



TesterHome社区孵化机构-霍格沃兹测试学院

移动app自动化测试技术

霍格沃兹测试学院

思寒

seveniruby@testerhome.com



About Me

- ❖ 黄延胜 (思寒 seveniruby@testerhome)
 - ❖ TesterHome社区联合创始人 (退休)
 - ❖ 霍格沃兹测试学院创始人
 - ❖ 开源工具AppCrawler作者
 - ❖ MTSC、GMTC大会讲师
 - ❖ 主要工作经历：阿里、百度、雪球
- ❖ 霍格沃兹测试学院
 - ❖ 领先的测试开发技术创新公司
 - ❖ 测试开发工程师的培训摇篮



黄延胜[思寒] 

中国 北京



扫一扫上面的二维码图案，加我微信



About TesterHome

- ❖ TesterHome.com 社区介绍
 - ❖ 中国最大的移动测试社区
 - ❖ 注册测试工程师数万名
 - ❖ 倡导用技术推动测试行业发展
- ❖ 社区活动
 - ❖ 一年一度移动互联网测试开发大会
 - ❖ 全国十多城市的线下技术交流沙龙
 - ❖ 积极推动开源项目的发展
 - ❖ 孵化众多特色测试组织





TesterHome社区首页

TesterHome

社区

Bug 曝光台

问答

社团

招聘

Wiki

酷站

搜索本站内容



社区精华帖

查看更多精华帖...

- | | |
|---|--|
|  <p>Macaca 自动遍历器 NoSmoke 发布公测 </p> <p>Samuel.ZhaoY</p> <p>43</p> |  <p>[小记] Android 获取详细功耗 </p> <p>shixue33 for 平安科技</p> <p>8</p> |
|  <p>支付笔记 (1) 之 PayPal 支付原理 </p> <p>Ningxw</p> <p>9</p> |  <p>[持续交付实践] 基于 sonarqube 的代码检查</p> <p>cay</p> <p>8</p> |
|  <p>ExtentReports 官方说明文档翻译 </p> <p>naraka</p> <p>5</p> |  <p>[持续交付实践] pipeline: pipeline 使用之快速入</p> <p>cay</p> <p>6</p> |
|  <p>实践：火线静态代码扫描以及定位错误代码</p> <p>captain_liu</p> <p>16</p> |  <p>[持续交付实践] 开篇：持续集成&持续交付综</p> <p>cay</p> <p>6</p> |
|  <p>脑洞小开-selenium, 动态运行日常调试代码</p> <p>hellomm</p> <p>8</p> |  <p>安全技能学习笔记——反射型 XSS、SQL</p> <p>chenhengjie123 for PPMoney</p> <p>9</p> |
|  <p>appium+python+jenkins 一套自动化测试框</p> <p>JinZhu</p> <p>14</p> |  <p>[开源] 基于 Nodejs 的 Restful Api 集成测试</p> <p>brookshi</p> <p>13</p> |
|  <p>fastmonkey 自我扫盲贴 </p> <p>yxys01</p> <p>19</p> |  <p>使用 Python 库 facebook-wda 完成网易云</p> <p>codeskyblue</p> <p>14</p> |

七日最热 Top10

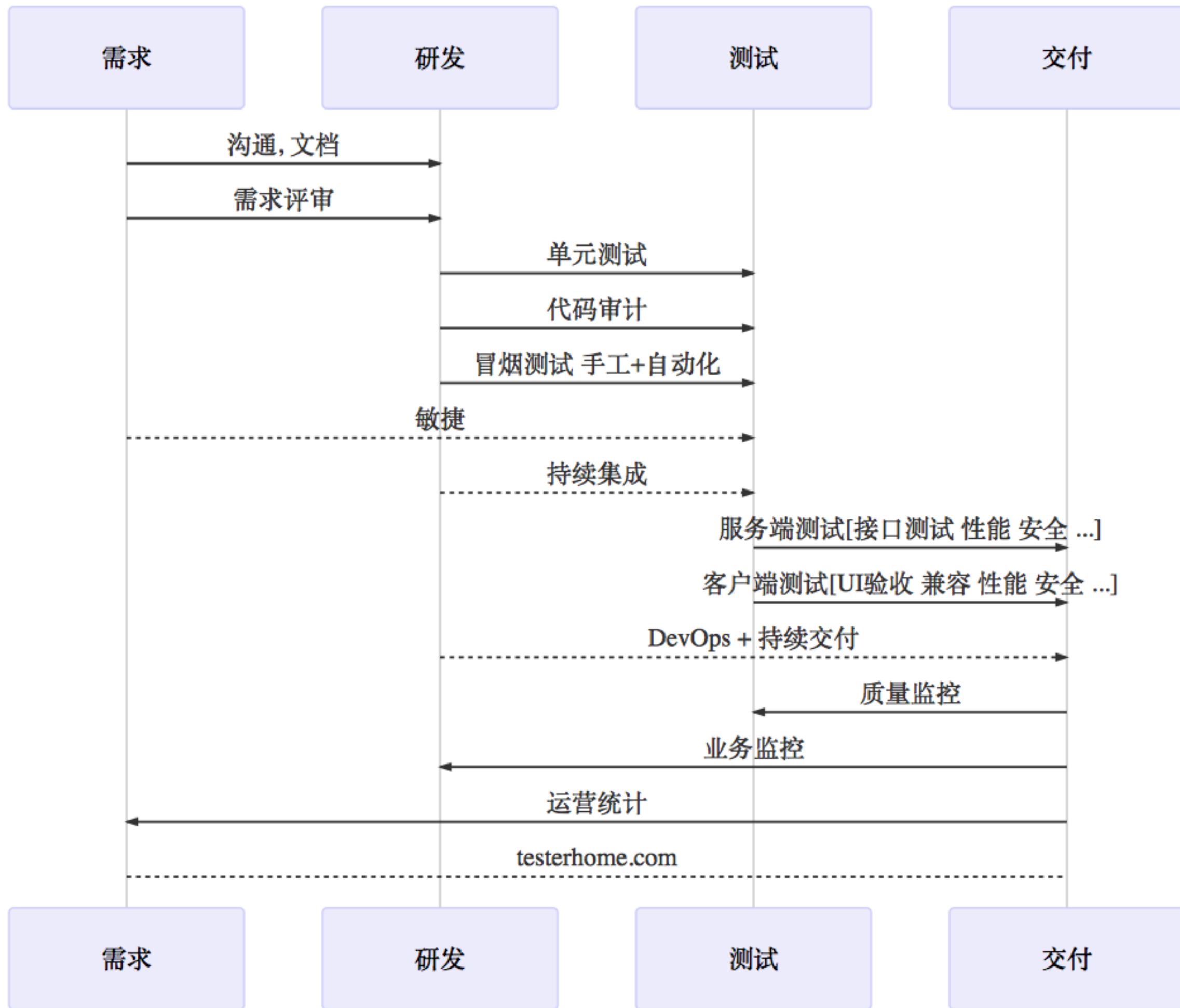
- Macaca 自动遍历器 NoSmoke 发...
- appium 调用不起程序，请大神看...
- 看了苹果发布会，突发奇想，Fac...
- [北京 / 杭州] 阿里巴巴-飞猪-高级...
- 如何量化测试覆盖率
- fiddler 在 iOS 手机上的使用问题
- 支付笔记 (1) 之 PayPal 支付原理
- [小记] Android 获取详细功耗
- 助力 RubyConf China，凭微信朋...
- [持续交付实践] 基于 sonarqube ...

