

CI/CD

📄 CI/CD📄 git.nju.edu.cn/xxxxxxxxxx

- [CI/CD 📄](#)
- [📄 Docker📄](#)

CI/CD 入门

简介

环境	工具	runner tag
本地 / 云服务器 / 虚拟机 / 容器 / 云原生	本地 job	docker runner
容器化环境	kaniko	docker runner
docker run / compose 运行 Docker in Docker	DinD	dind runner

tags 配置

- 本地 runner 配置 tags: [docker] job 名称 kaniko 运行
- 本地 DinD 配置 tags: [dind] 运行 tag 名称 runner job 名称 Docker
- 容器化 runner 配置 tags: [myrunner] 运行
- 容器化 runner 配置 "Run untagged jobs" 运行 tags: job

本地 job

```
“ 本地 runner 配置
tags: [docker]
job 名称 kaniko
运行

本地 DinD 配置
tags: [dind]
运行 tag 名称 runner job 名称 Docker

容器化 runner 配置
tags: [myrunner]
运行

容器化 runner 配置
tags: job
运行”
```

- 本地 runner 配置 Docker daemon
- 本地 docker build 运行 kaniko / DinD
- 容器化 runner 配置 Docker Hub / gcr.io / ghcr.io / quay.io / nvcr.io / registry.gitlab.com / registry.k8s.io

本地 / 容器化

```
build:
  stage: build
  image: node:lts
  script:
    - npm ci
    - npm run build
  artifacts:
    paths: [dist/]
```



```
test:
  stage: test
  image: node:lts
  needs: [build]
  script:
    - npm ci
    - npm test -- --reporter=junit --outputFile=report.xml
  artifacts:
    when: always
    reports:
      junit: report.xml
```



```
lint:
  stage: test
  image: node:lts
  script:
    - npm ci
    - npx eslint .
```



```
package:
  stage: package
```

```
image: alpine:latest
needs: [build]
script:
  - tar -czf myapp.tar.gz -C dist .
artifacts:
  paths: [myapp.tar.gz]
```

```
“ package stage stages: [build, test,
package, deploy] GitLab .pre / build / test / deploy / .post
```

deployment job

```
rsync scp sftp WebDAV kubectl .....
script
```

```
“ Token GitLab CI/CD
.gitlab-ci.yml
```

```
deploy:
  stage: deploy
  image: alpine:latest
  variables:
    GIT_STRATEGY: none
  environment:
    name: production
    url: https://myapp.example.com
  resource_group: production
  rules:
    - if: $CI_COMMIT_BRANCH == "main"
      when: manual
  before_script:
    - apk add --no-cache openssh-client rsync
    - mkdir -p ~/.ssh && chmod 700 ~/.ssh
    - install -m 600 "$SSH_PRIVATE_KEY" ~/.ssh/id_ed25519
    - install -m 644 "$SSH_KNOWN_HOSTS" ~/.ssh/known_hosts
  script:
```

```
- rsync -az --delete -e "ssh -i ~/.ssh/id_ed25519" dist/ deploy@my-server:/srv/myapp/
- ssh -i ~/.ssh/id_ed25519 deploy@my-server 'cd /srv/myapp && docker compose up -d'
```

Settings → CI/CD → Variables

SSH_PRIVATE_KEY	File	File \$SSH_PRIVATE_KEY install -m 600 "\$SSH_PRIVATE_KEY" ~/.ssh/id_ed25519
SSH_KNOWN_HOSTS	Variable File	ssh-keyscan my-server StrictHostKeyChecking=no
/Token	Variable	Masked Protected job

- job 600 job
-
- protected branch + when: manual + environment + resource_group
- deploy token / deploy key PAT

