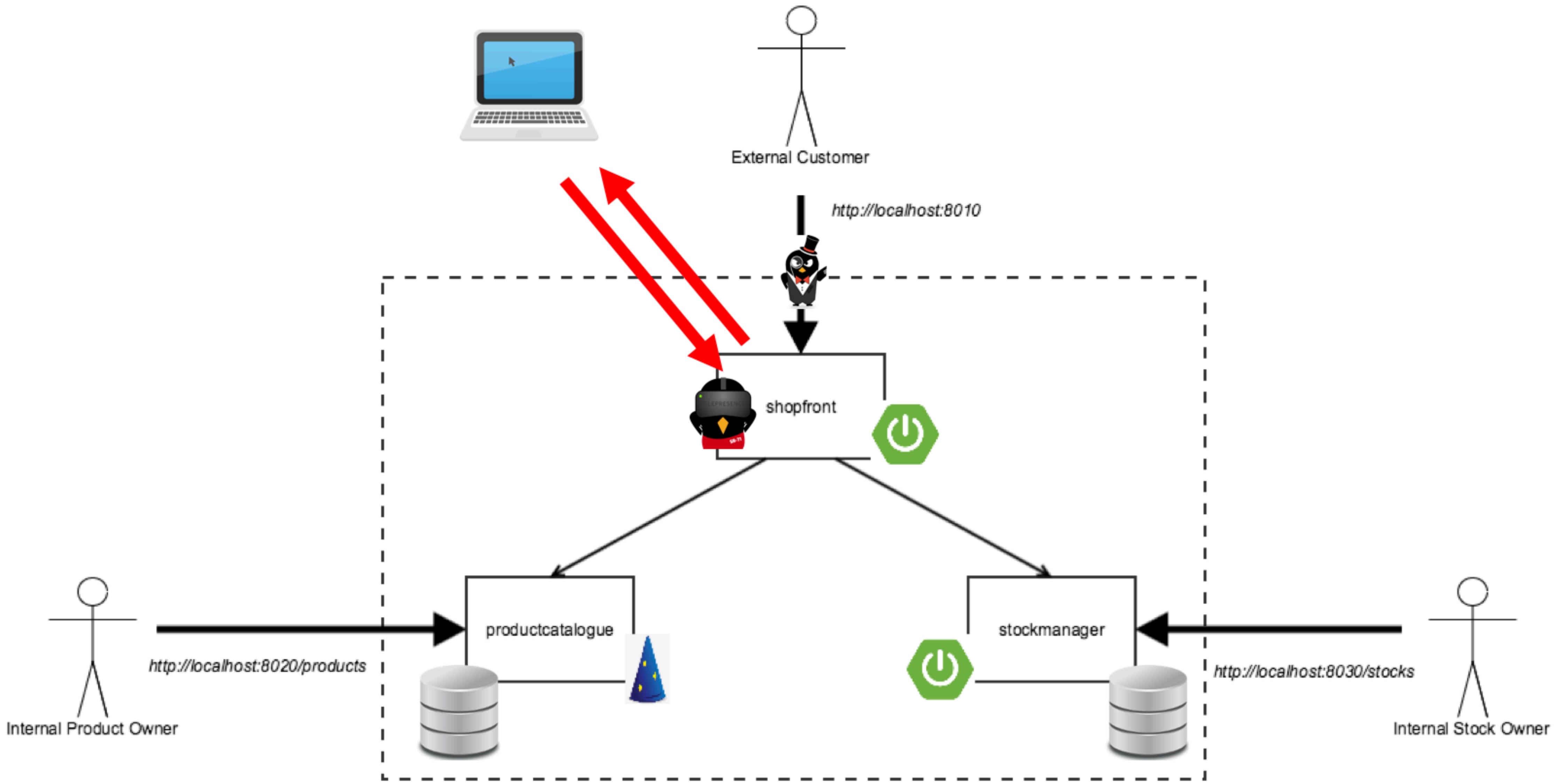
The “A” deployment is replaced by a proxy. Requests to the proxy are sent to the laptop.



