

Golang monorepo in the wild



Tu The Hien
WeBuild day
2020-12-12

About

- **Former Blk71 Singapore resident**
- **Running Go microservices in production, at scale, since 2016**
- **Current, Builder @ Begroup**



- Vietnamese startup. Ride hailing and more.
- Stay tuned for 2021

**WE'RE
HIRING!**

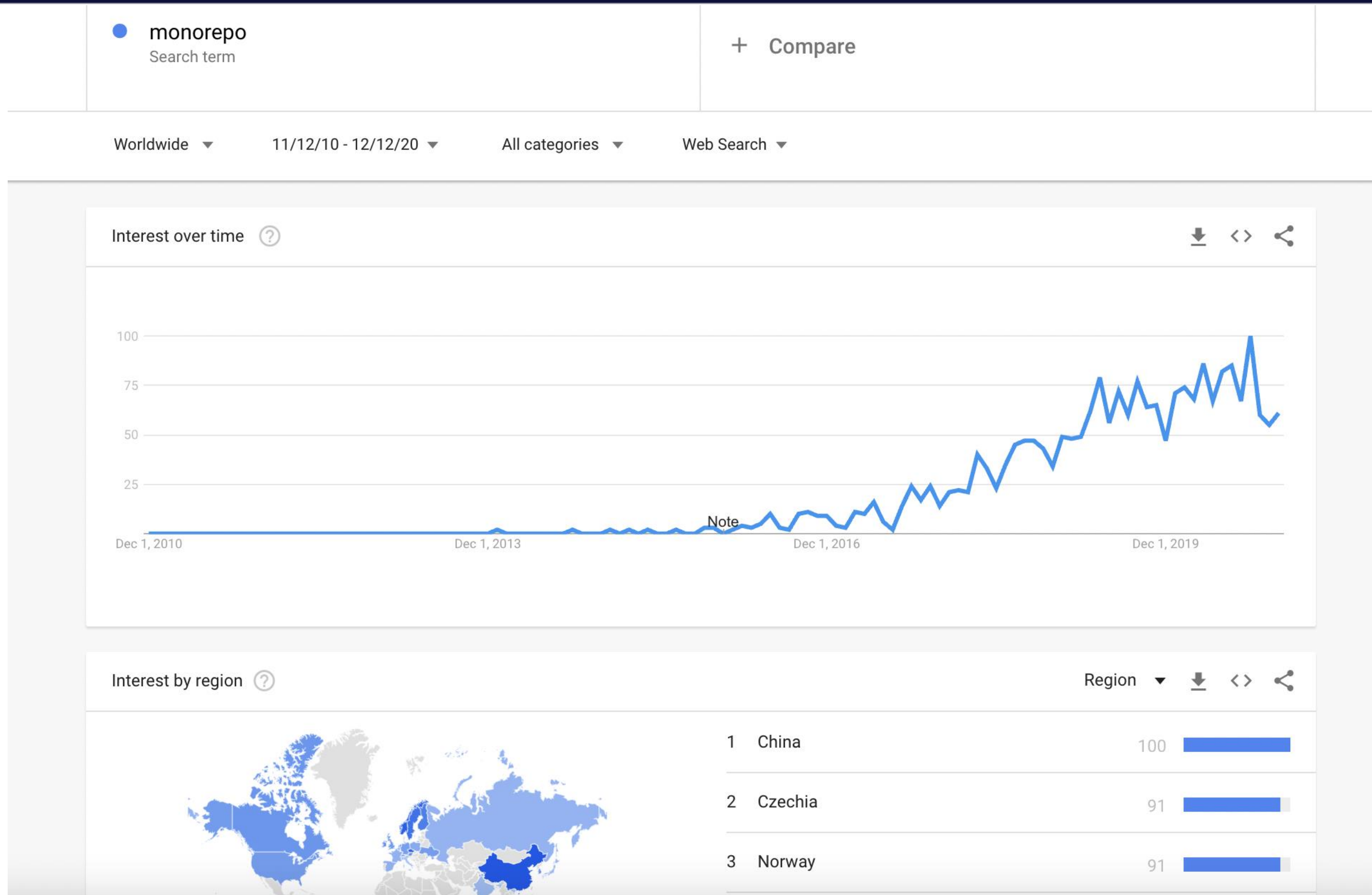
- Always have openings for excellent engineers
- Let's talk, we fit people to role, not the reverse

Table of contents

- **Background**
- **Setup**
- **Build**
- **Deploy**
- **Summary**
- **Q&A**



Trend



What is monorepo

- **Monorepo: single multi package repository,**
- **Polyrepo: multi single package repository**



