**VIVO广告平台deeplink广告主接入文档**

vivo 版权所有

|  |  |  |
| --- | --- | --- |
| 修改时间 | 修改内容 | 备注 |
| 2019-04-22 | deeplink调起新版规范初版 | 解决媒体端返回到自定义页面的诉求 |
| 2019-04-26 | 对未接入deeplink新规范的媒体端的兼容性处理 |  |
| 2019-05-06 | 广告主客户端不展示回调按钮处理 |  |

**目录**

[一、 Deeplink简介 1](#_Toc6839499)

[二、 接入说明 1](#_Toc6839500)

[1. 主要步骤 1](#_Toc6839501)

[2. 接入注意事项 1](#_Toc6839502)

[3. 返回按钮主要改动点 2](#_Toc6839503)

[4. 接入说明 2](#_Toc6839504)

[三、 联系我们 4](#_Toc6839505)

# Deeplink简介

DeepLink技术为广告主提供拉活广告服务。通过DeepLink跳转，用户可通过广告，直达其他App的活动页；可以为其他App带入用户，来提升产品活跃度。

# 接入说明

## 主要步骤

1. 有android App
2. 接入的App支持scheme调起
3. 接入App有对应的H5或Wap页（对应广告活动页面）
4. WAP与APP调起内容体验一致
5. 支持一键返回

## 接入注意事项

1. 接入广告主的App须提供支持的DeepLink跳转链接给广告平台。
2. vivo会在deeplink链接上拼接回调链接（backurl）。
3. 从vivo广告 DeepLink跳转到广告主应用时，调起链接为deeplink链接+回调链接，接入方须支持点击返回按钮，deepLink直接跳转回vivo相关媒体。
4. 接入App不能修改vivo给出的回调链接。
5. 接入App需考虑回调vivo失败情况。
6. 接入App根据本文档4-2），实现安全跳转。

## 返回按钮主要改动点

本次对接方案改动点主要在对返回按钮的要求上：

1. 返回按钮要求根据vivo广告平台拼接的不同“回调链接”返回到不同的媒体（如“返回vivo浏览器”、“返回vivo小视频”等）;
2. 返回按钮文案要求根据vivo 广告平台拼接的不同”按钮名称”显示不同的返回按钮文案；按钮样式由广告主自行设计。
3. 部分媒体不要求有返回按钮，需要接入广告主根据“按钮名称”、“回调链接”动态进行配置，不展示返回按钮；

## 接入说明

广告主接入流程如下：



* 1. **deeplink回调按钮配置介绍**

vivo广告平台调起广告主app的deeplink地址形式如：

ctrip://wireless/h5?guidtype=base=&type=2&allianceid&**btn\_name**=\_\_BTN\_NAME\_\_&**backurl**=\_\_BACKURL\_\_

示例：

ctrip://wireless/h5?guidtype=base=&type=2&allianceid&btn\_name=%E8%BF%94%E5%9B%9E%E6%B5%8F%E8%A7%88%E5%99%A8&backurl=vivobrowser%3A%2F%2Fbrowser.vivo.com%3Fa%3Dxx%26b%3Dyy%26ad\_id%3Dxx%26ad\_token%3Dxx%26ad\_position%3Dyy%26btn\_name%3D%E8%BF%94%E5%9B%9E%E6%B5%8F%E8%A7%88%E5%99%A8

上述deeplink地址由两部分组成：橙色部分是广告主的原始deeplink地址，蓝色部是经过vivo广告平台追加的两个参数：**btn\_name和backurl，**用于回调按钮的配置

* **backurl参数：** \_\_BACKURL\_\_宏表示广告主app回调vivo媒体端的目标deeplink地址
* vivo媒体deeplink调起广告主app前,对参数backurl的值\_\_BACKURL\_\_进行了 URLEncode操作,广告主app在获取backurl参数值\_\_BACKURL\_\_进行回调前，需要将\_\_BACKURL\_\_回调地址进行URLDecode操作.否则回调会失败.

