MonkeyRunner工具使用

http://www.android-doc.com/tools/help/monkeyrunner_concepts.html

本章大纲

- MonkeyRunner介绍

- MonkeyRunner常用API

- MonkeyRunner实战

- 录制与回放

MonkeyRunner介绍

- MonkeyRunner工具是使用Jython(使用Java编程语言实现的Python)写出来的，它提供了多个API，通过monkeyrunner API 可以写一个Python的程序来模拟操作控制Android设备app，测试其稳定性并通过截屏可以方便地记录出现的问题。

- MonkeyRunner是由Google开发、用于android系统的自动化测试工具，存在于android sdk中，在PC端通过api控制设备的运行

- MonkeyRunner通过运行python脚本，来执行脚本中预先定义好的一系列操作事件，如：安装卸载某个应用、进入任一模块、点击、按键、编辑文本、截图对比等.

- 目录 AppData\Local\Android\sdk\tools，
  AppData\Local\Android\sdk\tools\lib

- http://www.android-doc.com/tools/help/monkeyrunner_concepts.html

- http://androidxref.com/4.1.2/xref/sdk/monkeyrunner/

monkeyrunner

The monkeyrunner tool provides an API for writing programs that control an Android device or emulator from outside of Android code. With monkeyrunner, you can write a Python program that installs an Android application or test package, runs it, sends keystrokes to it, takes screenshots of its user interface, and stores screenshots on the workstation. The monkeyrunner tool is primarily designed to test applications and devices at the functional/framework level and for running unit test suites, but you are free to use it for other purposes.
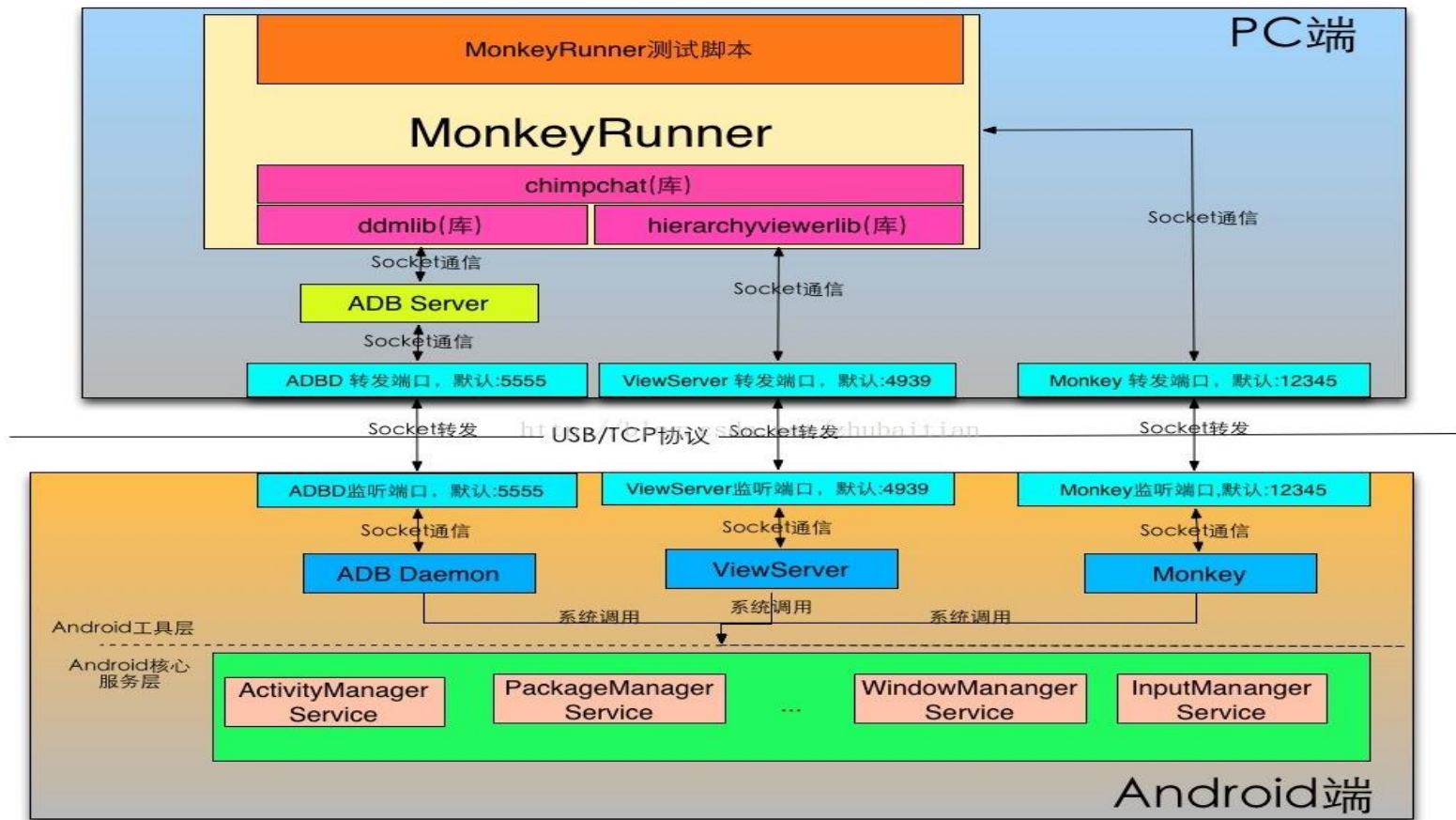
MonkeyRunner与Monkey区别

