Concourse

Overview

05/19/2016
Etourneau Gwenn
@The_shinji62

About me

Etourneau Gwenn

Sr Solution Architect

Bef.: Platform Architect

https://twitter.com/the_shinji62

https://github.com/shinji62

Pivotal

Pivotal®

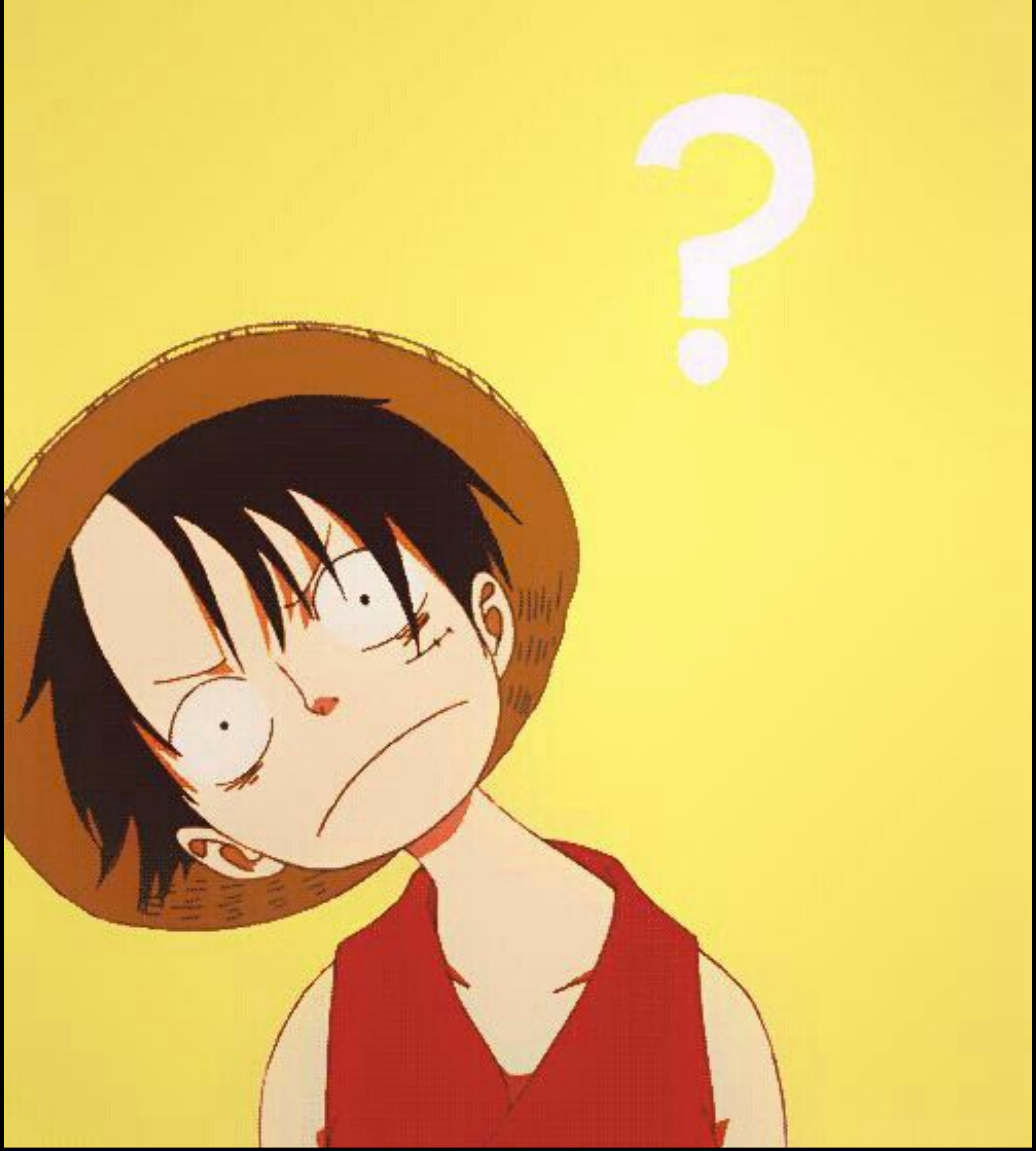
Pivotal Labs

- Cloud Foundry
- Pivotal Tracker
- Gemfire
- Spring Framework
- RabbitMQ
- Concourse

Agenda

- Yet Another CI / CD
- Architecture
- Usage
- Demo
- Q&A

Yet Another CI / CD

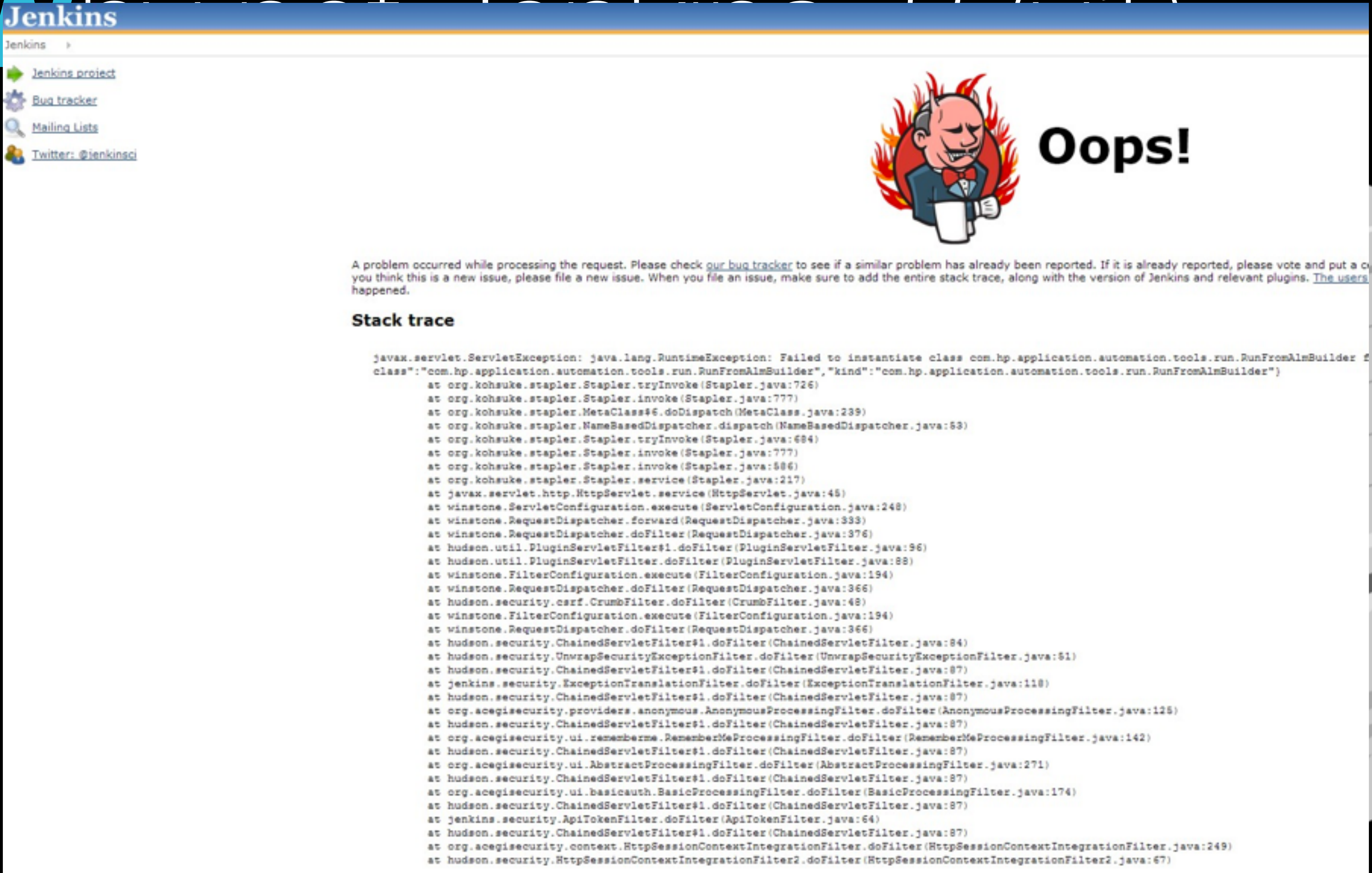


But

Why not Jenkins ? (v1)

- Snowflakes
- No strict isolation
- Too much dependency
- well java ….
- V2 really ?! pipeline yes ! but Groovy ..

Why not Jenkins? (v1)



A problem occurred while processing the request. Please check our bug tracker to see if a similar problem has already been reported. If it is already reported, please vote and put a c
you think this is a new issue, please file a new issue. When you file an issue, make sure to add the entire stack trace, along with the version of Jenkins and relevant plugins. The users
happened.