Telepresence:

1. Intercepts DNS
2. Environment variables & secrets
3. Volumes
4. TCP





DEMO



DATAWIRE

Benefits

- Use any tool that runs on your laptop: IDE, profiler, debugger, ...
- Connect to cloud-based resources
- Same network namespace (e.g., nslookup works)
- Very fast inner loop!
- Requirements
 - Network connection
 - kubectl access to cluster
 - Mac or Linux

Telepresence is a CNCF project, with a wide variety of users.





Hacker News new | threads | past | comments | ask | show | jobs | submit

- ▲ Why's that company so big? I could do that in a weekend (danluu.com)
712 points by [danluu](#) on Oct 3, 2016 | hide | past | web | favorite | 423 comments | add to buffer

TELEPRESENCE



- Capturing DNS and routing only part of it to the cluster is tricky
- No two laptops are alike
- Swapping out your (real) deployment for a (proxy) deployment is tricky

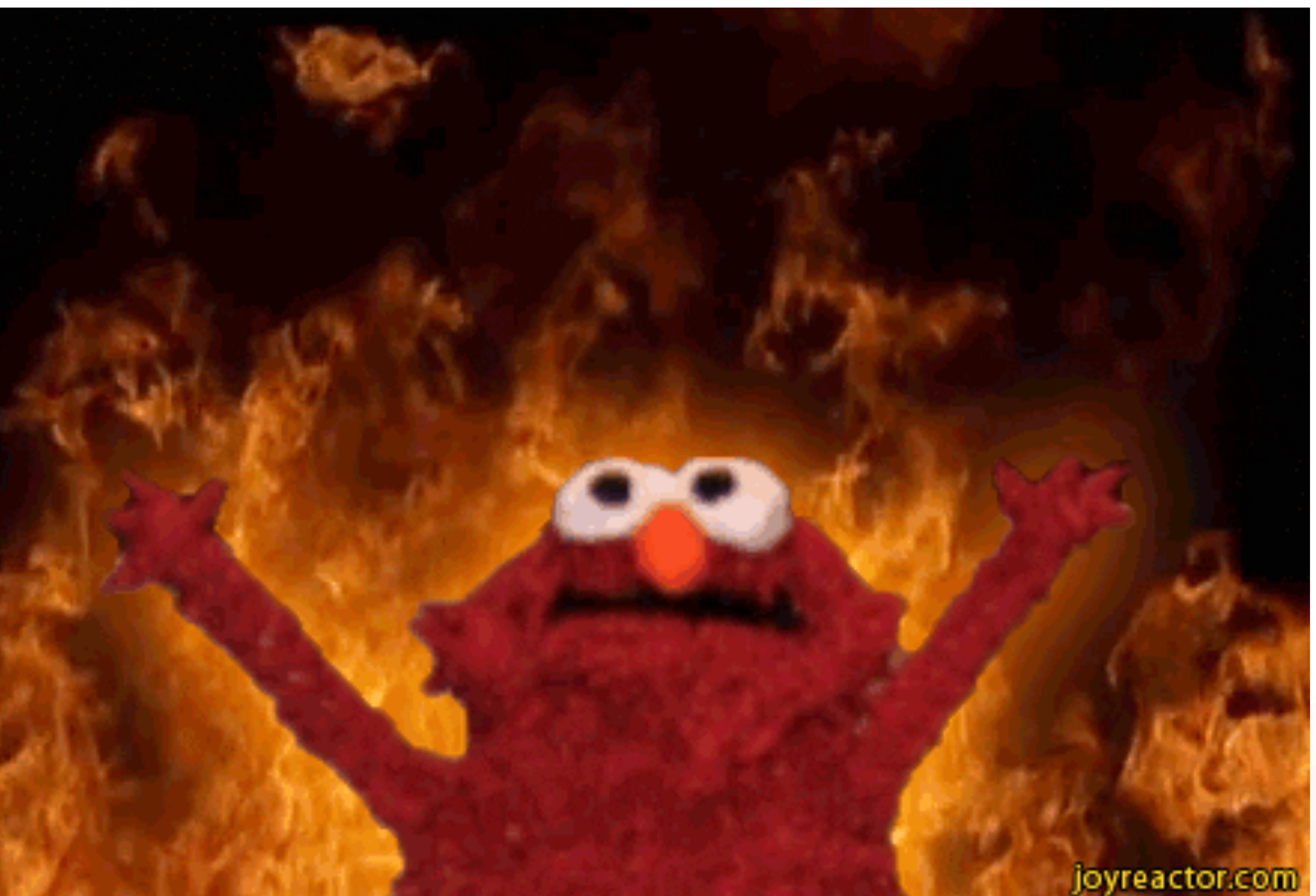
USING THIS IN YOUR WORKFLOW



Simple Workflow (YOLO)

- One (production) cluster for single-source-of-truth microservices
- Developer checks out code and works on microservices
- Swapping services on demand using Telepresence + IDE
- Telepresence is used in default (vpn-tcp) mode
- Once code is done, microservices are built and deployed via pipeline

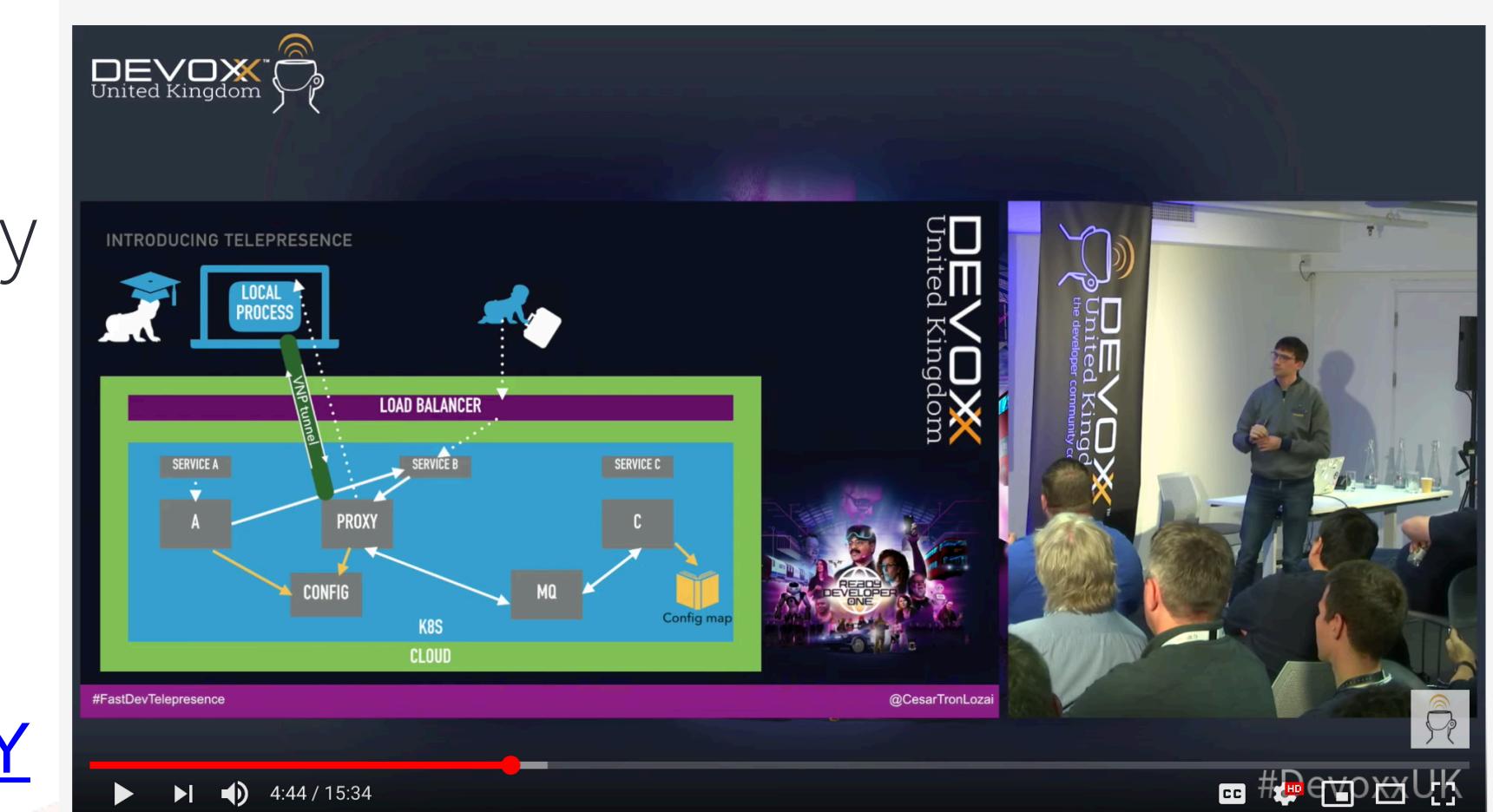
<https://www.youtube.com/watch?v=75soljochjY>