kaniko

job registry Docker daemon

- docker run / compose RUN --mount=type=secret
- DinD runner
- cache-repo registry
- tag debug tag latest scratch
- GoogleContainerTools/kaniko 2025-06 v1.24.0 martizih/kaniko osscontainertools/kaniko

```
build-image:
  stage: build
  image:
    name: martizih/kaniko:debug # tag shell
ghcr.io/osscontainertools/kaniko:debug
```

```

    entrypoint: [""]
script:
  - mkdir -p /kaniko/.docker
  - |
    cat > /kaniko/.docker/config.json <<CONF

{"auths":{"${CI_REGISTRY}":{"username":"${CI_REGISTRY_USER}","password":"${CI_REGISTRY_PASSWORD}"}}}

CONF
- /kaniko/executor
  --context "${CI_PROJECT_DIR}"
  --dockerfile "${CI_PROJECT_DIR}/Dockerfile"
  --destination "${CI_REGISTRY_IMAGE}:${CI_COMMIT_SHORT_SHA}"
  --destination "${CI_REGISTRY_IMAGE}:latest"
  --snapshot-mode=redo
  --compressed-caching=false
  --cache=true
  --cache-repo="${CI_REGISTRY_IMAGE}/cache"

```

DinD

job Docker build / run / compose

- privileged runner
- docker build / run / compose user-namespace binfmt_misc
- multi-arch / QEMU
- daemon job job registry /artifacts
- docker

```

build-and-test:
  stage: test
  tags: [dind] # ← Docker daemon runner
  image: docker:29-cli
  services:
    - name: docker:dind
      alias: docker
  variables:
    DOCKER_HOST: tcp://docker:2375

```

```
DOCKER_TLS_CERTDIR: "" # 2375 job
script:
  - docker build -t myapp:ci .
  - docker run --rm myapp:ci ./run-tests.sh
  - docker compose up -d --build && docker compose ps
```

runner

runner

GitLab runner

- Settings → CI/CD → Runners → New project runner
- Tags myrunner — job runner "Run untagged jobs" runner
- runner authentication token glrt-...

runner

Docker deb/rpm Kubernetes GitLab

- <https://docs.gitlab.com/runner/install/>
- <https://docs.gitlab.com/runner/register/>

--url https://git.nju.edu.cn/ --token glrt-... --executor

- tags: [myrunner] job runner
- GitLab → Runners online job



- `image: reg.nju.edu.cn/<namespace>/<project>`
`image: reg.nju.edu.cn/<namespace>/<project>`
- `registry: registry` `Masked + Protected` `CI`
- `.gitlab-ci.yml` `--build-arg SECRET=...` `dind` `RUN --`
`mount=type=secret` `kaniko`
- Docker Hub `Dependency Proxy`



- `tags: [dind]` `→ job` `Docker daemon` `runner`
- `services: [docker:dind]` `→` `tcp://docker:2375`
- `martizih/kaniko:latest` `→` `scratch` `shell` `CI`
`martizih/kaniko:debug`
- `.gitlab-ci.yml` `--build-arg SECRET=...` `→`
`CI/CD` `DinD` `RUN --`
`mount=type=secret` `kaniko`



Docker

“ ” CI/CD Docker HPC

<https://git.nju.edu.cn/escience/singularity-example>

- root
-
-

Docker Singularity

eScience

Conda Python Numpy

Dockerfile

Dockerfile Docker

```
FROM continuumio/miniconda3:22.11.1

# conda
COPY .condarc /root/.condarc

# 
RUN conda create -n my-env python=3.10 numpy

# 
SHELL ["/bin/bash", "--login", "-c"]
RUN conda init bash
RUN echo "source activate my-env" > ~/.bashrc
ENV PATH /opt/conda/envs/my-env/bin:$PATH
```

numpy

□□□□ Docker□□□ docker□□□□□

```
docker build -t escience/conda-numpy .
```

“ Docker [] [] [] [] Docker [] docker.nju.edu.cn []

```
docker run -i escience/conda-numpy python < test.py
```

CI/CD

CI/CD [] git.nju.edu.cn [] .gitlab-ci.yml [] CI/CD

[]

