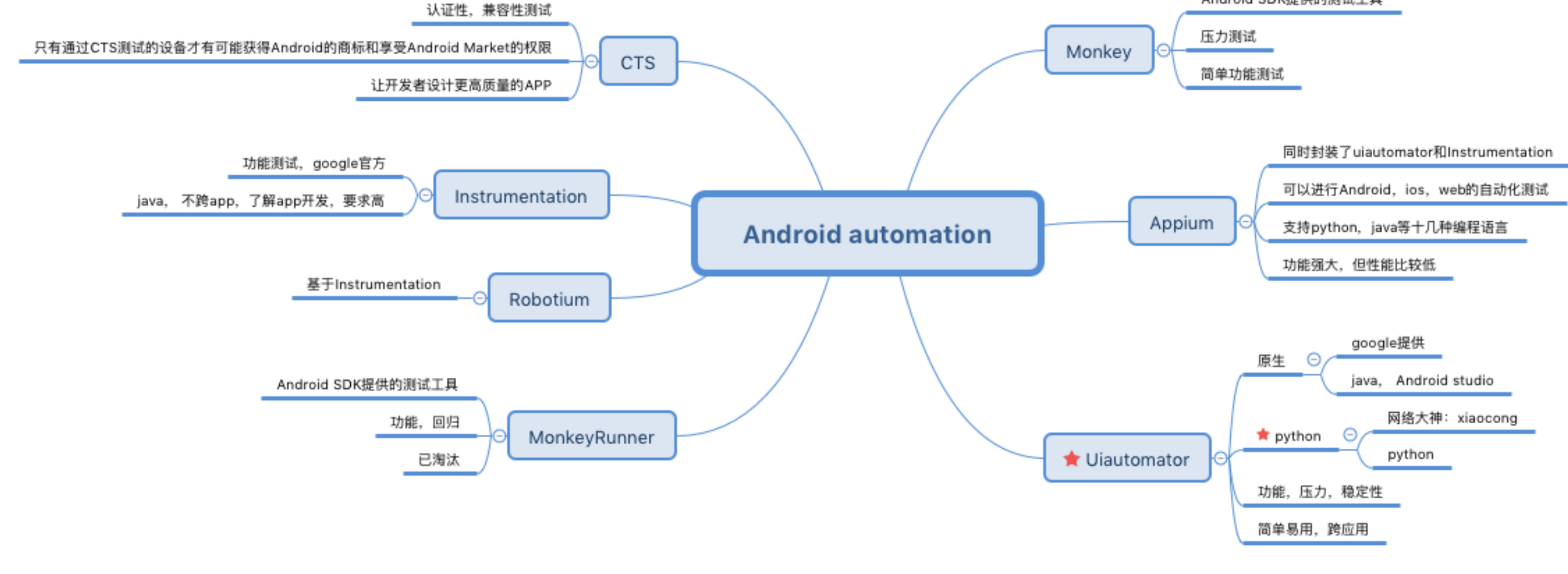
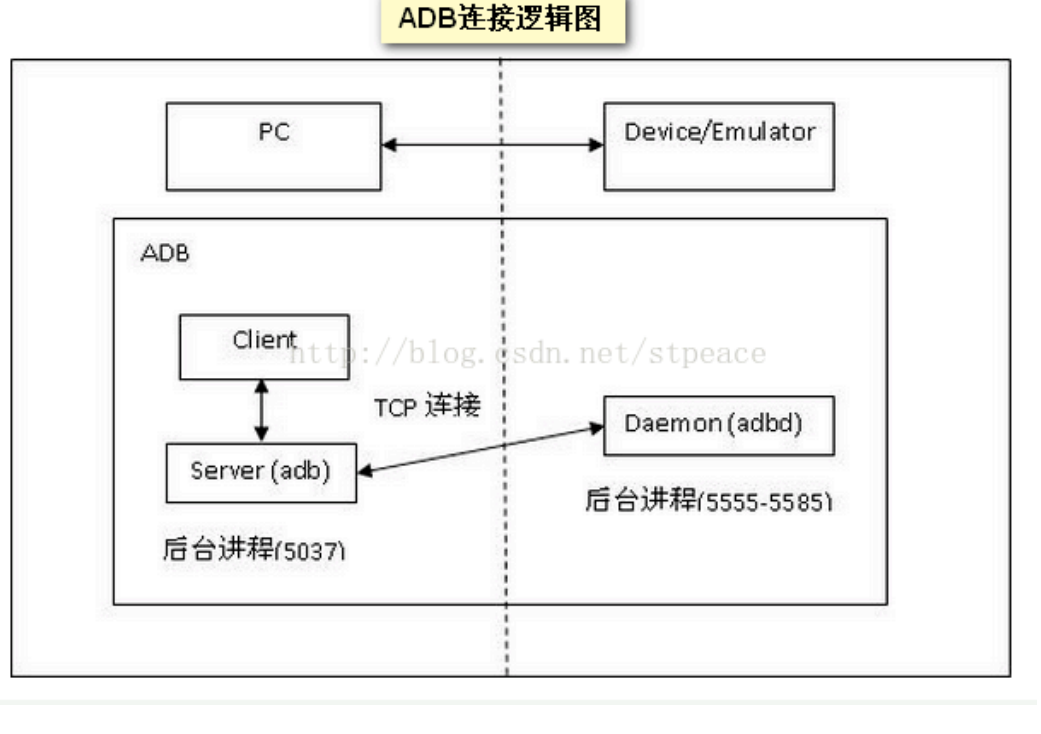