最新 bug

- testerhome 首页界面布局出错
- 支付宝切屏异常

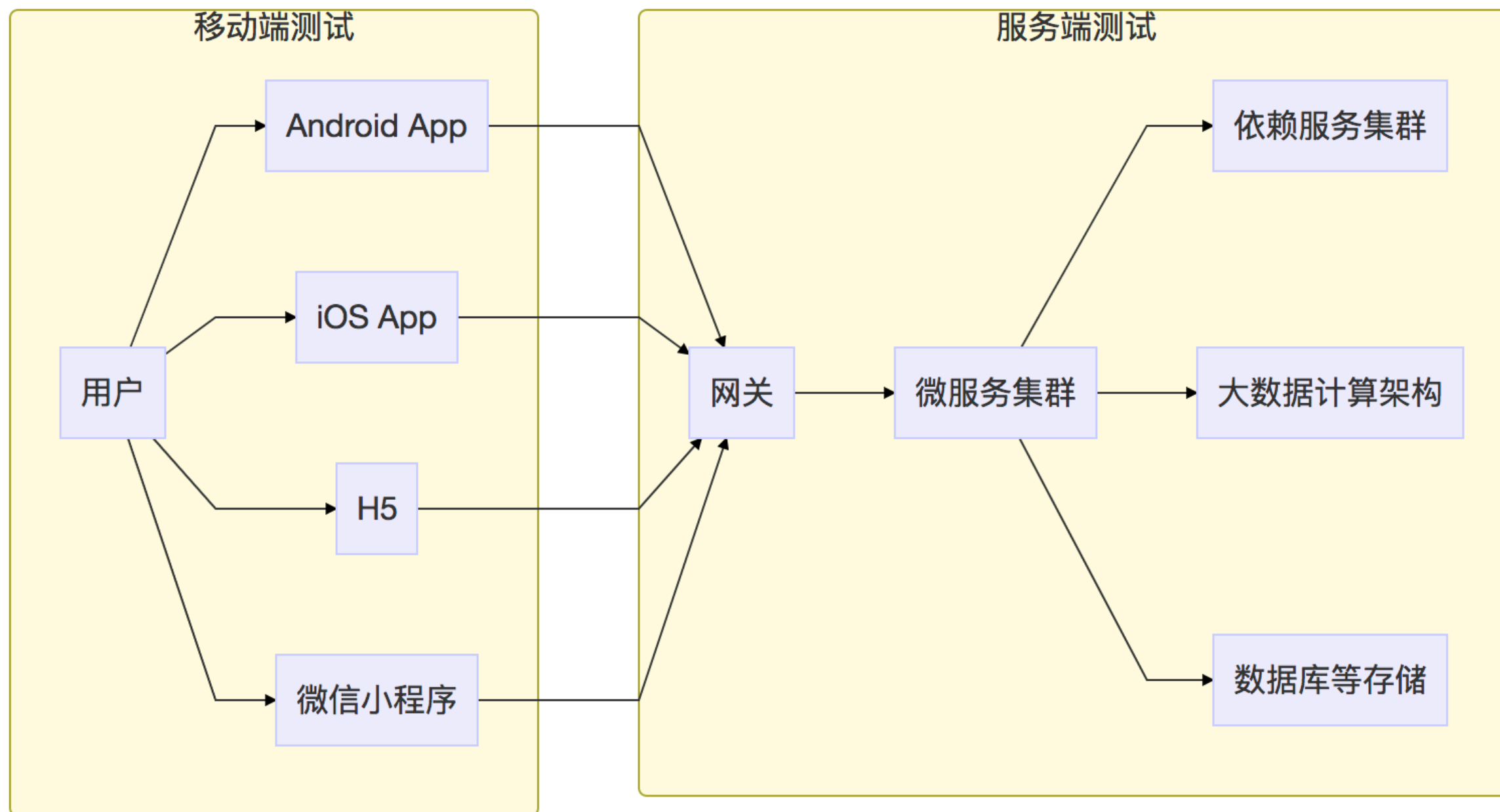


移动自动化测试



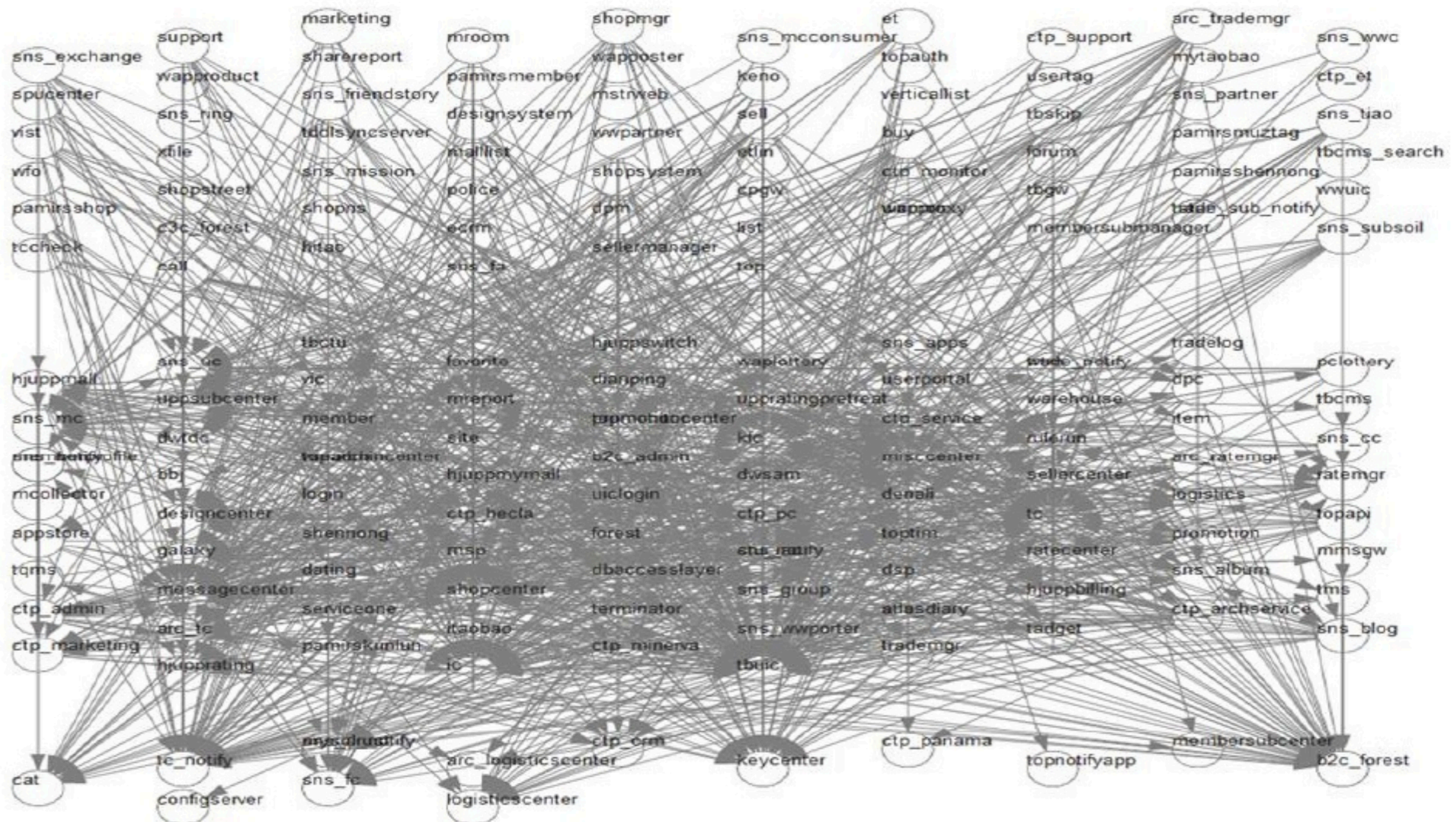


互联网公司的服务架构





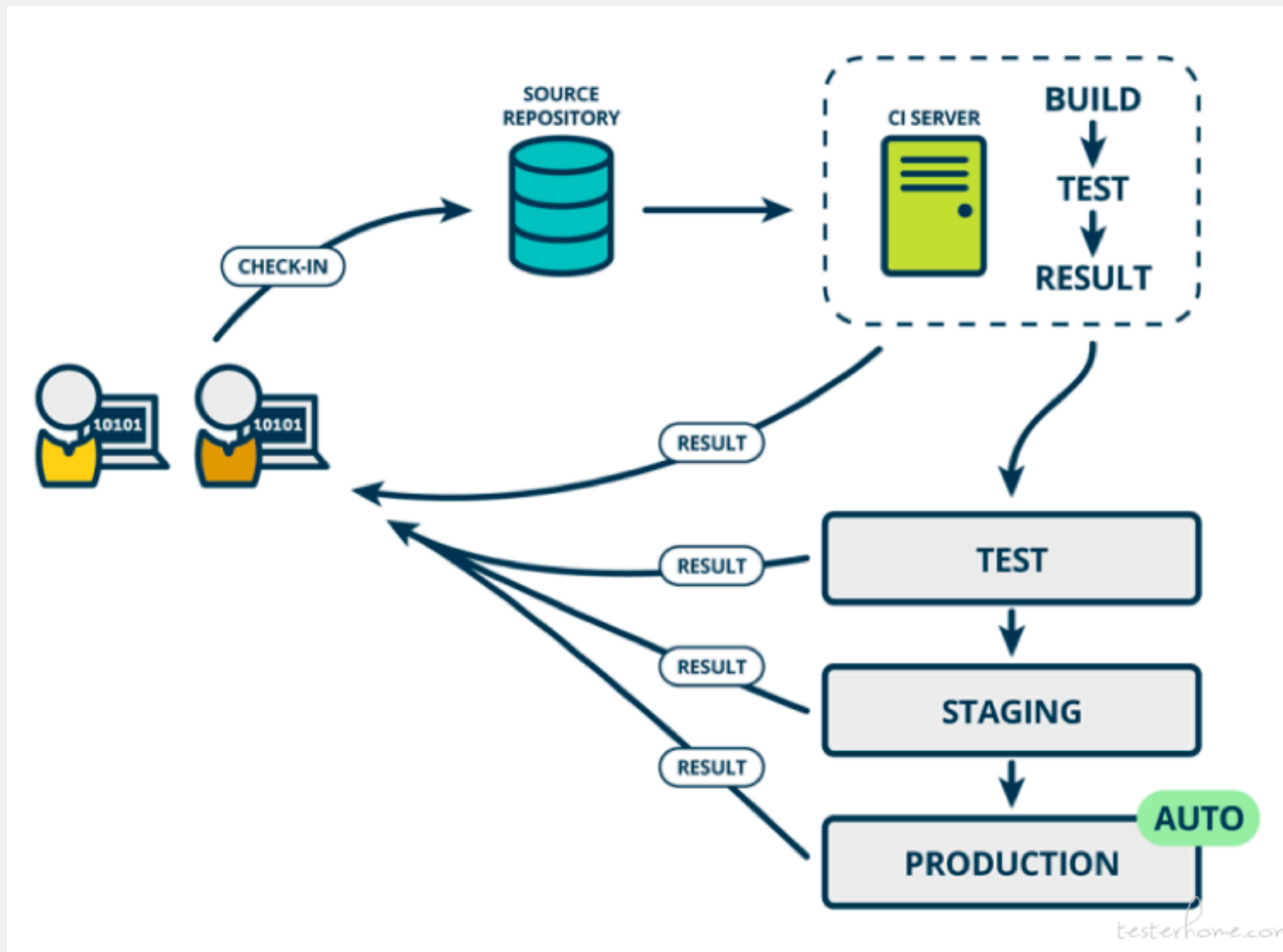
真实的后端服务



2012 淘宝核心链路应用拓扑图



持续交付与DevOps





我们面临的问题

- ❖ 按月发布->按周发布->按小时发布
- ❖ 多端发布: Android iOS 微信小程序 h5
- ❖ 多环境发布: 联调环境 测试环境 预发布环境 线上环境
- ❖ 多机型发布: 众多设备型号 众多系统版本
- ❖ 多版本共存: 用户群体中存在多个不同的版本
- ❖ 历史回归测试任务: 成百上千条业务用例如何回归
- ❖ 总结: 背锅+加班



