WEB CONF
<腾 讯 前 端 技 术 大 会/>

# 从 Jenkins 到 GitHub Actions

## CI/CD 现代史

演讲者 苗兆丰

0

# 个人介绍

.yml

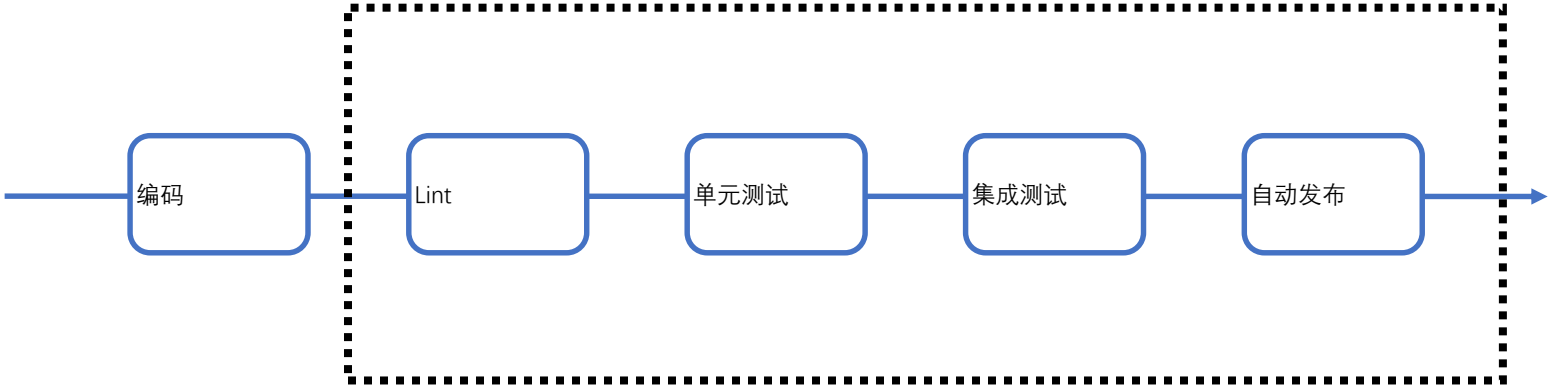
```
name: "苗兆丰"
nickName: "maple"
graduatedFrom: "西安交通大学"
graduatedAt: "2016/7"
workingIn: "腾讯 QQ-Web 团队"
workingOn: "研发效率提升工作"
```

# CI/CD 现代史

之

CI/CD 是什么？

# 1 CI/CD 是什么？




能够频繁、高质量地完成交付

对生活充满希望

# CI/CD 现代史

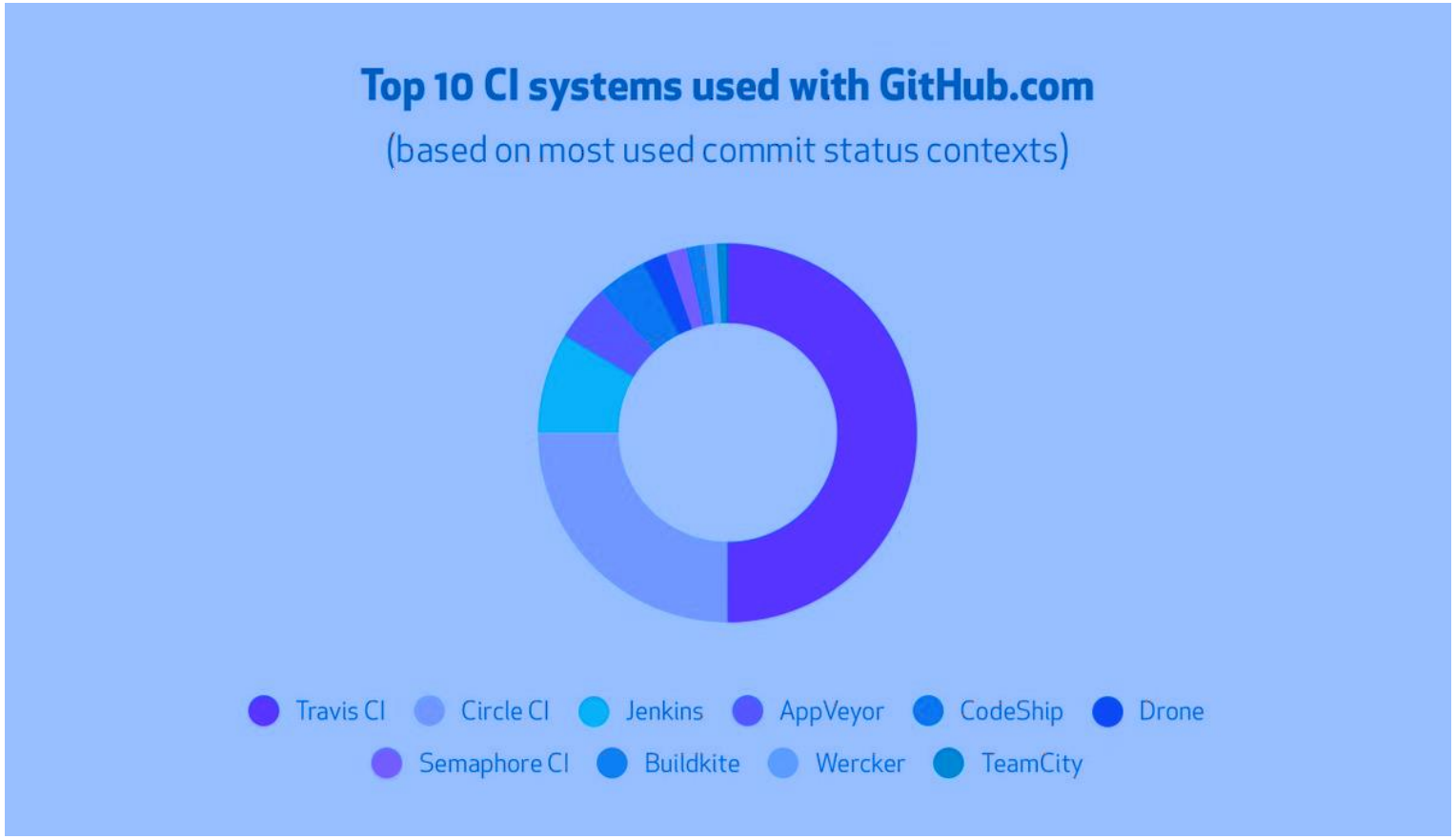
之

CI/CD 是什么？

2

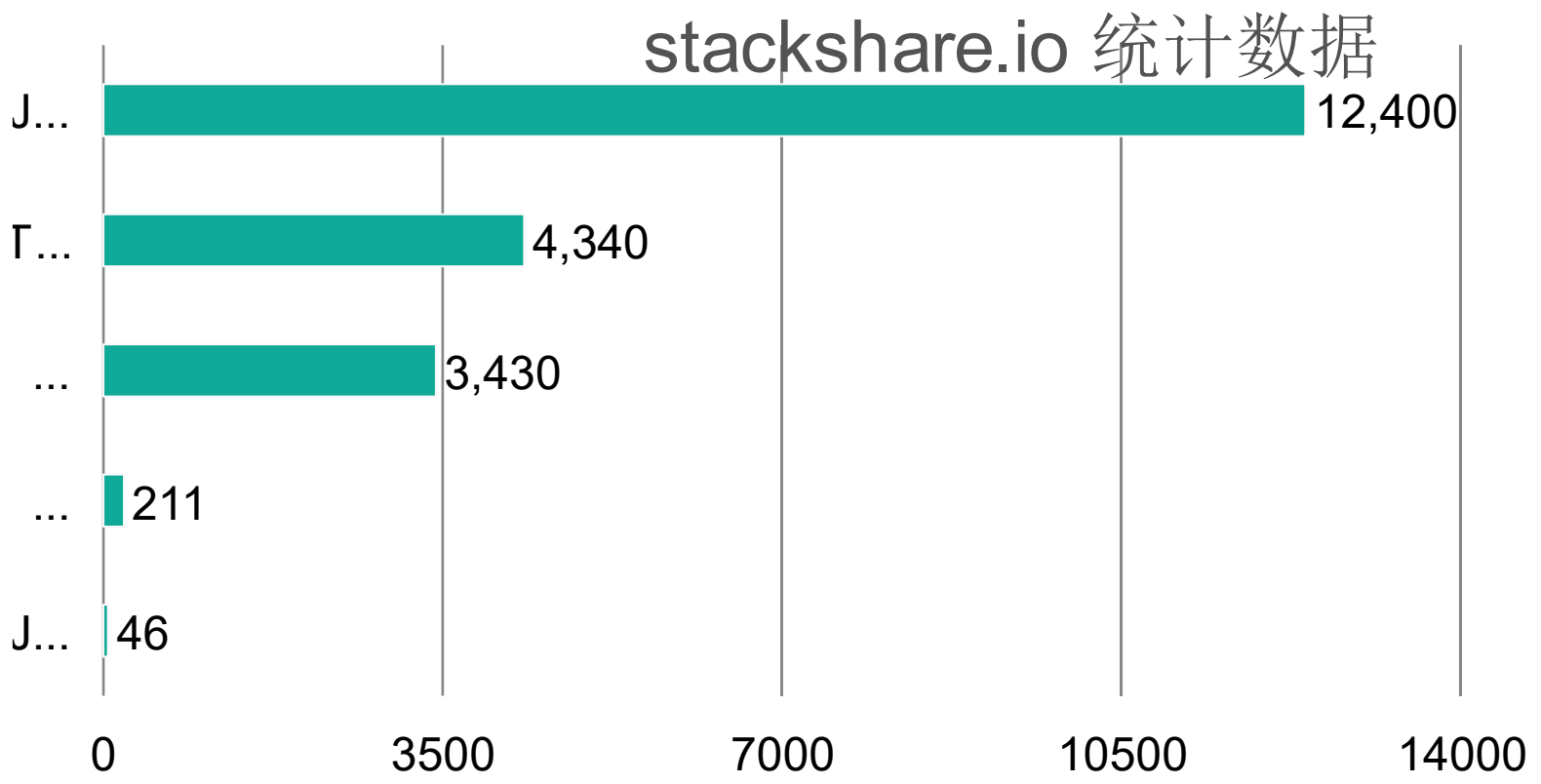
# 行业生态

公有云服务化几乎被 Travis CI/Circle CI 瓜分



# 行业生态

Jenkins 在私有化部署市场一家独大



# 行业生态

是不是数年间 CI 行业毫无进步?

