韩后起

2018-02-21

拆分大工程本地检测

目录

[业务背景 3](#_Toc1683246)

[业务流程 4](#_Toc1683247)

[业务需求 4](#_Toc1683248)

[checkService 4](#_Toc1683249)

[fixbotMain 5](#_Toc1683250)

[requestDeliver 6](#_Toc1683251)

[integration 6](#_Toc1683252)

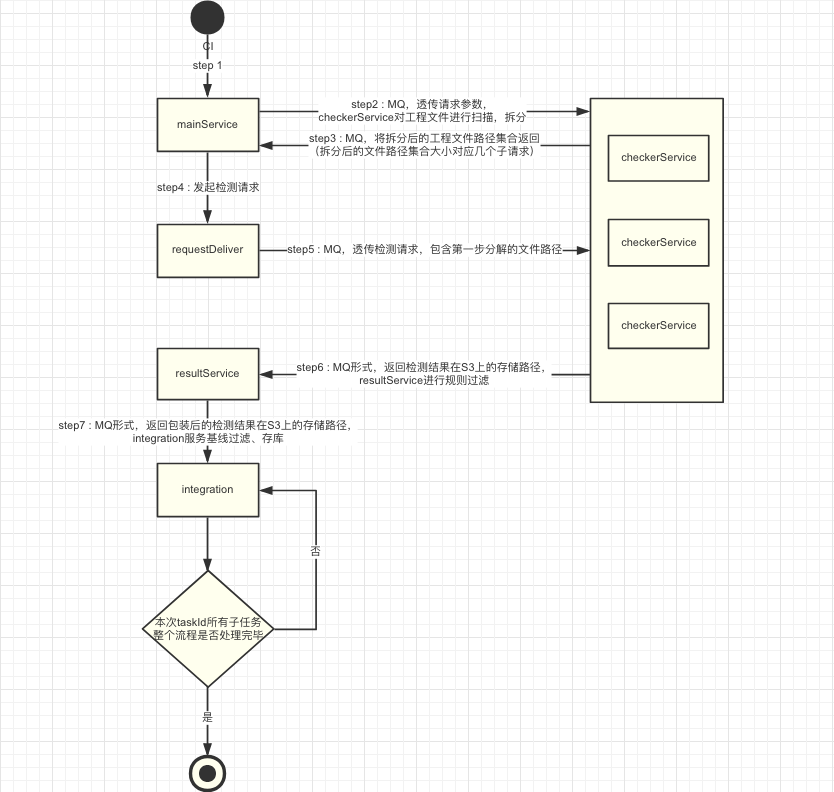
[测试流程 6](#_Toc1683253)

reviewbot大工程本地检测

# 业务背景

面对较大的工程，reviewbot整体，检测时间较长，无法满都用户的需求，针对检测效率低的情况，将reviewbot版本级进行调整，将大的工程按照文件拆分为多个文件列表，一个工程的检测，分为多次检测；

# 业务流程



# 业务需求

## checkService

checkService，部署在执行机上，使用shell脚本，遍历本地被检测工程，按照文件后缀，进行遍历，将遍历出的被检测工程文件的绝对路径，以一定的数量，写入到指定的目录文件下，获取到这些含有被检测文件绝对路径的文件，通过MQ的形式，发送给fixbotMain；

注意点：多少个被检测文件，写入到一个文件中；（数量，设置为系统变量，由fixbotMain透传给checkService，可进行修改，并且一个task检测，系统数量不发生变化，及时用户修改，此系统变量也只是在下一次task中生效，在测试的时候，按照cpu大小，内存大小，以及checkService的检测能力，对数量进行调整，选择）；面对cdt是否检测文件数量设置小，待调整；

checkService：每检测完一个文件列表，将缺陷上传至S3，发送MQ值fixbotMain；

## fixbotMain

监听checkService发送的文件列表MQ，保存task，subtask，对应的request，subtask；

子任务标识：taskId (uuid)+checkName+subTaskId (uuid);

mongo库：

request表：添加本次task整个流程的状态参数；flowStatus；

subTask表：添加本次subTask整个流程的状态参数：subFlowStatus；

fixbotMain：每接收到checkService发送的MQ修改原有状态值状态（保持现状）；

fixbotMain：每处理完一次checkService检测完成后发送的MQ，规则过滤，匹配规范，上传缺陷文件至S3；

（以下接口也可在integration内部实现）

接口定义：public boolean isFlowOver(String subTaskId);

确认本次task是否检测完毕，供integration调用；

请求参数 jsonString

{“ subTaskId”:” taskId+checkerName+subTaskId”};

处理逻辑：

1. 更新本次子任务subFlowStatus状态为完成；
2. 查询taskId对应的所有子任务是否都处理完毕

完成：更新flowStatus状态为完成，返回true；

未完成：返回false；

1. 返回值 Boolean：

true：本次task已检测完毕；

false：本次task未检测完毕；

## requestDeliver

保持现有业务逻辑

## integration

1、异步进行消费fixbotMain消息，每检测完成一个文件，上传至S3；

2、http调用fixbotMain接口isFlowOver，返回值true，向君辉发送MQ；返回值false，不发送；

以上第二步也可在integration内部查库判断；将isFlowOver接口在integration内部实现；

# 测试流程

TODO