分层测试

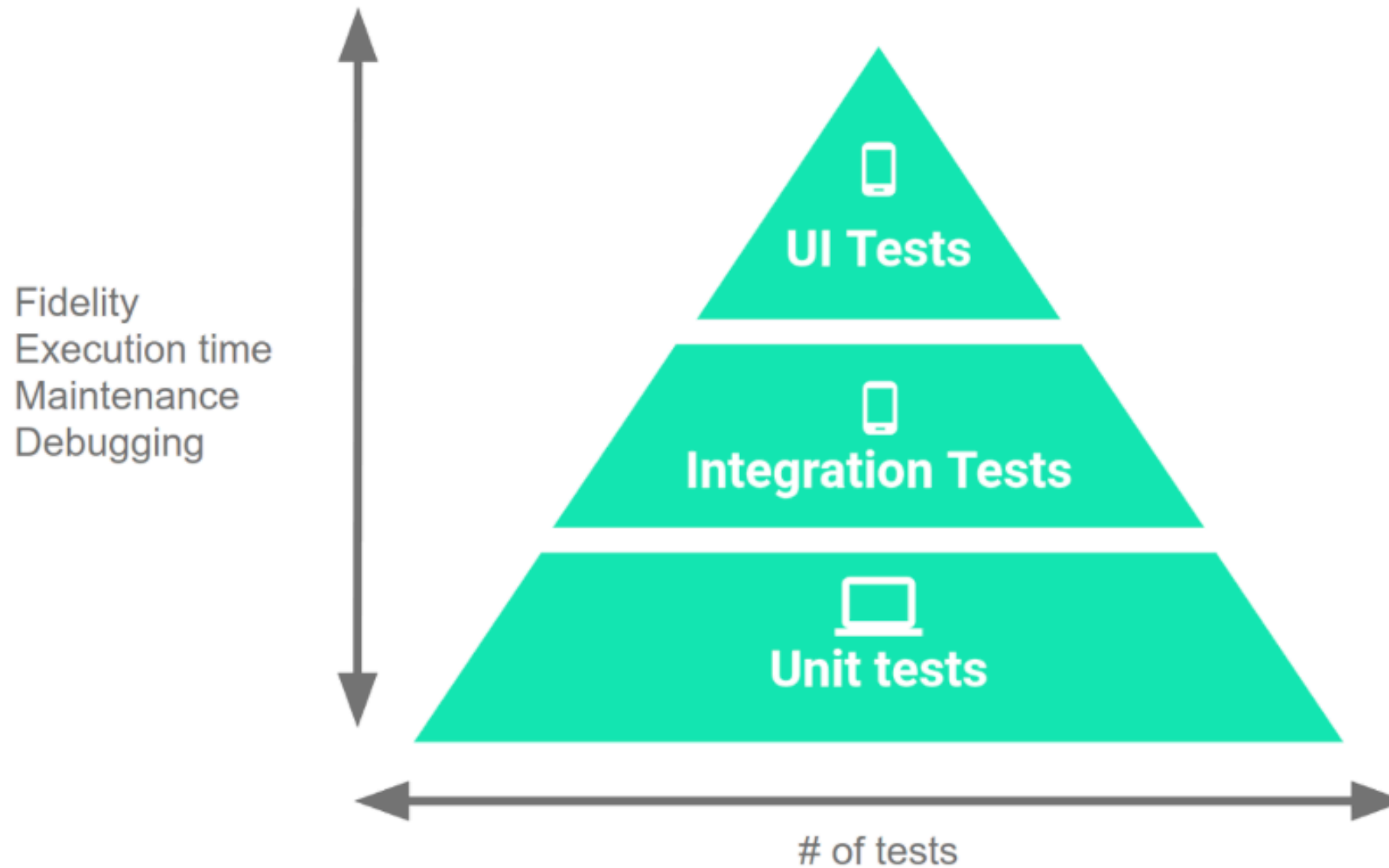


Figure 2. The Testing Pyramid, showing the three categories of tests that you should include in your app's test suite



快速迭代同样存在于前端

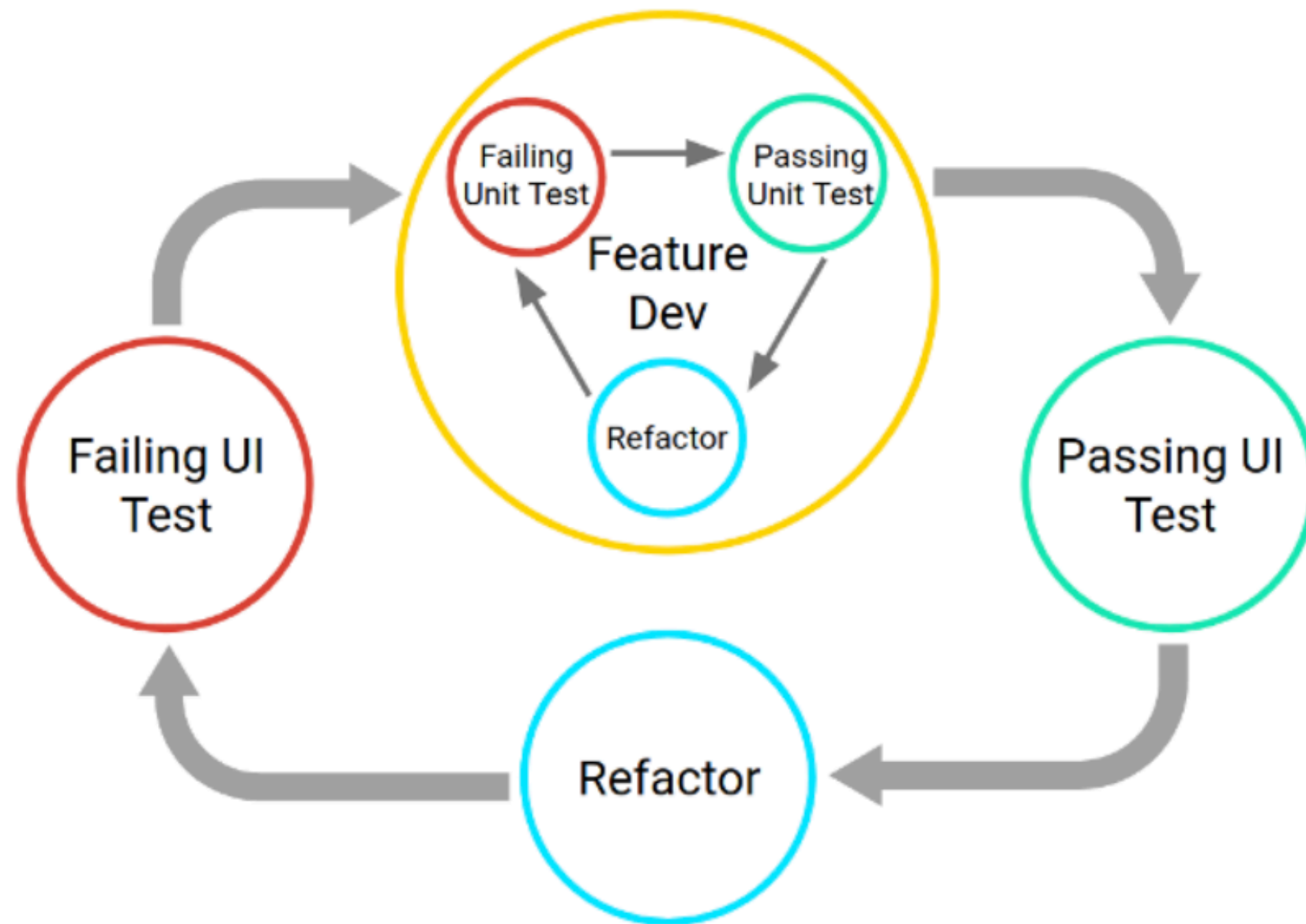


Figure 1. The two cycles associated with iterative, test-driven development



UI自动化测试方面的努力

- ❖ Google: Espresso、Uiautomator
- ❖ Facebook: WebDriverAgent
- ❖ Alibaba: Macaca
- ❖ Ebay: Selendroid
- ❖ Walmart: Robotium
- ❖ SauceLabs: Appium
- ❖ ThoughtWorks: Selenium



谁在做UI自动化测试

- ❖ 阿里：
 - ❖ macaca等多个内部UI测试框架
 - ❖ mqc等多个内部UI测试平台
- ❖ 腾讯：
 - ❖ WeTest、uTest
 - ❖ 刚开源微信小程序自动化测试工具
 - ❖ 众多测试部门在落地使用
- ❖ 百度：
 - ❖ MTC
 - ❖ 内部众多测试部门都在使用成熟开源工具
- ❖ 其他：几乎中大型的公司都有自动化测试建设



谁可以不做UI自动化

- ❖ 公司节奏慢：人工测试就可以满足公司需求
- ❖ 初创团队：业务未定型，产品重构频繁
- ❖ 业务无积累：大量小功能频繁上线
- ❖ 测试体系完备：单测、集成测试非常完备，完美金字塔