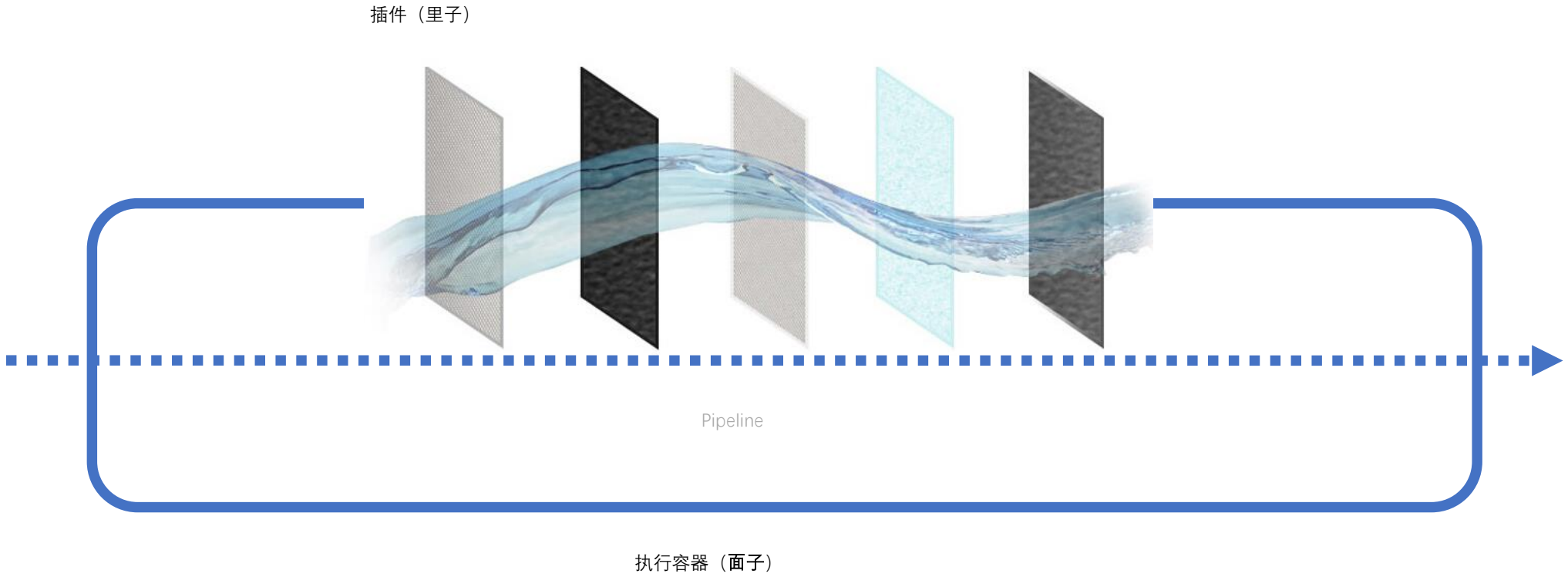
**Jenkins / Travis CI**

2011

# 行业生态

# 2 行业生态

CI 的“面子”和“里子”



# 2 行业生态

要么更新“面子”，要么更新“里子”，要么一起更新！

2

# 行业生态

以 VM 为主的“面子”和以 jar 包、Shell 为主的“里子”有哪些缺点？

## VM / 实体机

资源占用多
启动时间长
管理成本高
...

## jar 包 / Shell

对插件贡献者要求高
对运行环境有特定要求

2

# 行业生态

CI 的命运同样要考虑到历史的进程

2013

# 行业生态

CI 的命运同样要考虑到历史的进程

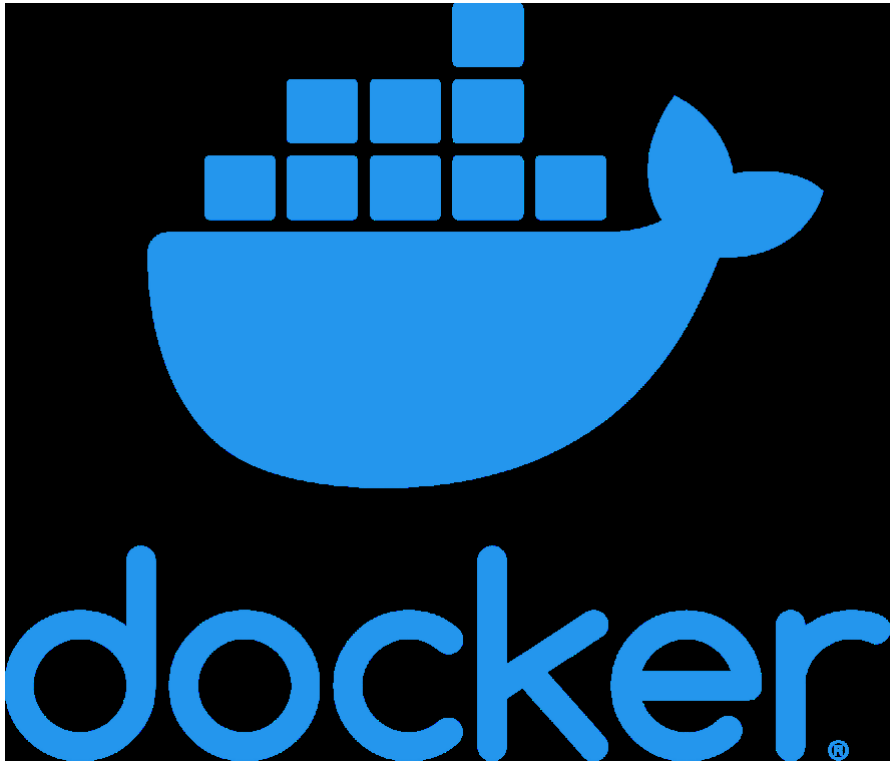
2013

2

# 行业生态

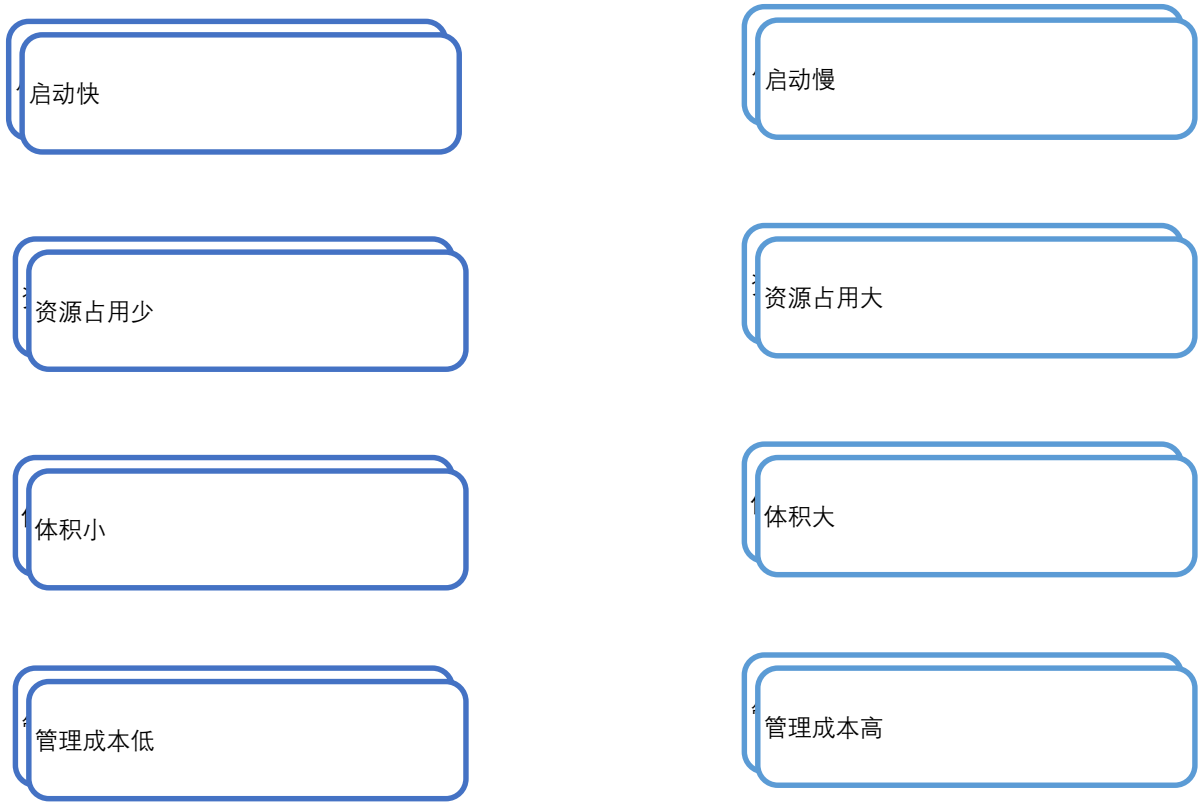
CI 的命运同样要考虑到历史的进程

2013



# 2 行业生态

Docker vs. VM



# 行业生态

有了原子弹，下一步怎么搞？

Jenkins X

2018

2011

9102

Jenkins / Travis CI

2011

2013 Docker
2015 CNCF

2

# 行业生态

Jenkins X

**Jenkins**

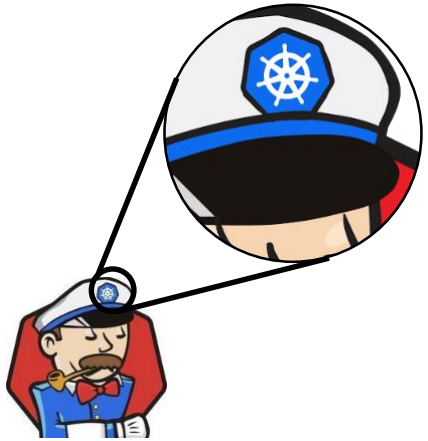
2011

# 2 行业生态

Jenkins X

**Jenkins**

2011



**Jenkins X**

2018

# 2 行业生态

Jenkins X

Jenkins

2011

Jenkins X

2018

“Jenkins X is a CI/CD [continuous delivery] solution for modern cloud applications on Kubernetes …”

# 行业生态

插件的形态

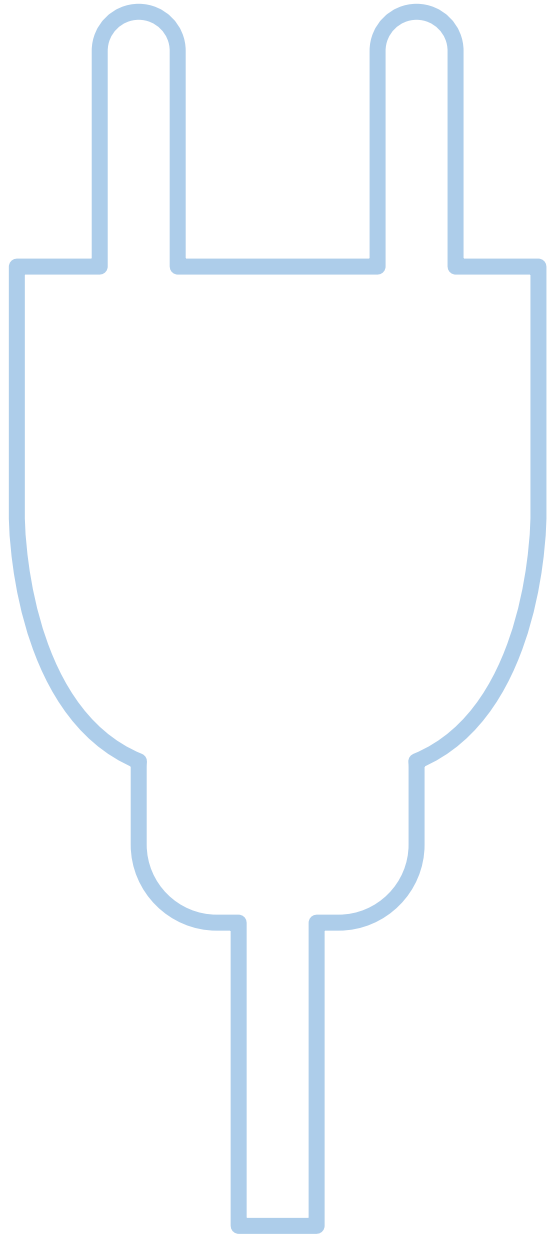
npm 包?

jar 包?

好像都没有本质差别?

一段 Shell?

Pip 包?