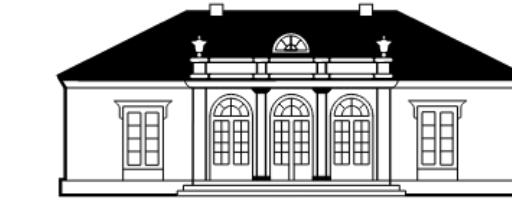
Typical Workflow (SME)

- One (dev) cluster for single-source-of-truth microservices
- Developers check out one of the microservices (and coordinate with other devs)
- Developers Telepresence + IDE
- Telepresence is used in default (vpn-tcp) mode
- Once code is done, CircleCI is used to run tests, build, and deploy

<https://www.youtube.com/watch?v=75soljoqhjY>



Advanced Workflow



ENGEL & VÖLKERS®

- One cluster with multiple namespaces for all single-source-of-truth microservices
- Developers check out one of the microservices
- Developers use Bazel + Telepresence + IDE
- Telepresence is used in container mode with Bazel images
- Once code is done, Jenkins is used to run tests, build, and deploy

<https://www.youtube.com/watch?v=tD0FlIxO1AQ>

The image shows a YouTube video player interface. The video is titled "Reproducible Development and Deployment with Bazel and Telepresence - Christian Roggia". It has 987 views and was uploaded on May 22, 2019. The video duration is 6:35 / 37:23. The video content is a presentation slide comparing Standard architecture, Containerized architecture, and Reproducible architecture. The slide includes logos for KubeCon and CloudNativeCon Europe 2019. The right side of the screen shows a speaker on stage at a conference.

Standard architecture	Containerized architecture	Reproducible architecture
Compilation is executed via scripts and Makefiles	Compilation is executed inside Docker via scripts and Makefiles	Compilation is executed via Bazel, Bazel itself can be dockerized
Tools and deps must be pre-installed and pre-configured in the system	Tools and deps must be manually downloaded and configured in the image	Tools and deps are automatically downloaded and configured by Bazel
Builds are hard to reproduce	Builds are not always reproducible	All builds are reproducible
Builds are not hermetic	Builds are not fully hermetic	All builds are hermetic
Implementation is different for each development environment	Implementation is the same for every development environment	Implementation is the same for every development environment

Reproducible builds, also known as deterministic compilation, always produce the same binaries.
Hermetic builds are insensitive to the libraries and other software installed on the build machine or image.