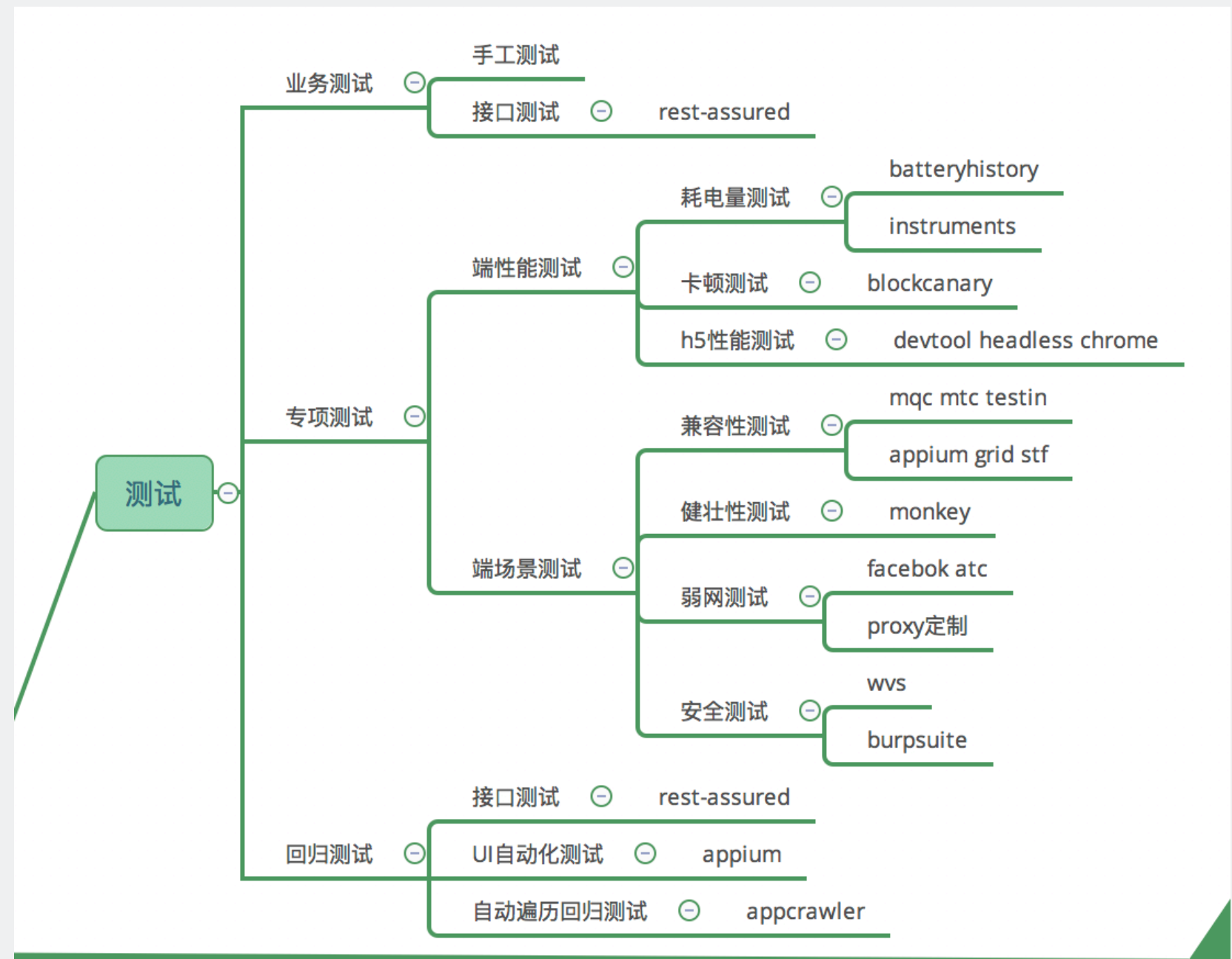
UI自动化测试价值在哪

- ❖ 提高效率：
 - ❖ 融入企业迭代流水线，与CI/CD/DevOps结合
 - ❖ 回归测试、功能测试加速
- ❖ 提高质量：
 - ❖ 兼容性测试
 - ❖ 专项/非功能测试
 - ❖ 自动化探索测试



UI自动化测试在企业中的用途

- ❖ 回归测试
- ❖ 自动探索测试
- ❖ 专项测试自动化
- ❖ 竞品分析





掌握UI自动化测试对个人的价值

❖ 技能提升

- ❖ 更深入的理解待测产品与平台
- ❖ 工作效率提升与方法论提升
- ❖ 融入行业一线公司的工作氛围

❖ 增加个人行业竞争力

- ❖ UI自动化测试在薪资中的收益是5000 / 月
- ❖ 接口测试在薪资中的收益是5000 / 月
- ❖ CI/CD/DevOps薪资收益是10000 / 月

❖ 总结：与行业优秀测试工程师、开发工程师、CTO将会有更多共同语言



手工测试难道就无用嘛

- ❖ 非常有用
- ❖ 但仅有手工测试是不够的
- ❖ 手工测试+自动化测试才是企业快速迭代的根基
- ❖ 各自占比与公司业务、发展阶段和行业技术进步相关
- ❖ 总结：现阶段的社会矛盾就是日益增长的快速迭代需求与落后的测试生产力之间的矛盾



UI自动化测试的瓶颈

❖ 成本高：

- ❖ 人力成本高：优秀测试开发工程师难招
- ❖ 技术门槛高：80%的自动化测试工程师在自动化能力上都不及格
- ❖ 复用率不高：UI和业务流程变更是最主要变更
- ❖ 稳定性不足：容易被干扰，执行慢

❖ 常见行业错误观点

- ❖ 错误观点一：UI自动化测试无用论
- ❖ 错误观点二：UI自动化测试只能模拟人工



合理的使用UI自动化

- ❖ 使用分层测试策略，控制UI自动化测试规模
 - ❖ 少数核心用例交给自动化测试
 - ❖ 大部分的基础回归测试交给自动遍历
 - ❖ 新功能测试交给人工测试
- ❖ 技术改进方案
 - ❖ 良好的维护模型：PageObject或者其他更简单的封装
 - ❖ 更好的框架支持：增加Watch，智能等待，失败重试等机制



自动化测试就是机器点点点?

- ❖ 传统自动化测试
- ❖ 自动探索测试
- ❖ code-less方向的自动化测试
- ❖ 自动化测试用例智能生成



如何高效实用的落地自动化

❖ 学习方向

- ❖ 了解待测产品和平台：Android、iOS、App
- ❖ 精通最好的测试框架：Appium、UiAutomator、WebDriverAgent
- ❖ 掌握自动化测试落地经验：PageObject、稳定性提升、数据驱动。。。

❖ 达到的结果

- ❖ 一天搞定一款app的自动化测试脚手架
- ❖ 一天完成单个功能的业务测试自动化用例集合
- ❖ N周内搞定公司的app自动化测试



课程预告

- ❖ 移动自动化测试入门
- ❖ Android自动化测试
- ❖ iOS自动化测试
- ❖ WebView与小程序自动化测试
- ❖ PageObject模式与数据驱动
- ❖ 自动化测试与持续集成结合
- ❖ 测试框架原理剖析
- ❖ 雪球自动化测试实战
- ❖ 企业微信自动化测试实战
- ❖ 自建智能设备实验室



入门篇： 自动化测试框架选择



目前mobile自动化的方案

iOS	Android
calabash-ios	calabash-android
Frank	MonkeyTalk
UIAutomation	Robotium
ios-driver	UiAutomator
KeepItFunctional	selendroid





业界top框架综合对比

	Robotium	uiautomator	Espresso	Appium	Calabash
Android	Yes	Yes	Yes	Yes	Yes
iOS	No	No	No	Yes	Yes
Mobile web	Yes (Android)	Limited to x.y clicks	No	Yes (Android & iOS)	Yes (Android)
Scripting Language	Java	Java	Java	Almost any	Ruby
Test creation tools	Testdroid Recorder	UI Automator viewer	Hierarchy Viewer	Appium.app	CLI
Supported API levels	All	16 =>	8, 10, 15-19	All	All
Community	Contributors	Google	Google	Active	Pretty quiet



框架发展趋势

calabash-...
搜索字词

robotium
搜索字词

appium
搜索字词

calabash-...
搜索字词

selendroid
搜索字词

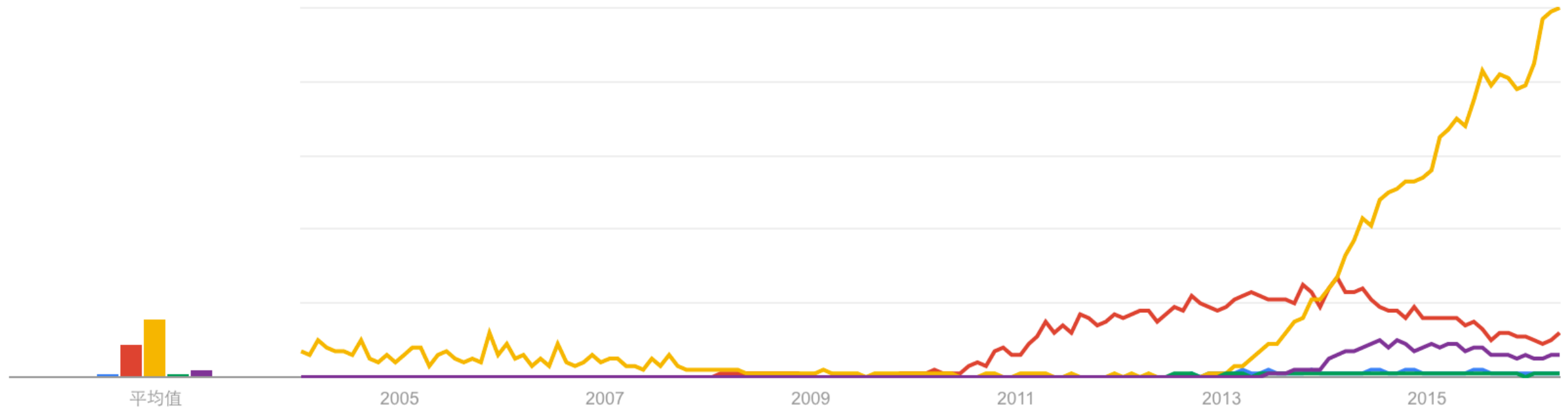
热度随时间变化的趋势



新闻头条



预测





自动化测试框架选择

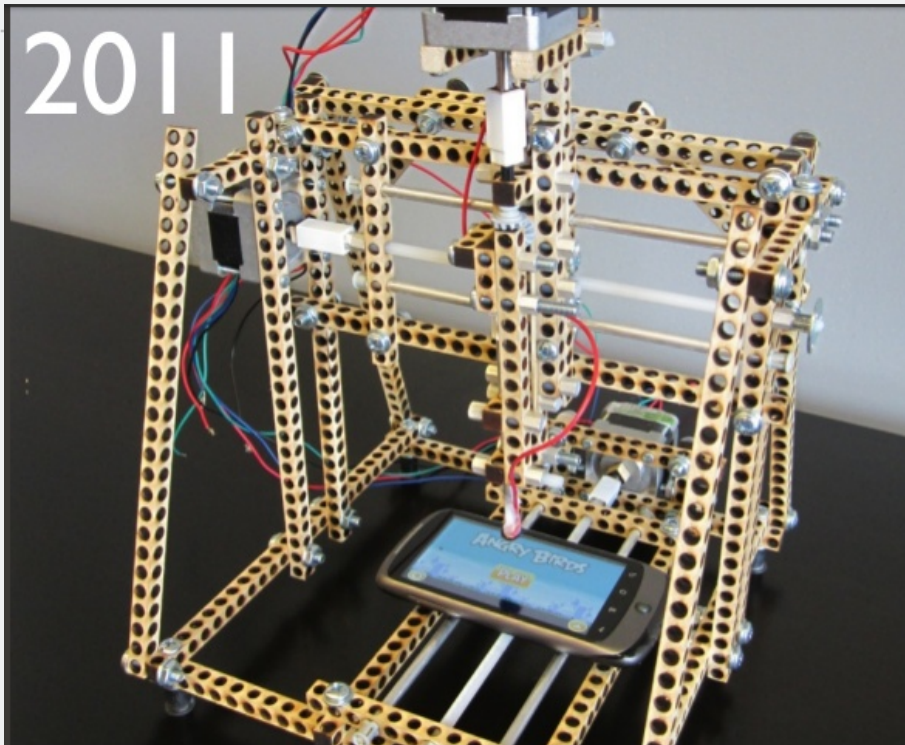
- ❖ 推荐Appium
 - ❖ 跨语言：Java、Python、nodejs等
 - ❖ 跨平台
 - ❖ Andoid、iOS
 - ❖ Windows、Mac
 - ❖ 底层多引擎可切换
 - ❖ 生态丰富，社区强大
- ❖ iOS：KIF WDA XCTest
- ❖ Android：Robotium Uiautomator
- ❖ 其他框架：calabash macaca atx