Stack trace

```
javax.servlet.ServletException: java.lang.RuntimeException: Failed to instantiate class com.hp.application.automation.tools.run.RunFromAlmBuilder f
class":"com.hp.application.automation.tools.run.RunFromAlmBuilder","kind":"com.hp.application.automation.tools.run.RunFromAlmBuilder")
        at org.kohsuke.stapler.Stapler.tryInvoke(Stapler.java:726)
        at org.kohsuke.stapler.Stapler.invoke(Stapler.java:777)
        at org.kohsuke.stapler.MetaClass$6.doDispatch(MetaClass.java:239)
        at org.kohsuke.stapler.NameBasedDispatcher.dispatch(NameBasedDispatcher.java:53)
        at org.kohsuke.stapler.Stapler.tryInvoke(Stapler.java:684)
        at org.kohsuke.stapler.Stapler.invoke(Stapler.java:777)
        at org.kohsuke.stapler.Stapler.invoke(Stapler.java:586)
        at org.kohsuke.stapler.Stapler.service(Stapler.java:217)
        at javax.servlet.http.HttpServlet.service(HttpServlet.java:45)
        at winstone.ServletConfiguration.execute(ServletConfiguration.java:248)
        at winstone.RequestDispatcher.forward(RequestDispatcher.java:333)
        at winstone.RequestDispatcher.doFilter(RequestDispatcher.java:376)
        at hudson.util.PluginServletFilter$1.doFilter(PluginServletFilter.java:96)
        at hudson.util.PluginServletFilter.doFilter(PluginServletFilter.java:88)
        at winstone.FilterConfiguration.execute(FilterConfiguration.java:194)
        at winstone.RequestDispatcher.doFilter(RequestDispatcher.java:366)
        at hudson.security.csrf.CrumbFilter.doFilter(CrumbFilter.java:48)
        at winstone.FilterConfiguration.execute(FilterConfiguration.java:194)
        at winstone.RequestDispatcher.doFilter(RequestDispatcher.java:366)
        at hudson.security.ChainedServletFilter$1.doFilter(ChainedServletFilter.java:84)
        at hudson.security.UnwrapSecurityExceptionFilter.doFilter(UnwrapSecurityExceptionFilter.java:51)
        at hudson.security.ChainedServletFilter$1.doFilter(ChainedServletFilter.java:87)
        at jenkins.security.ExceptionTranslationFilter.doFilter(ExceptionTranslationFilter.java:118)
        at hudson.security.ChainedServletFilter$1.doFilter(ChainedServletFilter.java:87)
        at org.acegisecurity.providers.anonymous.AnonymousProcessingFilter.doFilter(AnonymousProcessingFilter.java:125)
        at hudson.security.ChainedServletFilter$1.doFilter(ChainedServletFilter.java:87)
        at org.acegisecurity.ui.rememberme.RememberMeProcessingFilter.doFilter(RememberMeProcessingFilter.java:142)
        at hudson.security.ChainedServletFilter$1.doFilter(ChainedServletFilter.java:87)
        at org.acegisecurity.ui.AbstractProcessingFilter.doFilter(AbstractProcessingFilter.java:271)
        at hudson.security.ChainedServletFilter$1.doFilter(ChainedServletFilter.java:87)
        at org.acegisecurity.ui.basicauth.BasicProcessingFilter.doFilter(BasicProcessingFilter.java:174)
        at hudson.security.ChainedServletFilter$1.doFilter(ChainedServletFilter.java:87)
        at jenkins.security.ApiTokenFilter.doFilter(ApiTokenFilter.java:64)
        at hudson.security.ChainedServletFilter$1.doFilter(ChainedServletFilter.java:87)
        at org.acegisecurity.context.HttpSessionContextIntegrationFilter.doFilter(HttpSessionContextIntegrationFilter.java:249)
        at hudson.security.HttpSessionContextIntegrationFilter2.doFilter(HttpSessionContextIntegrationFilter2.java:67)
```

Why not ?

- No pipeline
- Difficult to debug, commit to debug

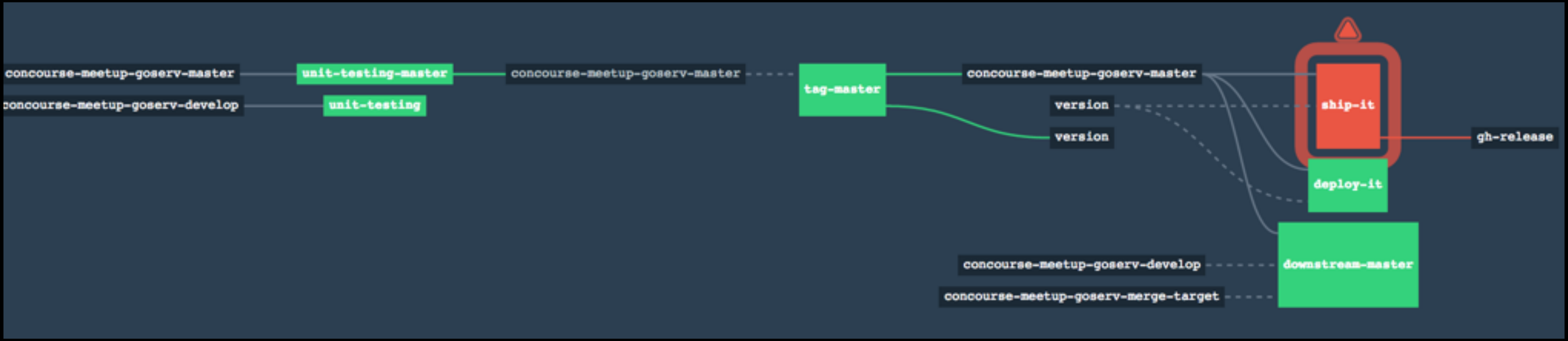
- Difficult to use / configure ..
- Learning curve is high
- Difficult to see everything

WE ARE ENGINEER

Concourse

- Pluggable Resource Interface
- Running builds in Containers Natively
- Zero Snowflake-able Configuration
- Submitting builds from the local file system up to run in CI
- Pipeline pipeline pipeline !!!
- Reproducible

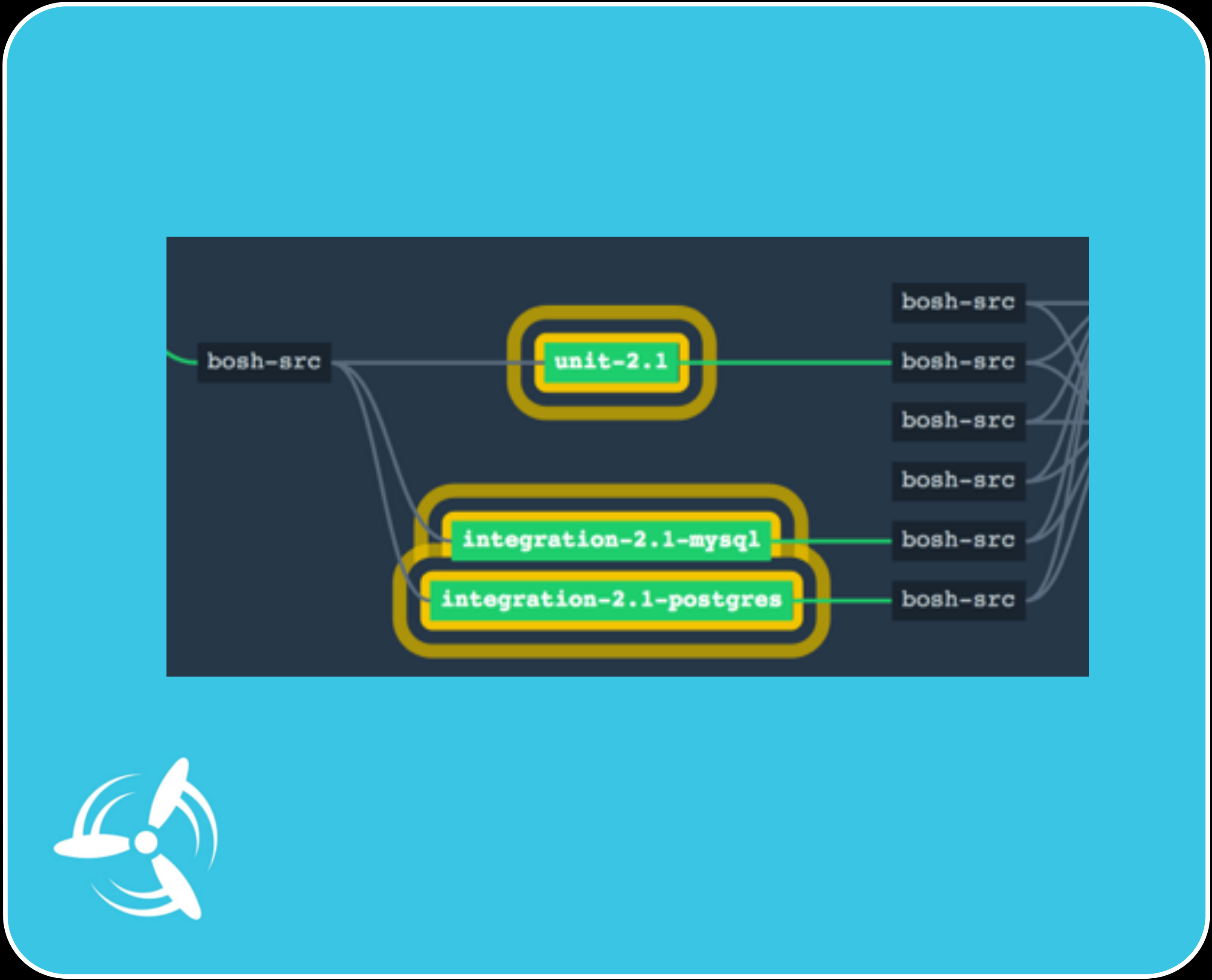
Concourse


concourse-meetup-goserv-master
unit-testing-master
concourse-meetup-goserv-develop
unit-testing
concourse-meetup-goserv-master
tag-master
concourse-meetup-goserv-master
version
version
ship-it
gh-release
deploy-it
concourse-meetup-goserv-develop
concourse-meetup-goserv-merge-target
downstream-master

