

# Step1 本地验证

## 目录

- 目标:
- 方案概览
  - ADB Shell & Service
- **ADB Shell 指令**
  - 2.1 控制台交互指令
  - 2.2 Shell脚本
    - 2.3.1 开始录制
    - 2.3.2 结束录制
    - 2.3.3 开始回放
- 平台查询日志&Diff
- 数据格式
  - 4.1. 录制
  - 4.2. 回放
- 问题
  - 5.1 代码&功能逻辑问题
  - 其他组件有相同的Hook点导致功能失败
- **TODO**
  - 6.1 提高回放成功率
  - 6.2 后续进展

本地生成录制脚本文件；进行本地的跨设备、版本、生命周期验证

# 目标：

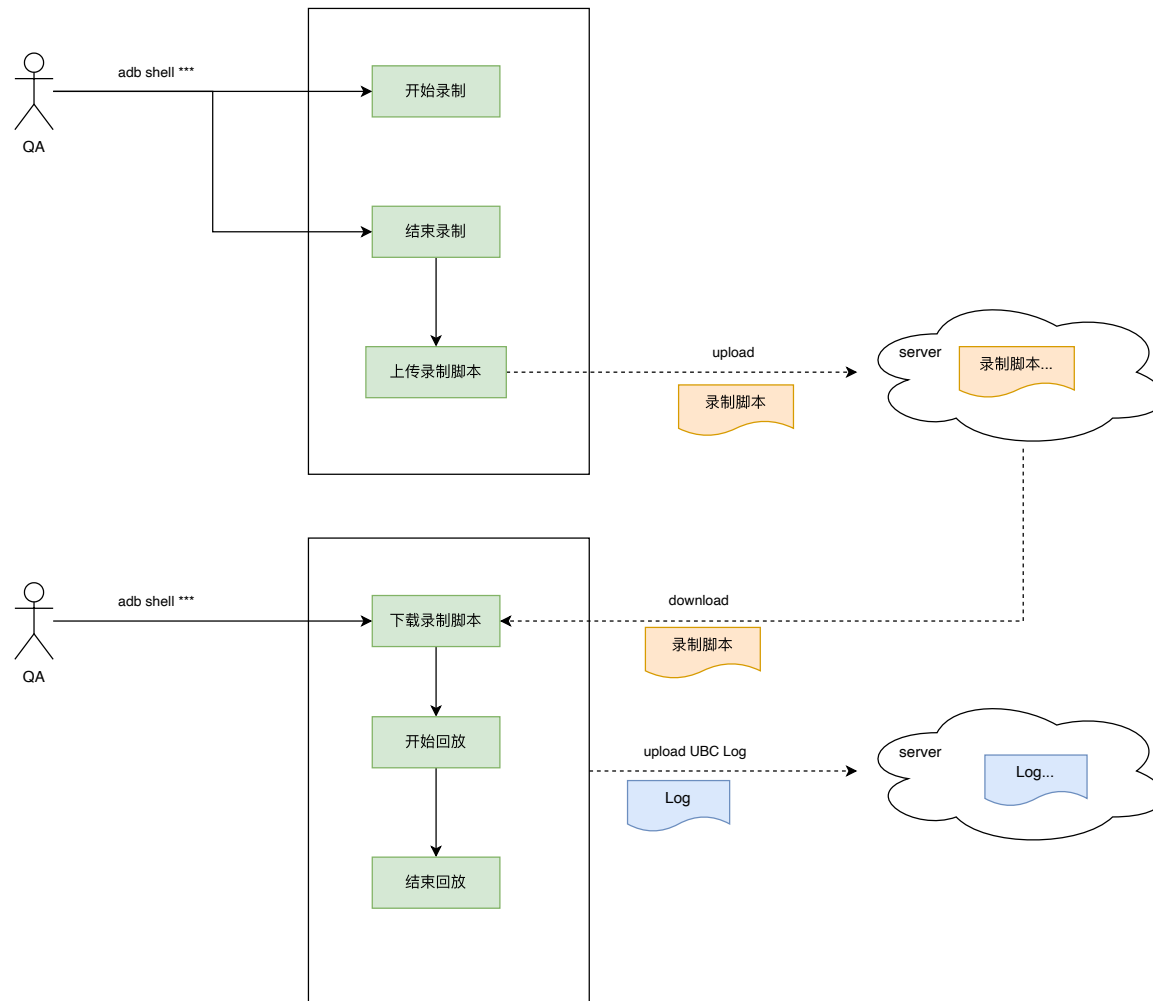
1. 探索录制起始点和自动化验证方式
  - a. 录制起始点：端能力
  - b. 自动化：`adb shell` + `Service`
2. 跨设备、版本、生命周期验证

