广告主deeplink接入规范

## 1 体验要求

## **广告deeplink调起体验要求**

**调起体验是指：点击广告响应deeplink内容时相关的一系列切换与返回过程**

1. 调起动画（具体是指：当用户点击广告跳转到deeplink页面的过程）：左滑（从左至右的内容加载，参考下载类广告调起商店的体验）
2. 调起过程体验需要流畅，不卡顿、闪黑等影响体验的现象出现。
3. 返回：页面Back、手机物理back返回原App，点击一次即返回原App，返回按钮不可关闭
4. 其他：首次调起可出现启动页，进程后台挂起时，二次点击调起希望不再出现app启动页，需要直达应用内指定页

## deeplink页面内容

需要添加回呼按钮让用户明确知晓返回原App的路径与入口，该按钮不能进行关闭。

**示例：**



* 返回按钮展示文案： 读取deeplink中的btn\_name字段进行显示
* 返回：读取deeplink中的back\_url字段，并通过deeplink的方式切换
* UI要求：暂无统一要求，广告主可自行定义
* 协议格式：scheme格式的链接

## 2 接入

## 2.1接入步骤

主要接入步骤

1. 接入App支持scheme调起
2. 接入App有对应的广告落地页
3. APP与广告落地页调起内容体验一致
4. 支持一键返回

## 2.2接入方案

* 1. 接入App须提供支持的deeplink跳转链接给浏览器广告平台。
  2. 广告服务在deeplink链接后拼接回调链接（参数back\_url）及文案(参数btn\_name)。
  3. 广告通过deeplink跳转到广告主应用后，接入方需支持点击返回按钮，deeplink直接跳转回原广告播放侧。
  4. 接入App不能修改广告给出的回调链接。
  5. 接入App需考虑回调deeplink失败的情况。
  6. 接入App根据本文档2.4，实现安全跳转。

## 2.3 deeplink配置说明

deeplink广告要求跳转目标App后，目标App需在跳转页上设置返回按钮，点击可直接返回，按钮样式可由App自行设计，返回按钮文案通过参数返回；

返回按钮包括回调链接地址与返回按钮文案两部分，都通过参数返回，需要广告主根据返回参数进行处理；

参数说明：

|  |  |  |
| --- | --- | --- |
| 参数名 | 作用 | 备注 |
| back\_url=\_\_BACK\_URL\_\_ | 回调链接 | 应用回调必须回调此链接 |
| btn\_name=\_\_BTN\_NAME\_\_ | 返回按钮文案 | 文案为空，代表不展示返回按钮，如果有值，则必须有返回按钮，且应用必须按照文案显示 |

### 2.3.1 回调链接示例

在App中点击返回按钮后，App需要按照回调链接进行回调；广告调起目标App时，会在deeplink链接后拼上回调浏览器的回调链接，由于会拼接一系列参数，回调链接每次会有不同；

目标应用需要动态获取每次调起时的回调链接，执行回调，透传参数；

示例

|  |  |  |
| --- | --- | --- |
|  | 链接示例 | 备注 |
| 原deeplink链接 | ctrip://wireless/h5?guidtype=base=&type=2&allianceid |  |
| 需展示返回按钮，广告调起目标应用时执行的调起链接，增加了回调链接back\_url和返回按钮文案btn\_name | ctrip://wireless/h5?guidtype=base=&type=2&allianceid&back\_url=oppobrowser%3A%2F%2Fresume%3Ffrom%3Dcom.ctrip.x&btn\_name=%E8%BF%94%E5%9B%9E%E6%B5%8F%E8%A7%88%E5%99%A8 | 返回按钮回调链接为oppobrowser://resum?From=comctrip.x，按钮文案为“返回浏览器”，标红部分为参数名，参数值经过url encode |
|  |  |  |

**Deeplink常见FAQ**

1. back\_url是否需要OPPO方提供？---是，从deeplink中取back\_url参数中的值
2. deeplink链接是什么形式的？---标准的schema格式
3. 页面返回按钮如何添加？---自定义实现的控件
4. 点击返回按钮如何实现返回原？相关代码如何实现？

点击返回按钮，监听返回按钮点击事件，调用以下方法就可以：

// 从deeplink中获取到back\_url

backUrl = 从deeplink的back\_url参数中获取到值

//从第三方返回

Uri data = Uri.parse(backUrl);

  Intent intent = new Intent(Intent.ACTION\_VIEW,data);

  //保证新启动的APP有单独的堆栈，如果希望新启动的APP和原有APP使用同一个堆栈则去掉该项

  intent.addFlags(Intent.FLAG\_ACTIVITY\_NEW\_TASK);

  try {

    startActivity(intent);

  } catch (Exception e) {

    e.printStackTrace();

  }

1. 手机物理返回键一键返回浏览器如何实现？---物理键需要客户方应用去检测的，检测到之后，调用以下代码：

// 从deeplink中获取到back\_url

backUrl = 从deeplink的back\_url参数中获取到值

//从第三方返回浏览器，后面有应用的包名

Uri data = Uri.parse(backUrl);

Intent intent = new Intent(Intent.ACTION\_VIEW,data);

//保证新启动的APP有单独的堆栈，如果希望新启动的APP和原有APP使用同一个堆栈则去掉该项

intent.addFlags(Intent.FLAG\_ACTIVITY\_NEW\_TASK);

try {

startActivity(intent);

} catch (Exception e) {

e.printStackTrace();

}