

GitLab CI

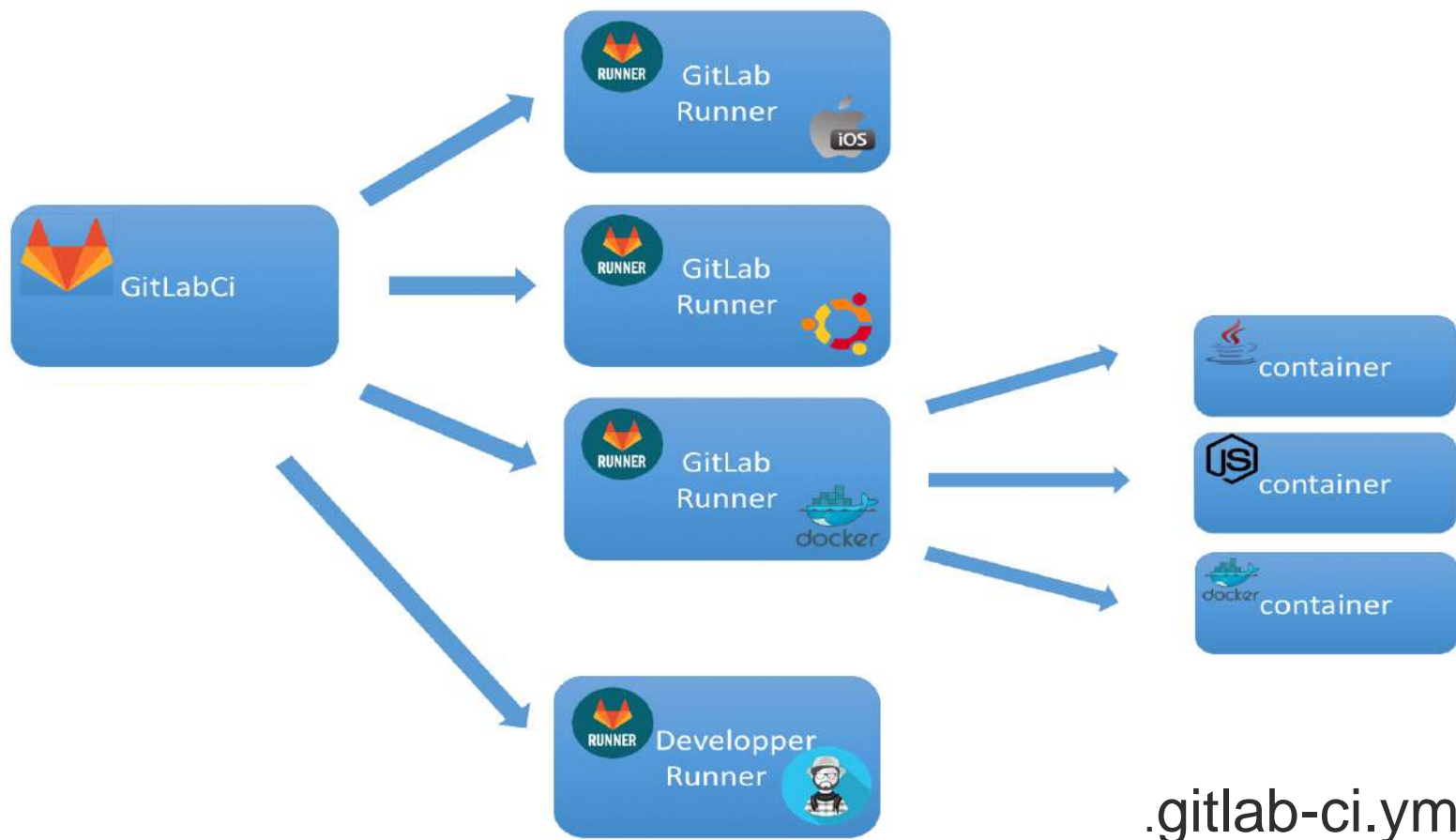
By shaowenchen

使用情况

- 8core CPU/15G MEM/500G DISK
- CPU 使用率在 5% 上下，内存使用率在 80%-90%



简介



背景

- 2017/10 Jenkins & GitLab CI 的一个小分享 (Dcloud)
 - 不久, gitlab-paas 新增一台服务器支持 CI
 - 需要 Git 推送 SVN, 向 PaaS 提需求
 - 慢! 自己提供机器
 - 全面使用 GitLab CI
-