6 | Christian Roggia, Engel & Völkers Technology GmbH

Videos brought to you by: Google Cloud

CC HD □ []

Reproducible Development and Deployment with Bazel and Telepresence - Christian Roggia
987 views • May 22, 2019

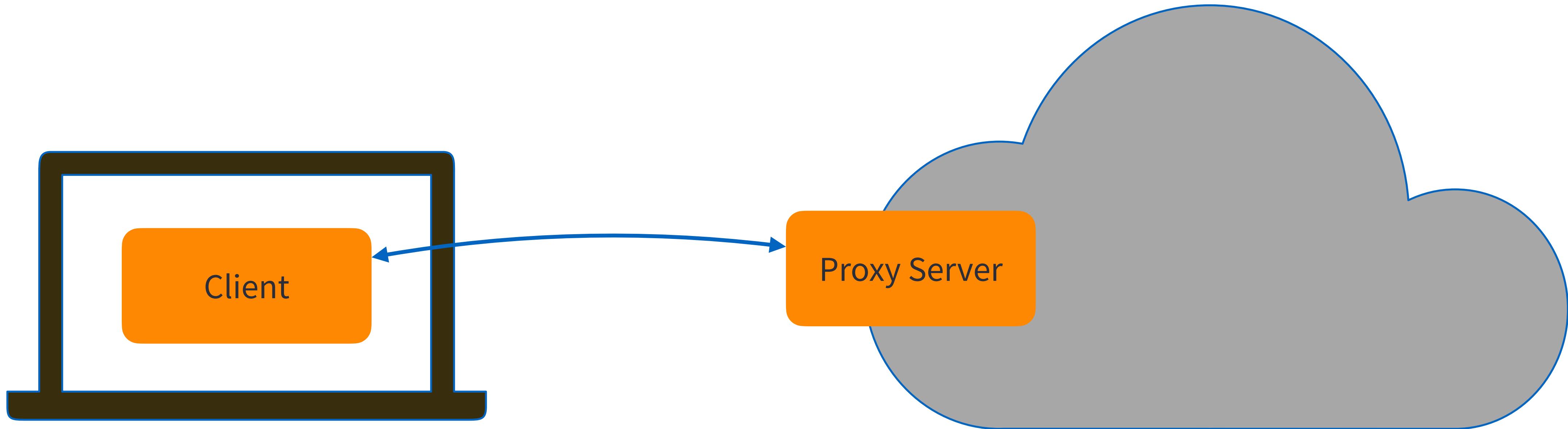
Like 18 Dislike 0 Share Save ...

WHAT'S NEXT?



DATAWIRE

Telepresence components



- Capture DNS requests for Kubernetes resources
- DNS resolution for Kubernetes resources
- Proxy TCP connections to cluster
- Proxy TCP connections to laptop
- Canonical DNS for Kubernetes resources