Concourse

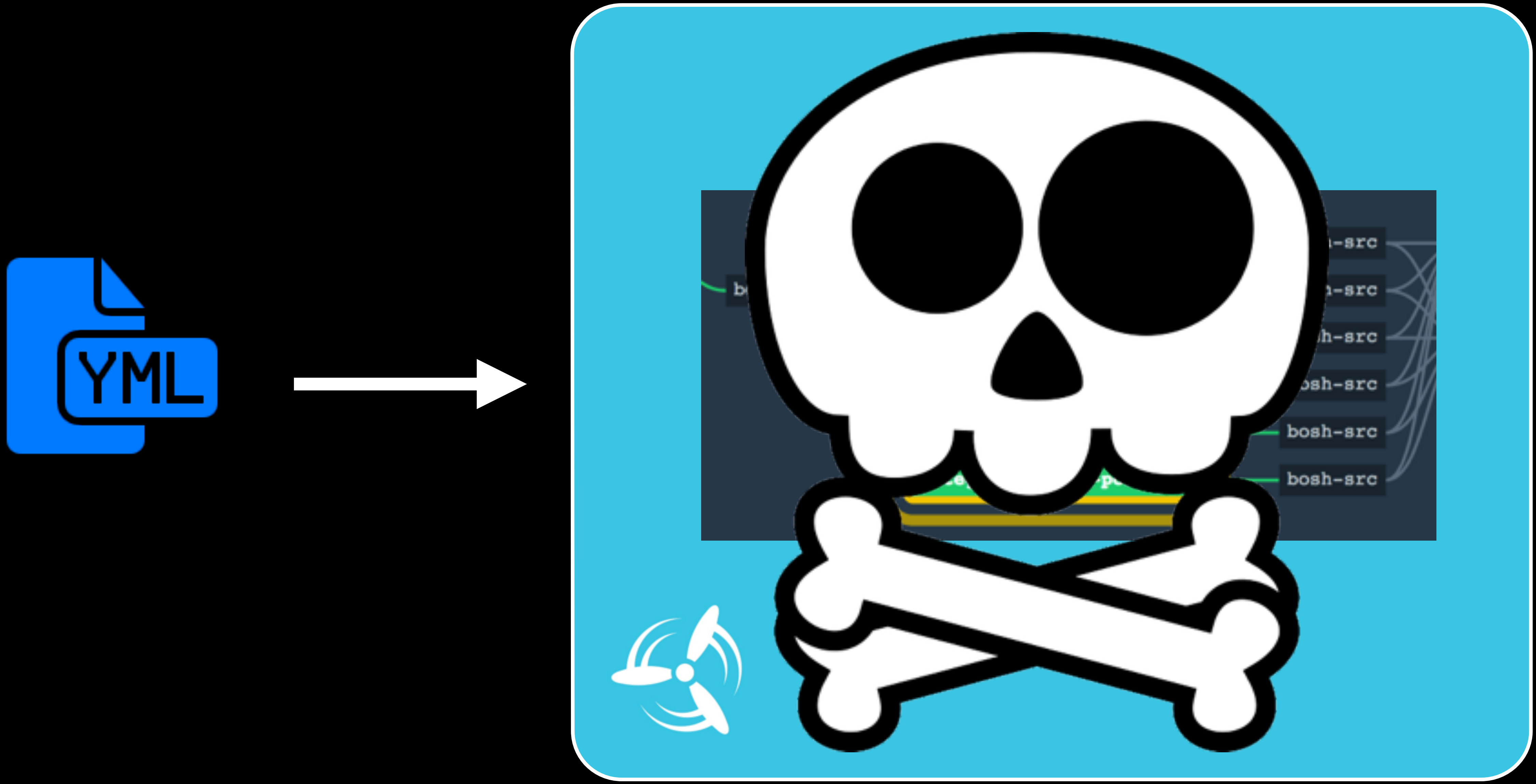
- Pluggable Resource Interface
- Running builds in Containers Natively
- Zero Snowflake-able Configuration
- Submitting builds from the local file system up to run in CI
- Pipeline pipeline pipeline !!! ✓
- Reproducible

Concourse

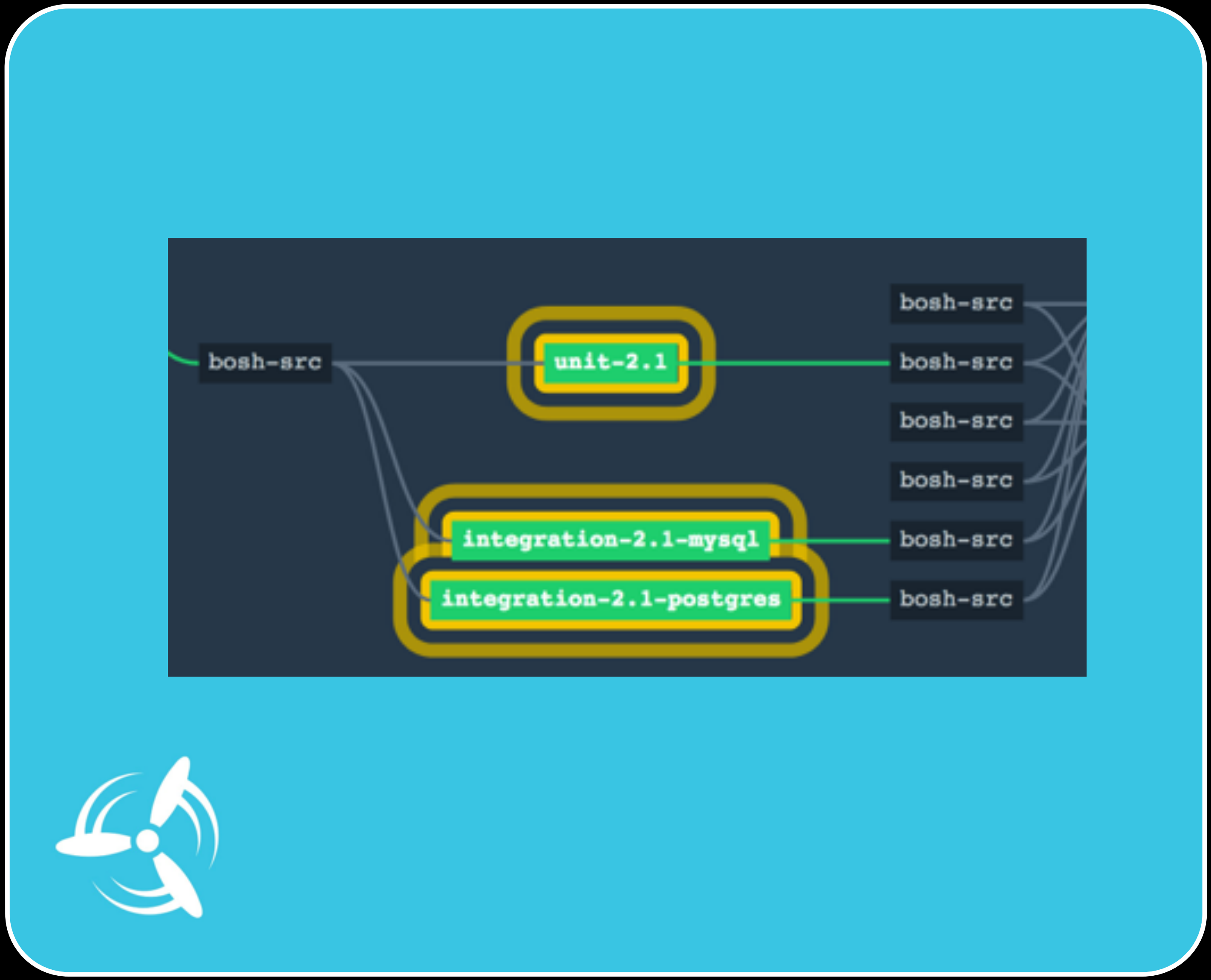


Yes yaml ….