开源硬件Arduino支持

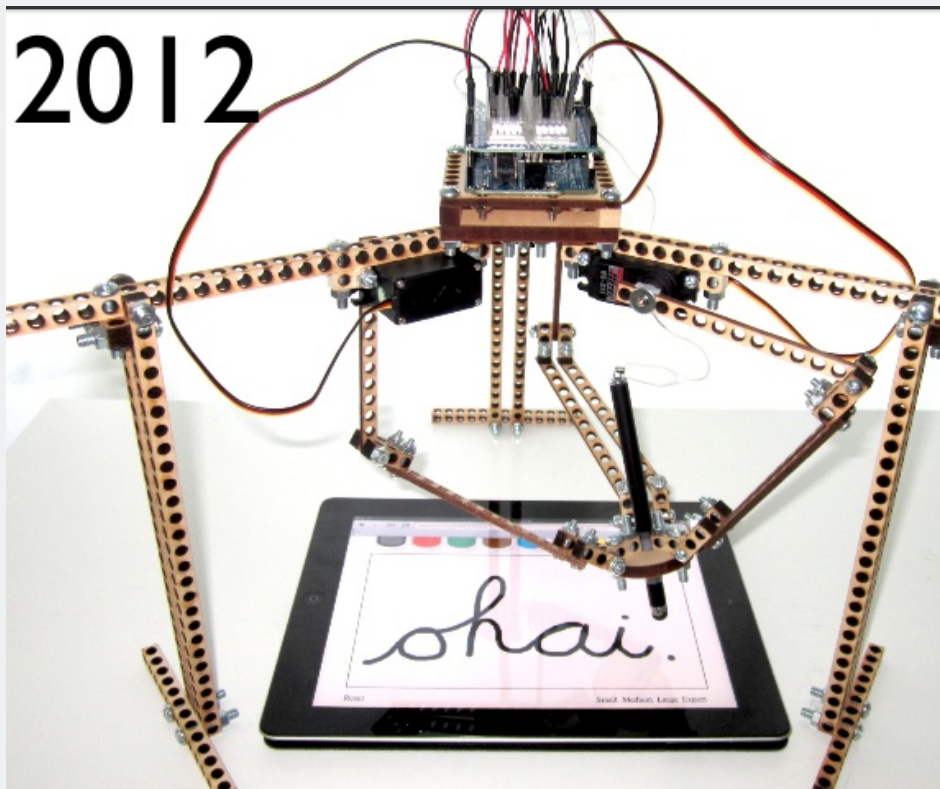
2011



2013



2012





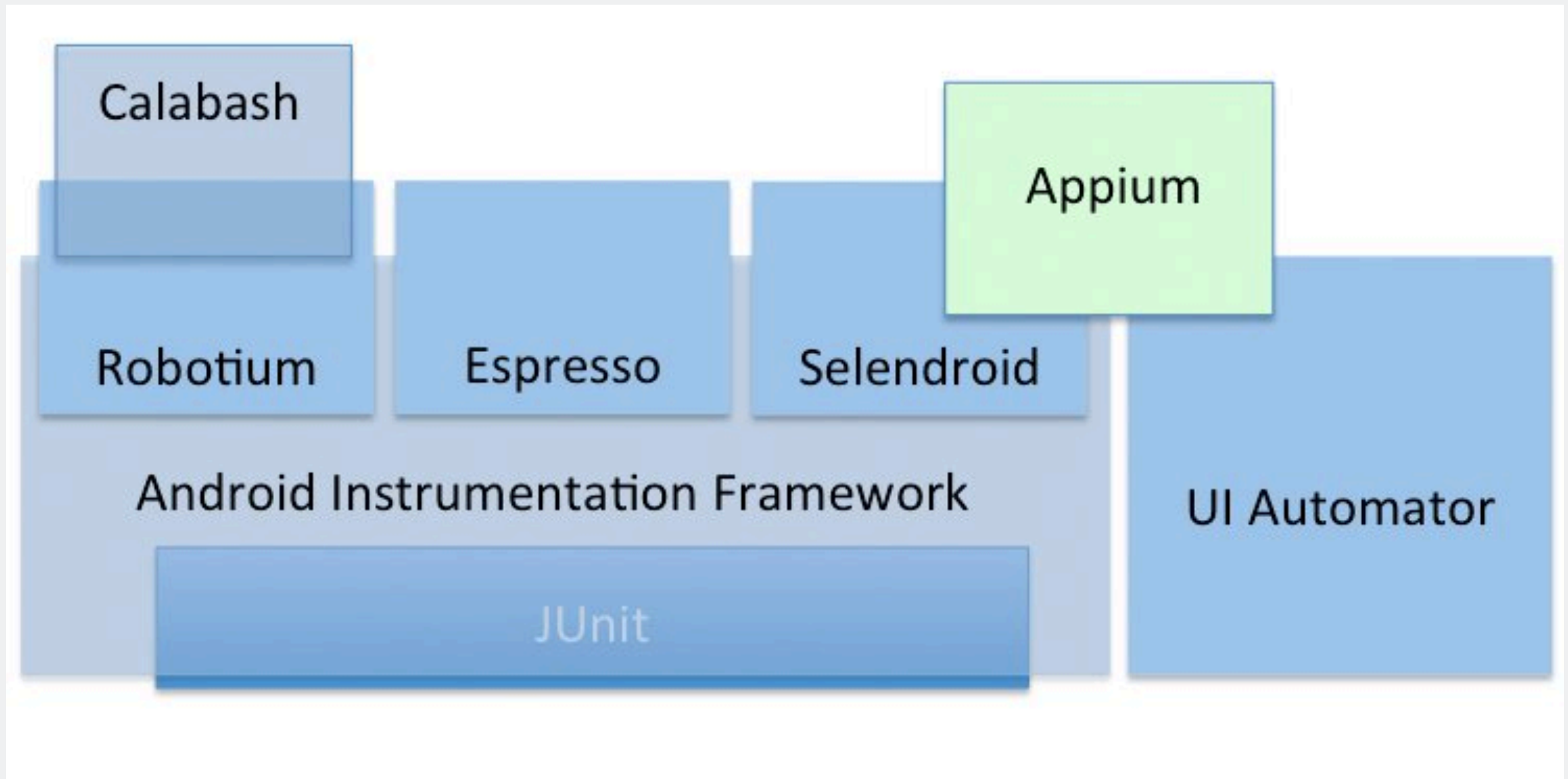
Appium引擎列表

- ❖ Android
 - ❖ espresso
 - ❖ selendroid
 - ❖ uiautomator
 - ❖ uiautomator2 [推荐]
- ❖ iOS
 - ❖ uiautomation
 - ❖ xcuitest [推荐]
- ❖ mac
- ❖ windows

- iOS
 - The [XCUI Test Driver](#)
 - (DEPRECATED) The [UIAutomation Driver](#)
- Android
 - (BETA) The [Espresso Driver](#)
 - The [UiAutomator2 Driver](#)
 - (DEPRECATED) The [UiAutomator Driver](#)
 - (DEPRECATED) The [Selendroid Driver](#)
- The [Windows Driver](#) (for Windows Desktop apps)
- The [Mac Driver](#) (for Mac Desktop apps)