Telepresence Client

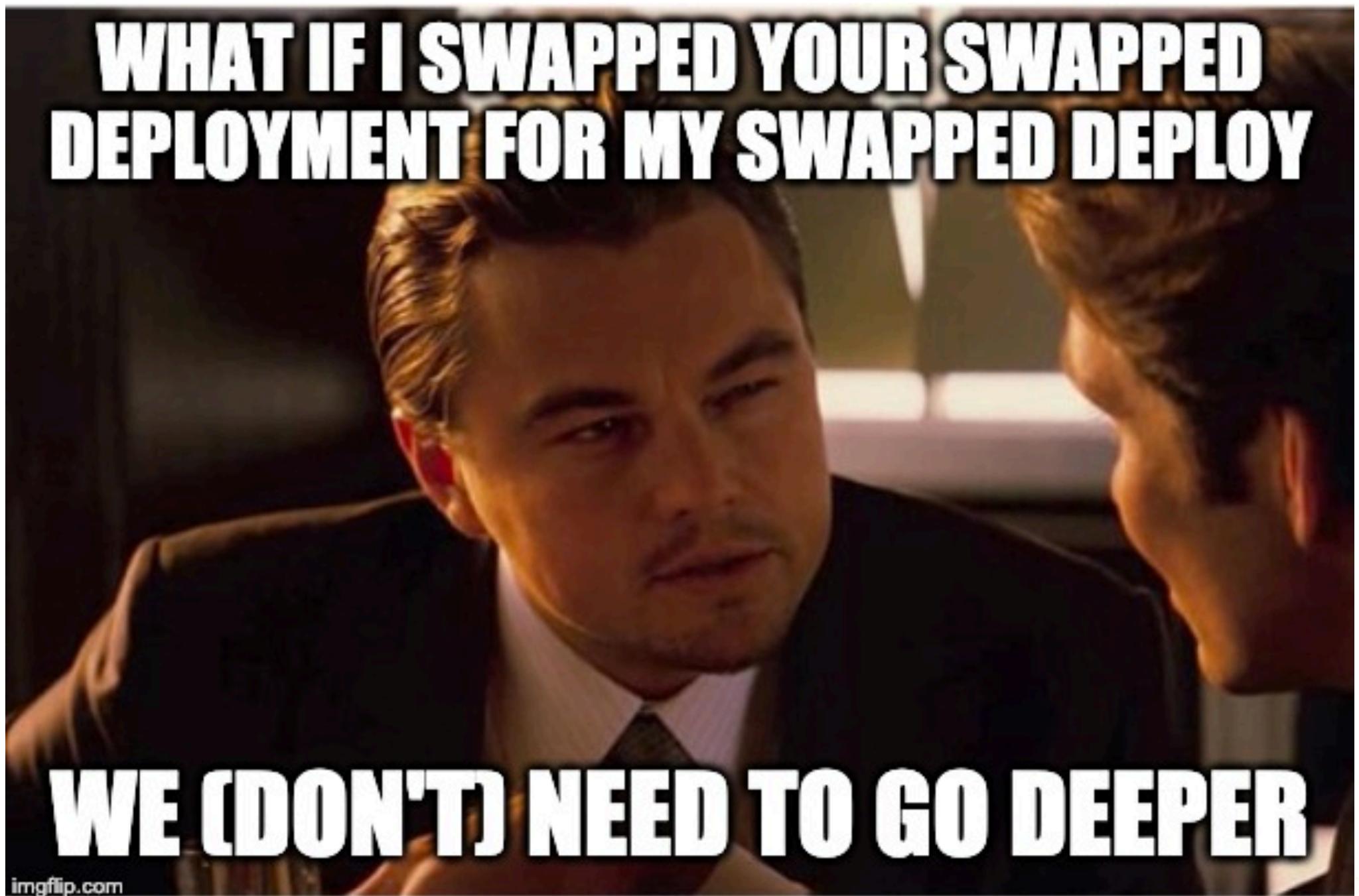
- Use kubectl port-forward to bootstrap sshuttle (ssh-based VPN)
- sshuttle has a lot of functionality that is not used by Telepresence (it's a VPN, after all)
- Replace sshuttle with something else — designed for Kubernetes
 - Capture DNS locally
 - Do DNS resolution intelligently
 - Redirect TCP connections to the right destination
- Already in progress, will replace VPN-TCP method
- Also support multiple simultaneous swaps

