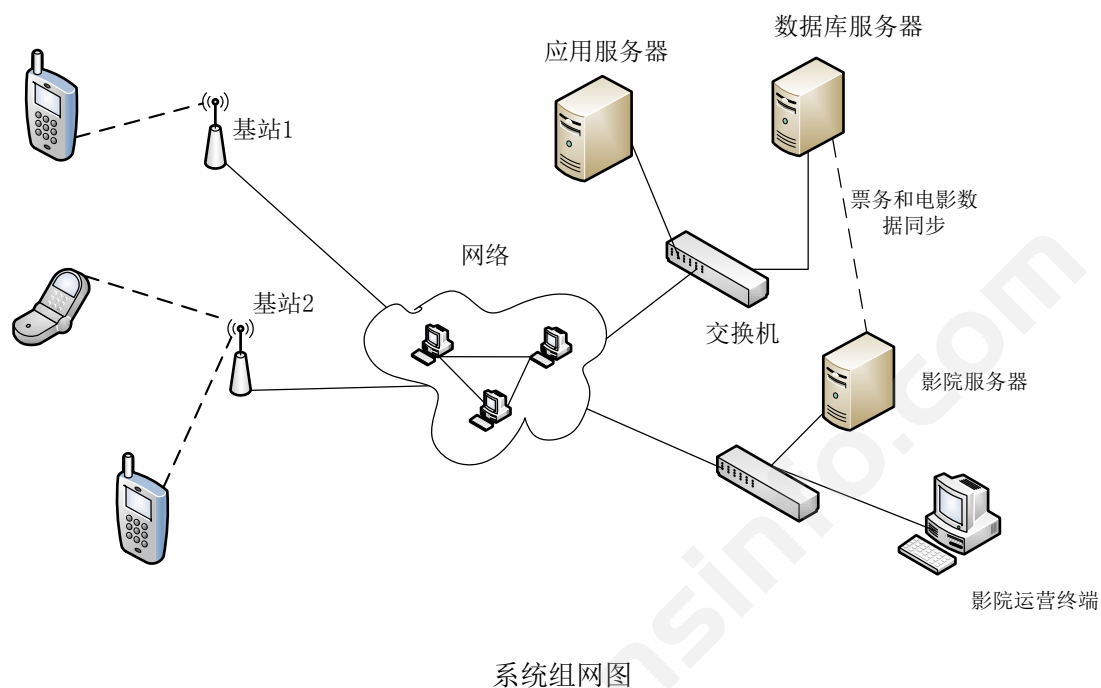


APP 测试工具和命令操作指南

1、常见的 APP 的组网图



2、SDK 介绍

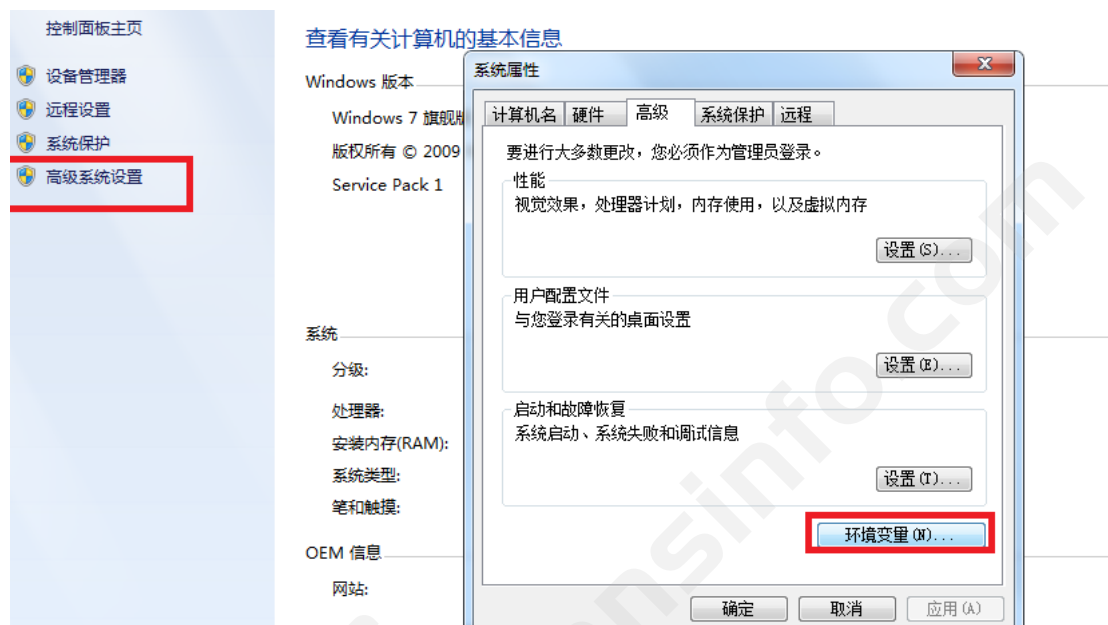
Android SDK (Software Development Kit, 软件开发工具包) 被软件开发工程师用于为特定的软件包、软件框架、硬件平台、操作系统等建立应用软件的开发工具的集合。它提供了 Android API 库和开发工具构建, 测试和调试应用程序。简单来讲, Android SDK 可以看做用于开发和运行 Android 应用的一个软件。

