

1. 无感切换目的

如果用户现在用的是clang或者clang++，将会使用毕昇编译器增加编译选项，如果用户使用的是gcc或者g++将会替换为clang或者clang++

2. 切换后的优势

(1)性能

会针对鲲鹏进行后端亲和的编译以及在不损精度的情况下的最优性能编译，性能比开源编译器平均高30%以上，也会针对关键场景进行极致性能优化，让性能再上一个台阶

(2)安全

相比gcc，毕昇编译器在语法检查，标准遵循上更加严格，能帮助用户更早地发现问题，规范代码，提高代码质量

3. 流水线代码示例

```

stages:          # List of stages for jobs, and their order of execution
  - build
  - test
  - deploy

build-job:       # This job runs in the build stage, which runs first.
  stage: build
  script:
    - CURDIR=$(pwd)
    - echo $CURDIR
    - source "${HOME}"/.local/wrap-bin/devkit_pipeline.sh # 如果想要使用毕昇编译器的相关
    能力, 请添加这条命令
    - cp -rf /root/.local/compatibility_testing/template.html.bak
    /root/.local/compatibility_testing/template.html
    - sudo /root/.local/lkp-tests/bin/lkp run /root/.local/lkp-
    tests/programs/compatibility-test/compatibility-test-defaults.yaml
    - cp -rf /root/.local/compatibility_testing/compatibility_report.html
    $CURDIR/compatibility_report.html
    - sudo sh /root/.local/compatibility_testing/Chinese/test_result.sh
    - echo "请去 '${CURDIR}'/compatibility_report.html 查看报告 "
  artifacts:
    paths:
      - compatibility_report.html # 文件后缀.html根据-r参数配置, 也可配置为 src-mig*.*
  tags:
    - dlj # 对应gitlab-runner注册时的标签, 可选择多个

```