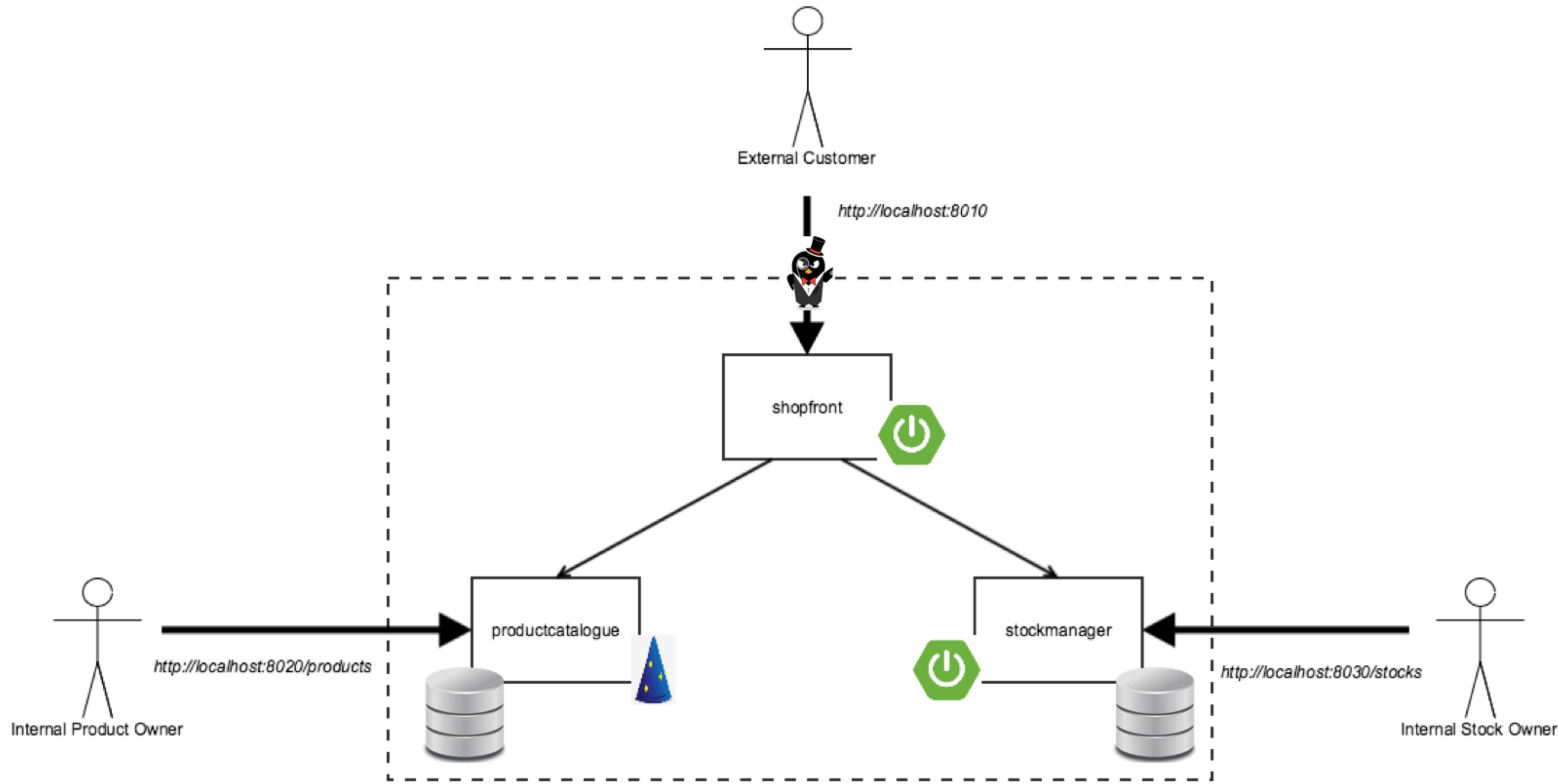
Telepresence Proxy Server

- Currently deployed every time you do a swap-deployment (and then deleted)
- Switch mode to be persistent (this is what Kubernetes is good for)
 - Speeds up startup
 - Support reconnect