2

# 行业生态

Docker as Plugin

```
FROM python:3-slim

WORKDIR /scfcli/

COPY . .

RUN pip install --no-cache-dir scf && \
    scf --version && \
    echo “finish"

ENTRYPOINT [ "/scfcli/entrypoint" ]
```

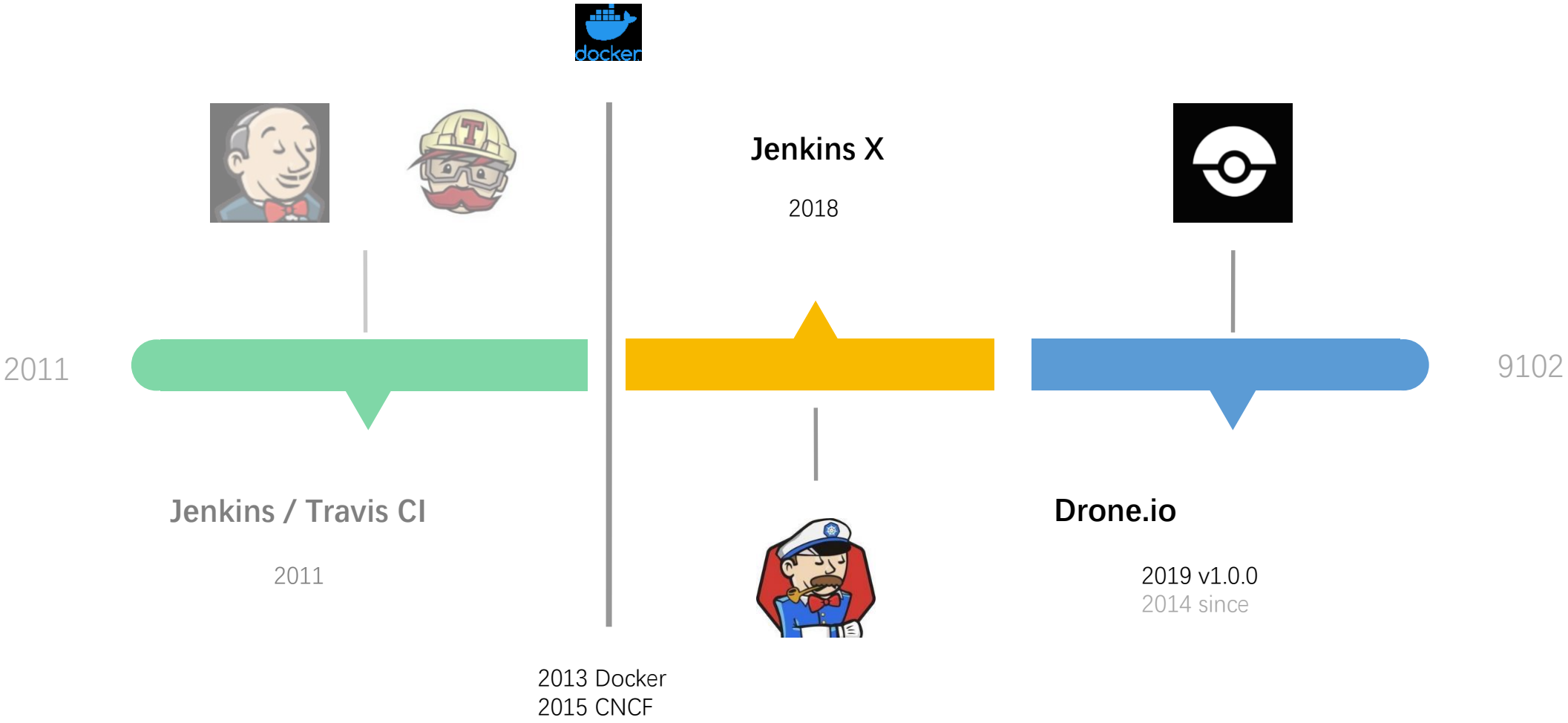
# 行业生态

革命性的开创

- 插件有了跨 CI 使用的可能性
- 插件自带执行环境
- 插件生态融入 Docker 生态

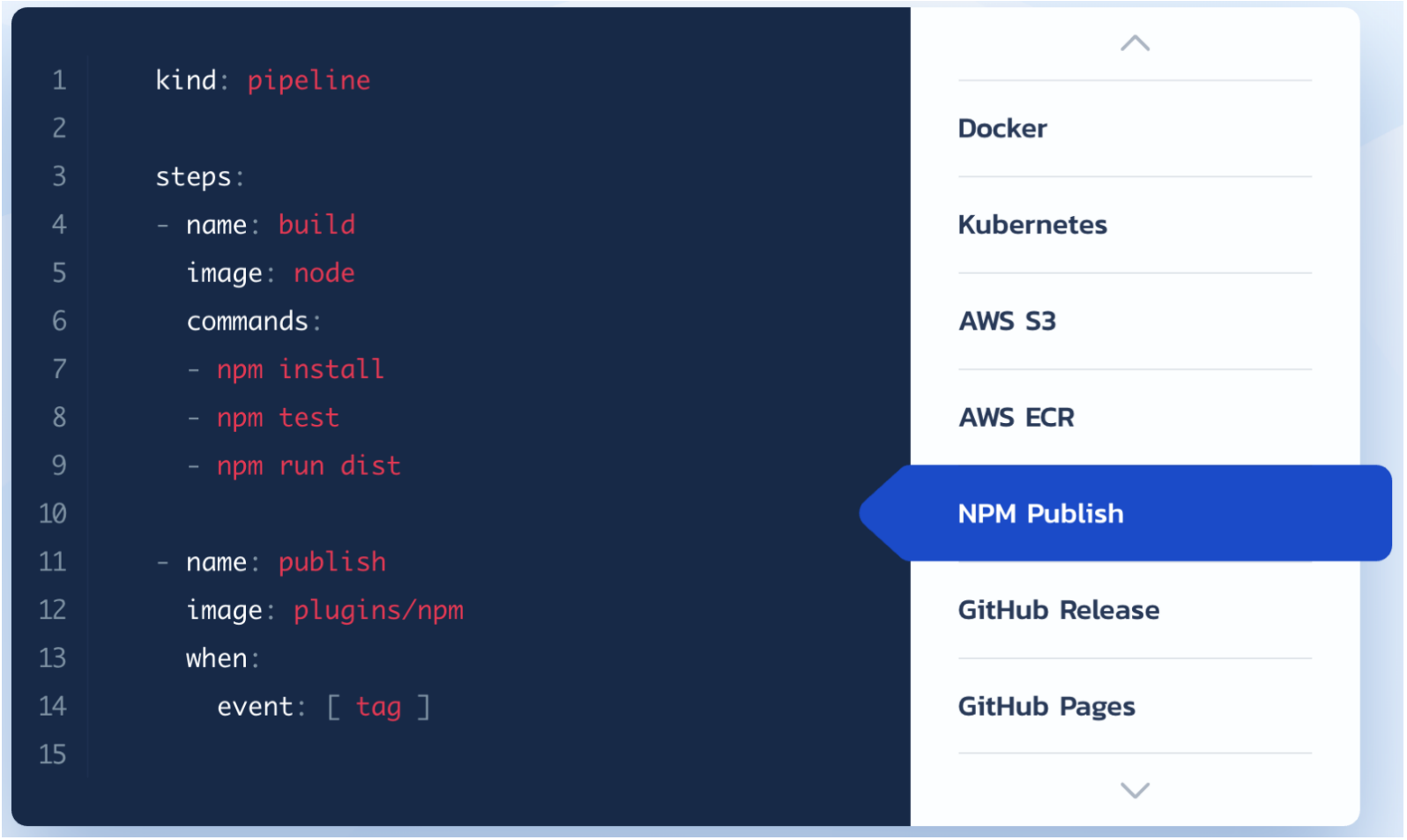
# 行业生态

全新的插件生态



# 行业生态

Drone.io



# 行业生态

新旧更迭

2011

VM Based

Jenkins / Travis CI

2011

Jenkins X

2018

Cloud Native / Docker based

9102

Drone.io

2019 v1.0.0

2014 since

2013 Docker
2015 CNCF

2

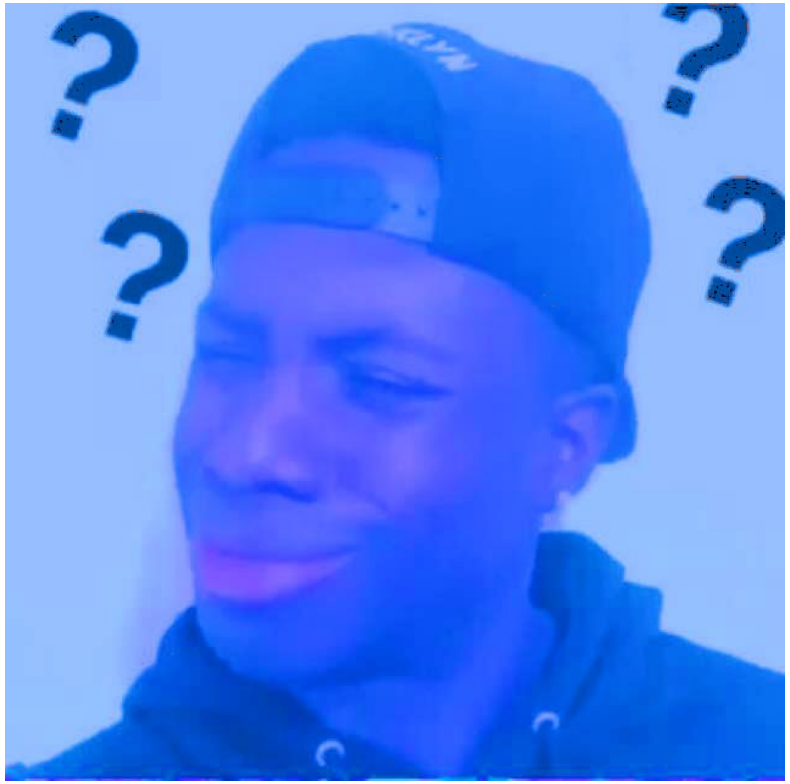
# 行业生态

一切都看起来那么完美，然而…

## 致命缺陷

# 行业生态

致命缺陷



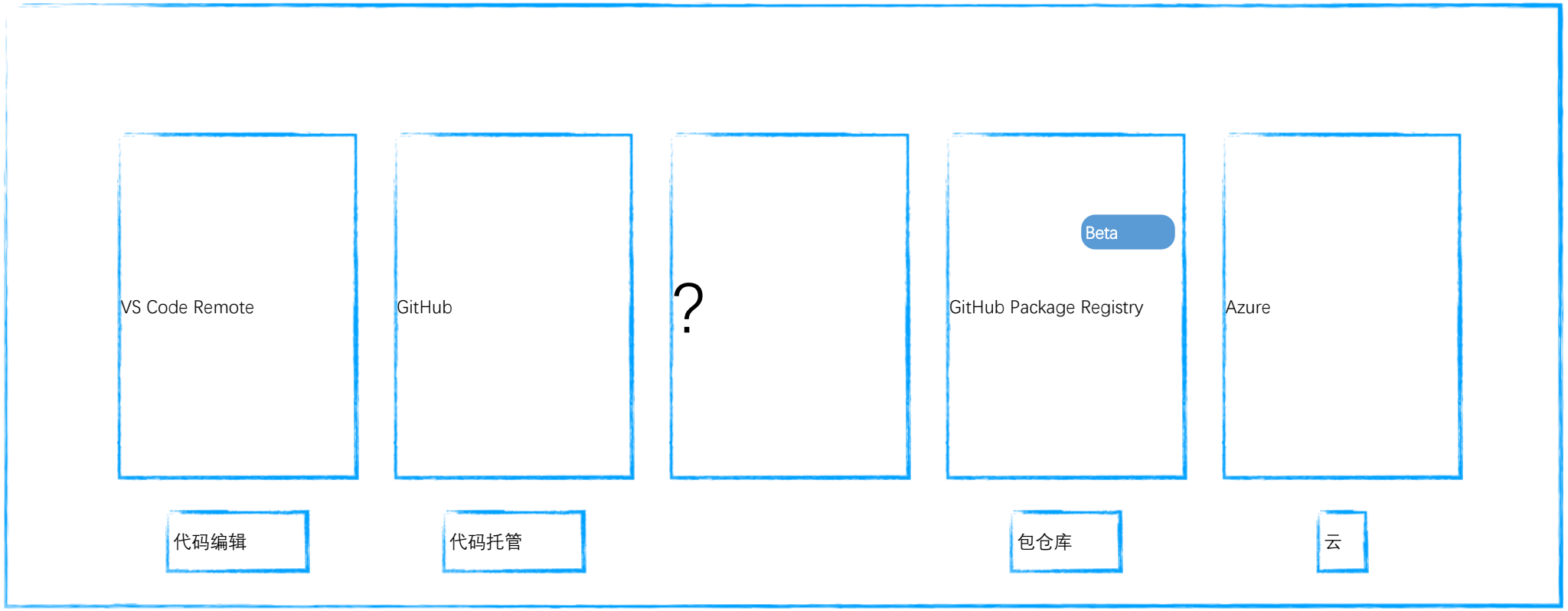
# CI/CD 现代史

之

“搅局者” GitHub Actions

# 搅局者 GitHub Actions

微软的野心



# 搅局者 GitHub Actions

怎么做？GitHub Actions 的艰难抉择

GitHub Universe 2018

# 搅局者 GitHub Actions

最开始，其实我也想全面上 Docker

GitHub Universe 2018

“An Action is a **Docker container** that runs as part of a Workflow.”

# 搅局者 GitHub Actions

后来，我...我都要

## Types of actions

You can build Docker container and JavaScript actions. Actions require a metadata file to define the inputs, outputs and main entrypoint for your action. The metadata filename must be `action.yml`.

| Type | Virtual environment |
| --- | --- |
| Docker container | Linux |
| JavaScript | Linux, MacOS, Windows |

GitHub Actions 2019 Beta Documents

# 搅局者 GitHub Actions

实现原理

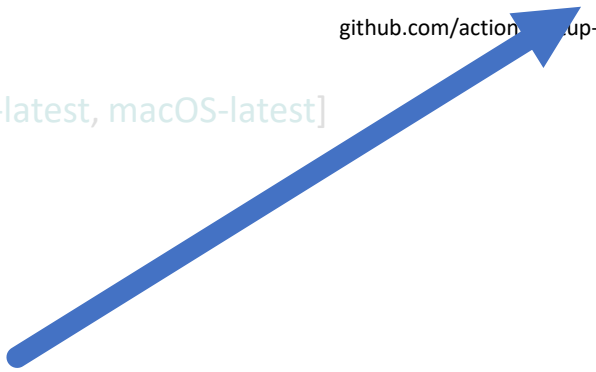
```yaml
jobs:
  test:
    name: GitHub Action Official Demo
    runs-on: ${{ matrix.os }}
    strategy:
      matrix:
        node_version: ['8', '10', '12']
        os: [ubuntu-latest, windows-latest, macOS-latest]

    steps:
    - uses: actions/checkout@v1
    - name: Use Node.js ${{ matrix.node_version }}
      uses: actions/setup-node@v1
      with:
        node-version: ${{ matrix.node_version }}

    - name: npm install, build and test
      run: |
        npm install
        npm run build --if-present
```

# 揽局者

# GitHub Actions

实现原理

```yaml
jobs:
  test:
    name: GitHub Action Official Demo
    runs-on: ${{ matrix.os }}
    strategy:
      matrix:
        node_version: ['8', '10', '12']
        os: [ubuntu-latest, windows-latest, macOS-latest]

    steps:
    - uses: actions/checkout@v1
    - name: Use Node.js ${{ matrix.node_version }}
      uses: actions/setup-node@v1
      with:
        node-version: ${{ matrix.node_version }}

    - name: npm install, build and test
      run: |
        npm install
        npm run build --if-present
```




```yaml
- name: Use Docker Hub
  uses: docker://alpine:3.8
```

# 搅局者 GitHub Actions

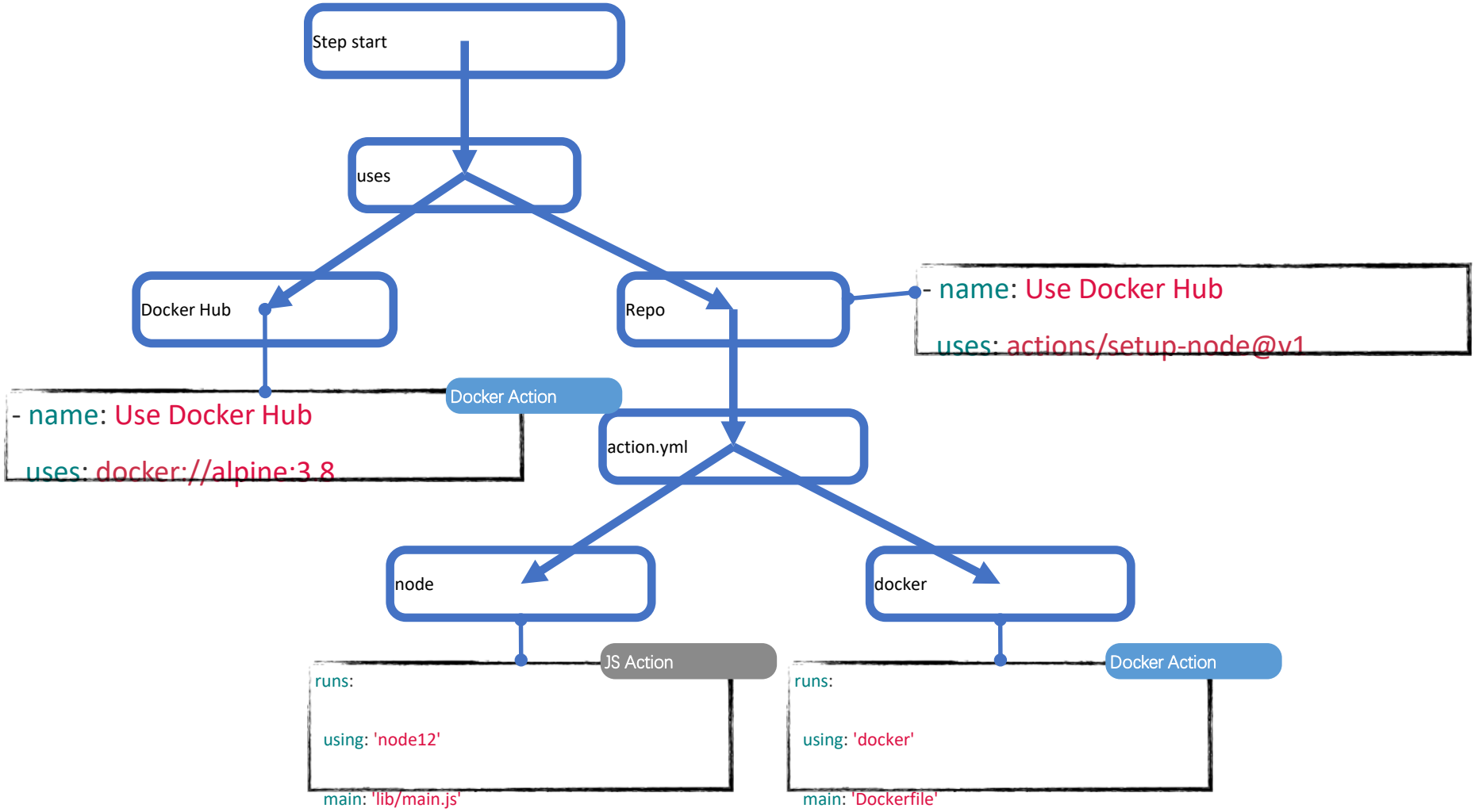
| | |
|---|---|
| .github | Initial Import |
| __tests__ | Add setting for always-auth (#48) |
| docs | Initial Import |
| externals | Initial Import |
| lib | Add setting for always-auth (#48) |
| node_modules | Consume toolkit from npmjs (#36) |
| src | Add setting for always-auth (#48) |
| .gitignore | Update .gitignore (#40) |
| .prettierrc.json | Initial Import |
| LICENSE | Initial Import |
| README.md | Update README.md |
| action.yml | Deprecate version property (#54) |
| jest.config.js | Initial Import |
| package-lock.json | Bump lodash from 4.17.11 to 4.17.15 (#45) |
| package.json | Consume toolkit from npmjs (#36) |
| tsconfig.json | Add auth support (#21) |

```
name: 'Setup Node.js environment'
description: 'Setup a Node.js environment'

author: 'GitHub'

inputs:
 node-version:
  description: 'Version to use.'
  default: '10.x'

runs:
```

```
runs:
 using: 'node12'

 using: 'docker'
 main: 'lib/setup-node.js'
  main: 'Dockerfile'
```

