CI PIPELINE IN ACTION

Junpeng Zhu

DaSE of ECNU

zjp.dase@stu.ecnu.edu.cn



CONTENTS

- Concourse Cl
 - What is Concourse CI?
 - Why is Concourse CI?
 - Concourse CI VS. Jenkins
 - Live Demo
- GitHub Actions
 - Live Demo
- Acknowledgement

CONCOURSE CI

WHAT IS CONCOURSE CI?

- Concourse is a CI/CD tool that treats build pipelines and artifacts as first-class citizens.
- It enables builds that run in containers, has a clean, usable UI and discourages snowflake build servers.
- It aims to provide an expressive system with as few distinct moving parts as possible.

WHY IS CONCOURSE CI?

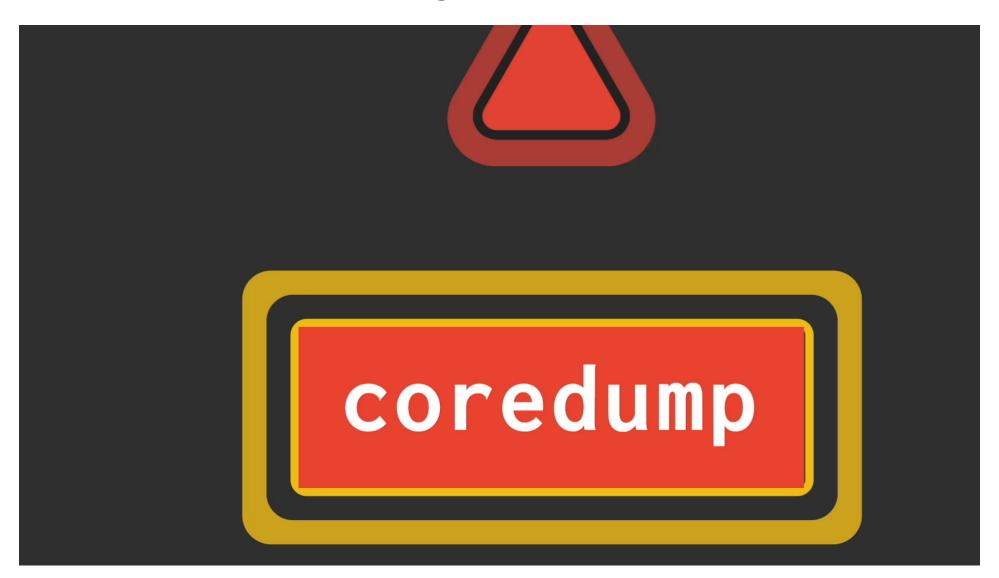
- Automation
- Streamlined
- Support parallel and serial workflows
- YML config file, easy to modify
- Job build in container