Concourse



Concourse



Download Yaml

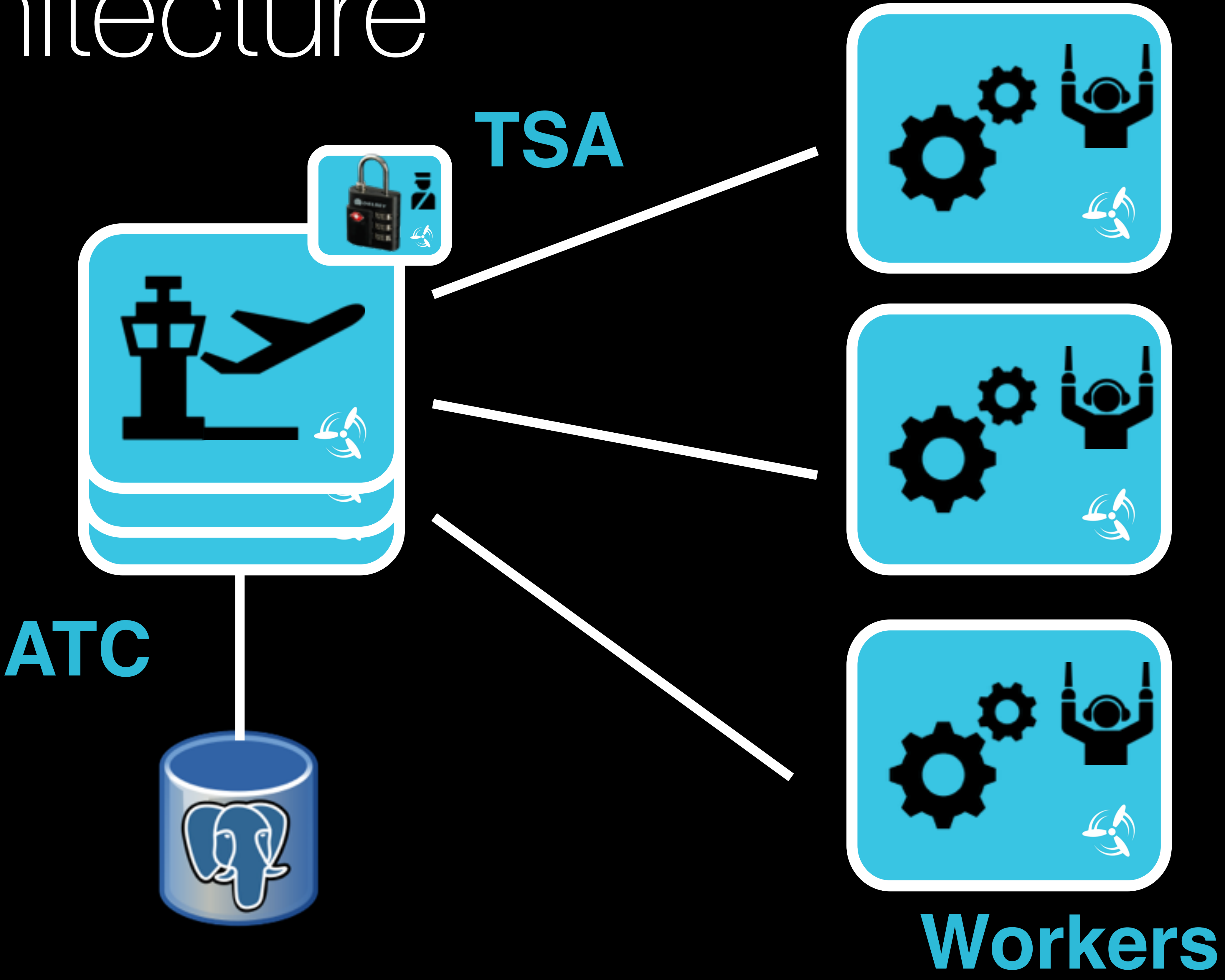
Set-pipeline in concourse

Concourse

- Pluggable Resource Interface
- Running builds in Containers Natively
- Zero Snowflake-able Configuration
- Submitting builds from the local file system up to run in CI
- Pipeline pipeline pipeline !!! ✓
- Reproducible ✓

Architecture

Architecture



Architecture

ATC Air Traffic Controller

- Brain
- **Web UI** (ELM)
- Responsible for Pipeline **scheduling**

ELM

Architecture

TSA Travel Security Agency

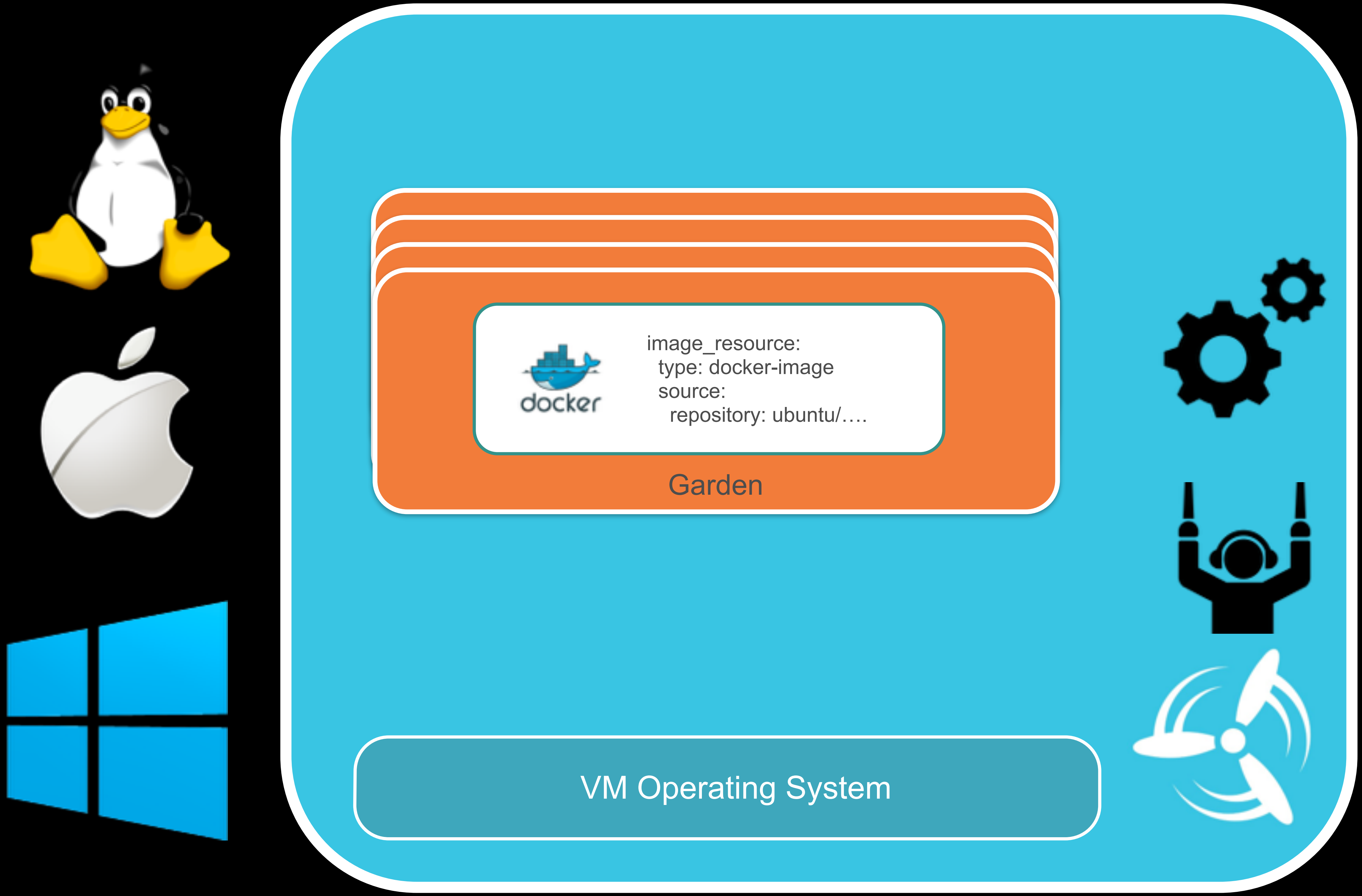
- **Register** workers against ATC
- **Unregister** workers against ATC

Architecture

Workers

- No State
- Run the task
- Container inside **Garden** container
- Register themselves to the TSA

Architecture Workers



```
image_resource:
  type: docker-image
  source:
    repository: ubuntu/….
```

Concourse

- Pluggable Resource Interface
- Running builds in Containers Natively ✓
- Zero Snowflake-able Configuration ✓
- Submitting builds from the local file system up to run in CI
- Pipeline pipeline pipeline !!! ✓
- Reproducible ✓

Concept

Three main concept

- Task

- Jobs
- Resource

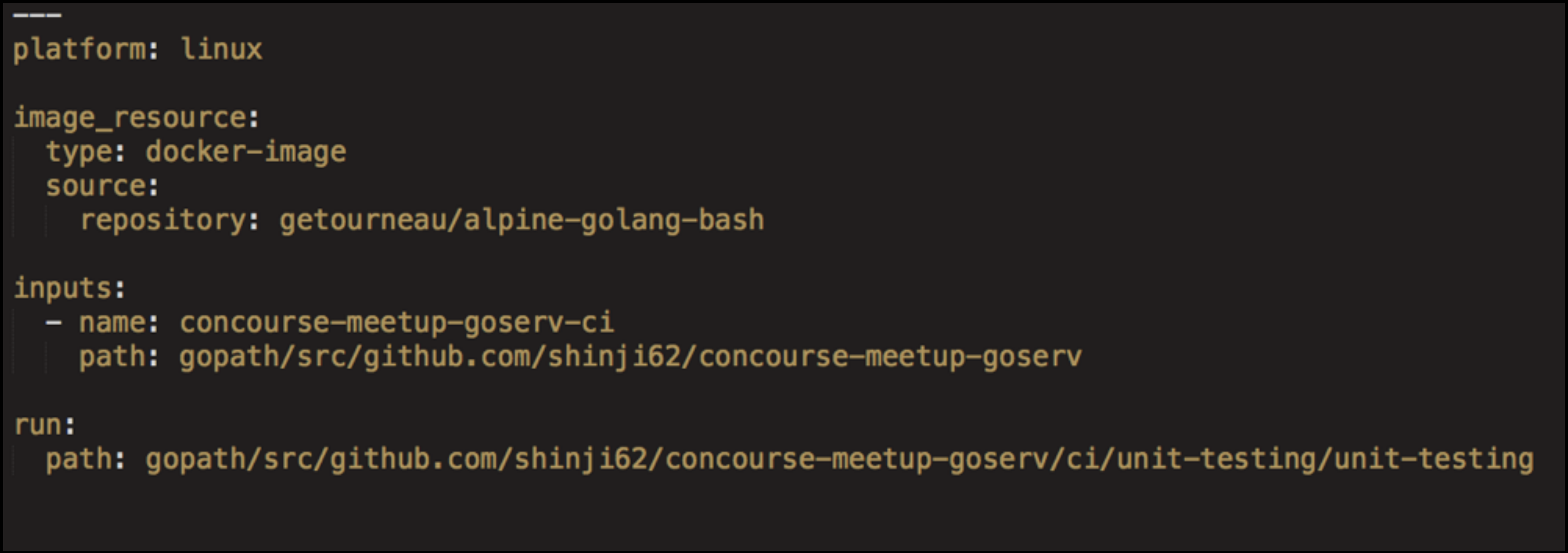
} PIPELINE

Task

- Smallest configurable unit in a Concourse pipeline
- Always behave same ways if inputs are the same result should be the same
- Basically run things into containers
- Input and output
- Exit 0 ==> **Success**
- Exit != 0 ==> **failed**