# 搅局者 GitHub Actions

GitHub Actions Overview



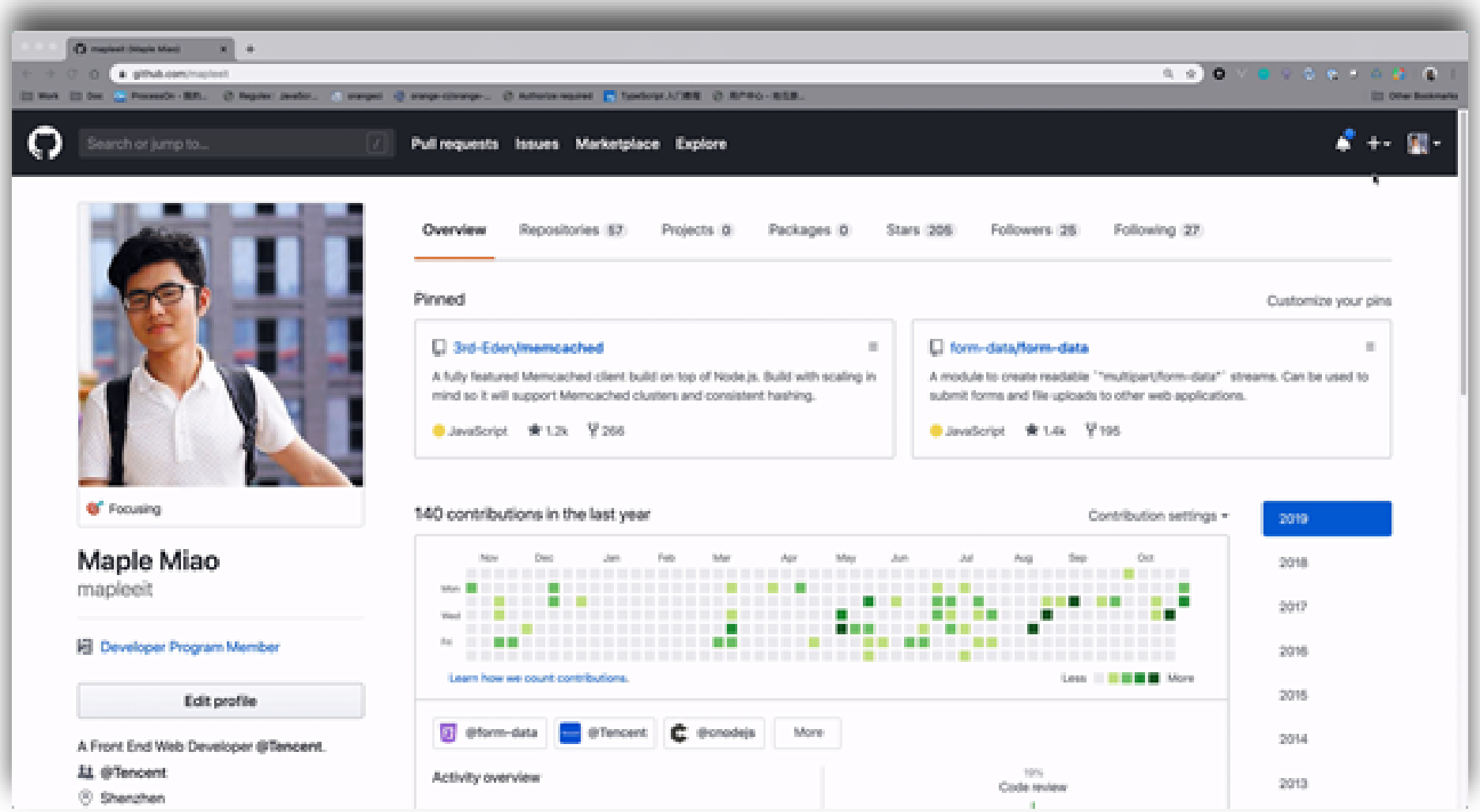
# Talk is Cheap

如何使用 GitHub Actions?

如何使用 JavaScript 开发一个 GitHub Action?