## 1. 方案概览

录制、结束录制、回放等操作逻辑通过 `Service` 组件调用。

【自动化】用户通过 `Terminal` 或 `shell` 脚本使用 `adb shell` 启动应用和 `Service` 组件，通过附加参数执行对应的逻辑以及传入所需要的参数值。

【录制起始点】通过`adb shell`可以冷启应用，录制和回放的起始点可以定位在手百首页。



## ADB Shell & Service

- `Intent$command = 1` 开启录制
  - `--ei command 1`
  - `--es scheme "xxxx"`
- `Intent$command = 2` 结束录制(并上传脚本)

- `--ei command 2`
- **Intent\$command = 3** 下载脚本并开启回放
  - `--ei command 3`
  - `--es record_id "xxxx"`
  - `--es scheme "xxxx"`

</> PrismCommandService.java

Java

```
1 /**
2  * AndroidManifest.xml配置:
3  *      <service
4  *          android:name="com.baidu.ubc.prism.PrismCommandService"
5  *          android:enabled="true"
6  *          android:exported="true" />
7  */
8 public class PrismCommandService extends Service {
9     private static final String EXTRA_COMMAND = "command";
10    private static final int VALUE_COMMAND_COLLECT_START = 1;
11    private static final int VALUE_COMMAND_COLLECT_END = 2;
12    private static final int VALUE_COMMAND_PLAYBACK = 3;
13    private static final String EXTRA_SCHEME = "scheme";
14
15    public PrismCommandService() {
16    }
17
18    @Override
19    public IBinder onBind(Intent intent) {
20        //      throw new UnsupportedOperationException("Not yet implemented");
21        return null;
22    }
```

```
23
24 @Override
25 public int onStartCommand(Intent intent, int flags, int startId) {
26     if (intent != null) {
27         int command = intent.getIntExtra(EXTRA_COMMAND, 0);
28         String scheme = intent.getStringExtra(EXTRA_SCHEME);
29         String recordId = intent.getStringExtra(EXTRA_DEBUG_IP);
30
31         switch (command) {
32             case VALUE_COMMAND_COLLECT_START:
33                 // 开启录制
34                 if (scheme != null) {
35                     SchemeRouter.invoke(AppUtils.getContext(), scheme);
36                 }
37                 PrismMonitorHelper.getInstance().startMonitor();
38                 break;
39             case VALUE_COMMAND_COLLECT_END:
40                 // 结束录制
41                 PrismMonitorHelper.getInstance().endMonitor();
42                 break;
43             case VALUE_COMMAND_PLAYBACK:
44                 // 开启回放
45                 // 下载回放脚本
46                 PrismService.getInstance().downloadPrism(s -> {
47                     try {
48                         JSONObject jsonObject = new JSONObject(s);
49                         JSONArray jsonArray = jsonObject.optJSONArray("data");
50                         if (scheme != null) {
51                             SchemeRouter.invoke(AppUtils.getContext(), scheme);
52                         }
53                     }
54                 });
55         }
56     }
57 }
```

```
53         PrismPlayBackHelper.getInstance().playback(recordId, jsonArray);
54     } catch (JSONException e) {
55         // log
56     }
57     });
58     break;
59 }
60 }
61 return super.onStartCommand(intent, flags, startId);
62 }
63 }
```