- 两者没有任何关系

- Monkey是一条命令，语言java，无断言。主要直接运行在设备或模拟器的adb shell中，生成用户的伪随机事件流。

- MonkeyRunner，存在于sdk中，通过API定义的命令和事件，控制设备，运行在PC端。

环境搭建

1. Android环境搭建（把platform-tools和tools路径添加到"path"环境变量中）
2. 安装python并配置环境变量到"path" 中
   https://www.python.org/downloads/
3. 验证：cmd输入python
4. 验证：cmd输入monkeyunner
5. 模拟器或root的真机

MonkeyRunner架构

MonkeyRunnerAPI

- MonkeyRunner 类：它是一个为MonkeyRunner 程序提供工具方法的类。这个类提供了连接MonkeyRunner至设备或模拟器的方法。它还提供了用于创建一个MonkeyRunner 程序的用户界面以及显示内置帮助的方法。

- MonkeyImage 是一个截图对象的类。这个类提供了截图、将位图转换成各种格式、比较两个对象MonkeyImage 以及写图像到文件的方法。

- MonkeyDevice 是一个设备或模拟器的类。这个类为安装和卸载包、开启Activity、发送按键和触摸事件、运行测试包等提供了方法。

MonkeyRunner常用库

- chimpchat库:chimpchat对于使用ADB进行通信的过程来说，相当于MonkeyRunner和ddmlib之间的代理中间层。但对于使用Monkey进行通信 的时候，chimpchat并不需要通过ddmlib来驱动Android目标机器的Monkey服务，而是chimpchat自己来完成socket建立和命令发送的功能

MonkeyRunner常用库

- ddmlib库:在MonkeyRunner框架中，ddmlib主要是帮忙处理跟ADB服务器通信的事宜

- hierarchyviewerlib库:当MonkeyRunner脚本需要用到控件相关的功能的时候就会使用到这个库来建立控件树和获取指定控件的相关属性

本章大纲

- MonkeyRunner介绍

- MonkeyRunner常用API

- MonkeyRunner实战

- 录制与回放

MonkeyRunner API

- `MonkeyDevice`

- `MonkeyImage`

- `MonkeyRunner`

MonkeyRunner常用API介绍

- MonkeyRunner API - alert

  - 警告框

  - void alert(string message,string title,string oktitle)