2.1 sdk 环境变量配置（前提 jdk 安装正常）

Step1 解压 SDK 包

Step2 设置环境变量

在桌面上右键点击“计算机”图标，选择“属性”，选择高级系统设置—>环境变量



其中，变量值 F:\soft\android_sdk 修改为本机 sdk 包解压的目录；

Step3 在 path 添加路径



%ANDROID_HOME%\platform-tools;

Step4 检查 sdk 环境是否正常

在 cmd 下输入命令

adb

如果可以正常显示命令，则说明配置成功

3、手机连接电脑

3.1 安卓手机通过数据线连接到电脑，并且打开开发者调试模式（在手机设置里，因手机型号而位置不同，详细可以通过百度自己手机型的开发者模式）；

手机是否连接上电脑，可以通过 adb devices 来查看，如果显示 no devices attached，则表示没有连接上，解决办法：建议在电脑和手机端各安装 360 手机助手

如果手机连接不上电脑，还可以通过手机模拟器来模拟手机，安装夜神模拟器在电脑上，打开 cmd，输入连接命令 adb connect ip:端口号，操作如下图：

```
C:\Users\Administrator>adb connect 127.0.0.1:62001
adb server is out of date. killing...
* daemon started successfully *
connected to 127.0.0.1:62001

C:\Users\Administrator>adb devices
List of devices attached
127.0.0.1:62001 device

C:\Users\Administrator>
```

表示连接模拟器成功，一样可以使用各种 adb 命令

4 常见的 adb 命令

ADB(Android Debug Bridge) 工具，我们可以管理设备或手机模拟器的状态。还可以进行很多手机操作，如安装软件、系统升级、运行 shell 命令等等。其实简而言说，就是连接 Android 手机与 PC 端的桥梁，可以让用户在电脑上对手机进行全面的操作。

1、查看连接的设备 adb devices

该命令可以查看连接的 adb 的设备的情况，如果连接多个设备，均显示在结果中

2、查看设备连接状态 adb get-state

结果可能有三种 offline 设备连接异常，device 正常 ,unknown 没有连接设备

3、查看日志 adb logcat

查看手机日志，ctrl+c 结束

可查看手机四个缓冲区日志，包括 radio, system,main,event，其中 radio 存放通信的日志，比如通话短信等；event 是表示手机按键输出等事件日志等；main 指的是手机的应用软件的日志，是应用层日志输出；system 是系统组件日志

