

# 腾讯程序化广告交易平台 接入说明文档

---

**V1.8 版**

## 法律声明

DSP 在与腾讯程序化广告交易平台签署了《腾讯程序化广告交易平台保密协议》后，腾讯程序化广告交易平台将向 DSP 提供本文档，作为其与腾讯程序化广告交易平台对接的指引文档。未经腾讯程序化广告交易平台书面许可，不得以任何形式向第三方披露、泄露有关本文档的任何内容。腾讯程序化广告交易平台拥有修改、调整、增补本文件的权利，并在法律允许范围内对本文档拥有最终解释权。

## 修订历史

文档版本	修订日期	修订描述
1.0Beta	2014-12-5	1. 初稿
1.1Beta	2015-3-24	1. Buyer API 增加广告主投放资质上传接口
1.2Beta	2015-4-13	1. Buyer API 增加报表获取接口 2. 修改 BidRequest/Response 示例
V1.0	2015-5-20	1. 增加 APP 下载类广告投放 2. 增加移动联盟开屏流量
V1.1	2015-6-1	1. 增加对第三方曝光监测的支持
V1.2	2015-6-19	1. Buyer API 消耗报表增加有效曝光量 2. BidRequest 增加 boss_qq 和 user
V1.3	2015-8-10	1. 增加 QQ 秀、QQ 音乐开屏和移动联盟原生流量 2. BidRequest 增加 app_bundle_id
V1.3.1	2015-9-21	1. BidRequest 增加 contract_code（用于程序化预留）和 multimedia_type_white_list 2. BidRequest.User.Demographic 增加 consume_level（消费能力）和 house_price（住房价格）
V1.4	2015-12-8	1. 增加跨屏频控功能 2. 增加应用直达（Deep Link）功能，广告创建接口增加 deep_link_uri 3. 商品类型（product_type）取值增加 4（微信品牌页），目前仅支持微信公众号文章页面底部 banner 4. 更新 BidRequest 中的一些枚举值定义
V1.4.1	2015-12-23	1. 更正文档包中《DSP 常见问题解答》6.1.10 微信频控方案（暂时不可用）
V1.5	2016-1-29	1. BidRequest 增加 idfa（idfa 原文） 2. BidRequest 增加 blocking_appname、blocking_keyword、advertiser_whitelist 和 advertiser_blacklist 3. 第三方曝光监测宏支持双下划线__（兼容原有%%形式）
V1.6	2016-2-18	1. BidRequest 增加 support_deep_link（广告位是否支持 Deep Link 广告）和 wx_flow_class（微信公众号分类） 2. BidResponse 中增加 DynamicCreative 类型和 ProductInfo 类型用以支持动态创意，详细说明见附录 4 “ <b>动态创意（Dynamic Creative）</b> ”
V1.7	2016-4-28	1. 增加对流量预定位（Pretargeting）功能的支持 2. 增加对频次控制（Frequency Capping）功能的支持 3. Bid Request 去掉 blocking_appname（广告位过滤的应用名称） 4. Bid Request 增加 manufacturer（设备制造商信息） 5. Bid Request 增加 android_id（加密后的 Android ID） 6. Bid Request 暂停提供微信公众号分类信息（wx_flow_class）

		<ul style="list-style-type: none"><li>7. Bid Response 增加 app_filter，用以支持 App 已安装过滤功能</li><li>8. 更新超时时间范围（<a href="#">5.5 超时时间</a>）</li><li>9. 附录 4 中 Deep Link 部分分 iOS 和 Android 细化逻辑描述</li></ul>
V1.8	2016-08-03	<ul style="list-style-type: none"><li>1. 更新广告主层级 qq 字段定义</li><li>2. 广告主审核状态获取接口增加 industry_id（审核员校正后的广告主行业 ID）和 union_only（广告主是否被设定为仅能投放腾讯广告联盟）</li><li>3. 报表中增加“竞价失败原因报表”（<a href="#">4.6.2 竞价失败原因报表</a>）</li><li>4. 消耗报表增加 valid_click（经反作弊处理后的点击数据）</li></ul>