Android 自动化框架



ADB

- 1 概念: **Android Debug Bridge**
- 2 工作原理



Adb命令

help 帮助信息

version 版本信息

服务启停

adb start-server 启动adb server

adb kill-server 停止adb server

设备相关

adb devices 显示所有设备（格式为 串号和状态）

• -l 显示qualifiers (如: emulator-5554 device product:sdk_google_phone_x86 model:Android_SDK_built_for_x86 device:generic_x86)

adb wait-for-device device online前一直阻塞，可以作为不需要系统完全启动就能执行的命令的前缀使用:如 adb wait-for-device shell getprop

adb get-state

adb get-serialno

adb get-devpath

adb status-window 持续显示设备状态（有变化时更新）

adb connect <host>[:<port>] 通过TCP网络连接设备，即上图中的方式一。（默认端口：5555）

adb disconnect <host>[:<port>] 断开通过网络连接的设备

adb root 以root权限重启设备上的adb服务

adb usb 重启adb服务，监听USB

adb tcpip <port> 重启adb服务，监听指定的TCP端口

adb指定目标设备选项（可以没有）

-a 所有设备

-d USB连接的设备

-e 模拟器设备

-s <Serial-No> 按串号指定设备

-H <server_host> 指定连接的adb server主机

-P <server_port> 指定连接的adb server端口

adb reboot 重启设备

文件传送

adb pull <path-dev> <path-pc> 把设备中的文件或目录拷贝到PC端

• -p 显示进度

adb push <path-pc> <path-dev> 把PC端文件或目录拷贝到设备

• -p 显示进度

• -a 拷贝时间戳和模式

adb sync [<directory>] 把PC端改动的内容拷贝到设备（默认为/system,/vendor,/data）

• -l 只列出，不拷贝

adb install <apkfile-pc> 安装应用

• -r 替换

• -s 安装到SD卡

可能的错误信息：

adb uninstall <package-name> 移除应用

• -k 保留数据和缓存目录

调用Android shell命令

adb shell 进入shell环境

adb shell [shell-cmd] 执行shell命令

android shell命令总结: <http://www.cnblogs.com/pixy/p/4741532.html>

网络相关

adb forward <spec-local> <spec-remote> socket定向绑定：

spec格式为：

tcp:<port>

localabstract:<unix domain socket name>

localreserved:<unix domain socket name>

localfilesystem:<unix domain socket name>

dev:<character device name>

jdwp:<process pid> (remote only)

--no-rebind 如果local端已经绑定则失败

adb forward --remove <spec-local> 移除指定的定向绑定

adb forward --remove-all

adb reverse --list 显示所有从设备发出的反向socket连接

adb reverse <spec-remote> <spec-local> 反向socket绑定

spec格式为：

tcp:<port>

localabstract:<unix domain socket name>

localreserved:<unix domain socket name>

localfilesystem:<unix domain socket name>

--norebind <spec-remote>已经绑定则失败

adb reverse --remove <spec-remote> 移除反向绑定

adb reverse --remove-all 移除所有反向绑定

adb ppp <tty> [parm] 通过USB运行PPP

logcat

adb logcat [option] [FILTER-SPECS] 输出日志数据。可以重定向到文件: adb logcat>d:\logcat.txt

options:

-s	沉默默认的过滤器，以下两种方式等效 logcat -s tagName:I logcat tagName:I *:s
-v <format>	指定输出格式，默认为
-f <file>	Log to file
-c	clear(flush) 所有log，然后退出
-d	dump log，退出
-g	获取log循环缓冲大小，退出
-b <buffer>	请求其他循环缓冲，main(default)，radio，events
-B	output log in binary

FILTER-SPECS: tagName:[PRIORITY]...

PRIORITY:

V	Verbose
D	Debug
I	Info
W	Warn
E	Error
F	Fatal
S	Silent (supress all output)

tagName和priority都可以用*作为通配符，logcat命令中必须使用-s选项或加*:s才能过滤掉无关的log

调试相关

adb bugreport 显示dumpsys, dumpstate, logcat信息

adb jdwp 显示有效地jdwp (java debug wire protocol) 进程

其他

adb emu <emu_command> 模拟器控制台命令

adb backup 备份系统数据到文件

- -f <file> 指定文件名（省确为backup.ab）
- -apk | -noapk 是否备份.apk（默认为-noapk）
- -obb | -noobb 是否备份apk扩展文件.obb，默认为noobb
- -shared | -noshared 是否备份Shared Storeage,SD 卡内容（默认为noshared）
- -all 备份所有安装的应用
- -system | -nosystem -all是否包括系统应用。（默认为system）
- [<packages...>] 要备份的应用列表。如果有-all或-shared，可以没有此项。

adb restore <file> 从备份文件恢复数据。

adb disable-verity

adb keygen <file> 生成密钥对，<file>文件中存放私钥，<file>.pub中存放公钥。已存在文件会被覆盖。

Python

- 1 简介： 简单易学强大
- 2 与java的比较
- 3 学习建议

自我定位： 测试， 自动化测试， web开发， 爬虫开发， 大数据开发， 人工智能， 【基础】

职业规划

学习程度