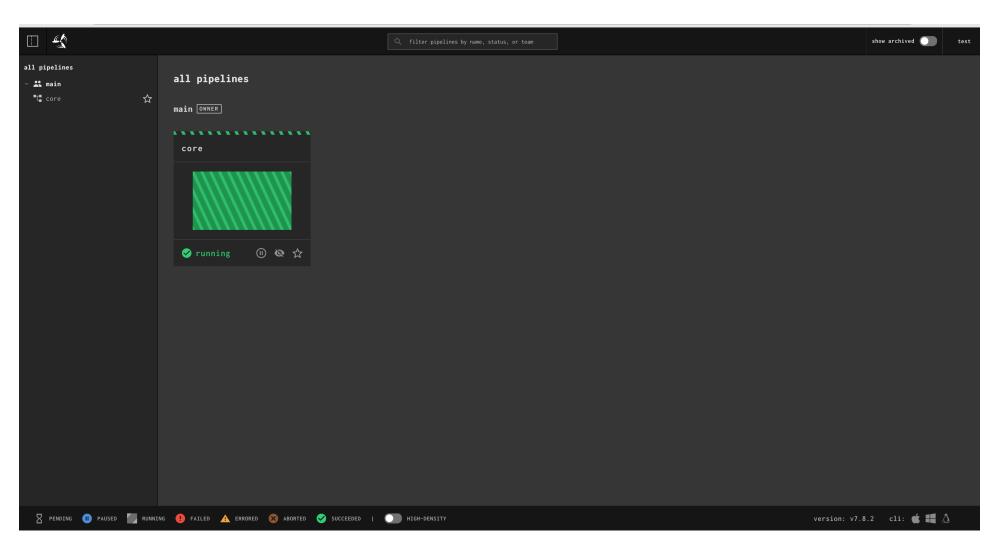
CONCOURSE VS. JENKINS

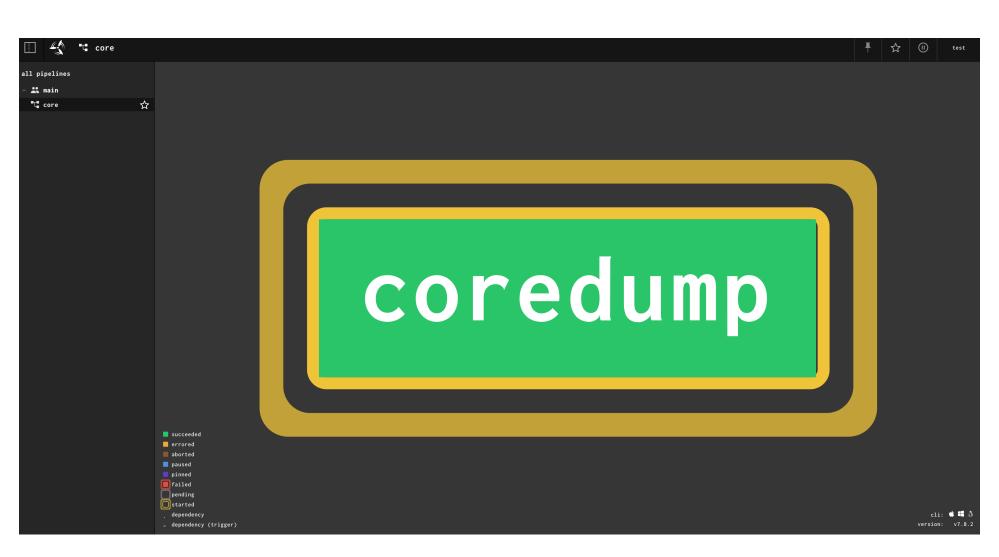
- Jenkins: As the scale of Pipeline expands, there are many copies and pastes, which increase maintenance costs
- Jenkins: With Job as the focus, when the order of Jobs becomes complicated, it is difficult to sort out the relationship clearly
- Jenkins: It does not directly support CD, it needs to be completed with the help of plug-ins
- Jenkins: Plug-in issues may cause the build environment to be polluted

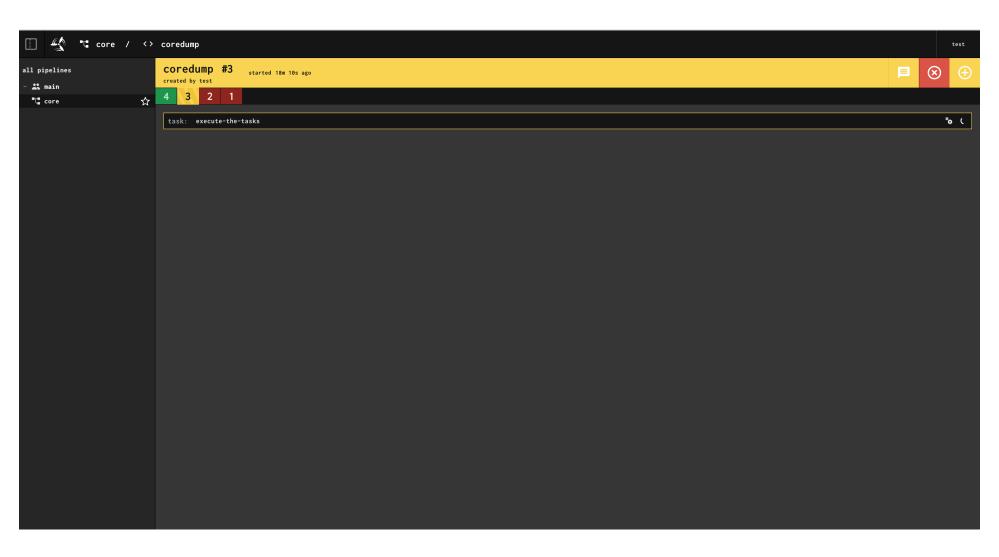
- Concourse: YML files to config and the duplicate content is quoted in multiple places by extracting the file
- Concourse: Each job only cares about its own valid input, local optimization achieves global optimization
- Concourse: Each job is built in a separate Container, and has no impact on other environments
- All files under VCS

Hang Job Demo









LIVE DEMO

The ubuntu/centos amd64 arch 🗹



The m1 pro mac due to the arm arch X



The arm64 ubuntu/centos X



Concourse CI Pipeline

YAML EXAMPLE

Requirements: Concourse will run CI jobs in a docker container. Please create a user-defined docker image and upload it to docker hub. Then Concourse can fetch and use this container. Docker Image requirement:

- base image: Ubuntu20.04
- install package libleveldb-dev
- generate a rsa key.
- [Plus] enable coredump

pipeline.yml Contents

```
jobs:
  - name: coredump
    public: true
    plan:
      - task: execute-the-tasks
        privileged: true
        config:
          platform: linux
          image resource:
            type: docker-image
            source: {
              repository: ubuntu,
              tag: 20.04
```

pipeline.yml Contents

```
run:
  user: root
  path: /bin/bash
  args:
    - "-e"
    - "-c"
      set -x
      ulimit -c unlimited
      echo 'ulimit -c unlimited' >> ~/.bash_profile
      echo '/tmp/core.%t.%e.%p' | tee /proc/sys/kernel/core pa
      apt-get update
      apt-get install git -y
      apt-get install gcc -y
      ant-get install vim -v
```

GITHUB ACTIONS

LIVE DEMO

Github Actions Pipeline

GitHub Actions YAML

```
name: CSAPP CI
on:
  push:
    branches: [ "main" ]
  pull_request:
jobs:
  job-2:
    runs-on: ubuntu-latest
    steps:
    - uses: actions/checkout@v3
    - name: Install dependencies
      run:
        sudo ant-get install gcc-multilih
```

ACKNOWLEDGEMENT

