





---- North America 2023 -

How We Optimized Developer Productivity with Telepresence

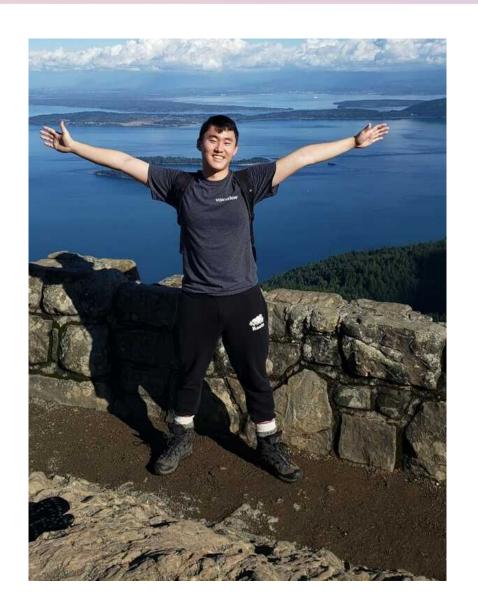
Frank Gu, Voiceflow

About Me



- 10 years of "full-stack" experience
- Licensed pilot
- Director of Engineering @ Voiceflow
- Hat enthusiast

Voiceflow



Agenda

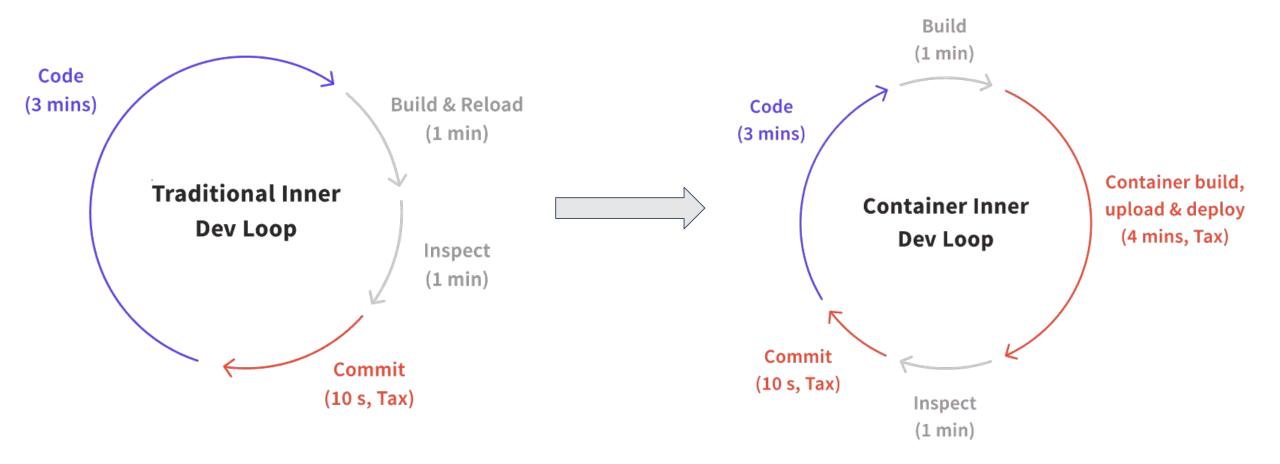


- The Problem
- Potential Solutions
- Telepresence into K8s
- Demo
- Key considerations

The Container Workflow







Our Problem(s)



"But this worked on my machine!"

"Can you send me that config/secret file you used?"

"I'm waiting for the CI pipeline to run my changes..."

"Can you give me a demo of your PR changes?"

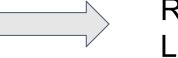
"My computer can't run all the required services!"