.gitlab-ci.yml

```
before_script:
  - source /data/gitlab-runner/paas-webfe/bin/activate
  - which node && node --version
  - alias tnpm="/data/gitlab-runner/node/bin/tnpm"
  - which tnpm && tnpm --version
  - LANG="zh_CN.utf8"
  - export LC_ALL=zh_CN.UTF-8

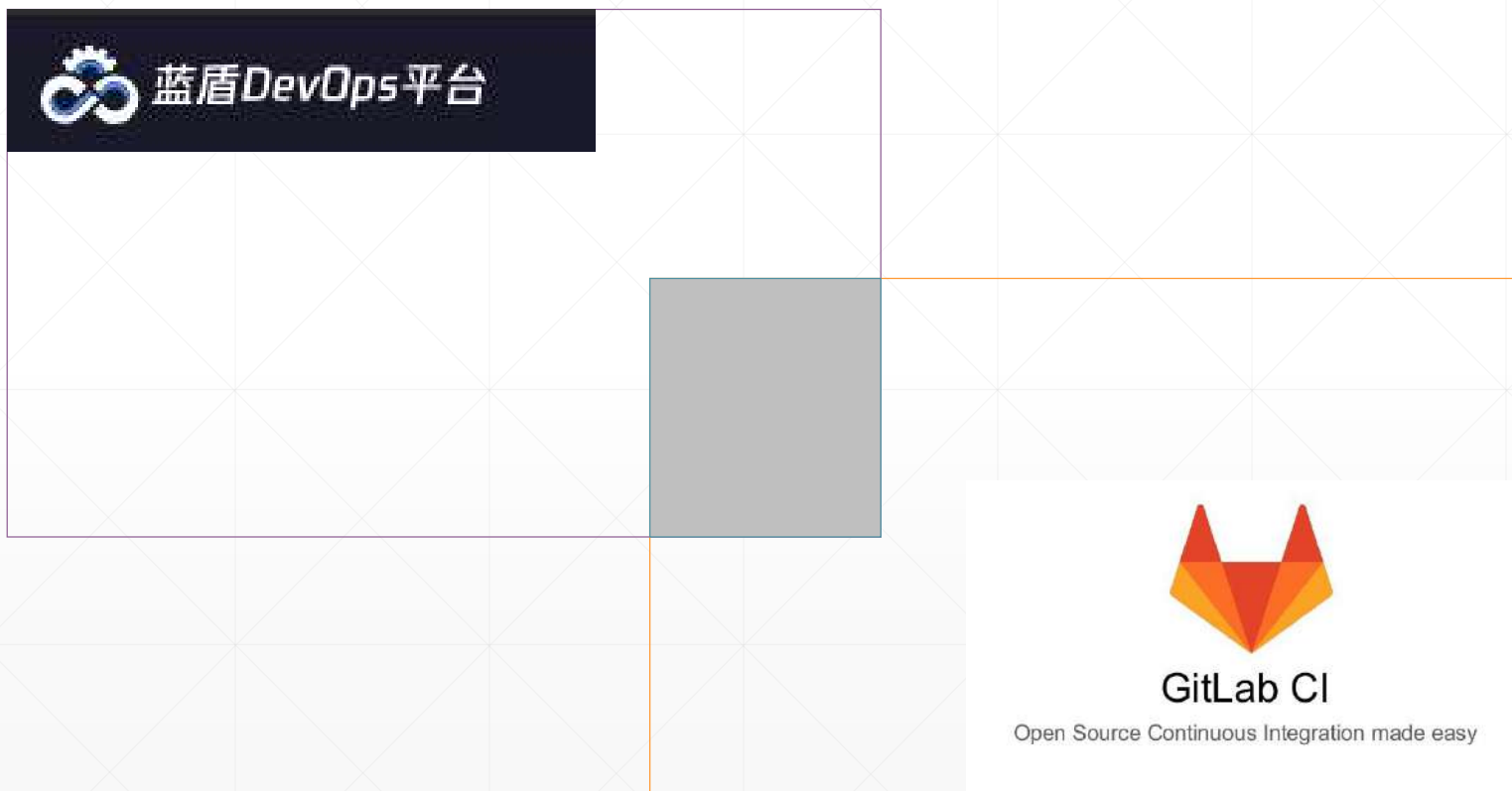
stages:
  - build

build-webpack:
  stage: build
  cache:
    untracked: true
    paths:
      - node_modules
  script:
    - echo "start build test"
    - rm -rf $CI_PROJECT_NAME ./static/test
    - cd ./webpack && tnpm install && tnpm run build-
      test && cd ..
    - git clone $CI_REPOSITORY_URL
    - rm -rf $CI_PROJECT_NAME/static/test && cp -r
      ./static/test $CI_PROJECT_NAME/static
    - cd $CI_PROJECT_NAME
    - git add static/test
    - date >> auto_commit.txt
    - git add auto_commit.txt
    - git commit -m "auto commit [ci skip]"
    - git push http://$GIT_USERNAME:$GIT_PASSWORD@gitlab-
      paas.open.oa.com/$CI_PROJECT_PATH.git master >/dev/null
      2>&1 &
    - echo "end build test"
  only:
    - master
  tags:
    - linux
    - shell
```

适用场景

- 从 GitLab 同步代码到 SVN
 - 前端代码服务器端打包
 - Python PEP8 检测
 - 前端 ESLint 检测
 - 单元测试/测试覆盖度
-

更好的使用蓝盾 & GitLab CI



蓝盾适合做什么

- 固化的流程
 - 安全检查
 - 敏感信息检查
 - 企业版部署
 - 低频任务
 - 交包&交包相关的操作
-

GitLab CI 适合做什么

- 高频任务
 - 代码检查
 - 经常性的 SVN 推送
 - 轻量级任务
 - 单元测试
 - 实验性的新功能
 - 落地开发规范
-

End