CI/CD gcr.nju.edu.cn

```
# []
stages:
  - build
  - test
# []
build:
  stage: build
  image:
    name: martizih/kaniko:debug
    entrypoint: [""]
  script:
    - /kaniko/executor
      --context "${CI_PROJECT_DIR}"
      --dockerfile "${CI_PROJECT_DIR}/Dockerfile"
      --destination "${CI_REGISTRY_IMAGE}:${CI_COMMIT_TAG}"
  rules: # []
    - if: $CI_COMMIT_TAG
# []
```

test



更新.gitlab-ci.yml文件
#73419 test 689731fe 最新

S

singularity-example

项目信息

仓库

议题0

合并请求0

CI/CD

安全与合规

部署

软件包与镜像库

软件包库

容器镜像库

Terraform 模块

模型实验

基础设施

监控

分析

Wiki

代码片段

设置

收起侧边栏

eScience > singularity-example > 容器镜像库 / singularity-example

singularity-example

1个标签清理已禁用创建于 8月 29, 2023 10:20

筛选结果

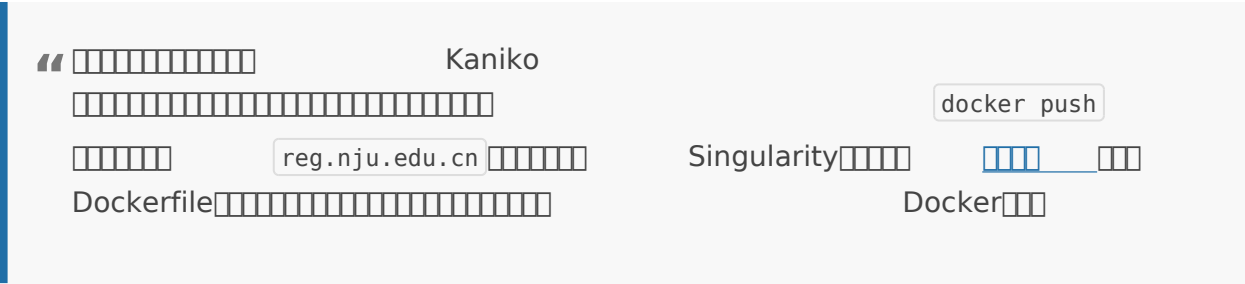
名称

删除所选

1个标签

test858.19 MiB

发布于17分钟前摘要: b4e0ace



hpc.nju.edu.cn

Singularity

```
singularity build conda-numpy.sif docker://reg.nju.edu.cn/escience/singularity-example:test
```

```
INFO: Starting build...
Getting image source signatures
Copying blob 3389e9eb8624 done
Copying blob 3f4ca61aafcd done
Copying blob baee49be4542 done
Copying blob 7b4354700ca4 done
Copying blob 69a5d9e1ecd6 done
Copying blob 8a7a15cee421 done
Copying blob 5f4e24b7e321 done
Copying config 51871566f8 done
Writing manifest to image destination
Storing signatures
2023/08/29 21:46:55 info unpack layer: sha256:3f4ca61aafcd4fc07267a105067db35c0f0ac630e1970f3cd0c7bf552780e985
2023/08/29 21:46:57 info unpack layer: sha256:69a5d9e1ecd6566da53d0978004bdf37dddfaba1d8a6117966f397b41cbbc529
2023/08/29 21:46:59 info unpack layer: sha256:7b4354700ca480732ead22a553cc45916dc5466709ca64d964c4647b5b9343e9
2023/08/29 21:47:02 info unpack layer: sha256:baee49be454261f20f9770566da694b7e7845cf7d279cc2421c6b3eed68c012c
2023/08/29 21:47:02 info unpack layer: sha256:8a7a15cee4219b244df17b401664119dfd5b7e52a5659107f1ee3ec210722373
2023/08/29 21:47:20 info unpack layer: sha256:3389e9eb8624f21330c272ba23defc59ba31d46273e38cde475e7256edea80cb
2023/08/29 21:47:20 info unpack layer: sha256:5f4e24b7e32113995e0417eb5c4de5cbef9be70a4a3841a19cdb2fa1f3f12a34
INFO: Creating SIF file...
INFO: Build complete: conda-numpy.sif
```

conda-numpy.sif

job.lsf

```
#BSUB -q 6140ib
#BSUB -n 1

module load singularity/latest

SINGULARITY="singularity run --env MKL_NUM_THREADS=$LSB_DJOB_NUMPROC conda-numpy.sif"
${SINGULARITY} python test.py
```

bsub < job.lsf

bjobs

bpeek



CI/CD

Docker

HPC

eScience

- /

docker.nju.edu.cn

gcr.nju.edu.cn

mirror.nju.edu.cn
- CI/CD

git.nju.edu.cn

reg.nju.edu.cn
- hpc.nju.edu.cn