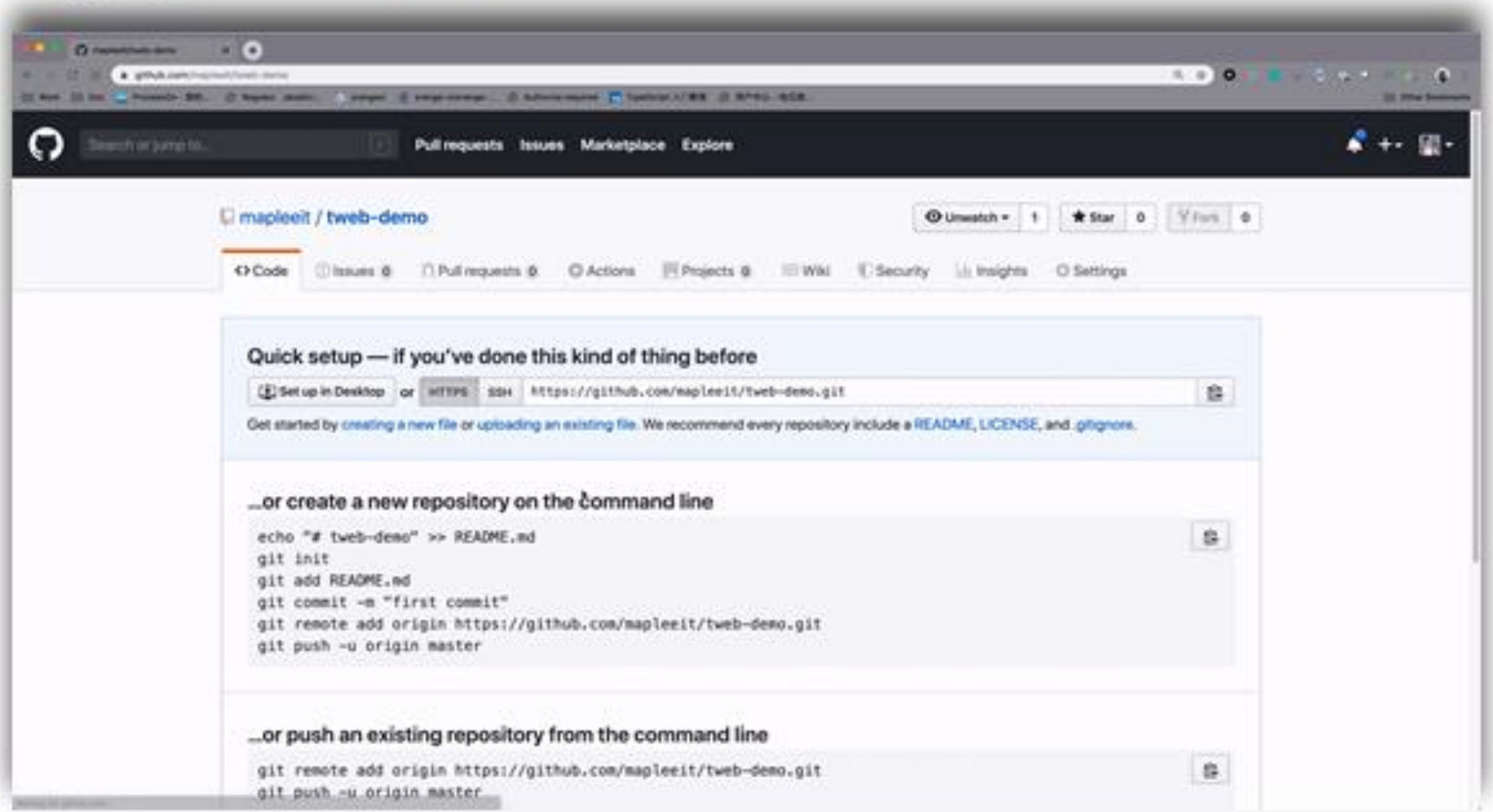
# Talk is Cheap

1. 在 GitHub 上新建 Demo 项目



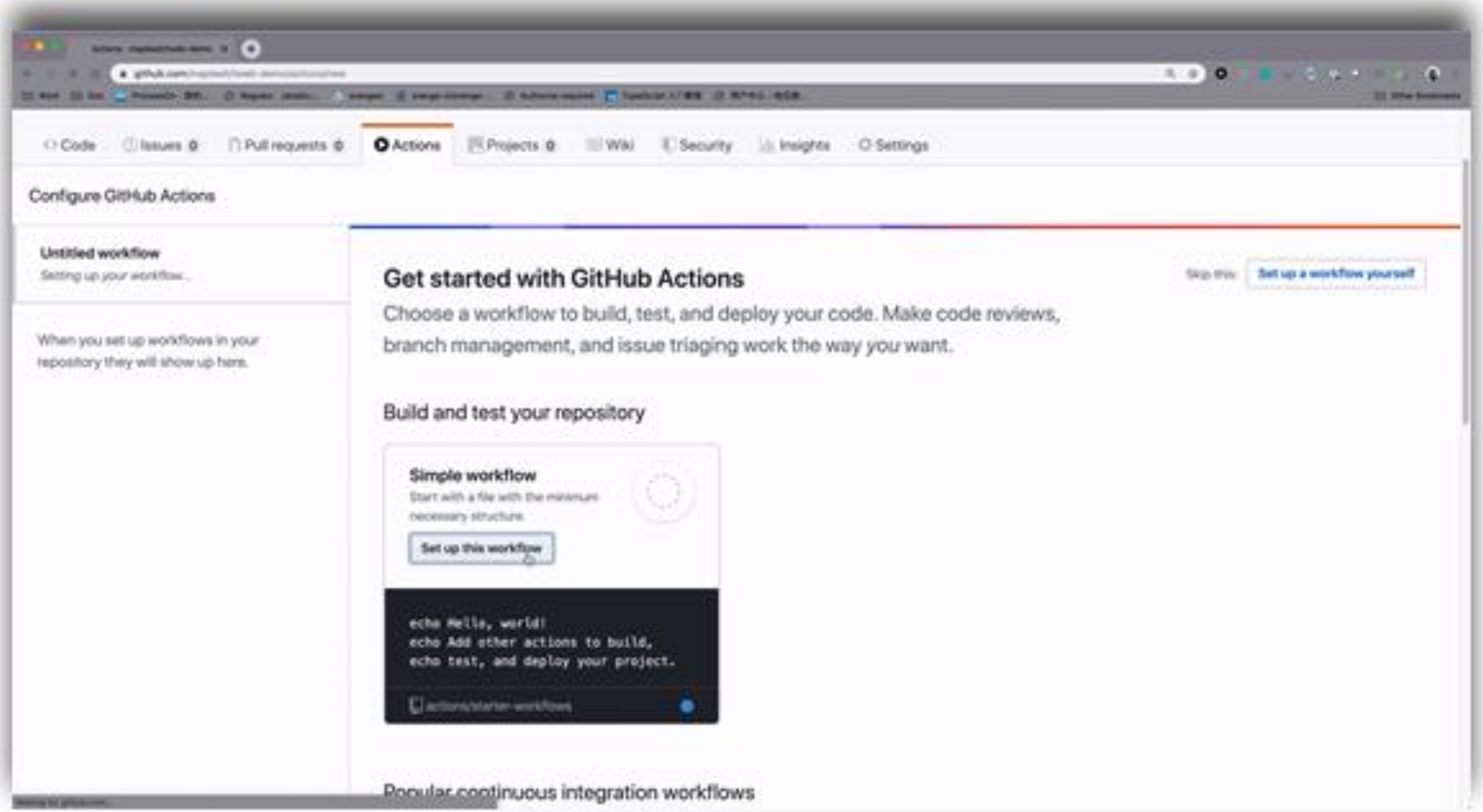
# Talk is Cheap

2. 在该项目上新建一个 GitHub Actions Workflow



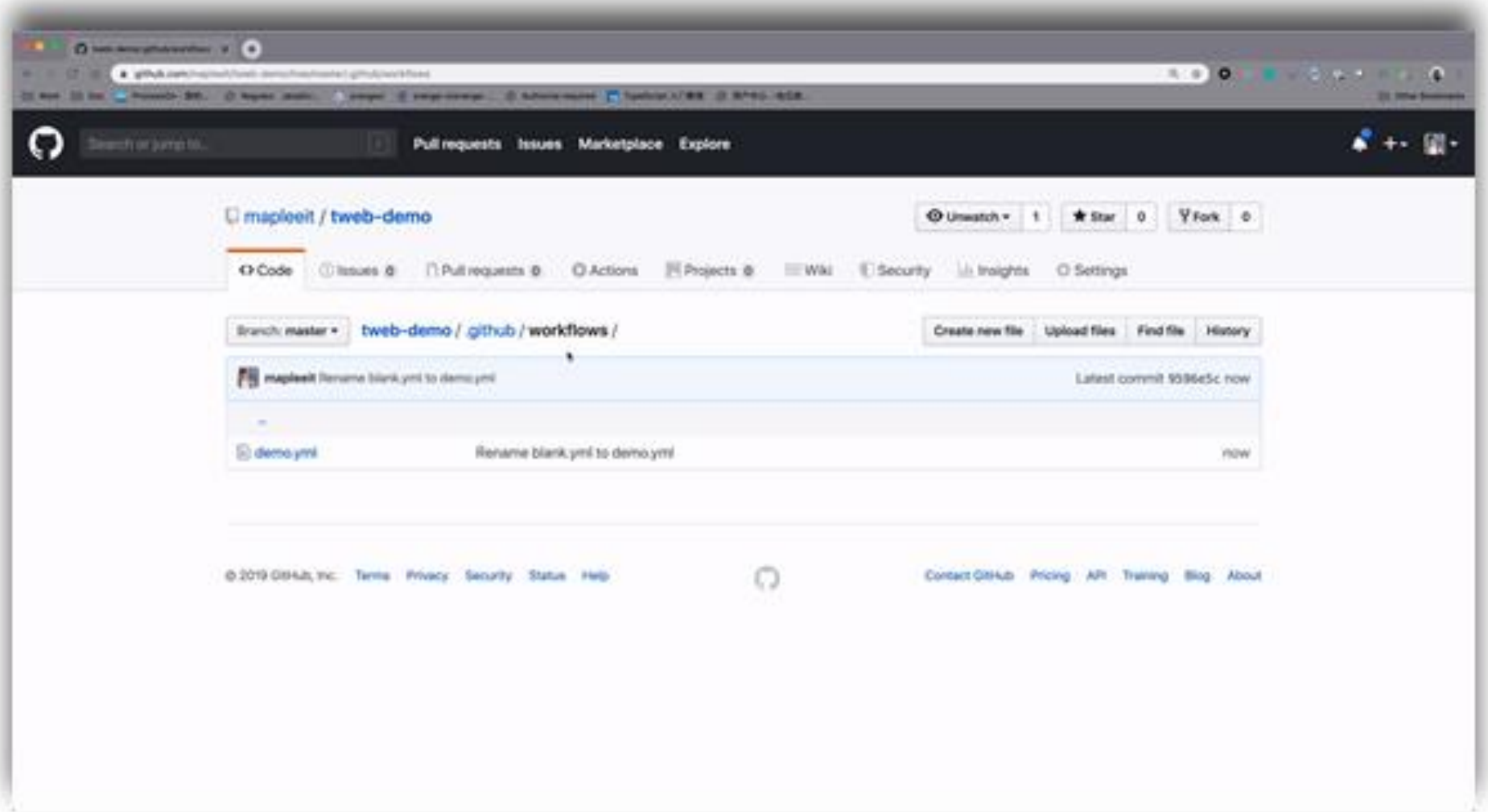
# Talk is Cheap

3. 编辑该 GitHub Actions Workflow



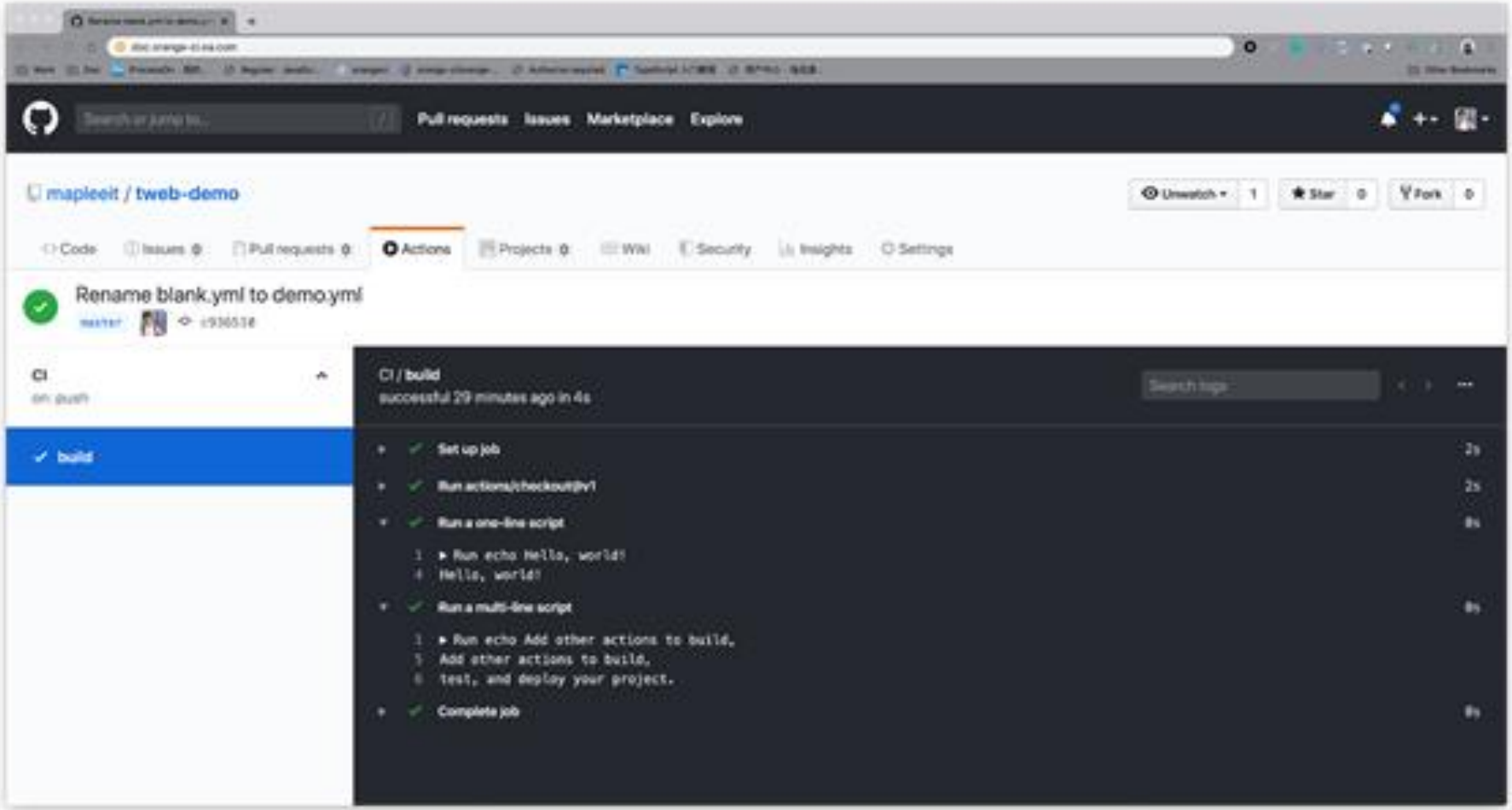
# 4 Talk is Cheap

4. 查看效果



# Talk is Cheap

5. 等待一段时间后…

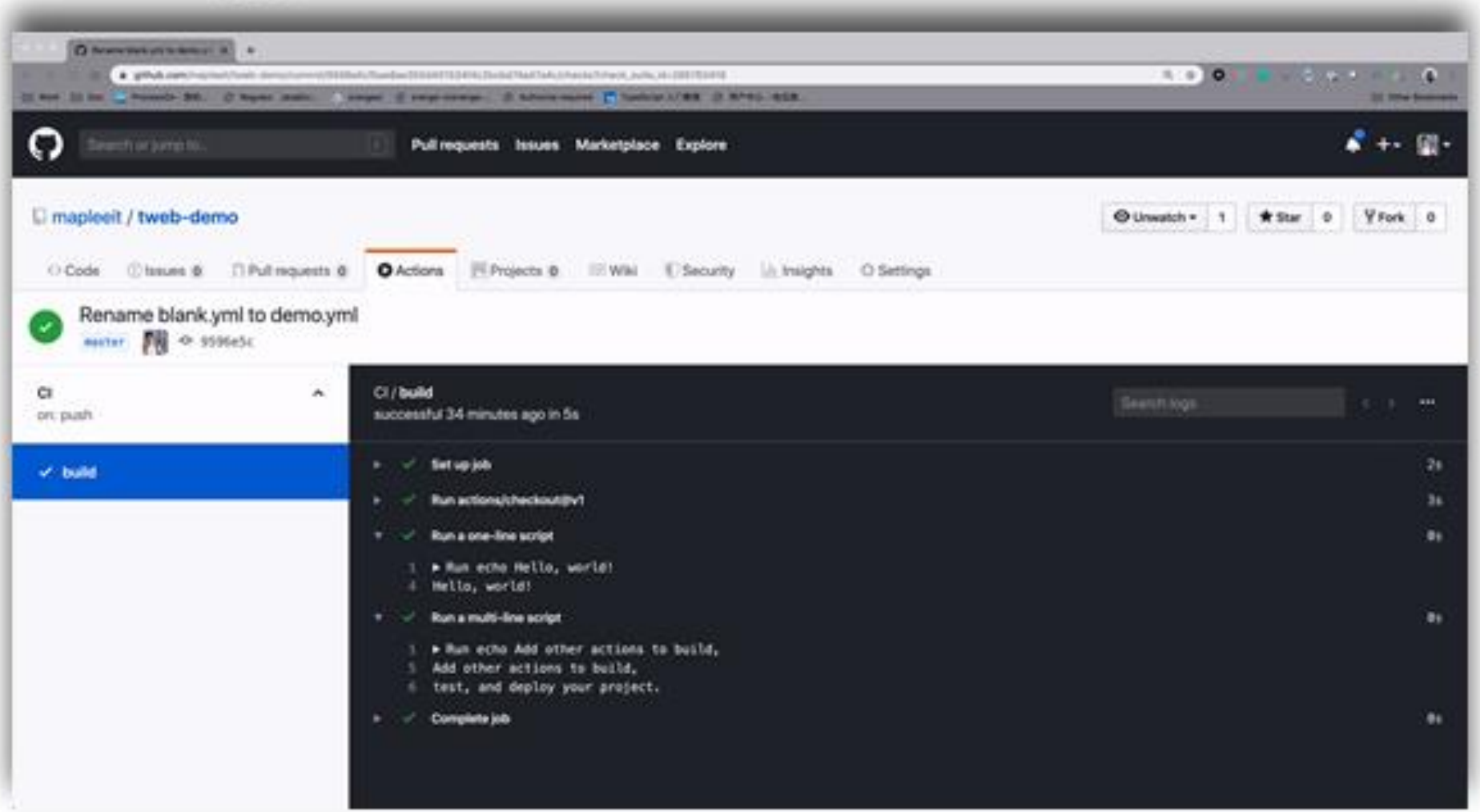


# Talk is Cheap

写一个自己的 JavaScript Action 吧

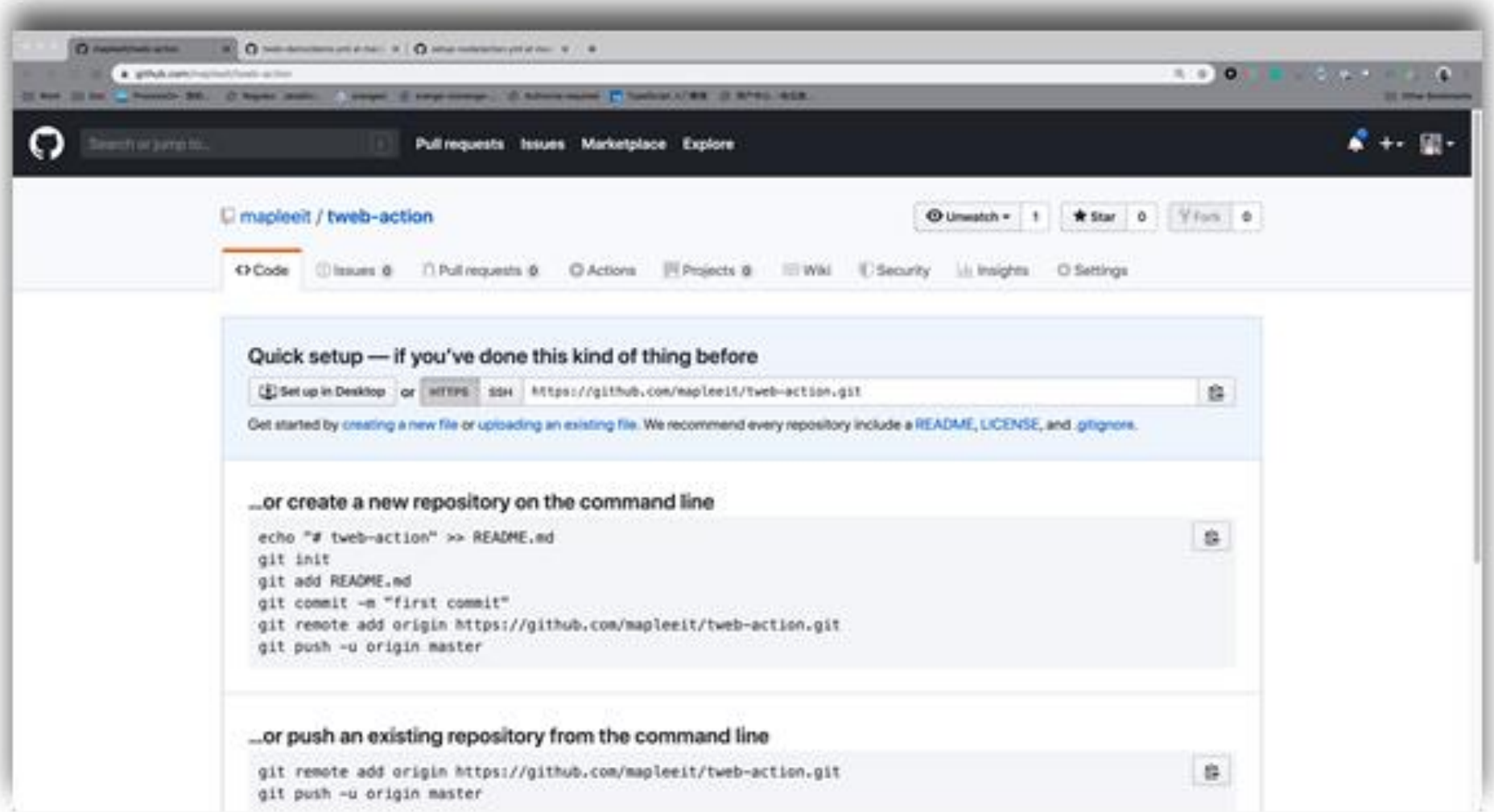
# 4 Talk is Cheap

1. 创建一个 tweb-action 的仓库



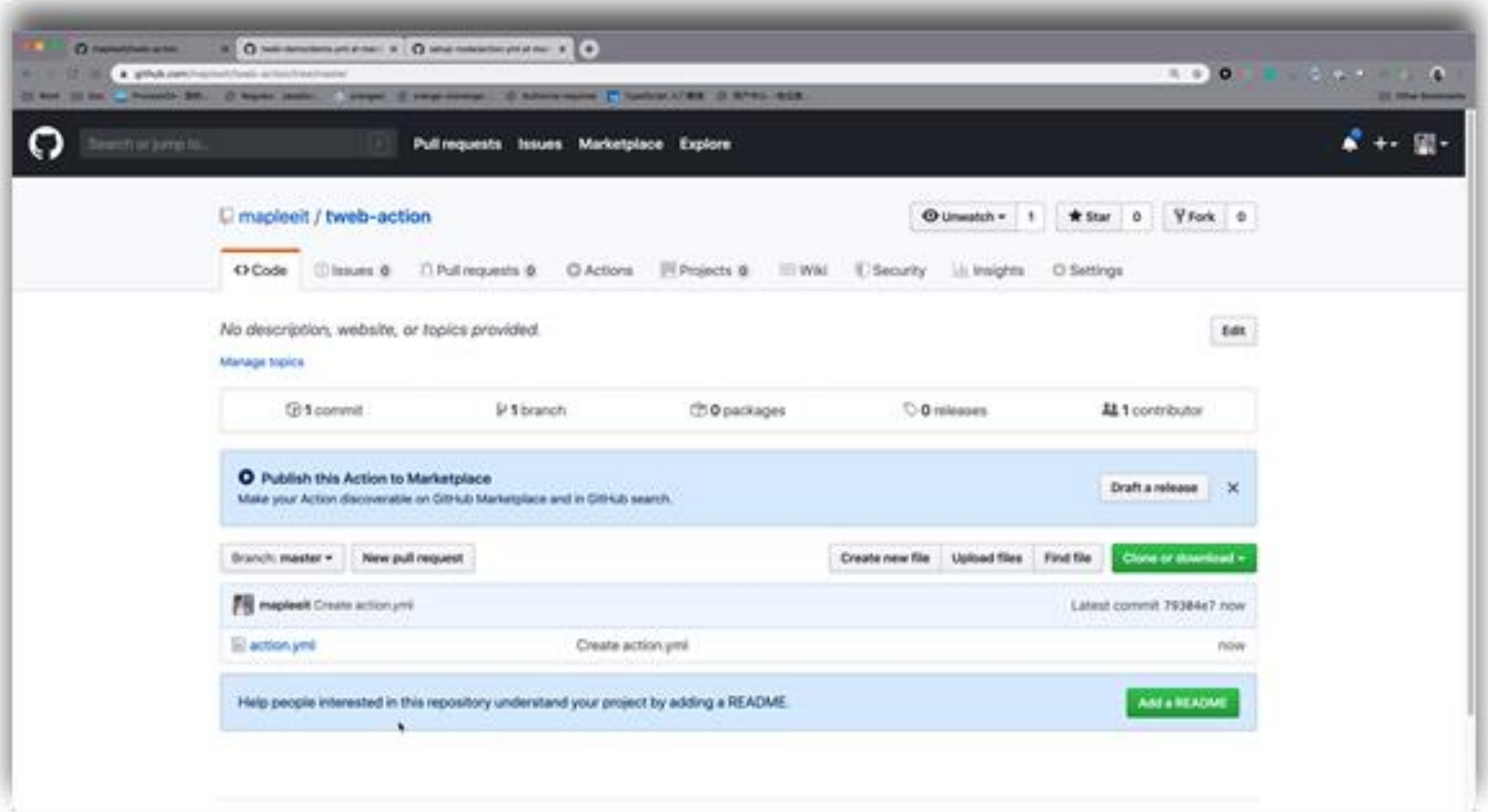
# 4 Talk is Cheap

2. 写 action.yml