## 2. ADB Shell 指令

### 2.1 控制台交互指令

<https://developer.android.com/tools/adb?hl=zh-cn#IntentSpec>

</>

Plain Text

```
1 adb shell am startservice -n com.example.demolocal/.PrismTestService
2 adb shell am startservice -n com.example.demolocal/.PrismTestService --ei command 1 --es scheme
  "'baiduboxapp://v11/appTab/select?item=home&upgrade=0&params=%7B%22channel%22%3A%221%22%2C%22to_default%22%3A%221%22%7D'"
3
4 # 1.开启应用
5 adb shell am start com.baidu.searchbox/com.baidu.searchbox.SplashActivity
6
7 2. 录制、回放
8 # 开始录制
```

```
9 adb shell am startservice -n com.baidu.searchbox/com.baidu.ubc.prism.PrismCommandService --ei command 1
10 # 结束录制
11 adb shell am startservice -n com.baidu.searchbox/com.baidu.ubc.prism.PrismCommandService --ei command 2
12 # 开始回放
13 adb shell am startservice -n com.baidu.searchbox/com.baidu.ubc.prism.PrismCommandService --ei command 3
14
15
16 # scheme (&要转义或者使用'')
17 1. 打开首页/Feed Tab
18 baiduboxapp://v11/appTab/select?item=home\&upgrade=0
19 baiduboxapp://v11/appTab/select?item=home\&upgrade=0\&params=%7b%22channel%22%3a%221%22%2c%22ceiling%22%3a%221%22%7d
20
21
22 # 组合命令
23 1. 打开应用跳转到Feed推荐频道并开启录制
24 adb shell am start com.baidu.searchbox/com.baidu.searchbox.SplashActivity & adb shell am startservice -n
  com.baidu.searchbox/com.baidu.ubc.prism.PrismCommandService --ei command 1 --es scheme "'baiduboxapp://v11/appTab/select?
  item=home&upgrade=0&params=%7b%22channel%22%3a%221%22%2c%22ceiling%22%3a%221%22%7d'"
25
26 2. 结束录制
27 adb shell am startservice -n com.baidu.searchbox/com.baidu.ubc.prism.PrismCommandService --ei command 2
28
29 3. 打开应用跳转到Feed推荐频道并开启回放
30 adb shell am start com.baidu.searchbox/com.baidu.searchbox.SplashActivity & adb shell am startservice -n
  com.baidu.searchbox/com.baidu.ubc.prism.PrismCommandService --ei command 3 --es scheme "'baiduboxapp://v11/appTab/select?
  item=home&upgrade=0&params=%7b%22channel%22%3a%221%22%2c%22ceiling%22%3a%221%22%7d'"
```

## 2.2 Shell脚本

## 2.3.1 开始录制

</> start\_monitor.sh

Java

```
1 #!/bin/bash
2 source ~/.bash_profile
3
4 # 打开手百
5 adb shell am start com.baidu.searchbox/com.baidu.searchbox.SplashActivity
6 sleep 1
7 adb shell am startservice -n com.baidu.searchbox/com.baidu.ubc.prism.PrismCommandService --ei command 1 --es scheme
  "'baiduboxapp://v11/appTab/select?item=home&upgrade=0&params=%7b%22channel%22%3a%22%2c%22ceiling%22%3a%22%22%7d'"
8 echo "Finished!"
```

## 2.3.2 结束录制

</> stop\_monitor.sh

Java

```
1 #!/bin/bash
2 adb shell am startservice -n com.baidu.searchbox/com.baidu.ubc.prism.PrismCommandService --ei command 2
```



Feed推荐频道录制.mp4

10.5MB

## 2.3.3 开始回放

</> prism\_playback.sh

Java

```
1 #!/bin/bash
```




```
2 source ~/.bash_profile
3
4 # 打开手百
5 adb shell am start com.baidu.searchbox/com.baidu.searchbox.SplashActivity
6 sleep 5
7 adb shell am startservice -n com.baidu.searchbox/com.baidu.ubc.prism.PrismCommandService --ei command 3 --es record_id null
  --es scheme "'baiduboxapp://v11/appTab/select?
  item=home&upgrade=0&params=%7b%22channel%22%3a%221%22%2c%22ceiling%22%3a%221%22%7d'"
8 echo "Finished!"
```




Feed推荐频道自动回放.mp4

13.4MB


## 3. 平台查询日志&Diff

 需要依赖于扫码日志校验才能查询到对应回放日志

点位查询curl

 curl

Java

 python

Java

## 4. 数据格式

### 4.1. 录制

详细内容见见2.2 [☰ 打点场景回归-小桔棱镜Android SDK调研](#)

1. 被点击view所在窗口信息
2. 被点击view在ViewTree上的路径信息
3. 被点击view自身可提取的信息
4. `i` : 脚本回放序列下标

```
</> sample Java  
  
1 i_&_2_w_&_com.xiaojuchefu.prism.TestActivity_&_1_&_vi_&_btn_&_vcn_&_FeedOneImgView_&_vp_&_btn/*/*listview/xml_root/content[0  
1]/decor_content_parent/_&_vl_&_l:1,1_&_vr_&_BUTTON 1_&_vq_&_100
```