URLEncode示例：

ctrip://wireless/h5?guidtype=base=&type=2&allianceid&btn\_name%3D%E8%BF%94%E5%9B%9E%E6%B5%8F%E8%A7%88%E5%99%A8&backurl=vivobrowser%3A%2F%2Fbrowser.vivo.com%3Fa%3Dxx%26b%3Dyy%26ad\_id%3Dxx%26ad\_token%3Dxx%26ad\_position%3Dyy

URLDecode示例:

ctrip://wireless/h5?guidtype=base=&type=2&allianceid&btn\_name%3D%E8%BF%94%E5%9B%9E%E6%B5%8F%E8%A7%88%E5%99%A8&backurl=vivobrowser://browser.vivo.com?a=xx&b=yy&ad\_id=xx&ad\_token=xx&ad\_position=yy

* vivo广告平台在\_\_BACKURL\_\_回调地址上拼接了广告投放相关的信息,如需获取该信息,需将回调地址 URLDecode操作之后才能获取正确的参数。具体添加信息如下：

|  |  |  |
| --- | --- | --- |
| 参数名 | 说明 | 备注及校验规范 |
| enter\_from | 应用id | 数字 |
| ad\_id | 广告id | a46854593ba94cf091acd790a2baecff(数字+字母) |
| ad\_token | 请求id | 554809174015\_a4689b593ba94cf091acd790a2baecff(数字+下划线+字母) |
| ad\_position | 广告位id | a4689b593ba94cf091acd79afdbaecff(数字+字母) |

* **btn\_name参数：**\_\_BTN\_NAME\_\_宏表示展示在广告主app客户端返回按钮上的文案
* vivo媒体deeplink调起广告主app客户端前,对参数btn\_name的值\_\_BTN\_NAME\_\_进行了 URLEncode操作,广告主app在获取btn\_name参数值\_\_BTN\_NAME\_\_进行展示前，需要将\_\_BTN\_NAME\_\_文案进行URLDecode操作.否则悬浮按钮文案显示不正确

URLEncode示例：

ctrip://wireless/h5?guidtype=base=&type=2&allianceid&**btn\_name=%E8%BF%94%E5%9B%9E%E6%B5%8F%E8%A7%88%E5%99%A8**

URLDecode示例：

ctrip://wireless/h5?guidtype=base=&type=2&allianceid&**btn\_name=返回浏览器**

* 1. **新老版本兼容处理**

由于版本升级率因素,媒体客户端会存在部分用户版本升级不上来，这部分用户采用的是老的deeplink接入规范，其调起广告主app时，不会携带btn\_name参数，此时需要广告主app的回调按钮文案统一显示为:“返回vivo”,比如以下链接就是未升级版本媒体版本调起广告主app时的deeplink地址：

ctrip://wireless/h5?guidtype=base=&type=2&allianceid&backurl=vivobrowser%3A%2F%2Fbrowser.vivo.com%3Fa%3Dxx%26b%3Dyy%26ad\_id%3Dxx%26ad\_token%3Dxx%26ad\_position%3Dyy

**4.3、媒体端不需要回调处理**

由于部分媒体客户端不需要被回调,此时广告主客户端app不能添加回调按钮,以下情况广告主客户端不能展示回调按钮：

* backrul参数不存在时
* backurl参数值为空时
* backurl参数值仍为\_\_BACKURL\_\_宏时

**4.4、DeepLink跳转安全性**

Intent scheme url的引入，给恶意攻击页面通过intent-based攻击终端上已安装应用提供了便利。

攻击场景和方法描述见http://bbs.m.qq.com/thread-70939-1-1.html。

应对方法，intent做严格的安全限制：

// convert intent scheme URL to intent object

Intent intent = Intent.parseUri(uri);

// forbid launching activities without BROWSABLE category

intent.addCategory("android.intent.category.BROWSABLE");

// forbid explicit call

intent.setComponent(null);

// forbid intent with selector intent

intent.setSelector(null);

// start the activity by the intent

context.startActivityIfNeeded(intent, -1);

# 联系我们

接入过程中遇到的问题请联系vivo浏览器合作人员，或向vivo浏览器合作人员要开发人员联系方式。