Dev / Prod Parity



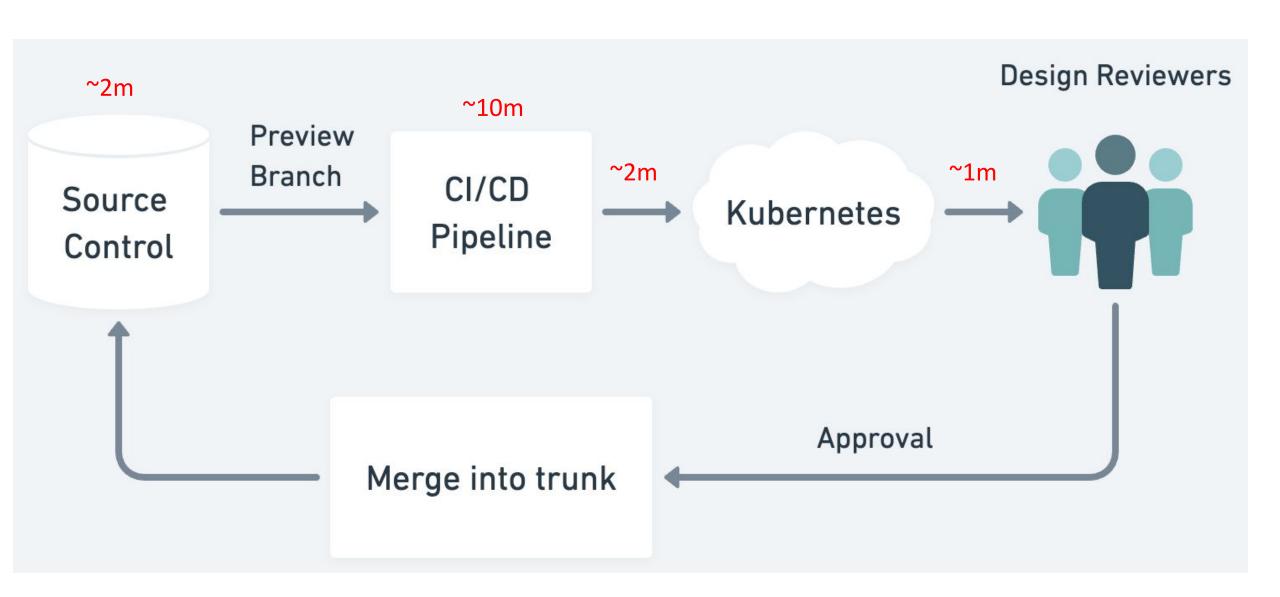
Change Review Process



Resource Limitation

Our (old) Dev Loop - 15m / it





Potential Solutions



- Using a vendor's code preview solution -> insufficient customization
- Running a cluster locally -> others can't (easily) access the instance
- Map local files into the a dummy pod in the cloud and execute the code
 - -> lack of dev-prod parity



What We Need



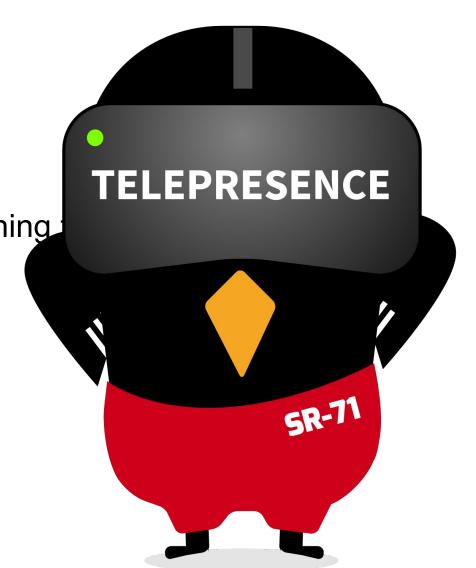
Run a service locally

Map cloud configurations into local service

Communicate with other cloud services

Pretend that my local machine was a pod running.

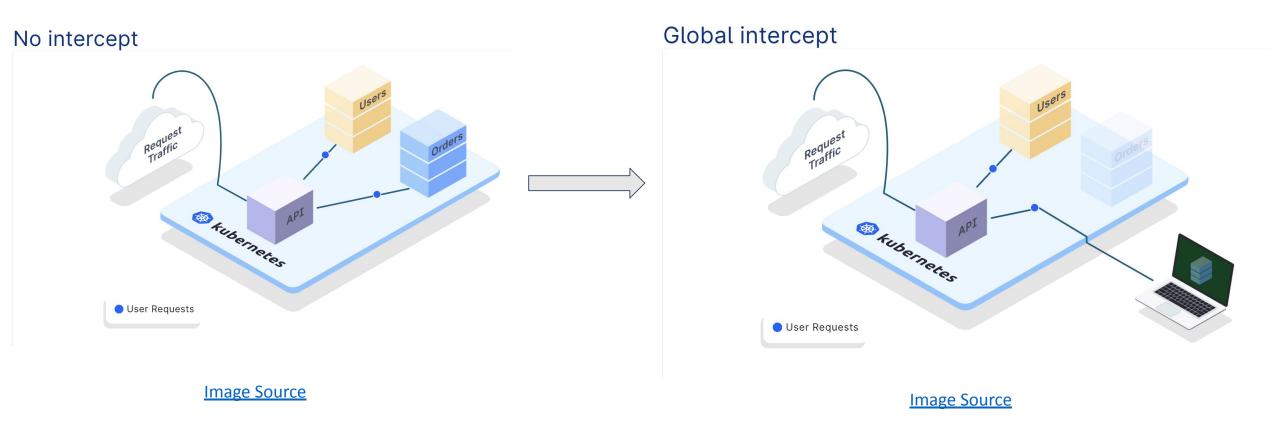
If only there was a way to proxy traffic between K8s and local machine...