Task

unit-testing.yml



```yaml
---
platform: linux

image_resource:
  type: docker-image
  source:
    repository: getourneau/alpine-golang-bash

inputs:
  - name: concourse-meetup-goserv-ci
    path: gopath/src/github.com/shinji62/concourse-meetup-goserv

run:
  path: gopath/src/github.com/shinji62/concourse-meetup-goserv/ci/unit-testing/unit-testing
```

platform

Worker type

image_resource

Docker image used to run the task

inputs

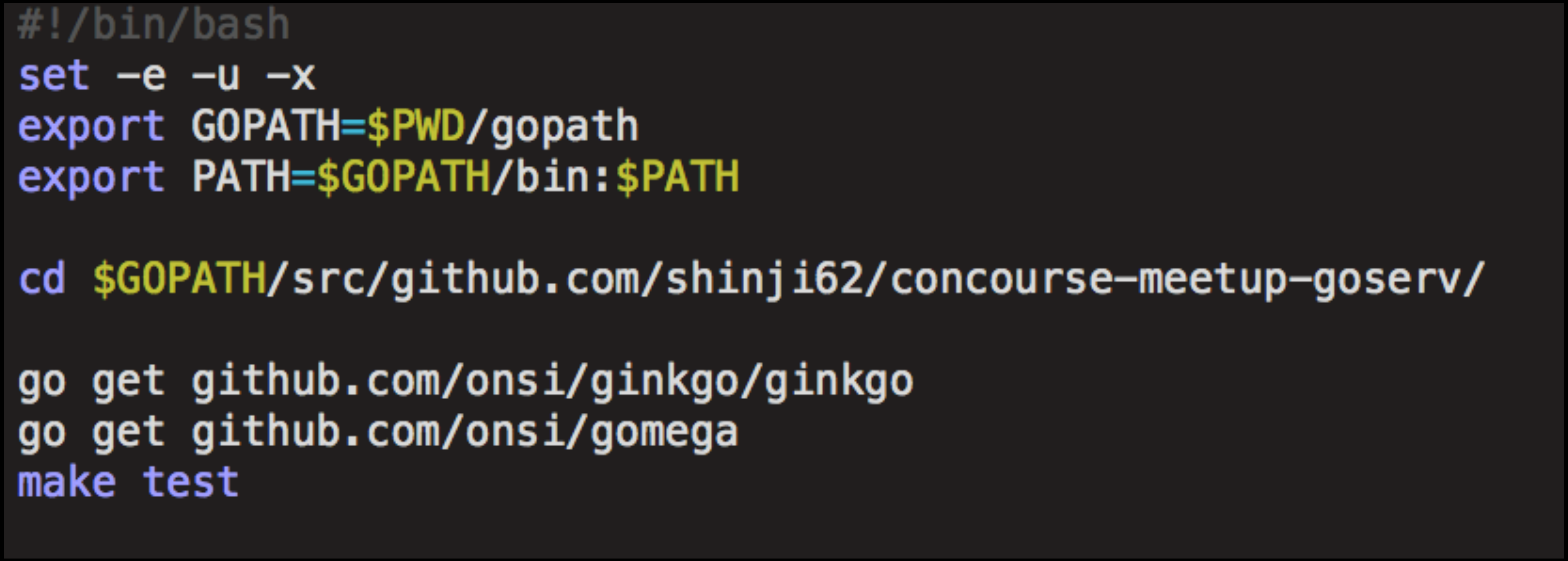
input used by the task

run

what to do

Task

unit-testing

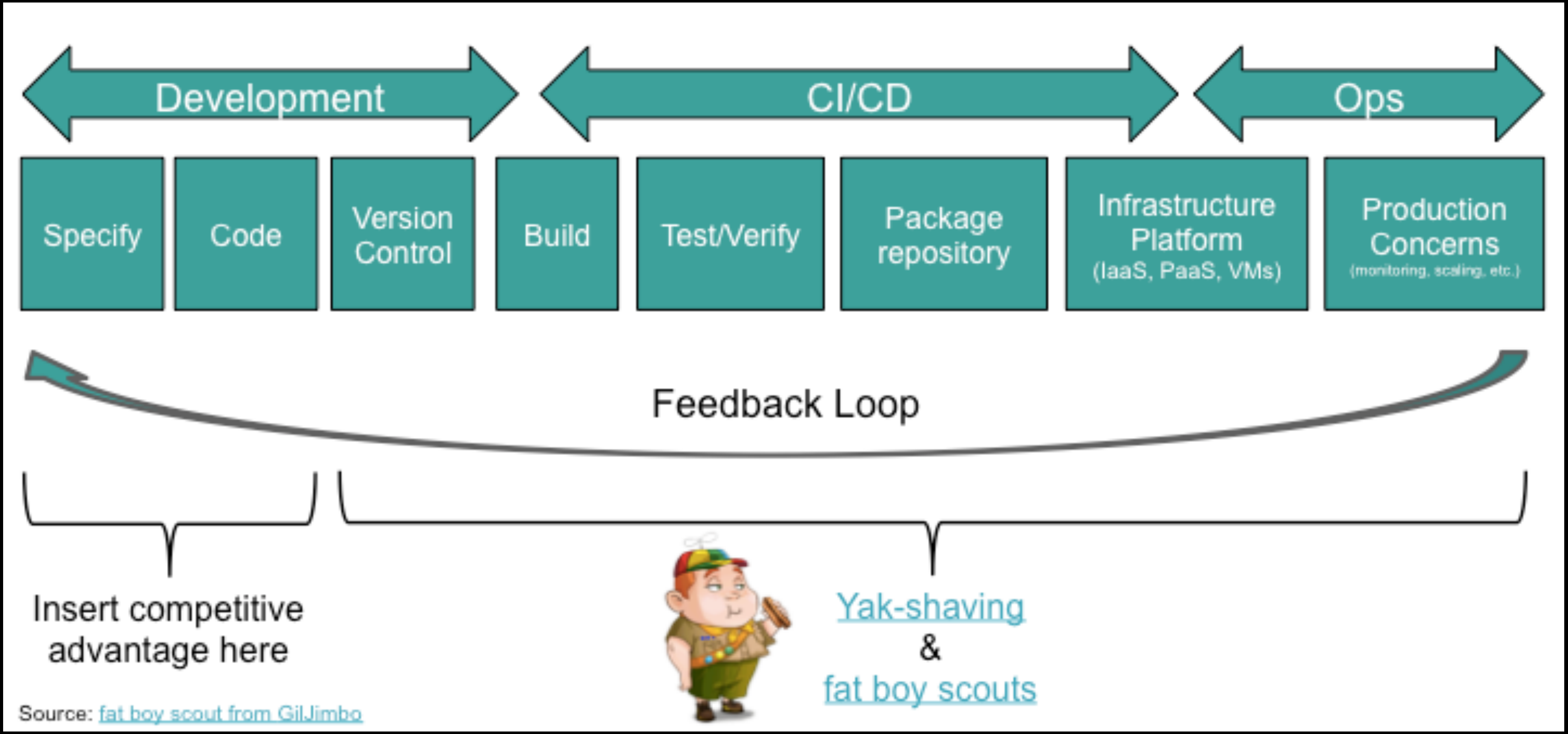


```
#!/bin/bash
set -e -u -x
export GOPATH=$PWD/gopath
export PATH=$GOPATH/bin:$PATH

cd $GOPATH/src/github.com/shinji62/concourse-meetup-goserv/

go get github.com/onsi/ginkgo/ginkgo
go get github.com/onsi/gomega
make test
```

Concourse

- Pluggable Resource Interface
- Running builds in Containers Natively ✓
- Zero Snowflake-able Configuration ✓
- Submitting builds from the local file system up to run in CI ✓
- Pipeline pipeline pipeline !!! ✓
- Reproducible ✓

Pipeline



Development

CI/CD

Ops

Specify

Code

Version Control

Build

Test/Verify

Package repository

Infrastructure Platform (IaaS, PaaS, VMs)

Production Concerns (monitoring, scaling, etc.)