Android 自动化测试



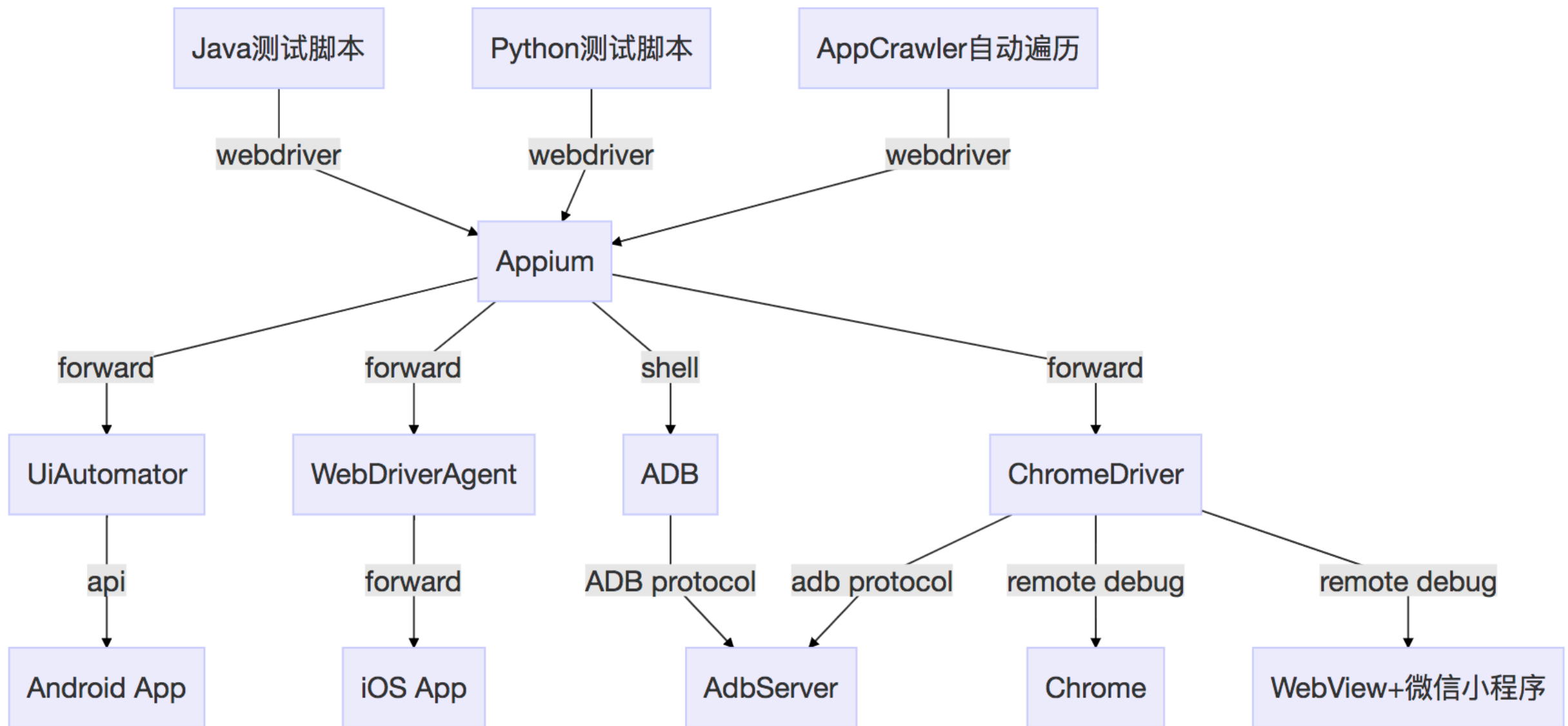


iOS 自动化测试

Automation framework	Xcode version			iOS version		
	6.x	7.x	8.x	8.x	9.x	10.x
UIAutomation	✓	✓		✓	✓	
XCTest (KIF)		✓	✓	✓	✓	✓
XCUITest		✓	✓		✓	✓
Appium < 1.6	✓	✓		✓	✓	
Appium >= 1.6			✓	✓	✓	✓
			Xcode iOS support	iOS version		
				8.x	9.x	10.x
			Xcode 6.x	✓		
			Xcode 7.x	✓	✓	
			Xcode 8.x	✓	✓	✓



多端架构与自动化





入门篇：Appium环境安装



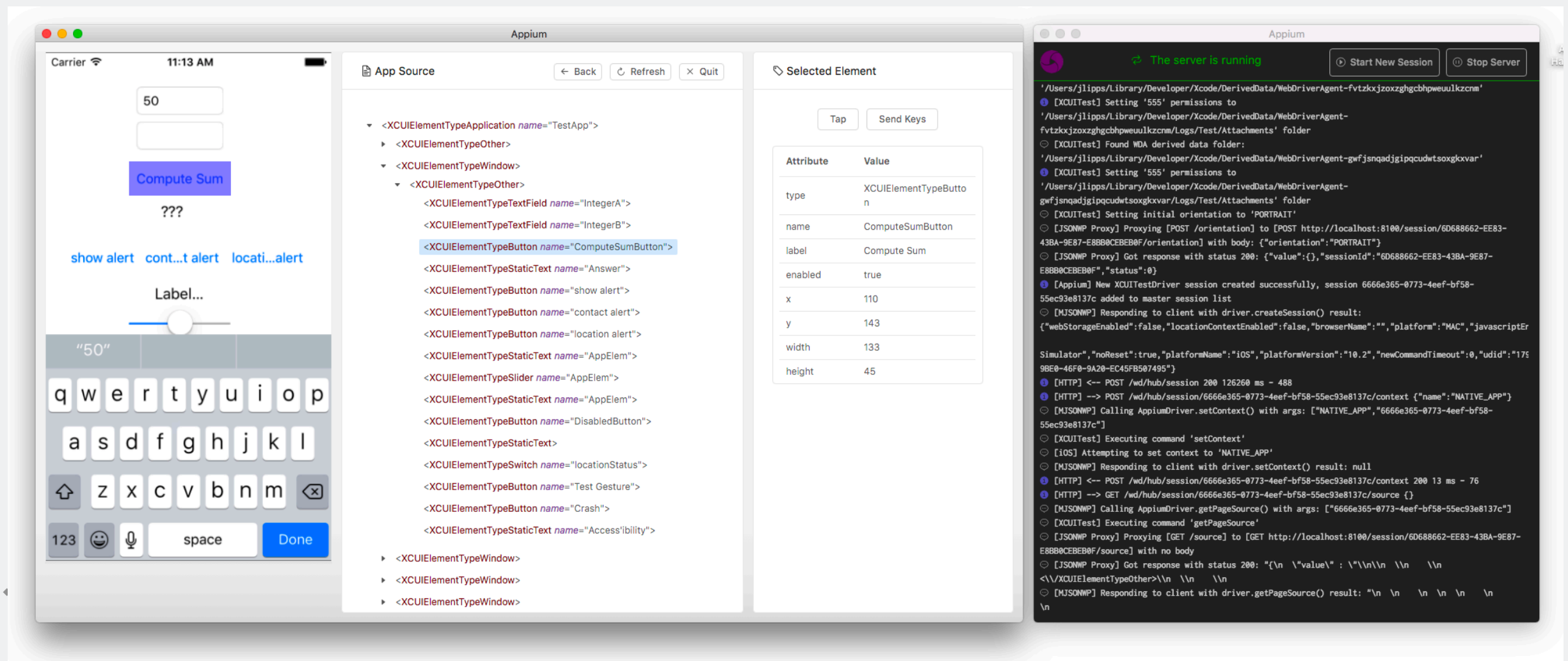
Appium生态工具

- ❖ adb: android的控制工具, 用于获取android的各种数据和控制
- ❖ Appium Desktop: 内嵌了appium server和inspector的综合工具
- ❖ Appium Server: appium的核心工具, 命令行工具
- ❖ Appium client: 各种语言的客户端封装库, 用于连接appium server
 - ❖ python
 - ❖ java
 - ❖ ruby
 - ❖ robotframework-appium
- ❖ AppCrawler 自动遍历工具



appium desktop 工具包

- ❖ Appium Server and Inspector in Desktop GUIs for Mac, Windows, and Linux
- ❖ 下载地址: <https://github.com/appium/appium-desktop/releases>





appium server环境安装

❖ 安装方法

❖ 安装nodejs，推荐LTS版本

❖ 安装appium

❖ `npm install -g appium`

❖ 学院专属工具 `bashgems : proxy npm install -g appium`

❖ 淘宝cnpm

❖ `npm install -g cnpm --registry=https://registry.npm.taobao.org`

❖ `cnpm install -g appium`

❖ <https://testerhome.com/topics/6040>



6040

- ❖ `https://testerhome.com/topics/6040`
- ❖ `npm view appium versions --json`
- ❖ `npm install -g appium`
- ❖ `proxy=http://112.126.81.122:6$(date +%m%d)`
- ❖ `http_proxy=$proxy https_proxy=$proxy npm --proxy $proxy --https-proxy $proxy install -g appium --verbose`



为什么Appium Server安装困难

❖ node问题

- ❖ node安装不要用root权限安装
- ❖ node版本不宜太低，也不宜太高
- ❖ npm需要更新自己，使用npm version观察node相关配置

❖ 网络问题

- ❖ npm安装需要科学上网
- ❖ cnpm需要从github上下载相关jar和apk也需要科学上网，但是设置代理会导致cnpm本来的加速无效，只能等。。。

❖ 路径问题

- ❖ PATH变量要设置 adb java等路径，可以用appium-doctor验证
- ❖ iOS还需要解决WebDriverAgent的编译和依赖下载问题



入门篇：多语言测试用例录制



android 自动化前提依赖

- ❖ android sdk: Android Studio可辅助安装
- ❖ 模拟器
 - ❖ Android Studio自带emulator [推荐]
 - ❖ Genymotion
 - ❖ BlueStacks
- ❖ 真机
- ❖ Appium Desktop: 入门学习工具



获取app的信息

❖ app信息

- ❖ 获取当前界面元素: `adb shell dumpsys activity top`
- ❖ 获取任务列表: `adb shell dumpsys activity activities`

❖ app入口

- ❖ `adb logcat | grep -i displayed`
- ❖ `aapt dump badging mobike.apk | grep launchable-activity`
- ❖ apkalyzer 最新版本的sdk中才有

❖ 启动应用

- ❖ `adb shell am start -W -n com.xueqiu.android/.view.WelcomeActivityAlias -S`



配置待测应用

Automatic Server

Custom Server

SAUCE LABS

TestObject
A PART OF SAUCE LABS

headspin

BrowserStack

Will use currently-running Appium Desktop server at **http://localhost:4723**

> Advanced Settings

Desired Capabilities

Saved Capability Sets (20)

Attach to Session...

platformName	text	android	
deviceName	text	hogwarts	
automationName	text	uiautomator2	
appPackage	text	com.xueqiu.android	
appActivity	text	.view.WelcomeActivity	
+			

JSON Representation

```
{
  "platformName": "android",
  "deviceName": "hogwarts",
  "automationName": "uiautomator2",
  "appPackage": "com.xueqiu.android",
  "appActivity": ".view.WelcomeActivityAlias"
}
```

[Desired Capabilities Documentation](#)

Save As...

Start Session



利用Appium Desktop生成用例模板

The screenshot displays the Appium Desktop interface. On the left, a mobile application is running, showing a login screen with fields for '账号' (Account) and '密码' (Password), and buttons for '注册' (Register), '登录' (Login), and '忘记密码' (Forgot Password). The '登录' button is highlighted. The Recorder tool is active, showing a list of actions: Select, Swipe, and Tap. The 'Recorder' tab is selected, and a dropdown menu is open, showing the selected language as 'Python'. The 'App Source' tab is also visible, showing the XML structure of the app's UI. The 'Selected Element' tab on the right shows the selected element's details, including its ID, XPath, and Attributes.