# 目录

1. 关于文档.....	1
2. 背景知识.....	1
3. 商务对接流程.....	1
4. Buyer API.....	2
4.1 通用声明 .....	2
4.1.1 权限认证 .....	2
4.1.2 编码方式.....	2
4.1.3 URL 定义.....	2
4.1.4 API 请求 .....	3
4.1.5 API 响应 .....	3
4.1.6 标准处理流程.....	4
4.2 广告主 .....	5
4.2.1 广告主创建 .....	6
4.2.2 广告主修改 .....	8
4.2.3 广告投放资质 .....	10
4.2.4 广告主审核状态获取 .....	11
4.3 广告 .....	13
4.3.1 广告创建 .....	13
4.3.2 广告修改.....	17
4.3.3 广告审核状态获取.....	19
4.4 Pretargeting .....	20
4.4.1 创建 Pretargeting 配置 .....	21
4.4.2 更新 Pretargeting 配置 .....	23
4.4.3 删除 Pretargeting 配置 .....	24
4.4.4 根据 ID 获取 Pretargeting 配置 .....	24
4.4.5 获取所有的 Pretargeting 配置 .....	25
4.5 频次控制 .....	27
4.5.1 创建频次配置 .....	27
4.5.2 根据 ID 获取频次配置.....	28
4.5.3 删除频次配置.....	29

4.6 报表 .....	29
4.6.1 竞价报表 .....	30
4.6.2 竞价失败原因统计报表 .....	31
4.6.3 消耗报表 .....	33
5. 实时竞价 RTB 接口 .....	35
5.1 通信方式 .....	35
5.2 编码方式 .....	36
5.3 HTTP 请求头 .....	36
5.4 HTTP 响应 .....	36
5.5 超时时间 .....	36
5.6 消息定义 .....	37
5.6.1 Bid Request .....	37
5.6.2 Bid Response .....	45
5.6.3 Win Notice .....	48
5.6.4 成交价加解密 .....	48
5.6.5 宏替换 .....	48
附录 .....	50
附录 1、Buyer API 基础数据类型定义 .....	50
附录 2、枚举值定义 .....	51
商品类型 (ProductType) .....	51
审核状态 (ReviewStatus) .....	51
频控周期类型 (PeriodType) .....	51
原生广告组件 (Impression.Native.Fields) .....	51
原生广告类型 (Impression.Native.Type) .....	52
设备类型 (Device.DeviceType) .....	52
设备运营商 (Device.Carrier) .....	52
设备联网方式 (Device.ConnectionType) .....	52
移动设备操作系统 (Device.OperatingSystem) .....	53
用户性别 (User.Demographic.Gender) .....	53
用户学历 (User.Demographic.Education) .....	53
用户消费能力 (User.Demographic.ConsumptionAbilityLevel) .....	53
附录 3 返回码 (Buyer API) .....	54

系统返回码 .....	54
认证错误.....	54
通用业务错误 .....	54
广告主 .....	55
广告创意.....	55
附录 4、功能使用说明.....	57
应用直达（Deep Link） .....	57
动态创意（Dynamic Creative） .....	58
流量预定位（Pretargeting） .....	62
频次控制（Frequency Capping） .....	63

# 1. 关于文档

本文档作为 DSP 与腾讯程序化广告交易平台（以下简称“Tencent Ad Exchange”）进行对接的指引文档。主要包括三个部分：

**商务对接流程：**阅读对象为 DSP 的商务人员。该部分说明了 DSP 与 Tencent Ad Exchange 对接的流程和步骤。

**Buyer API：**阅读对象为 DSP 的产品和开发人员。该部分详细描述了 DSP 向 Tencent Ad Exchange 上传/修改广告主/广告信息以及查询广告主/广告审核状态的接口。

**实时竞价 RTB 接口：**阅读对象为 DSP 的产品和开发人员。该部分详细描述了 DSP 参与 Tencent Ad Exchange 实时竞价的接口协议。

# 2. 背景知识

本 API 文档所涉及接口均遵循 HTTP 协议，请求和响应数据格式如无特殊说明均为 JSON，您可以使用任何支持 HTTP 协议和 JSON 格式的编程语言开发应用程序。