- 代码实现

  - from com.android.monkeyrunner import MonkeyRunner

  - MonkeyRunner.alert('Hello','title','OK')

MonkeyRunner常用API介绍

- MonkeyRunner API - help

  - 生成帮助文档

  - help (*string* format)

- 命令行代码实现

  - text = MonkeyRunner.help("html");
  - f = open('help.html', 'w')
  - f.write(text)
  - f.close()

MonkeyRunner常用API介绍

- MonkeyRunner API - choice

  - 选择项

  - choice ($string$ message, $iterable$ choices, $string$ title)

- 代码实现

  - MonkeyRunner.choice('please select' ,['A' , 'B'],' title')

MonkeyRunner常用API介绍

- MonkeyRunner API – input

  - 生成帮助文档

  - input ($string$ message, $string$ initialValue, $string$ title, $string$ okTitle, $string$ cancelTitle)

  - 代码实现

  - MonkeyRunner.input('message input','uname','title','OK','Cancel'),

MonkeyRunner常用API介绍

- MonkeyRunner API – waitForConnection

  – 等待设备连接，有多个device id，需要指明哪个设备。

  – waitForConnection(float timeout,string deviceid)

- 代码实现

  – device =

    MonkeyRunner.waitForConnection(3,'127.0.0.1:62007')

MonkeyDevice常用API介绍

- MonkeyDevice API - installPackage

  - 安装

  - installPackage ($string$ path)

- 代码实现

  - device.installPackage('d:\meituan.apk')

MonkeyDevice常用API介绍

- MonkeyDevice API - drag

  - 拖动

  - drag(tuple start, tuple end, float duration, integer steps)

  start 起点位置

  end 终点位置

  duration 手势持续的时间

  steps 插值点的步数，默认10

MonkeyDevice常用API介绍

- MonkeyDevice API - press

  - 按键事件

  - press (string name, dictionary type)

      keycode 值：Down、UP、DOWN_AND_UP
  - KEYCODE_ENTER 回车键
  - KEYCODE_MENU 菜单键
  - KEYCODE_HOME Home键
  - KEYCODE_BACK 返回键

- 代码实现

  - device.press('KEYCODE_ENTER','DOWN_AND_UP')

MonkeyDevice常用API介绍

- MonkeyDevice API - startActivity

  - 启动应用

  - startActivity (package+'/'+activity)

- 代码实现

  - device.startActivity(component='com.example.android.myapplication.MainActivity')

MonkeyDevice常用API介绍

- MonkeyDevice API - touch

  – 点击

  – touch (integer x, integer y, integer type)

MonkeyDevice常用API介绍

- MonkeyDevice API - type

  - 输入

  - type（string message）

MonkeyDevice常用API介绍

- MonkeyDevice API - takeSnapshot()

  - 截屏

  - MonkeyImage takeSnapshot ()

MonkeyImage常用API介绍

- MonkeyImage API - sameAs

  - 图像对比

  - boolean sameAs（ MonkeyImage otherImage, float percent ）

MonkeyRunner

- MonkeyImage API - writeToFile

  - 图像对比

  - void writeToFile (string filename, string format)

本章大纲

- MonkeyRunner介绍

- MonkeyRunner常用API

- MonkeyRunner实战

- 录制与回放

MonkeyRunner实战

1. 安装包 *.apk放到指定目录Demo下

2. 新建monkeyrunnerDemo.py

3. 在cmd中执行monkeyrunner E:\android-code\MonkeyRunnerDemo.py

4. 如果没有错误则说明成功，并在该目录下生成一个截图

5. 模拟器正常安装

```python
# -*- coding: UTF-8 -*-
from com.android.monkeyrunner import MonkeyRunner, MonkeyDevice
device = MonkeyRunner.waitForConnection()
device.installPackage('myproject/bin/MyApplication.apk')
package = 'com.example.android.myapplication'
activity = 'com.example.android.myapplication.MainActivity'
runComponent = package + '/' + activity
device.startActivity(component=runComponent)
device.press('KEYCODE_MENU', MonkeyDevice.DOWN_AND_UP)
result = device.takeSnapshot()
result.writeToFile('myproject/shot1.png','png')
```