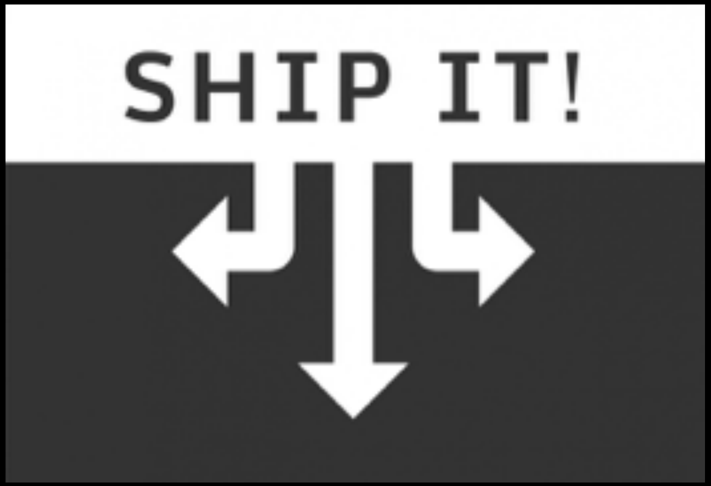
Feedback Loop

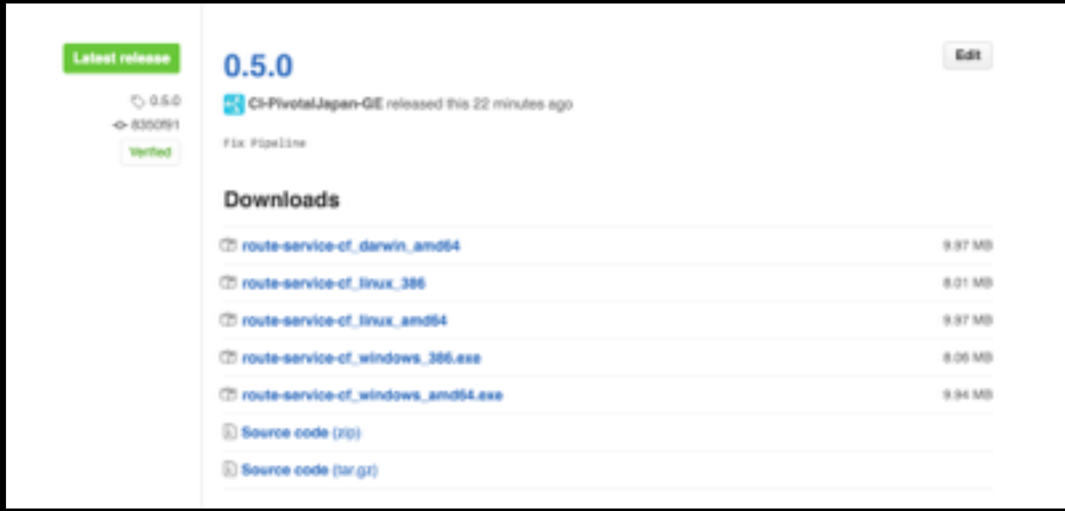
Insert competitive advantage here

Yak-shaving & fat boy scouts

Source: fat boy scout from GilJimbo

Pipeline





0.5.0

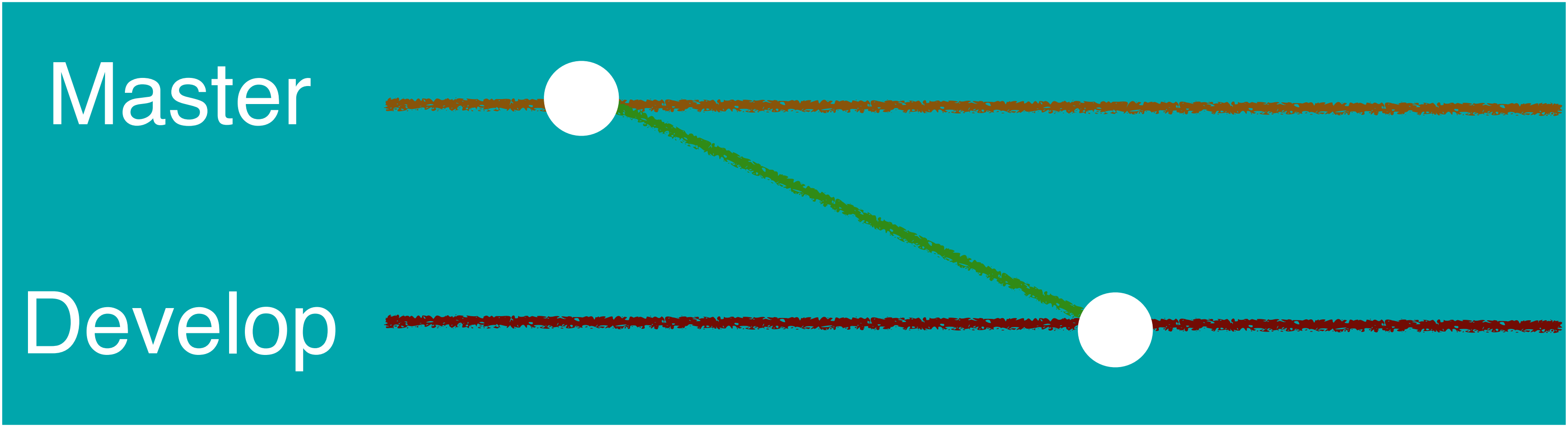
⊘ 0.5.0
-o- 8350f91

CI-PivotalJapan-GE released this 22 minutes ago

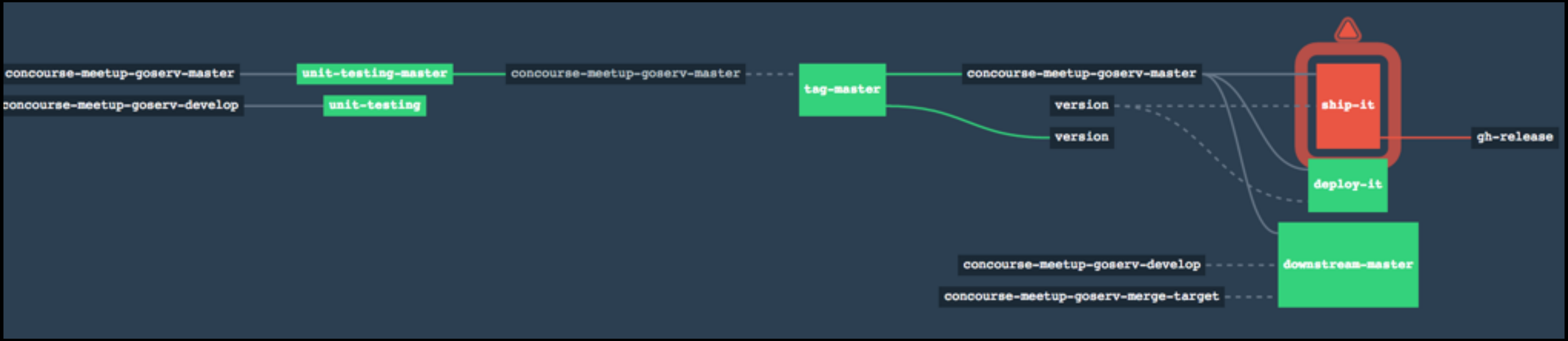
```
Fix Pipeline
```

Downloads

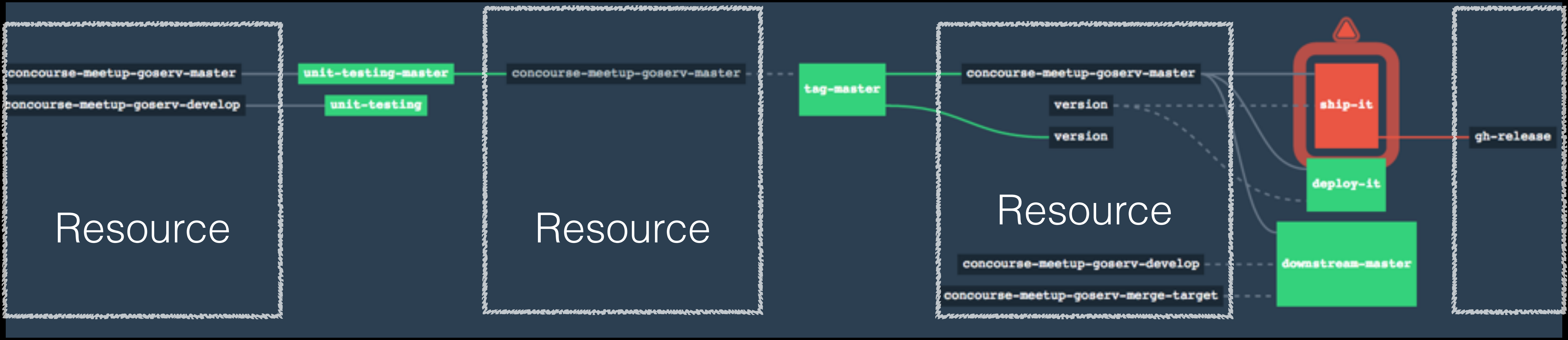
route-service-cf_darwin_amd64	9.97 MB
route-service-cf_linux_386	8.01 MB
route-service-cf_linux_amd64	9.97 MB
route-service-cf_windows_386.exe	8.06 MB
route-service-cf_windows_amd64.exe	9.94 MB
Source code (zip)	
Source code (tar.gz)	



Pipeline


concourse-meetup-goserv-master
concourse-meetup-goserv-develop
unit-testing-master
unit-testing
concourse-meetup-goserv-master
tag-master
concourse-meetup-goserv-master
version
version
ship-it
gh-release
deploy-it
downstream-master
concourse-meetup-goserv-develop
concourse-meetup-goserv-merge-target

Resource


concourse-meetup-goserv-master
concourse-meetup-goserv-develop
unit-testing-master
unit-testing
Resource
concourse-meetup-goserv-master
Resource
tag-master
concourse-meetup-goserv-master
version
version
Resource
concourse-meetup-goserv-develop
concourse-meetup-goserv-merge-target
ship-it
deploy-it
downstream-master
gh-release