Who do what

- **Monorepo**
 - Google (kubernetes, angular, ...)
 - Facebook
 - Uber
 - China internet company
 - Singapore based Green ride hailing company
- **Polyrepo**
 - Amazon
 - Netflix
 - Spotify
 - Rest of the world



Microservices

- Promote by tech startup, global internet company: Netflix, Twitter.
- Solving organization problem
- Create more technical problems

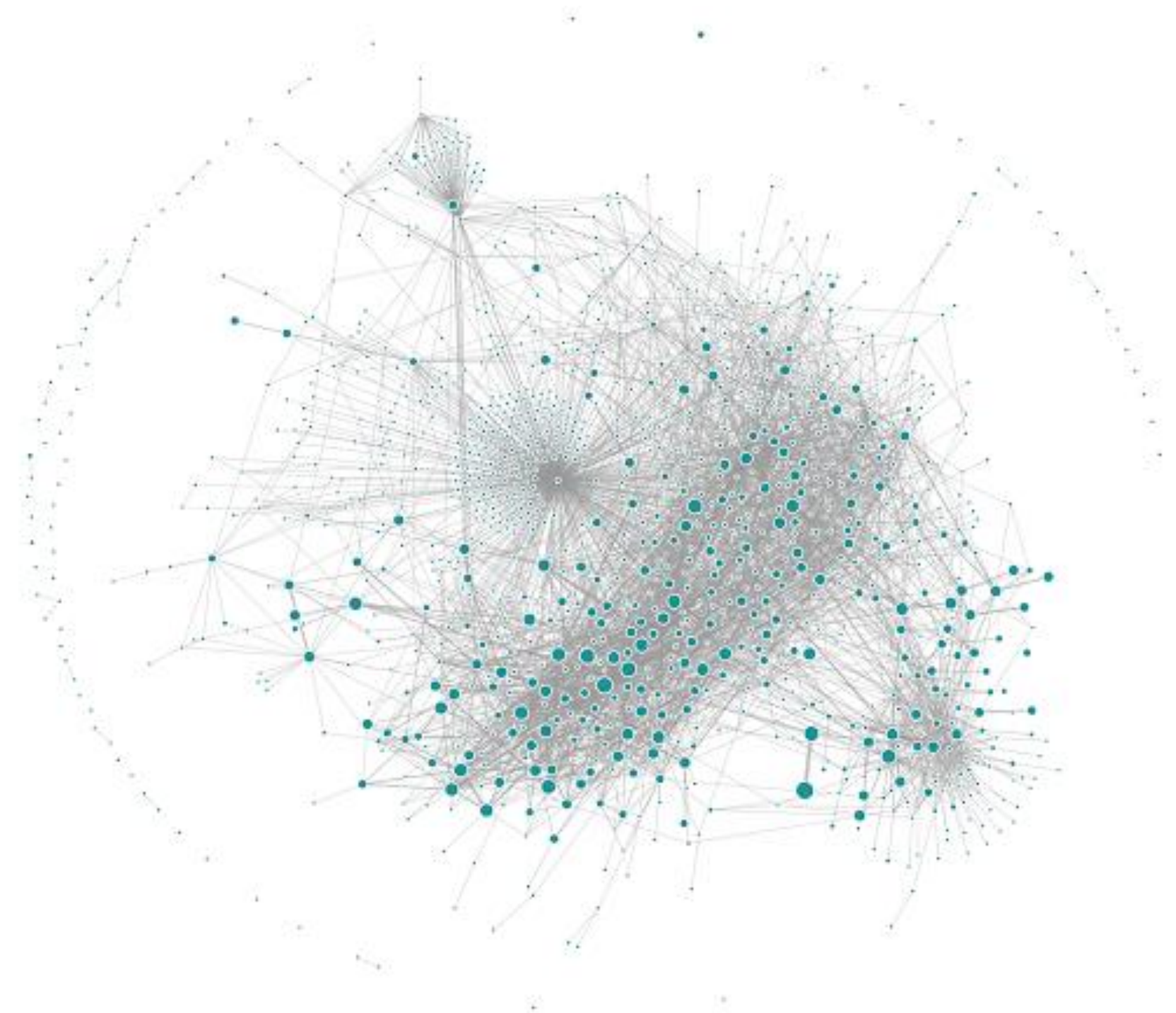
Microservices Freedom

- Freedom
- Easy create new
- Hard to make changes globally
- Hard to manage dependencies



Context

- **Project tend to grow**
- **Split into more sub-projects**
- **Create separate repo**



Polyrepo

- **Pros:**
 - **Flexible**
 - **Fine grained acl**
- **Cons**
 - **Maintainability**
 - **Hard to browse**
 - **Dependencies**
 - **Support multiple versions**
 - **Consumers upgrade at will**