如果使用 logcat 不指定输出哪个缓冲区的日志，默认输出 main 和 system 的日志，如果需要指定，则使用

adb logcat -b 指定缓冲区如 adb logcat -b main

也可以输出到文件 adb logcat -b main > d:\main.log

-c 清理日志

logcat 是实时日志，即在输出日志过程中，如果你操作软件，同时也会输出日志

4 adb bugreport

输出日志包括 logcat 日志，同时也包括功耗，cpu 等信息日志

5、安装 app 应用

adb install d:/xxx.apk

其中 xxx.apk 是安装包名

6、卸载 app 应用

adb uninstall [-k] <packagename>

packagename 是应用包名，该包名是在开发过程中确定的，可以通过命令查看要卸载的应用包名，也可以从开发人员处获取；

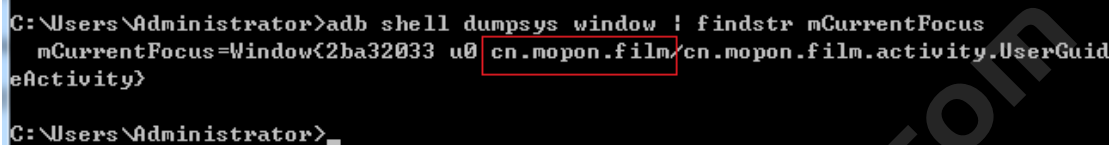
adb shell pm clear <packagename>

7 查看应用列表

adb shell pm list packages 查看所有的应用包名

adb shell dumpsys activity | find "mFocusActivity" 查看当前活动的包名

adb shell dumpsys window | findstr mCurrentFocus 查看正在运行的包名



```
C:\Users\Administrator>adb shell dumpsys window | findstr mCurrentFocus
mCurrentFocus=Window{2ba32033 u0 cn.mopon.film/cn.mopon.film.activity.UserGuideActivity}
C:\Users\Administrator>
```

8、获取 cpu 信息

adb shell cat /proc/cpuinfo 获取 cpu 信息（processor 的编号是 0 到 3，表 4 核，采用的架构是 ARMv7 Processor rev 0 (v71)）

9 手机电脑相互拷贝文件

adb push d:\pushlog /dev/log 从电脑上推送到手机

adb pull /dev/log/error.log d:\ 从手机上下载文件到电脑

10 adb root 进入 root 模式

11 adb shell 进入 shell 命令状态

Android 是基于 linux 内核，因此在 shell 状态下可以使用 linux 的命令

查看分辨率 adb shell wm size

输出电池信息 adb shell dumpsys battery

adb shell dumpsys batterystats cn.mopon.film 获取 cn.monpon.film 应用的耗电情况

adb shell getprop ro.build.version.release 获取系统版本号

12 adb kill-server

adb start-server

重启设备

12、截屏

```
adb shell screencap -p /sdcard/sc.png
```

```
adb pull /sdcard/sc.png
```

 把图片传到电脑

13、查看进程

```
adb shell ps
```

14、查看连接过的 WiFi 信息

```
adb shell
```

```
su
```

```
cat /data/misc/wifi/*.conf
```

注：需要 root 权限

15、获取 mac 地址（物理地址）

```
adb shell cat /sys/class/net/wlan0/address
```

16、查看当前内存占用

```
adb shell cat /proc/meminfo
```

16、重启手机

```
adb reboot
```

17、查看 adb 版本

```
adb version
```

18、清除应用数据与缓存

```
adb shell dumpsys activity activities | grep mFocusedActivity
```

```
adb shell dumpsys activity activities | ps mFocusedActivity
```

19、强制停止应用

```
adb shell am force-stop <packagename>
```

```
adb push <电脑上的文件路径> <设备里的目录>
```

20、查看 cpu 运行情况

```
adb shell top -m 10 -s cpu
```

（-t 显示进程名称，-s 按指定行排序，-n 在退出前刷新几次，-d 刷新间隔，-m 显示最大数量）

21 windows 下查看端口号

更多技术干货请关注公众号：川石信息或者知乎号：Foreknew

唯一官方网站：www.chuansinfo.com

```
netstat -ano | findstr "5037"
```

Linux 命令状态下查看 `netstat -anp | grep 5037`

22 `adb shell am start -n` 包名/入口

启动 app

23 `adb shell pm clear` 包名

清除应用的数据和缓存

24 `adb shell input tap x 轴坐标 y 轴坐标`

坐标点击

5 常见错误及解决办法:

在输入 adb 命令时, 显示: more than one device and emulator

解决办法:

查看设备 `adb devices`

```
C:\Users\administrator>adb devices
```

```
List of devices attached
```

```
emulator-5554 device
```

```
4dfadcb86b00cf05 device
```

发现还真是多个设备, 那就需要为 ADB 命令指定设备的序列号了。

```
C:\Users\administrator>adb -s emulator-5554 shell
```

也就是如上所示, 给命令加上-s 的参数就可以了!

如果实际上只有一个设备或模拟器, 并且查到有 `offline` 的状态;

那就说明是 ADB 本身的 BUG 所导致的, 就需要用如下的方法处理下了:

```
C:\Users\administrator>adb kill-server
```

```
C:\Users\administrator>taskkill /f /im adb.exe
```