Resource

- Objects used for jobs in the pipeline.
- Base on the resource type, ex : git, s3, github-release, semver, slack…. even your own
- As well as input or output

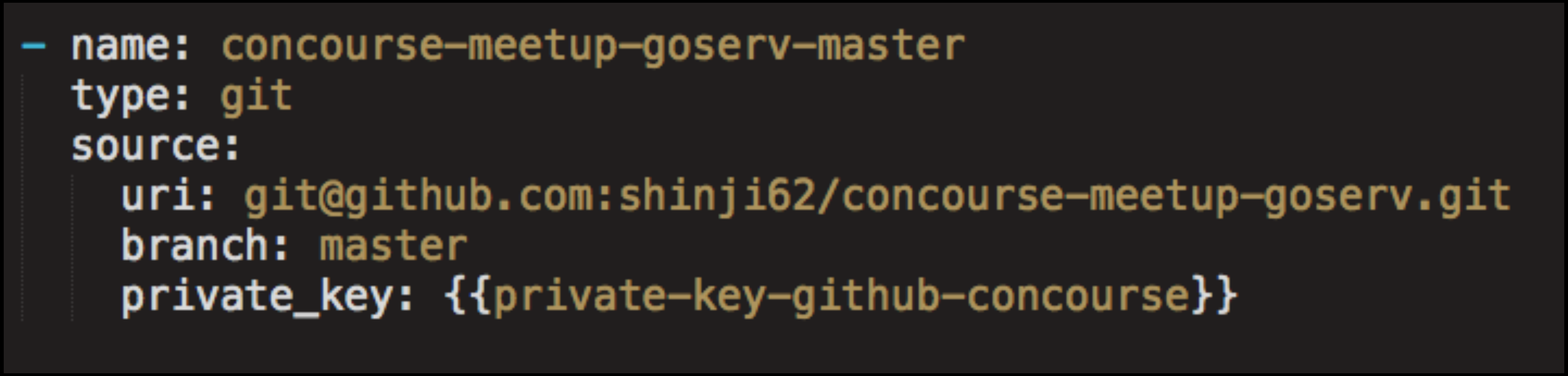
Official

- Git
- Git-release
- Docker Image
- S3
- Semver
- Etc ..

Community

- Slack
- Twitter
- Perforce
- Bintray
- Flowdoc
- Etc …

Resource



```
- name: concourse-meetup-goserv-master
  type: git
  source:
    uri: git@github.com:shinji62/concourse-meetup-goserv.git
    branch: master
    private_key: {{private-key-github-concourse}}
```

name

Name of resource

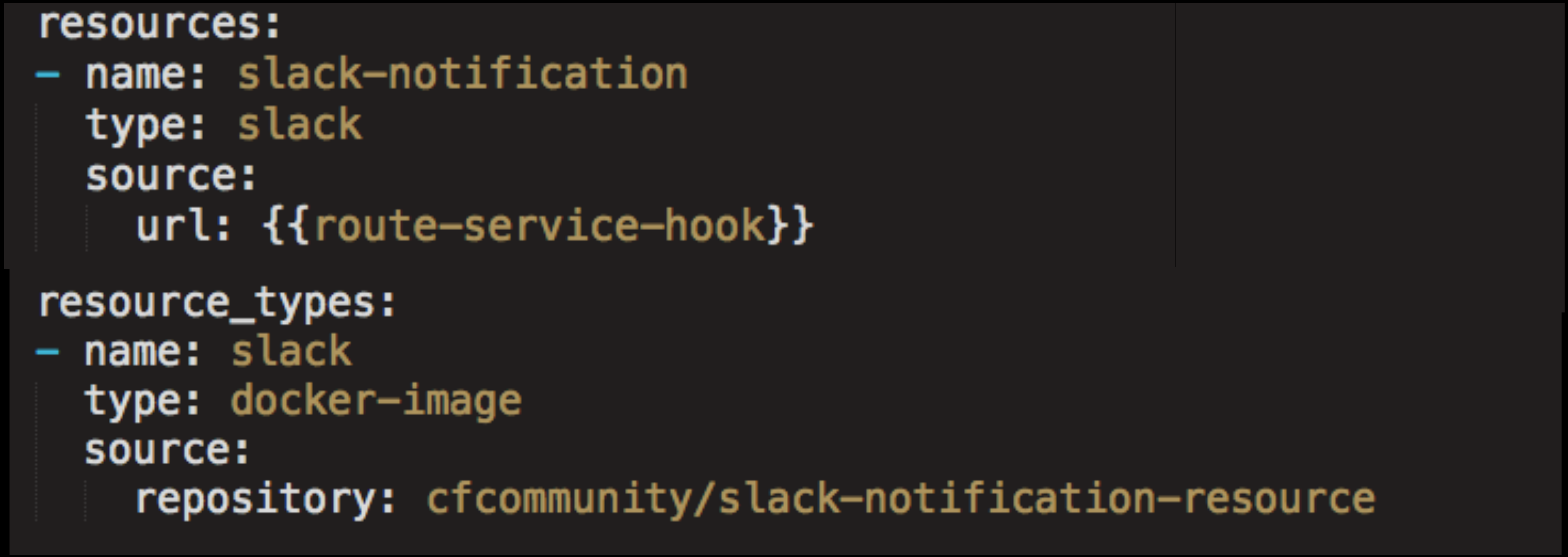
type

resource type in this case git resource

Source, etc…

depend of the resource

Resource (own resource type)



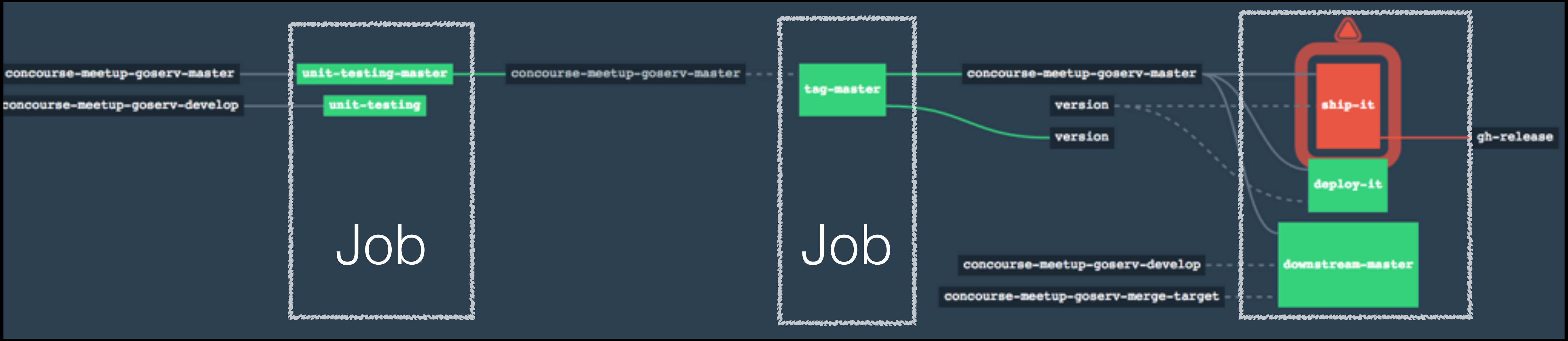
```
resources:
- name: slack-notification
  type: slack
  source:
    url: {{route-service-hook}}

resource_types:
- name: slack
  type: docker-image
  source:
    repository: cfcommunity/slack-notification-resource
```

Can import new resource type directly into the pipeline

Concourse

- Pluggable Resource Interface ✓
- Running builds in Containers Natively ✓
- Zero Snowflake-able Configuration ✓
- Submitting builds from the local file system up to run in CI ✓
- Pipeline pipeline pipeline !!! ✓
- Reproducible ✓

Job


concourse-meetup-goserv-master
concourse-meetup-goserv-develop
unit-testing-master
unit-testing
Job
concourse-meetup-goserv-master
tag-master
Job
concourse-meetup-goserv-master
version
version
ship-it
gh-release
deploy-it
downstream-master
concourse-meetup-goserv-develop
concourse-meetup-goserv-merge-target

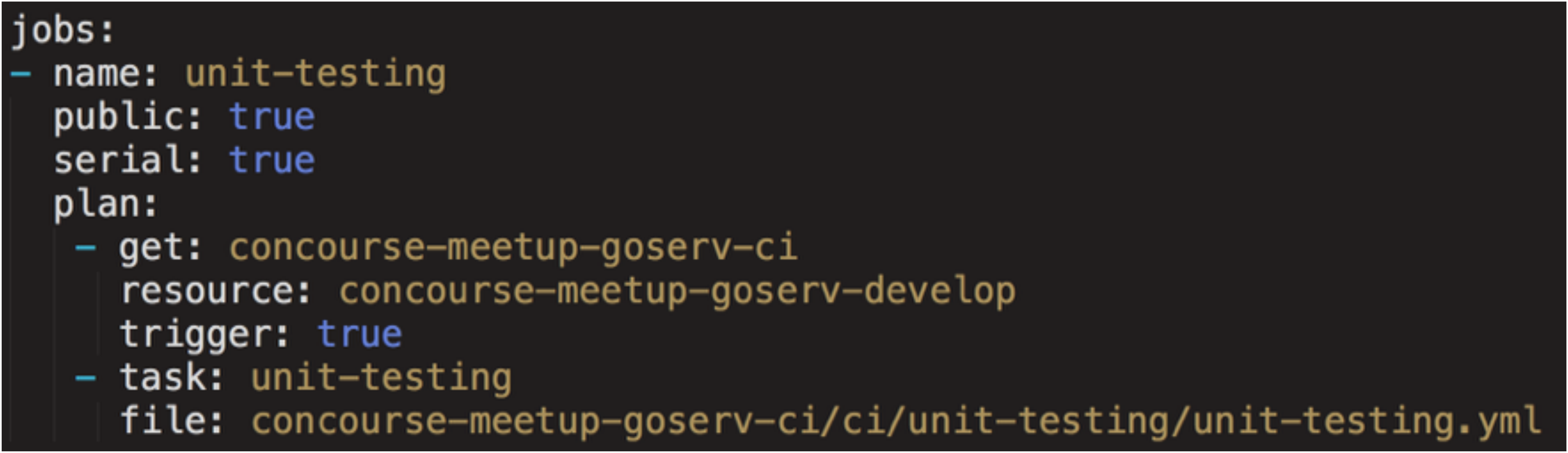
Job

Job

- Determine “actions” of the pipeline
- Plan, which resource, which task and in which order
- Basically
    - What I need to do ? ==> Task
    - What I need to use ? ==> Resource
    - What I need to produce ? ==> Resource