Motivation

- Facilitate **code sharing, knowledge discovery**
- **Standardize** and **speed up** development
- Better manage **dependencies**
- Toward true **CD**



Evaluate Monorepo as a solution

- **Pros:**
 - Encourage make changes
 - Global view on microservices meshes
 - Short feedback loop
- **Cons**
 - More dependency
 - Larger code base
 - Fine-grained access



Community enabler

- **Opensource and growing community**
- **Big adopters. Almost all google opensource projects.**
- **Kubernetes is a great learning example**



Bazel

- **Opensource from Google's internal Blaze**
- **Support multiple languages, multiple repo**
- **Extensible, via rules**
- **Fast**
 - **Automatically detect changes**
 - **Caching, incremental build**
 - **Multi-core, highly parallel execution**



{Fast, Correct} - Choose two

- **WORKSPACE**
- **BUILD**
- **src_files**
- **go_library/go_library_test**
- **go_binary**
- **container_image**
- **container_push**



{Fast, Correct} - Choose two

Gazelle

- <https://github.com/bazelbuild/bazel-gazelle>
- **Build file generator**
- **Go tool compatible**

Codeowners

- **GitHub and Gitlab built-in solutions**
- **Opensource solution**



- **Kubernetes open source**

solution: <https://github.com/kubernetes/community/blob/master/contributors/guide/owners.md>

- Fork k8s solution
- Chatops style
- Flow
 - Request a PR
 - **Bot** automatically assigned required *reviewers/approver*
 - *Approver* indicates his approval in discussion/approval button
 - **Bot** indicates if all required approvers **approved**

Build once, run everywhere

- Consistency of many environments
- Which backend version are we having on dev/staging/prod
- Major **Atomic** changes

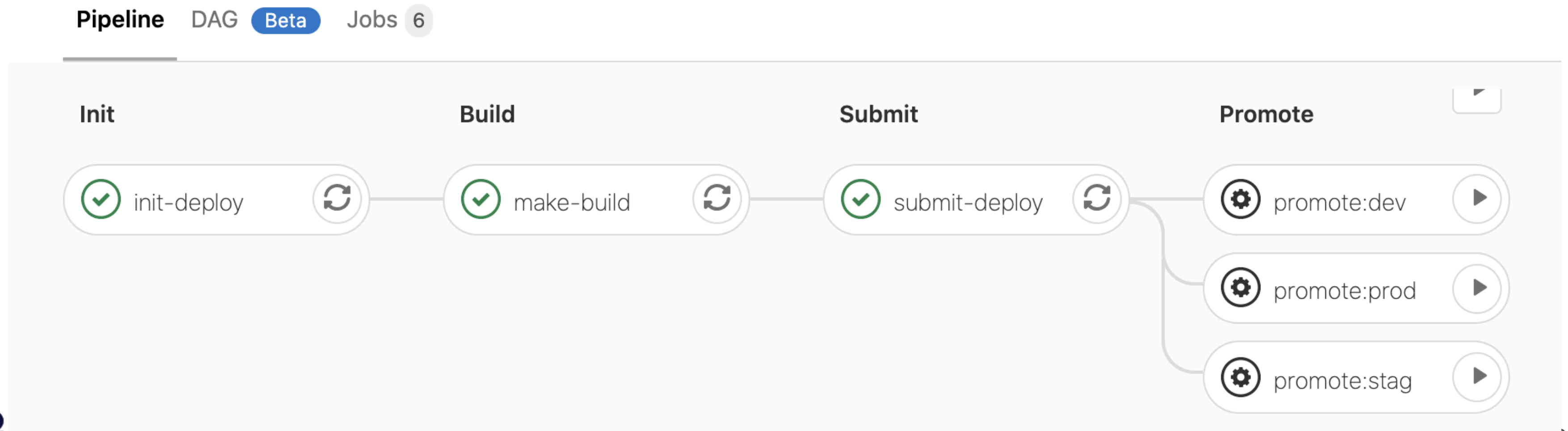


Version/tagging

- All image version, is ***monotonically increasing***, based on number of commits, count from trunk base (master branch)
- Image is **immutable**, and **reproducible** at the exact point on commit trunk
- Format: registry/{*domain*}/{*service-name*}:**version**
- If **one** PR or commit trigger **multiple** image changes , **all** resulted images will have the same **tag**

Deployment migration

- Reuse the same **deployment** pipeline
- Each change on monorepo, will result in 1 single commit, with potential **multiple** changes, corresponding to binary changes.
- Use **Promote** concept: promote 1 single **state** , to **many** environments



Must have

- **Automation tests is a must**
- **Solid dedicated platform team**
- **Synchronous code review**
- **Feature flag**



Questions