Telepresence into K8s



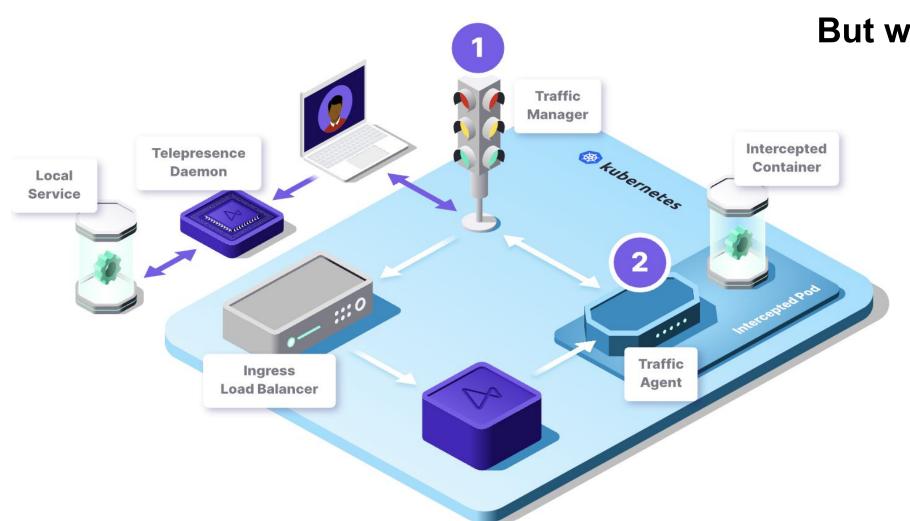
At a glance



Traffic can now flow between K8s cluster and local machine!

Telepresence Architecture





But wait, there's more!

K8s-aware DNS

Volume mount

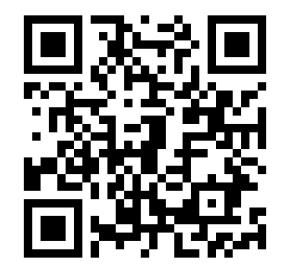
Environment Variable injection





North America 2023

DEMO TIME!!!