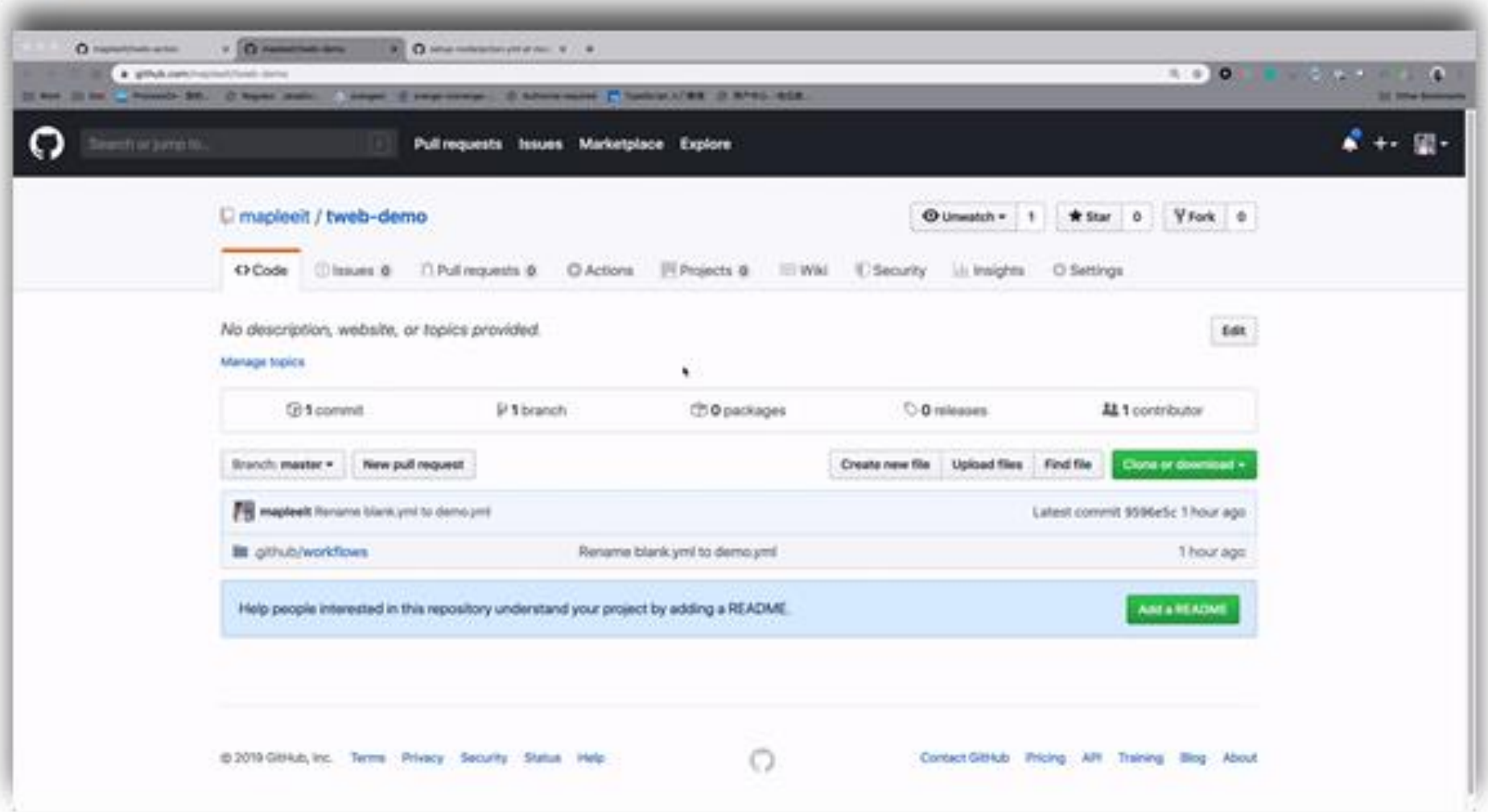
# Talk is Cheap

3. index.js



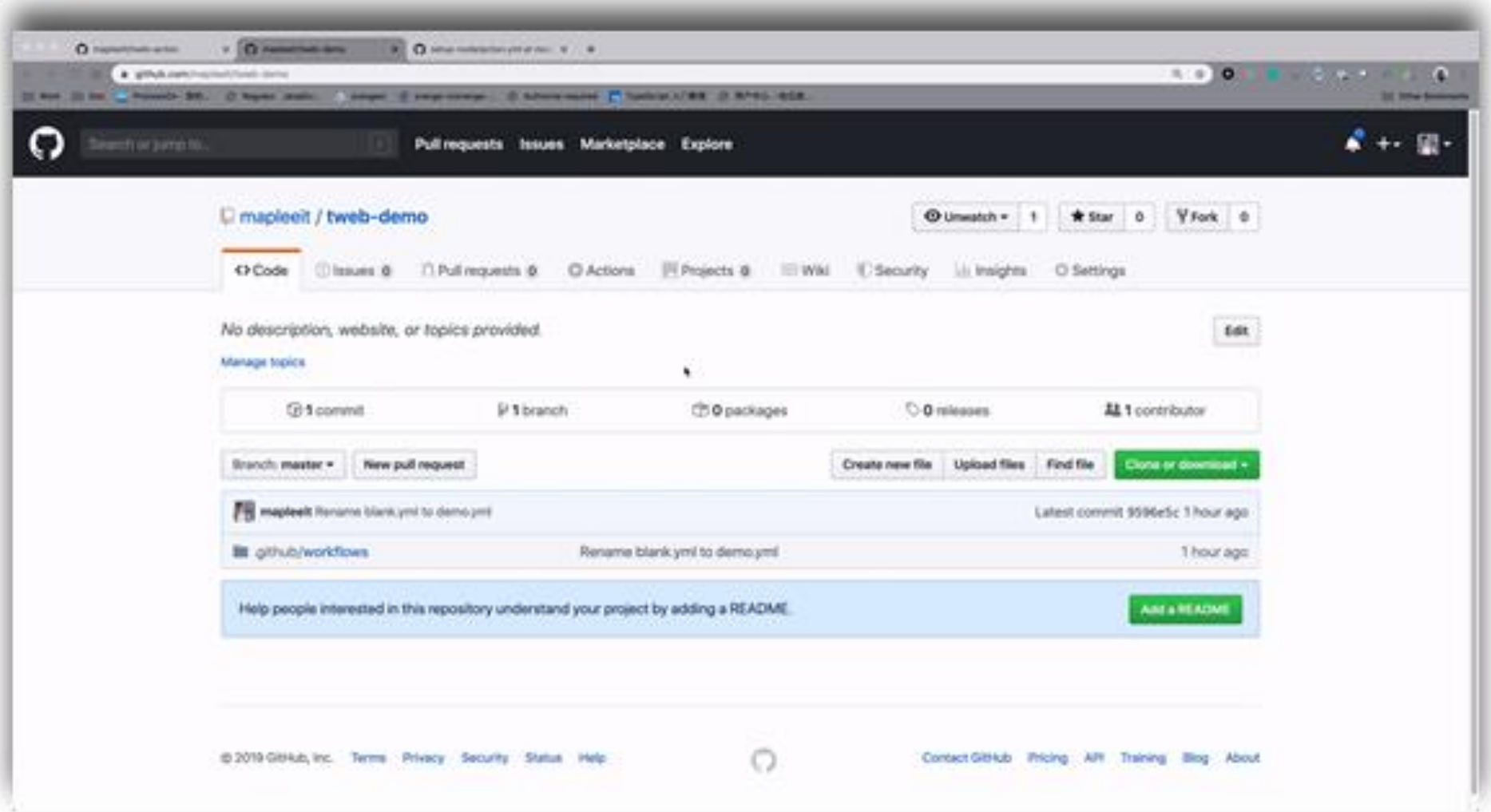
# 4 Talk is Cheap

4. 使用 my first action



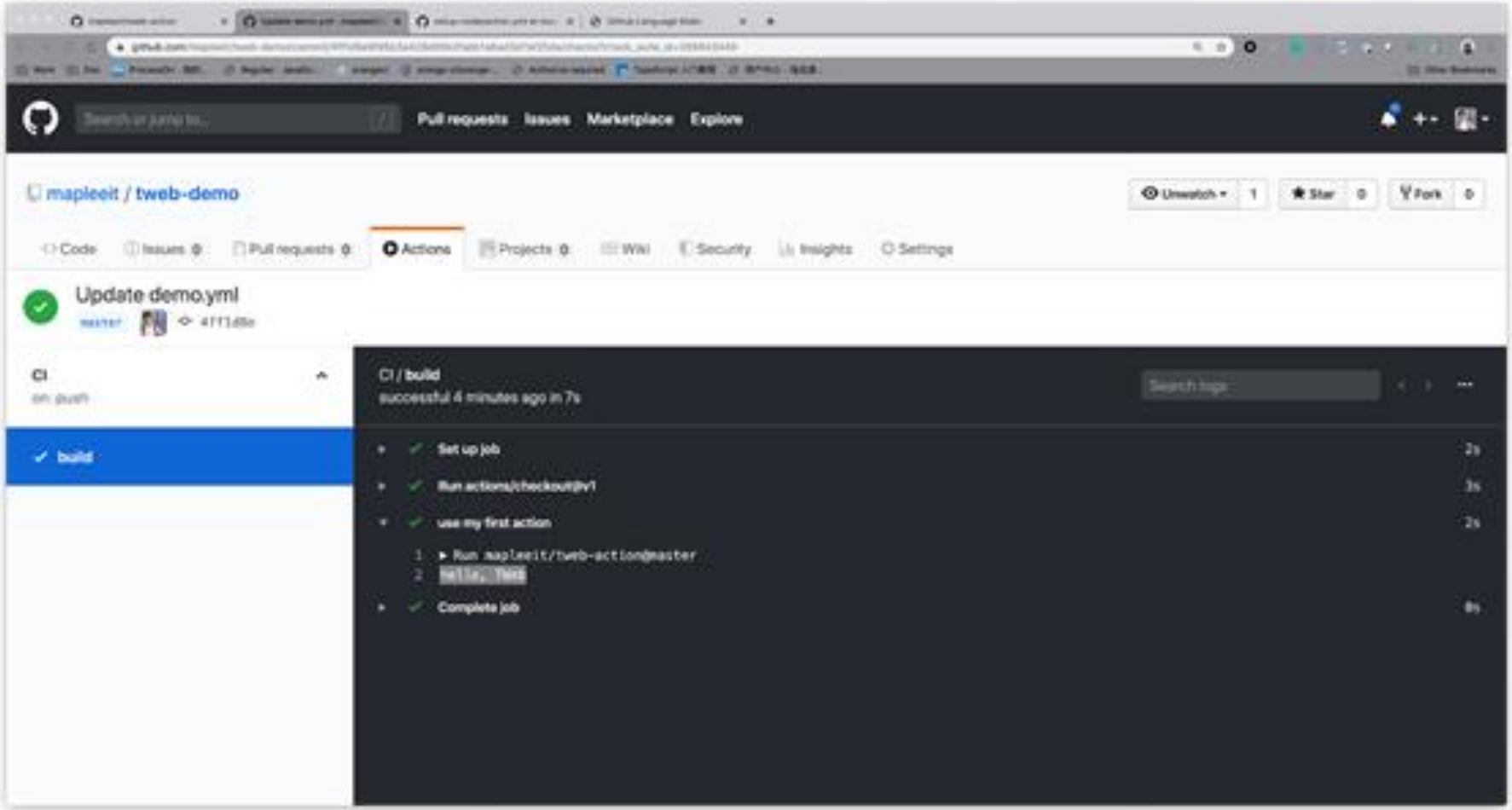
# 4 Talk is Cheap

4. 使用 my first action



# 4 Talk is Cheap

5. 大功告成!

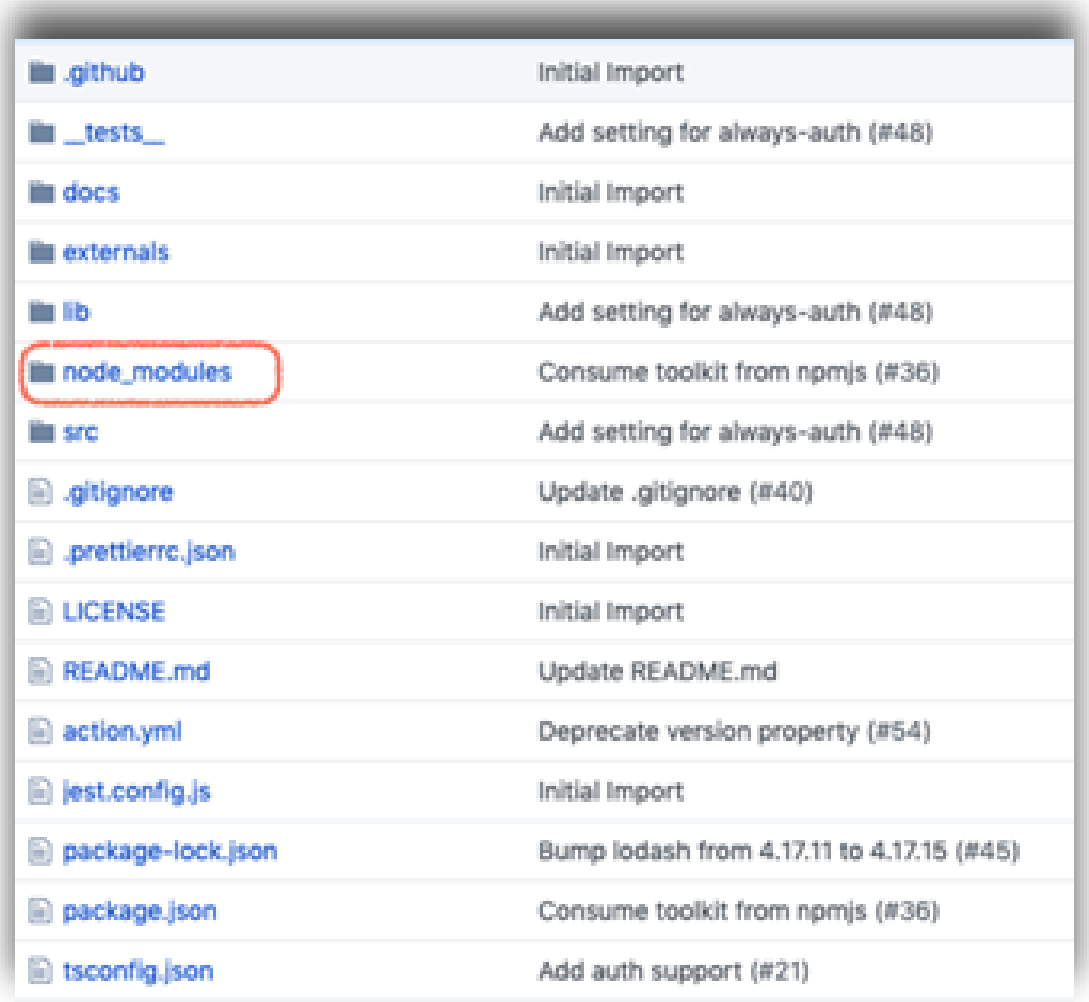


# 拥抱 GitHub Actions

# 拥抱 GitHub Actions

GitHub Actions 目前可能存在的问题

- node_modules 需要提交代码库



| | |
|---|---|
| .github | Initial import |
| __tests__ | Add setting for always-auth (#48) |
| docs | Initial import |
| externals | Initial import |
| lib | Add setting for always-auth (#48) |
| node_modules | Consume toolkit from npmjs (#36) |
| src | Add setting for always-auth (#48) |
| .gitignore | Update .gitignore (#40) |
| .prettierrc.json | Initial import |
| LICENSE | Initial import |
| README.md | Update README.md |
| action.yml | Deprecate version property (#54) |
| jest.config.js | Initial import |
| package-lock.json | Bump lodash from 4.17.11 to 4.17.15 (#45) |
| package.json | Consume toolkit from npmjs (#36) |
| tsconfig.json | Add auth support (#21) |

# 拥抱 GitHub Actions

GitHub Actions 的新想法

- GitHub Repo as Plugin，更广泛的去中心化
- 拥抱 JS 生态

# Q & A

mapleat@github
maplemiao@tencent.com

TWEB CONF
<腾讯前端技术大会/>

# <Thanks/>

主办方 Organizer Tencent TWeb