

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

```

stage('lkp test') {
    steps {
        script{
            echo '===== lkp test ====='
            sh '''
                CURDIR=$(pwd)
                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
/xxx/compatibility_testing/template.html.bak /xxx/compatibility_testing/template.html
                sudo lkp run /xxx/lkp-
tests/programs/compatibility-test/compatibility-test-defaults.yaml
                cp -rf
/xxx/test/compatibility_testing/compatibility_report.html $CURDIR
                ...

                sh(script: "sudo bash
/xxx/compatibility_testing/report_result.sh", returnStdout:true).trim()

            }
        }
        post {
            always {
                publishHTML(target: [allowMissing: false,
                                alwaysLinkToLastBuild:
false,
                                keepAll           :
true,
                                reportDir         :
'..',
                                reportFiles       :
'compatibility_report.html',
                                reportName       :
'compatibility test Report']
                )
            }
        }
    }
}

```