Recorder

Python

```
id/login_password")
el2 = driver.find_element_by_id(login_account")
el2.send_keys("15600534760")
el3 = driver.find_element_by_id(login_password")
```

App Source

```
<android.widget.FrameLayout>
  <android.view.View resource-id="android:id/action_bar_overlay_layout">
    <android.widget.FrameLayout resource-id="android:id/content">
      <android.widget.LinearLayout>
        <android.widget.FrameLayout resource-id="com.xueqiu.android:id/sky_cc">
          <android.widget.LinearLayout resource-id="com.xueqiu.android:id/login_r">
            <android.widget.LinearLayout>
              <android.widget.TextView>
                <android.widget.EditText resource-id="com.xueqiu.android:id/login">
```

Selected Element

Tap Send Keys Clear

Find By	Selector
id	com.xueqiu.android:id/button_next
xpath	/hierarchy/android.widget.FrameLayout/android.view.View/android.widget.FrameLayout/android.widget.LinearLayout/android.widget.TextView

Attribute	Value
index	3



desktop主要功能

- ❖ UI分析
- ❖ 录制用例
- ❖ 元素查找测试
- ❖ Attach已有的session
- ❖ 云测试



入门篇： 自动化用例编写

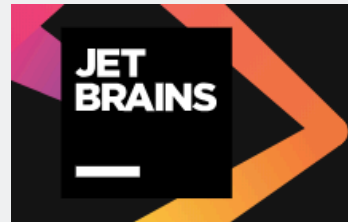


appium客户端安装

Language/Framework	Github Repo and Installation Instructions
Ruby	https://github.com/appium/ruby_lib
Python	https://github.com/appium/python-client
Java	https://github.com/appium/java-client
JavaScript (Node.js)	https://github.com/admc/wd
Objective C	https://github.com/appium/selenium-objective-c
PHP	https://github.com/appium/php-client
C# (.NET)	https://github.com/appium/appium-dotnet-driver
RobotFramework	https://github.com/jollychang/robotframework-appiumlibrary



IDE环境



Tools

Languages

Solutions



Find a tool for you

Whichever technologies you use there's a JetBrains tool to match

[find a tool](#)

IDES

AppCode

CLion

DataGrip

GoLand

IntelliJ IDEA

PhpStorm

PyCharm

Rider

RubyMine

WebStorm

.NET & VISUAL STUDIO

ReSharper

Rider

ReSharper C++

dotCover

dotMemory

dotPeek

dotTrace



java客户端安装

- ❖ IDE推荐：IntelliJ IDEA、Eclipse
- ❖ 配置maven项目
- ❖ 在src/test下编写用例

```
<dependency>
  <groupId>io.appium</groupId>
  <artifactId>java-client</artifactId>
  <version>${version.you.require}</version>
  <scope>test</scope>
</dependency>
```





python客户端安装

- ❖ 客户端安装: `pip install Appium-Python-Client`
- ❖ IDE: PyCharm
- ❖ Python多版本隔离工具: VEnv
- ❖ 国内依赖源: <http://pypi.douban.com/simple/>
- ❖ 测试框架: unittest、pytest、nose



各语言代码示例

❖


[appium](#) / [sample-code](#)

 Watch ▼

197

 Star

1,121

 Fork

1,898

<> Code

 Issues 48

 Pull requests 5

 Projects 0

 Wiki

 Insights

Branch: master ▼

[sample-code](#) / [sample-code](#) / [examples](#) /

Create new file

Upload files

Find file

History



padmakarsrinivas Changed Indentation and set it to 2 from 4.

Latest commit 44776f4 3 days ago

..

 C#/CalculatorTest	Adding Windows samples to Appium sample code repo (#97)	2 years ago
 RobotFramework	Add sample code for RobotFramework with WinAppDriver, based on https:...	a year ago
 dotnet	Fix broken Markdown headings (#116)	a year ago
 java	Changed Indentation and set it to 2 from 4.	3 days ago
 node	SPDX-compliant license in package.json	7 months ago
 perl	update the other file's app path	7 months ago
 php	update the other file's app path	7 months ago
 python	update the other file's app path	7 months ago
 ruby	Update appium.txt	7 months ago

❖



Android 自动化测试用例编写



Python案例

```
class SimpleAndroidTests(unittest.TestCase):
    def setUp(self):
        desired_caps = {}
        desired_caps['platformName'] = 'Android'
        desired_caps['platformVersion'] = '4.2'
        desired_caps['deviceName'] = 'Android Emulator'
        desired_caps['app'] = PATH(
            '../../../sample-code/apps/ApiDemos/bin/ApiDemos-debug.apk'
        )

        self.driver = webdriver.Remote('http://localhost:4723/wd/hub', desired_caps)

    def tearDown(self):
        # end the session
        self.driver.quit()

    def test_find_elements(self):
        el = self.driver.find_element_by_accessibility_id('Graphics')
        el.click()
        el = self.driver.find_element_by_accessibility_id('Arcs')
        self.assertIsNotNone(el)

        self.driver.back()

        el = self.driver.find_element_by_accessibility_id("App")
        self.assertIsNotNone(el)

        els = self.driver.find_elements_by_android_uiautomator("new UiSelector().clickable(true)")
        self.assertGreaterEqual(12, len(els))

        self.driver.find_element_by_android_uiautomator('text("API Demos")')
```

```
import os
from time import sleep
```

```
import unittest
```

```
from appium import webdriver
```

```
# Returns abs path relative to this file and not cwd
PATH = lambda p: os.path.abspath(
    os.path.join(os.path.dirname(__file__), p)
)
```

```
if __name__ == '__main__':
    suite = unittest.TestLoader().loadTestsFromTestCase(SimpleAndroidTests)
    unittest.TextTestRunner(verbosity=2).run(suite)
```




Java案例

@Before

```
public void setUp() throws Exception {  
    // set up appium  
    File appDir = new File(System.getProperty("user.dir"), "../../apps/TestApp/build/release-iphonesimulator");  
    File app = new File(appDir, "TestApp.app");  
    DesiredCapabilities capabilities = new DesiredCapabilities();  
    capabilities.setCapability("platformVersion", "9.3");  
    capabilities.setCapability("deviceName", "iPhone 6");  
    capabilities.setCapability("app", app.getAbsolutePath());  
    driver = new IOSDriver<WebElement>(new URL("http://127.0.0.1:4723/wd/hub"), capabilities);  
    values = new ArrayList<Integer>();  
}
```

@After

```
public void tearDown() throws Exception {  
    driver.quit();  
}
```

```
import static org.junit.Assert.assertEquals;  
import static org.junit.Assert.assertTrue;  
import io.appium.java_client.AppiumDriver;  
import io.appium.java_client.MobileBy;  
import io.appium.java_client.ios.IOSDriver;
```

@Test

```
public void testUIComputation() throws Exception {  
    // populate text fields with values  
    populate();  
    // trigger computation by using the button  
    WebElement button = driver.findElement(By.className("UIButton"));  
    button.click();  
    // is sum equal ?  
    WebElement texts = driver.findElement(By.className("UIStaticText"));  
    assertEquals(String.valueOf(values.get(0) + values.get(1)), texts.getText());  
}
```



测试用例的重要部分

- ❖ 导入依赖:
 - ❖ Python from import
 - ❖ Java import
- ❖ capabilities 设置
 - ❖ Python dict
 - ❖ Java Class
- ❖ 初始化driver:
 - ❖ Python webdriver.remote
 - ❖ Java new AppiumDriver
- ❖ 元素定位与操作 find+action
- ❖ 断言 assert



capabilities设置

- ❖ app apk地址
- ❖ appPackage 包名
- ❖ appActivity Activity名字
- ❖ automationName 默认使用uiautomator
- ❖ noReset fullReset 是否在测试前后重置相关环境
- ❖ unicodeKeyBoard resetKeyBoard 是否需要输入非英文之外的语言并在测试完成后重置输入法



capabilities示例代码

```
# Android environment
import unittest
from appium import webdriver
```

```
desired_caps = {}
desired_caps['platformName'] = 'Android'
desired_caps['platformVersion'] = '4.2'
desired_caps['deviceName'] = 'Android Emulator'
desired_caps['app'] = PATH('../..../apps/selendroid-test-app.apk')
```

```
self.driver = webdriver.Remote('http://localhost:4723/wd/hub', desired_caps)
```

```
import java.io.File;
import org.openqa.selenium.remote.DesiredCapabilities;
import io.appium.java_client.AppiumDriver;
import io.appium.java_client.android.AndroidDriver;
import io.appium.java_client.MobileElement;
import java.net.URL;

...
File app = new File("The absolute or relative path to an *.apk file");
DesiredCapabilities capabilities = new DesiredCapabilities();
capabilities.setCapability(MobileCapabilityType.DEVICE_NAME, "Android Emulator");
capabilities.setCapability(MobileCapabilityType.APP, app.getAbsolutePath());
capabilities.setCapability(MobileCapabilityType.PLATFORM_NAME, MobilePlatform.ANDROID);
//you are free to set additional capabilities
AppiumDriver<MobileElement> driver = new AppiumDriver<>(
new URL("http://target_ip:used_port/wd/hub"), //if it needs to use locally started server
//then the target_ip is 127.0.0.1 or 0.0.0.0
//the default port is 4723
capabilities);
```



控件定位与交互



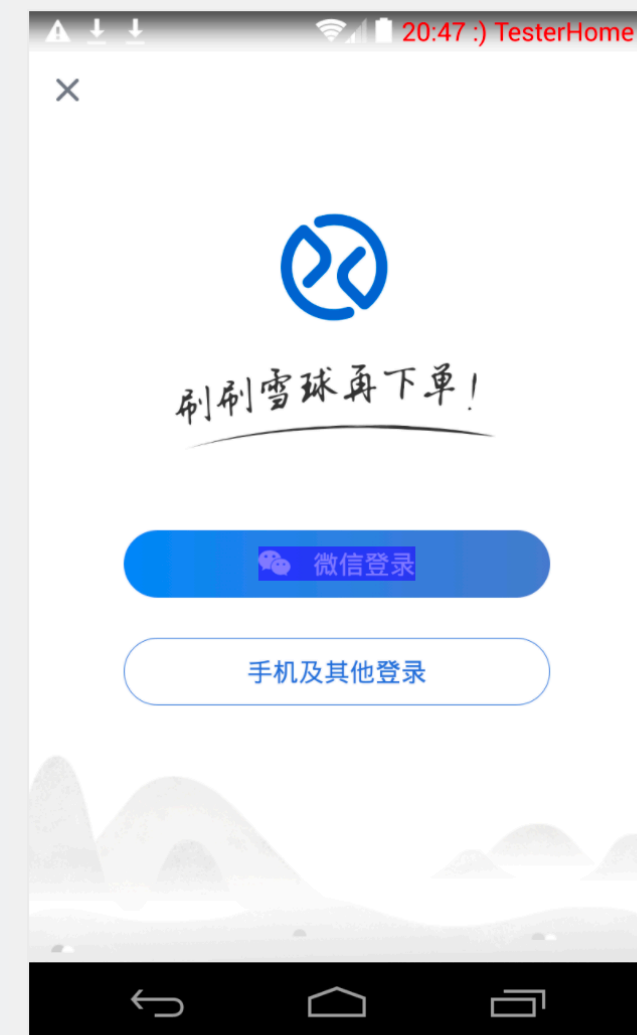
控件基础知识

- ❖ dom: Document Object Model 文档对象模型
- ❖ dom应用: 最早应用于html和js的交互。界面的结构化描述, 常见的格式为html、xml。核心元素为节点和属性
- ❖ xpath: xml路径语言, 用于xml中的节点定位





app dom结构解析



App Source

```
<android.widget.FrameLayout>
  <android.widget.LinearLayout>
    <android.widget.FrameLayout>
      <android.view.View resource-id="com.xueqiu.android:id/decor
        <android.widget.FrameLayout resource-id="android:id/cont
          <android.view.View resource-id="com.xueqiu.android:id,
        <android.widget.RelativeLayout>
          <android.widget.ImageView resource-id="com.xueqiu
          <android.widget.ImageView resource-id="com.xueqiu
          <android.widget.RelativeLayout resource-id="com.xu
            <android.widget.TextView resource-id="com.xuec
          <android.widget.TextView resource-id="com.xueqiu.
          <android.widget.ImageView>
```