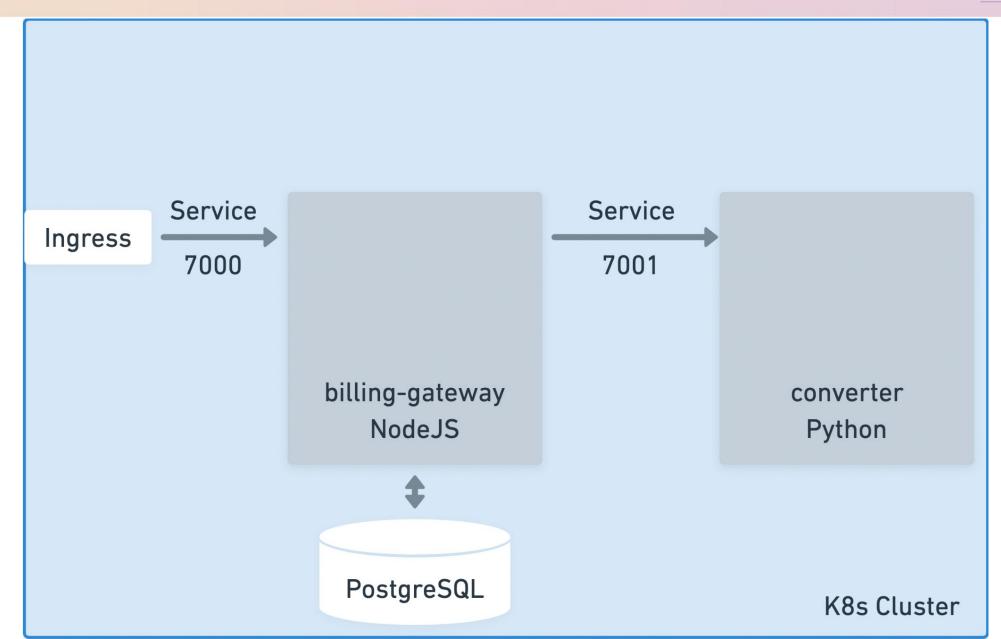


frankqu968/kubecon2023

Problem Setup



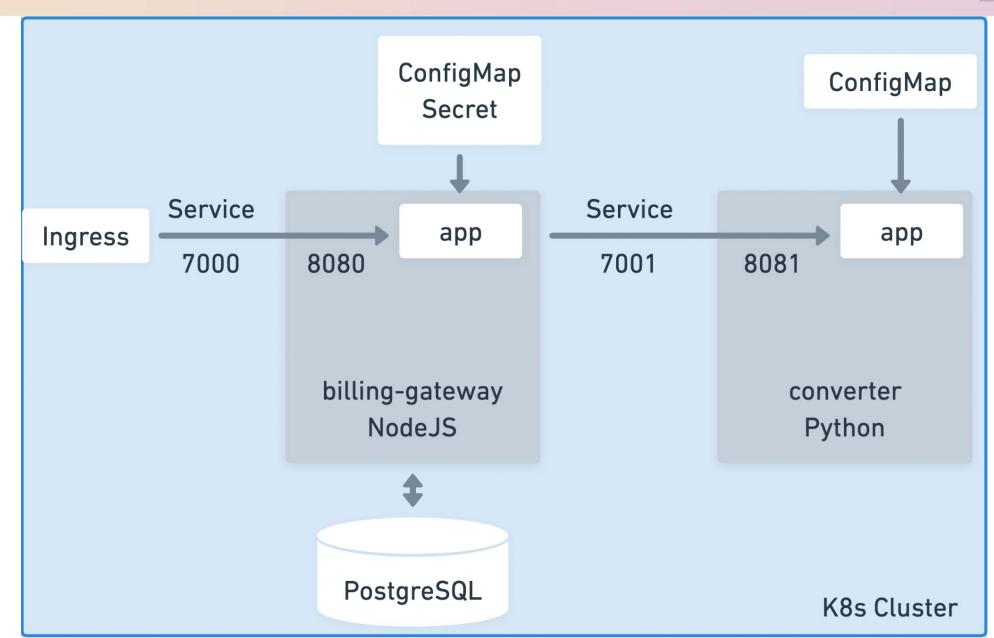




Initial Deployment



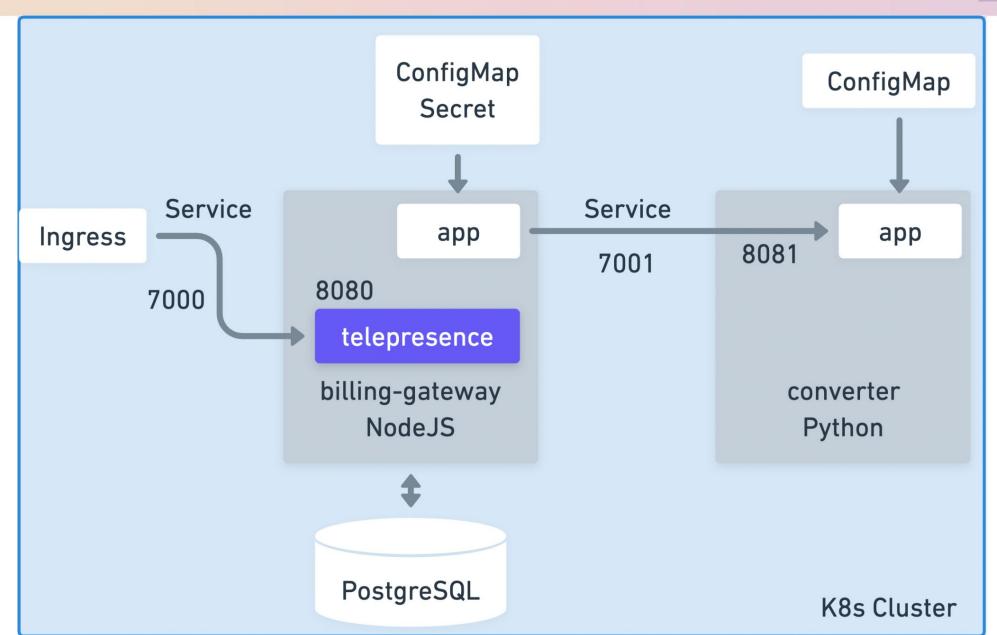




Telepresence Agent Injection



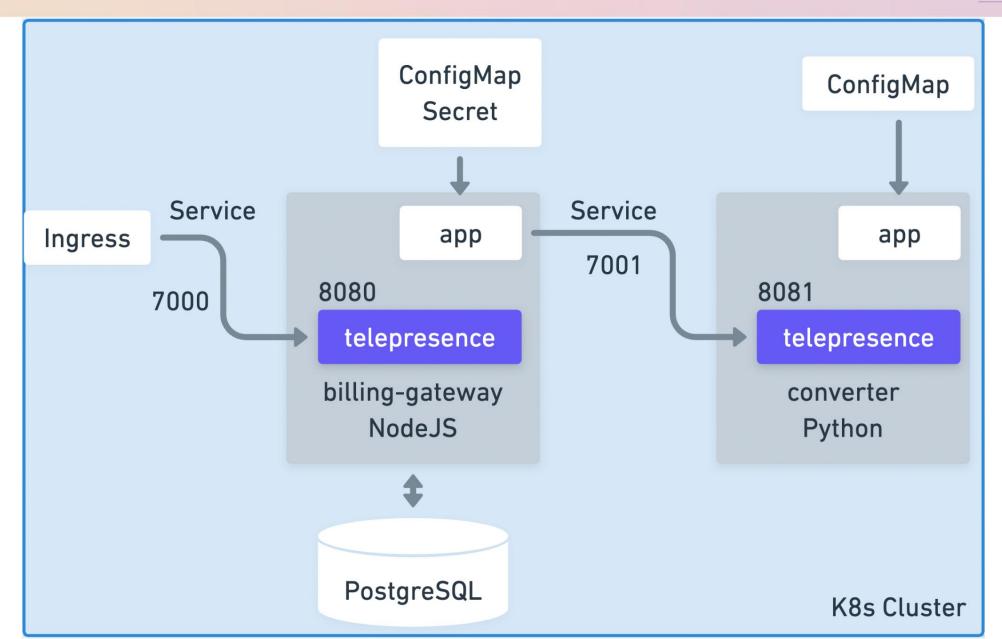




One more Intercept!!!



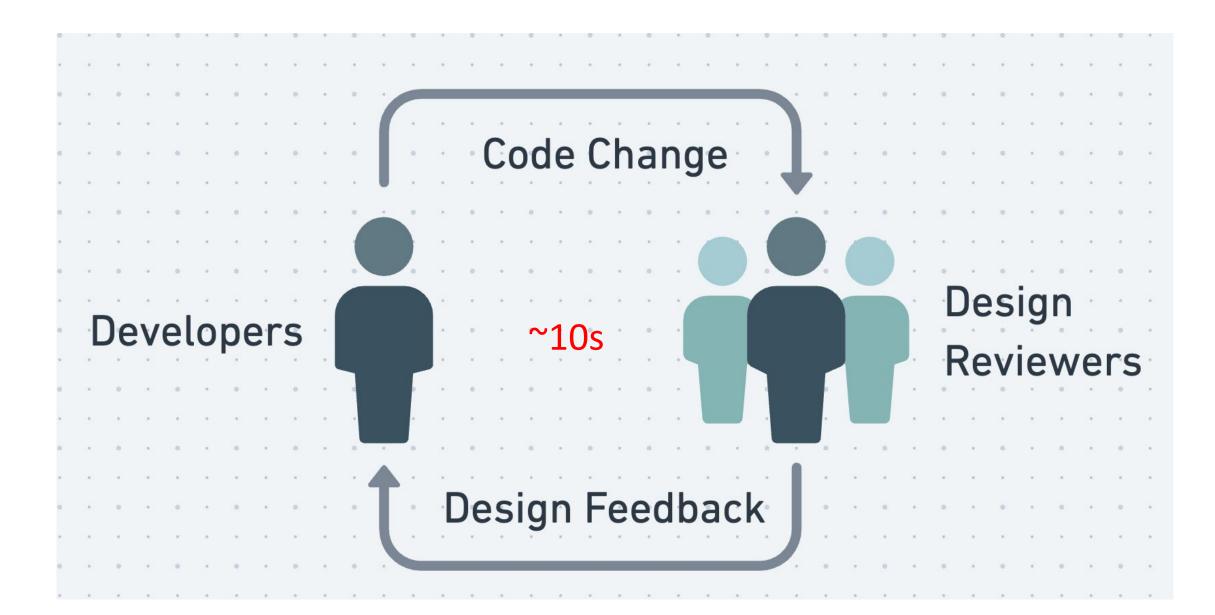




Our (new) Dev Loop - 10s / it







Telepresence Considerations



- Deployment Model
- Scale
- Security
- Robustness/Support
- Latency

Conclusion

 Dev/Prod parity + Resourcing + Deploy/Review velocity are major concerns in k8s development workflows

 Telepresence proxies traffic and configurations between local machine and the k8s cluster

 Reducing the build/upload/deploy overhead enhances developer experience

Thank you for attending this session!



Questions?