## 4.2. 回放

录制脚本回放过程需要在日志中携带对应的录制脚本信息，添加到日志 `content` 同级别字段（`vi`）

- `rid`: recordId, 录制脚本ID
- `pid`: playbackID, 回放标识，用于区分不同轮次的回放
- `i` : 脚本回放序列下标（录制脚本中添加）

```
</> sample ECharts-JSON  
  
1 {  
2     "appv": "13.85.0.0",  
3     "content": "  
    {\"session_id\": \"1739933441837\", \"click_id\": \"7c32c46d58132394b9d85f9ff4dd37f3\", \"from\": \"feed\", \"page\": \"new_home\"  
    , \"source\": \"feedtab_click\", \"value\": \"32\"}",  
4     "id": "5961",  
5     "idtype": "0",  
6     "isreal": "1",  
7     "sid": "1739933441917",
```

```

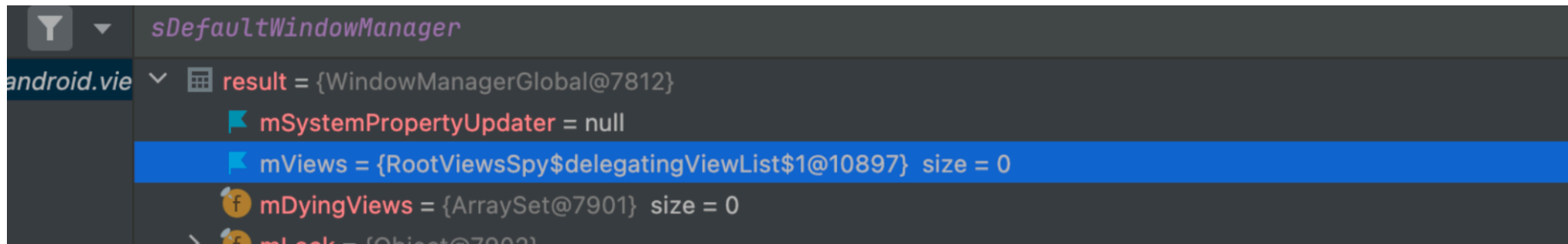
8     "timestamp": "1739933471940",
9     "type": "0",
10    "uuid": "ba54fa356ffb467d9e6f2ef2a6169481",
11    "vi":
      "rid_&_1739433327227_&_pid_&_1739433327447_&_i_&_2_&_w_&_com.xiaojuchefu.prism.TestActivity_&_1_&_vi_&_btn_&_vcn_&_FeedOneI
mgView_&_vp_&_btn_*/listview/xml_root/content[01]/decor_content_parent/_&_vl_&_l:1,1_&_vr_&_BUTTON 1_&_vq_&_100"
12 }
13

```

## 5. 问题

### 5.1 代码&功能逻辑问题

#### 1. 其他组件有相同的Hook点导致功能失败



原因：RootViewsSpy 是 LeakCanary 组件库引入的类。Hook 时机也是 ContentProvider.onCreate

- 目标在线下包开启功能，可以考虑延后 hook 时机，避免被篡改
  - 为 provider 指定 initOrder 为 -1，延迟hook时机；
    - 问题：LeakCanary 的Hook逻辑被覆盖
      - 兼容性处理【已完成】

- 不做兼容处理，使用开关控制组件是否 Hook 【暂不采用】

</> AndroidManifest.xml

Java

```
1 <provider
2     android:name="com.baidu.ubc.prism.PrismInitProvider"
3     android:authorities="com.baidu.ubc.prism.PrismInitProvider"
4     android:initOrder="-1"
5     android:enabled="true"
6     android:exported="true" />
```

## 6. TODO

### 6.1 提高回放成功率

- 列表项点击事件的采集回放 【已解决】
- ViewPager 控件的回放 【已解决】
- PopupWindow 或输入框控件的交互 【未解决】
- 滑动操作 【未解决】
- H5页面元素 【不支持】
- 业务逻辑可能存在首次、二次进入时的弹窗或引导遮罩
- 搜索落地页Tab切换时会有可能跳转到 AI页

### 6.2 后续进展

拉Feed 手百BP (兴雅)聊一下，演示一下效果

QA 发散讨论下场景，看能否真正提效，平台功能

出一个简单的产品形态，比如H5页面，整合这块功能（后续可以走流水线）

- 通过 H5 +后端调用 adb 命令来控制测试机，技术实现上有一定门槛，目前没有特别好的方案。
- adb shell
- 通过端外调起 startActivity 和 startService
  - 给一个安装包
  - 给出如 Leak Canary 的 Leak 形态的应用