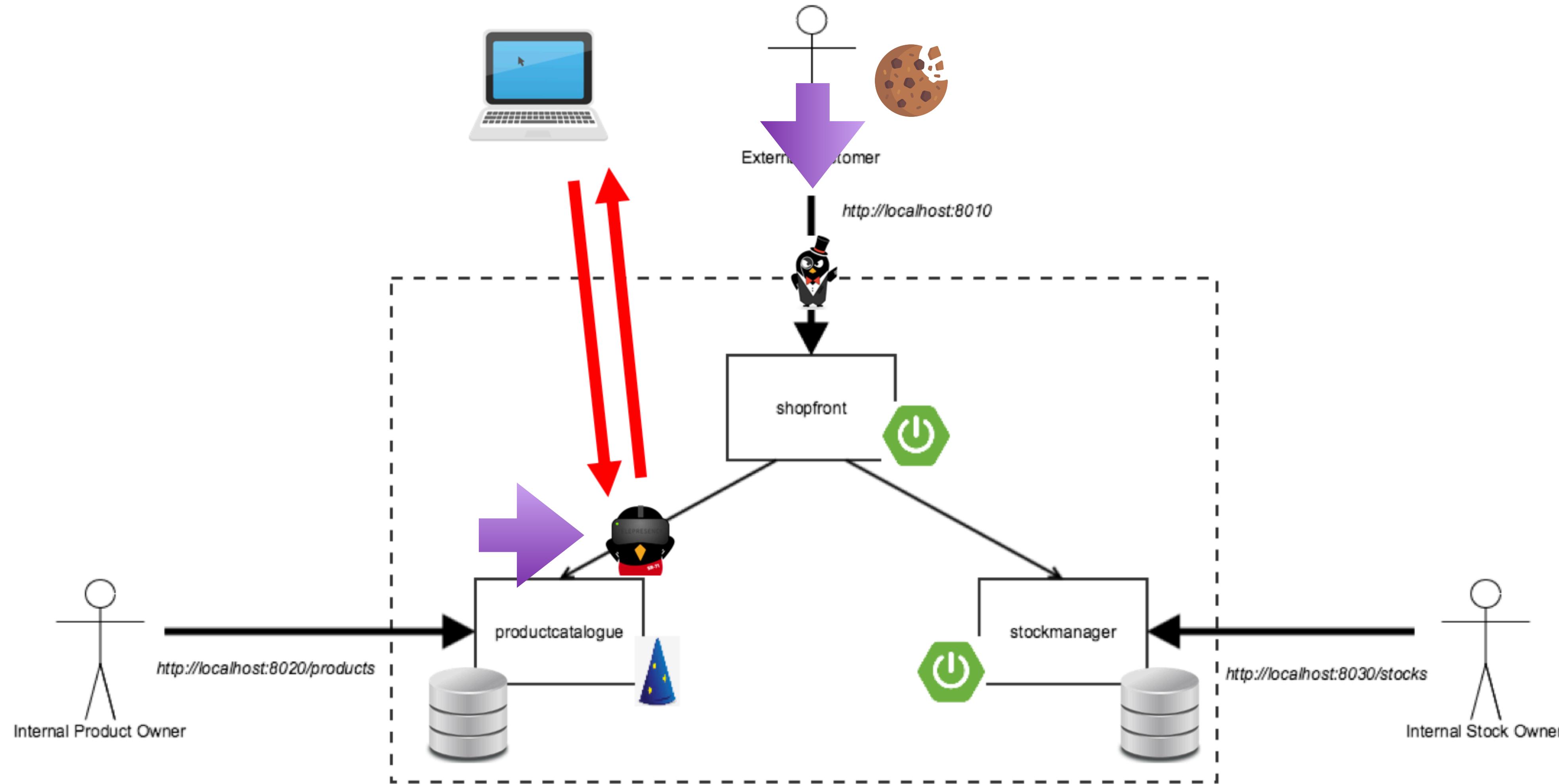
Multi-User Development

- Swap deployment doesn't work well for multi-user development
- Solution today is to use namespaces and/or individual clusters
 - Expensive (more hardware & setup)
 - Delays integration testing until after commit





Service Preview



Wrapping Up



DATAWIRE

Contributing to Telepresence

→ Documentation and workflows

- * how you integrate Telepresence into your development workflow

→ Let us know if you're a user!

→ Triage

- * Lots of folks use GitHub for support, which doesn't scale well

→ Join Slack to help other users <https://d6e.co/slack> #telepresence

Thank you! Questions?

- db@datawire / @danielbryantuk
- www.telepresence.io
- <https://github.com/telepresenceio>
- Check out the Ambassador Edge Stack (with service preview)
 - * <https://www.getambassador.io/>

