

有米原生广告API文档（开源）

授权认证

在以下所有API请求中，都需要携带用于作为身份验证的Token。在请求过程中，需要在HTTP Header中携带Token信息。目前Token即是应用的appid。

HTTP Header中携带Token信息的示例。

```
Authorization: Bearer <Token>
```

请求广告

请求URL

<https://native.umapi.cn/ios/v1/oreq>

参数列表（GET）

字段	类型	必须	描述
reqtime	string	是	发起请求的Unix时间戳，精确到秒
slotid	string	是	所需的广告位ID
adcount	string	是	所需要的广告数，不填默认为1，实际返回的广告数小于等于adcount
gender	string	否	性别，M=男性，F=女性
age	string	否	年龄，如24
cont_title	string	否	内容的标题
cont_kw	string	否	内容的关键词，多个关键词用逗号分隔
reqid	string	否	这次请求的唯一id，可不填写
idfa	string	否	iOS设备的IDFA，明文不加密
brand	string	否	制造厂商，如“apple”“Samsung”“Huawei”，默认为空字符串
model	string	否	型号，如“iphoneA1530”，默认为空字符串
mac	string	否	设备的mac地址，明文不加密
imei	string	否	设备的imei码，明文不加密
androidid	string	否	设备的android id，明文不加密
ip	string	否	当前请求的IP地址，如果是从移动终端发起请求则可以不填写
ua	string	否	UserAgent
os	string	否	操作系统，可选（android，ios）
osv	string	否	操作系统描述的系统版本号
connntype	string	否	网络类型，空=无，0=未知/其他，1=wifi，2=2g，3=3g，4=4g，5=5g
carrier	string	否	网络运营商，空=无，0=未知/其他，1=wifi，2=移动，3=联通，4=电信
pk	string	否	安卓为App的包名，iOS为App的BundleIdentifier
language	string	否	用户设置的语言，如zh
countrycode	string	否	用户设置的国家编码，如CN

返回值

正确返回值

```
{
```

```

"c": 0, // [int] 0
"rsd": "TADS234211435343f", // [string] RequestSeed
"ad": [
  {
    "id": 55, // [int] id
    "slotid": 1, // [int] id
    "name": "", // [string]
    "icon": "https://xxx.png", //
    "pic": [{
      "url": "https://xxx.png",
      "w": 600,
      "h": 400
    }, {
      "url": "https://yyyxxx.png",
      "w": 600,
      "h": 400
    }], //
    "slogan": "", // [string]
    "subslogan": "?Appstore", // [string]
    "url": "https://itunes.apple.com/cn/app/id1053012308?mt=8", //
    [string]
    "uri": "huangshizhanzheng://", // [string] deeplink
    "pt": 0, // [int] 0:APP1:WAP
    "track": {
      "show": [
        "http://track1.xxx",
        "http://track2.xxx"
      ], // [array] URL
      "click": [
        "http://track3.yyy",
        "http://track4.yyy"
      ] // [array] URL
    },
    "app": {
      "storeid": "1053012308", // [string] id
      "bid": "com.huangshizhanzheng", // [string] BundleIdentifier
      "description": "<>",
      "size": "223M", // [string]
      "screenshot": [
        "http://aaa.png",
        "http://ccc.png",
        "http://bbb.png"
      ], //
      "score": 4.5, // [float] 1.0~5.0
      "category": "" // [string]
    }, //
    "ext": {
      "io": 0, // 0:1:
      "delay": 1, //
      "sal": 0, // 0:1:
      "pl": 0 // 0:1:
    } //
  },

```

```
{
    // ...
}
] // [array]
}
```

错误返回值

```
{
  "c": -1 // 0
}
```

效果监控上报

注意：曝光监控和点击监控十分重要，需要贵司确保App内已经能够很好的支持曝光和点击监控的上报功能，否则可能会导致结算问题。

1. 广告曝光时，调用track结构体中的show列表里的url上报曝光监控，曝光链接需要从客户端发起。
2. 用户点击时，调用track结构体中的click列表里的url上报点击监控，需要从客户端发起请求。若需要发起点击监控，需要等待点击监控全部发送完成后再跳转到落地页。
3. 效果上报的链接请使用请求接口中（正确返回值）里面返回的数据，如下

```
"show": [
    "http://track1.xxx",
    "http://track2.xxx"
], // [array] array
"click": [
    "http://track3.yyy",
    "http://track4.yyy"
] // [array] array
```

点击跳转到逻辑页的逻辑

注意：为了防止跳出App后点击记录无法发送而出现结算问题，需要在效果监控上报完成后，再发起点击跳转。点击逻辑较为复杂，建议贵司直接使用有米封装好的开源逻辑类库简化开发工作。

1. 点击链接有url和uri两个。
2. url可以用于打开一个WAP页面，或者跳转到AppStore上。
3. uri用于直接打开指定应用内的落地页，可以用于deeplink推广。
4. 如果是iOS应用推广，一般情况app下的storeid字段会记录对应的itunesid，当这个字段有值时，也可以通过这个字段在应用内打开AppStore。

应用内打开AppStore

在应用内直接打开AppStore的详情页，可以有效提高用户体验与转化率，从而提高应用收入。建议在应用中支持打开AppStore的逻辑。

1. 当返回的广告为一个App广告时，app下的storeid字段记录对应的itunesid，注意storeid字段不一定都有值，有值的情况下才能处理。
2. InApp内打开Store的流程需要特殊处理，并且不能使用正常的url来打开，因此这种情况下url字段可能为空，点击记录由“效果监控上报”负责处理。
3. deeplink的逻辑不受影响。

deeplink推广

deeplink链接可以直接打开目标应用的某个特定页面，这种推广模式对于电商等品类的广告有显著效果，支持deeplink推广可以显著提高应用收入，贵司可以在应用中提供对deeplink推广的支持。

1. deeplink推广不能使用正常url来打开，因此点击监控由“效果监控上报”负责处理。
2. 当返回值中uri字段不为空时，表明该广告可以进行deeplink推广。
3. 首先通过uri尝试唤醒目标应用，唤醒成功直接跳转到目标App上，不需要再调用url字段。
4. 若唤醒失败，则调用url字段打开AppStore，引导用户下载应用。
5. 若唤醒失败，且app下的storeid字段有值，则可以尝试不调用url字段，直接在应用内打开AppStore引导用户下载。

普通链接推广

普通链接推广一般指点击打开一个普通的WAP页面，可以是跳转到外部Safari，也可以在App内打开一个内置浏览器，直接调用url打开对应的落地页即可。