Job

pipeline.yml



```
jobs:
- name: unit-testing
  public: true
  serial: true
  plan:
  - get: concourse-meetup-goserv-ci
    resource: concourse-meetup-goserv-develop
    trigger: true
  - task: unit-testing
    file: concourse-meetup-goserv-ci/ci/unit-testing/unit-testing.yml
```

name

Name of jobs

public

Visibility

serial

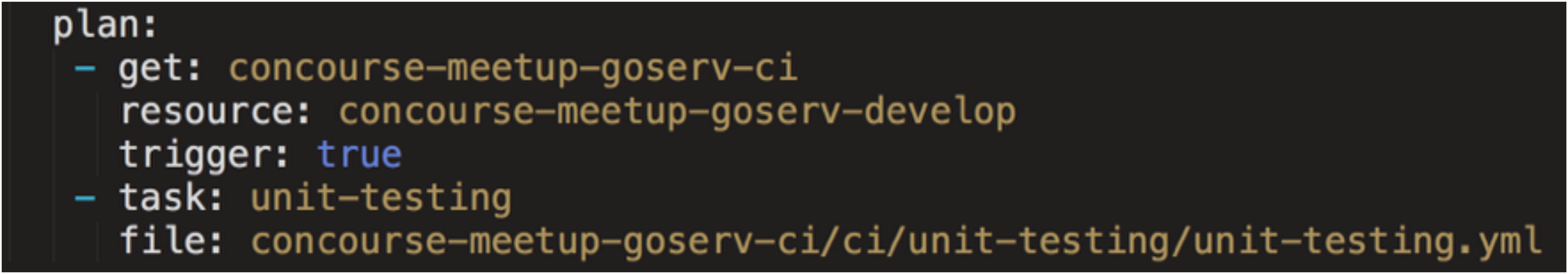
input used by the task

plan

builds_plan

Job Build-plans

pipeline.yml



```
plan:
- get: concourse-meetup-goserv-ci
  resource: concourse-meetup-goserv-develop
  trigger: true
- task: unit-testing
  file: concourse-meetup-goserv-ci/ci/unit-testing/unit-testing.yml
```

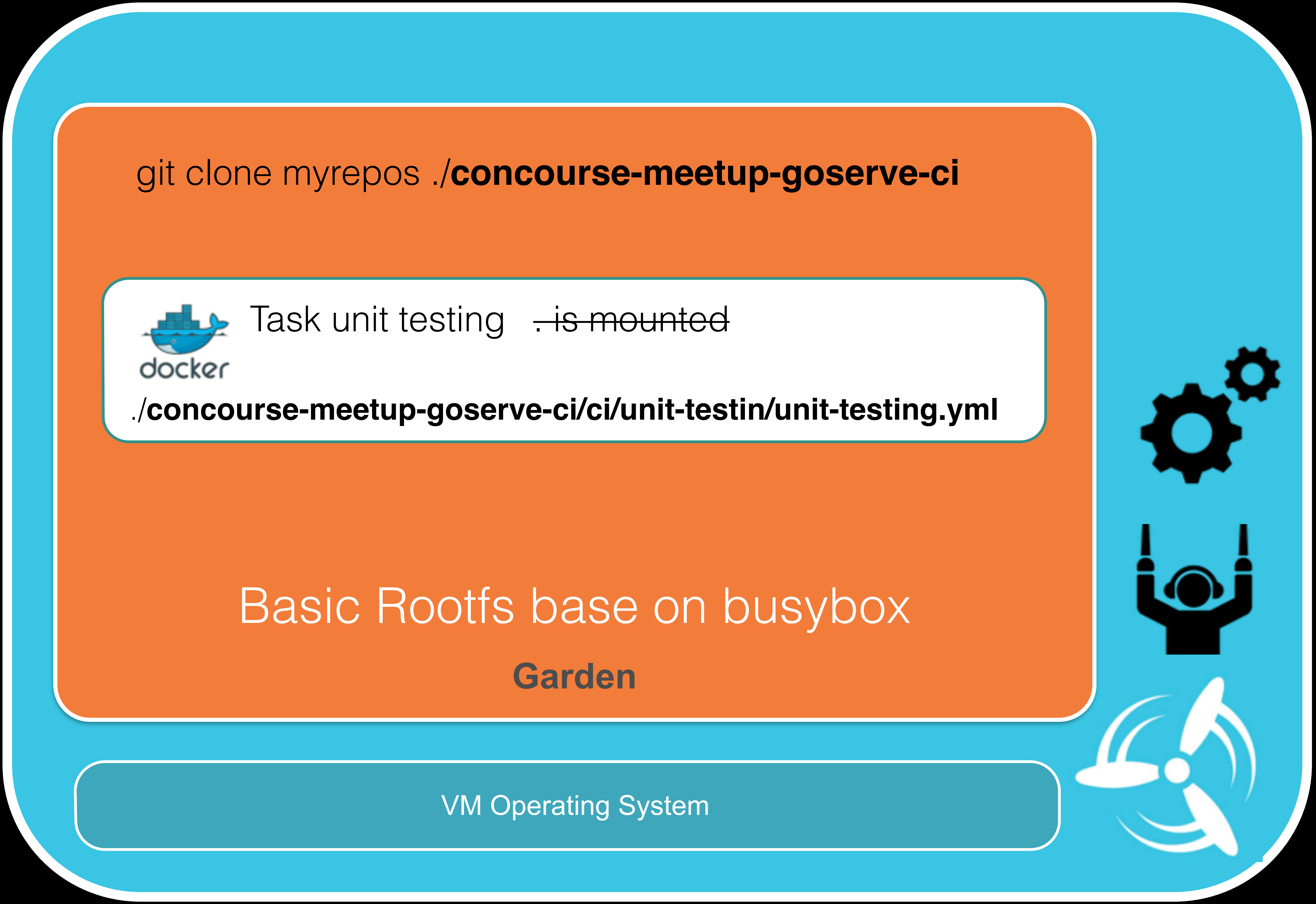
1 Job = 1 build-plan

Get

Input resource trigger the job new version of the resource

Task

Run the task unit-testing



git clone myrepos ./**concourse-meetup-goserve-ci**

Task unit testing ~~. is mounted~~

./**concourse-meetup-goserve-ci/ci/unit-testin/unit-testing.yml**

Basic Rootfs base on busybox

Garden

VM Operating System

Usage

Installation

Vagrant for local deployment

```
vagrant init concourse/lite
vagrant up
```

Bosh for Cluster deployment, aws, azure, gce

Configure your deployment file
Deploy your cluster

docker, chef etc… you can use the concourse binary

Installation

Concourse Binary

You can use the concourse binary with your favorite tool.
Or no tool at all

https://github.com/concourse/bin

First Step

login login and create target to be use for all command

```
$ fly -t blite login --concourse-url https://ci.example.com
```

---

sync To be sure your client is on the same version of your Concourse deployment

```
$ fly -t blite sync
```

One off-Task

execute Run one-off task from the local directory

```
$ fly -t blite execute -c my-task.yml --input my-input-res=.
```

I want to execute the task my-task.yml using my local directory as input my-input-res

Pipeline

set-pipeline Set or update the pipeline

```
$ fly -t blite set-pipeline -p pipeline-name -c ci/pipeline.yml
        --load-vars-from=/secret/credentials.yml
```

I want to setup the pipeline defined in ci/pipeline.yml
and I will use the file credentials.ml for variable

Demo

One off-Task

Running unit-testing task

```
$fly -t clocal execute -c ci/unit-testing/unit-testing.yml
--input concourse-meetup-goserv-ci=.
```

Pipeline

set-pipeline Set or update the pipeline

```
$ fly -t clocal set-pipeline -p -p concourse-meetup-c ci/pipeline.yml
```

Online Resource

Documentation

- Official documentation http://concourse.ci

Tutorials

- S&B https://github.com/starkandwayne/concourse-tutorial
- Flight-School http://concourse.ci/flight-school.html

Public Pipeline

- Buildpack
  - http://buildpacks-ci.cfapps.io/
- CF Release
  - https://runtime.ci.cf-app.com/pipelines/cf-release
- Bosh
  - https://main.bosh-ci.cf-app.com/
- Concourse
  - https://ci.concourse.ci/

Slack

http://concourseci.slack.com/

Q & A

Thank You !!