如何获取apk的包名和入口activity

- cmd下执行如下命令

cd D:\Users\think\AppData\Local\Android\sdk\build-tools\23.0.3

- aapt dump badging E:\ ContactManager.apk

```
D:\Users\think\AppData\Local\Android\sdk\build-tools\23.0.3>aapt dump badging E:
\ContactManager.apk
package: name='com.example.android.contactmanager' versionCode='1' versionName='
1.0' platformBuildVersionName=''
sdkVersion:'5'
targetSdkVersion:'5'
uses-permission: name='android.permission.GET_ACCOUNTS'
uses-permission: name='android.permission.READ_CONTACTS'
uses-permission: name='android.permission.WRITE_CONTACTS'
application-label:'Contact Manager'
application-icon-120:'res/drawable-ldpi/icon.png'
application-icon-160:'res/drawable-mdpi/icon.png'
application-icon-240:'res/drawable-hdpi/icon.png'
application: label='Contact Manager' icon='res/drawable-mdpi/icon.png'
application-debuggable
launchable-activity: name='com.example.android.contactmanager.ContactManager' l
abel='Contact Manager' icon=''
uses-permission: name='android.permission.READ_CALL_LOG'
```

常见错误

- 报错 1：

```
E:\android-code>monkeyrunner monkeyrunnerDemo.py
Can't open specified script file
Usage: monkeyrunner [options] SCRIPT_FILE

    -s          MonkeyServer IP Address.
    -p          MonkeyServer TCP Port.
    -v          MonkeyServer Logging level <ALL, FINEST, FINER, FINE, CONFIG, INFO,
WARNING, SEVERE, OFF>
```

解决办法：MonkeyRunner   program.py的绝对路径

- 找不到图片

result.writeToFile('./shot1.png','png')   #将截图
存放到和MonkeyRunner.bat同一个目录下的shot1.png中

```
result.writeToFile('E:/demo/shot1.png','png')
pic=MonkeyRunner.loadImageFromFile('E:/demo/shot.png')
if (result.sameAs(pic,1.0)):
    print('success')
else:
    print('fail')
```

或者
```
result = device.takeSnapshot()
result.writeToFile('E:/demo/shot11.png','png')

r1=result.getSubImage(200,200,400,400)
pic=MonkeyRunner.loadImageFromFile('E:/shot.png')
r2=pic.getSubImage(200,200,400,400)
if (r1.sameAs(r2,1.0)):
    print('success')
else:
    print('fail')
```

EasyMonkeyDevice

EasyMonkeyDevice   By

启动View Server:

adb shell service call window 1 i32 4939

查看 View Server:

  adb shell service call window 3

- GetHierarchyViewer

本章大纲

- MonkeyRunner介绍

- MonkeyRunner常用API

- MonkeyRunner实战

- 录制与回放

录制与回放

拷贝到sdk\tools目录下

1. 输入 monkeyrunner monkey_recorder.py
2. 操作 app，导出文件demo.mr
3. 输入 monkeyrunner monkey_playback.py

Wait：相当于sleep，每一步操作中间如果需要等待，以防止点击速度过快有些UI还未跳转成功。

Press a Button：相当于device.press的命令，用来模拟"点击回车，back"等事件

Type Something：相当于device.type 输入

Fling：模拟滑动事件

Export Actions：导出代码

Reflesh Display：刷新当前界面，与手机端屏幕同步

MonkeyRunner录制回放

- monkeyrunnerRecord.py

```
from com.android.monkeyrunner import MonkeyRunner
from com.android.monkeyrunner.recorder import import
MonkeyRecorder
device = MonkeyRunner.waitForConnection()
MonkeyRecorder.start(device)
```