



工作回顾

完成基于Mac Os环境的测试平台的上线

负责的服务

- Test-case-service
- Test-device-service
- Test-report-site
- Test-site
- Test-api
- Test-daemon

完成的核心功能

- 配置测试对象
- 解析测试用例
- 发现测试设备
- 监控测试设备
- 创建测试计划
- 运行测试任务，管理测试执行器的生命周期
- 监控测试进度
- 管理测试结果

亮点

- 解析测试用例的速率提升到1~2s
- 支持并行测试：每台Mac Mini支持3台虚拟机的并行测试

完成集成Appcenter测试到测试平台的上线

负责的服务

- Appcenter-service

完成的核心功能

- 获取App Center的设备&设备集
- 创建Test Run

亮点

- 创建Test Run 所需时间从7min 提升到1min左右

从Appcenter 下载App：3s左右

上传App-file到appcenter: 3s左右

上传Test-file到appcenter: 使用包含10个线程的线程池，异步上传文件，每个线程上传55个文件左右，时间是35s左右

Maven打包速度：10s左右（cpu 4核）

提升设备资源的利用率：对于接入测试平台的操作系统，平台会主动去发现该操作系统上已安装或已连接的Ios和android设备，包括虚拟机和真机，完成测试设备种类的可扩展性和设备的可插拔性。

提升测试速率：因为每个迭代留给测试开发团队的测试时间只有2天，随着测试用例的增多，测试时间就会延长，采用并行测试（多台不同类型的设备并行 & 同类型多台设备的并行）完成测试效率质的提升。

提早发现App的缺陷：在以往的测试过程中，测试开发人员需要手动到打包服务器下载app包到本地，再放到测试脚本中进行测试，在测试平台进行测试，可以保证每次都以系统中最新的app包进行测试，完成测试结果准确性的提升。

提升自动化程度：因为测试过程手工参与度过高，容易出现操作失误，如接收测试报告地址选错、测试设备选错等因素导致测试需要重新执行，提供可视化UI操作，完成杜绝手工操作失误带来的影响。

提升测试用例的解析速率：原先使用类加载的方式，耗费内存资源和性能，需要花费至少1分钟，到现在通过asm字节码解析class文件获取测试用例，达到秒级别完成解析。

成果

问题分析

个人

文档写作能力有待提升

团队

需求描述清晰

未来规划

技术

K8s

项目

集成Web Automation Test

基于Windows Os环境的测试平台