Selected Element

Attribute	Value
index	0
text	微信登录
class	android.widget.TextView
package	com.xueqiu.android
content-desc	
checkable	false
checked	false
clickable	false
enabled	true
focusable	false
focused	false
scrollable	false
long-clickable	false
password	false
selected	false
bounds	[286,666][482,709]
resource-id	com.xueqiu.android:id/tv_login_by_wx
instance	0



app dom为例

- ❖ node
- ❖ attribute
 - ❖ clickable
 - ❖ content-desc
 - ❖ resource-id
 - ❖ text
 - ❖ bounds
- ❖ iOS与Android的区别
 - ❖ dom属性和节点结构类似
 - ❖ 名字和属性的命名不同

```
<android.widget.LinearLayout
  bounds="[198,65][656,131]" checkable="false"
  checked="false"
  class="android.widget.LinearLayout"
  clickable="true" content-desc="" enabled="true"
  focusable="false" focused="false" index="2"
  instance="3" long-clickable="false"
  package="com.xueqiu.android" password="false"
  resource-id="com.xueqiu.android:id/home_search"
  scrollable="false" selected="false" text="">
  <android.widget.ImageView
    bounds="[212,80][248,116]" checkable="false"
    checked="false" class="android.widget.ImageView"
    clickable="false" content-desc="" enabled="true"
    focusable="false" focused="false" index="0"
    instance="3" long-clickable="false"
    package="com.xueqiu.android" password="false"
    resource-id="" scrollable="false"
    selected="false" text=""/>
  <android.widget.TextView
    bounds="[256,79][453,117]" checkable="false"
    checked="false" class="android.widget.TextView"
    clickable="false" content-desc="" enabled="true"
    focusable="false" focused="false" index="1"
    instance="0" long-clickable="false"
    package="com.xueqiu.android" password="false"
    resource-id="com.xueqiu.android:id/tv_search"
    scrollable="false" selected="false" text="雪球Club南京站"/>
</android.widget.LinearLayout>
<android.widget.FrameLayout NAF="true"
  bounds="[680,50][744,146]" checkable="false"
  checked="false" class="android.widget.FrameLayout"
  clickable="true" content-desc="" enabled="true"
```



元素定位

❖ 测试步骤三要素：

❖ 定位、交互、断言

❖ 定位

❖ id (重要)

❖ xpath (重要)

❖ accessibilityId: content-desc

❖ 不推荐：class -ios -android

```
private Object findElement(By by) throws InvalidSelectorException, ElementNotFoundException, ParserConfigurationException,
    if (by instanceof ById) {
        String locator = getElementLocator((ById)by);
        return getInstance().findObject(android.support.test.uiautomator.By.res(locator));
    } else if (by instanceof By.ByAccessibilityId) {
        return getInstance().findObject(android.support.test.uiautomator.By.desc(by.getElementLocator()));
    } else if (by instanceof ByClass) {
        return getInstance().findObject(android.support.test.uiautomator.By.class(by.getElementLocator()));
    } else if (by instanceof By.ByXPath) {
        return getXPathUiObject(by.getElementLocator(), null /* AndroidElement */);
    } else if (by instanceof By.ByAndroidUiAutomator) {
        return getInstance().findObject(findByUiAutomator(by.getElementLocator()));
    }
    String msg = String.format("By locator %s is currently not supported!", by.getClass().getSimpleName());
    throw new UnsupportedOperationException(msg);
}
```

```
switch (strategy) {
    case XPATH:
        try {
            selectors.addAll(getXPathSelectors(text, many, contextId));
        } catch (final ElementNotFoundException ignore) {
        }
        break;
    case CLASS_NAME:
        sel = sel.className(text);
        if (!many) {
            sel = sel.instance(0);
        }
        selectors.add(sel);
        break;
    case ID:
        // There are three types of ids on Android.
        // 1. resourceId (API >= 18)
        // 2. accessibility id (content description)
        // 3. strings.xml id
        //
        // If text is a resource id then only use the resource id selector.
        if (API_18) {
            if (resourceIdRegex.matcher(text).matches()) {
                sel = sel.resourceId(text);
            }
        }
    }
}
```



uiautomator1的定位逻辑

```
switch (strategy) {  
    case XPATH:  
        try {  
            selectors.addAll(getXPathSelectors(text, many, contextId));  
        } catch (final ElementNotFoundException ignore) {}  
        break;  
    case CLASS_NAME:  
        sel = sel.className(text);  
        if (!many) {  
            sel = sel.instance(0);  
        }  
        selectors.add(sel);  
        break;  
    case ID:  
        // There are three types of ids on Android.  
        // 1. resourceId (API >= 18)  
        // 2. accessibility id (content description)  
        // 3. strings.xml id  
        //  
        // If text is a resource id then only use the resource id selector.  
        if (API_18) {  
            if (resourceIdRegex.matcher(text).matches()) {  
                sel = sel.resourceId(text);  
            }  
        }  
        selectors.add(sel);  
        break;  
}
```



uiautomator2的定位逻辑

```
private Object findElement(By by) throws InvalidSelectorException, ElementNotFoundException, ParserConfigurationException,
    if (by instanceof ById) {
        String locator = getElementLocator((ById)by);
        return getInstance().findObject(android.support.test.uiautomator.By.res(locator));
    } else if (by instanceof By.ByAccessibilityId) {
        return getInstance().findObject(android.support.test.uiautomator.By.desc(by.getElementLocator()));
    } else if (by instanceof ByClass) {
        return getInstance().findObject(android.support.test.uiautomator.By.classz(by.getElementLocator()));
    } else if (by instanceof By.ByXPath) {
        return getXPathUiObject(by.getElementLocator(), null /* AndroidElement */);
    } else if (by instanceof By.ByAndroidUiAutomator) {
        return getInstance().findObject(findByUiAutomator(by.getElementLocator()));
    }
    String msg = String.format("By locator %s is currently not supported!", by.getClass().getSimpleName());
    throw new UnsupportedOperationException(msg);
}
```



定位与操作的代码示例

```
def test_simple_actions(self):
    el = self.driver.find_element_by_accessibility_id('Graphics')
    el.click()

    el = self.driver.find_element_by_accessibility_id('Arcs')
    el.click()

    self.driver.find_element_by_android_uiautomator('new UiSelector().text("Graphics/Arcs")')
```

```
@Test
public void apiDemo(){
    WebElement el = driver.findElement(By.xpath(".*[@text='Animation']"));
    assertEquals("Animation", el.getText());
    el = driver.findElementByClassName("android.widget.TextView");
    assertEquals("API Demos", el.getText());
    el = driver.findElement(By.xpath(".*[@text='App']"));
    el.click();
    List<WebElement> els = driver.findElementsByClassName("android.widget.TextView");
    assertEquals("Activity", els.get(2).getText());
}
```




wait等待

- ❖ 精确等待: Sleep 不推荐
- ❖ 全局隐式等待
 - ❖ `driver.manage().timeouts().implicitlyWait(10, TimeUnit.SECONDS);`
 - ❖ 默认值是0, 也就是不等待
- ❖ 显式等待
 - ❖ `WebDriverWait wait = new WebDriverWait(driver, 10);`
 - ❖ `WebElement element = wait.until(ExpectedConditions.elementToBeClickable(By.id("someid")));`





常见自动化动作支持

- ❖ click
- ❖ sendKeys
- ❖ sendKeyEvent

```
# python  
driver.swipe(start_x=75, start_y=500, end_x=75, end_y=0, duration=800)
```

```
// java  
driver.swipe(75, 500, 75, 0, 0.8)
```





手势操作

❖ press

```
TouchAction().press(el0).moveTo(el1).release()
```

❖ release

❖ moveTo

❖ tap wait

❖ longPress

❖ cancel

❖ perform

In pseudocode, both of the following are equivalent:

```
TouchAction().tap(el).perform()
```

```
driver.perform(TouchAction().tap(el))
```

```
action0 = TouchAction().tap(el)
```

```
action1 = TouchAction().tap(el)
```

```
MultiAction().add(action0).add(action1).perform()
```



常用功能

- ❖ <https://github.com/appium/appium/blob/master/docs/en/writing-running-appium/appium-bindings.md>
- ❖ 系统操作
 - ❖ lock background hideKeyBoard openNotifications shake
 - ❖ startActivity currentActivity getCurrentPackage
- ❖ app操作
 - ❖ installApp removeApp isInstalled closeApp launchApp reset getAppStrings
 - ❖ getContextHandles getContext context



演练app

- ❖ <https://github.com/appium/sample-code/raw/master/sample-code/apps/ApiDemos/bin/ApiDemos-debug.apk>





调试分析方法

- ❖ Appium Log
 - ❖ 清晰记录了所有的请求和结果
- ❖ getPageSource
 - ❖ 界面的完整dom结构.xml文件
- ❖ 脚本内调试
 - ❖ 利用xpath获取所有匹配的元素
 - ❖ `driver.findElementsByXPath("// /*")`
- ❖



Q&A

- ❖ 欢迎关注TesterHome测试社区
- ❖ 技术交流请上 testerhome.com
@seveniruby 即可
- ❖ 加好友请注明姓名和公司



黄延胜[思寒] 
中国 北京



扫一扫上面的二维码图案，加我微信