有关标准 HTTP 协议，可参考 [RFC2616](#) 或 [维基百科-HTTP](#) 相关介绍。有关 JSON 数据格式，可参考 [JSON.ORG](#) 或 [维基百科-JSON](#) 相关介绍。

# 3. 商务对接流程

**申请并签署保密协议：**DSP 与 Tencent Ad Exchange 签署《腾讯程序化广告交易平台保密协议》；

**开户&提供文档：**联系运营接口人并提供相关信息后进行开户，获取《腾讯程序化广告交易平台接入说明文档》；

**开发对接：**对接过程中如果遇到疑问，也可联系运营接口人进行相关的产品和技术答疑；

**测试联调：**在开发完成后，Tencent Ad Exchange 提供测试环境与外部 DSP 进行测试环境联调，包括 Buyer API 和实时竞价 RTB 接口两部分的测试，测试过程中将会向 DSP 提供测试环境 DSPID 及授权令牌（TOKEN）；

**服务器部署和网络测试：**DSP 部署好服务器后可联系 Tencent Ad Exchange 进行网络测试；

**合同签署：**合同签署可在步骤 1 完成后同步进行，正式开启流量必须在合同签署完毕后进行；

**充值：**支持预付款方式及账期后付款两种方式，其中账期后付款请参照《第三方 DSP 授信政策》申请；

**正式环境上线：**测试环境联调通过后，向运营接口人申请线上正式的 DSPID 和授权令牌（TOKEN），正式环境开启流量，将按照合同约定进行扣费。

以上各环节的运营接口人联系方式：[gdtadx@tencent.com](mailto:gdtadx@tencent.com)。



## 4. Buyer API

本章节将详细介绍 DSP 如何使用 Buyer API 与 Tencent Ad Exchange 进行交互，提交/修改广告主/广告信息，查询广告主/广告审核状态。为确保广告内容的有效性与合法性，所有参与 Tencent Ad Exchange 竞价的 DSP 平台必须将广告主和广告信息提前上传到 Tencent Ad Exchange 进行审核，审核通过后方可正常投放，如果 DSP 返回未经审核通过的广告，将直接被过滤不会被展现。

### 4.1 通用声明

#### 4.1.1 权限认证

API 通过授权令牌（TOKEN）和使用方的服务器 IP 列表进行双重权限控制。TOKEN 需要在每次请求时通过请求头传入，IP 需要在正式发布前提供给运营接口人，由运营接口人在系统中注册后，方能正常使用。

#### 4.1.2 编码方式

若无特殊说明或响应头中的 Content-Type 未指定编码，请求和响应中的字节编码均使用 UTF-8（无 BOM 头）。

#### 4.1.3 URL 定义

请求 URL 参考 RESTful 风格，约定了使用的协议、域名、模块、版本、资源及动作，详细定义如下：

`https://api.e.qq.com/adx/<API_VERSION>/<RESOURCE_NAME>/<RESOURCE_ACTION>`

其中：

- ✧ 协议：必须使用 HTTPS
- ✧ API\_VERSION：版本号，当前版本号码为 v1
- ✧ RESOURCE\_NAME：表示要操作的资源，如 creative
- ✧ RESOURCE\_ACTION：表示对资源的动作，如 add

RESOURCE\_NAME 目前支持如下资源：

资源	描述
advertiser	广告主
creative	广告

RESOURCE\_ACTION 目前支持如下基本动作：

操作	描述
----	----

add	新增
update	修改
get_review_status	获取审核状态

## 4.1.4 API 请求

### HTTP Method

调用方应设置 HTTP Method 为 POST。

### HTTP Header

调用方应遵循 HTTP 协议设置相应的 Header, 目前支持的 Header 有: Authorization、Content-Type, Authorization 用于传递授权令牌 (TOKEN), Content-Type 用于指定数据格式。本章节中 Content-Type 应为 application/json。

### HTTP Header 示例

```
Content-Type: application/json
Authorization: Bearer <TOKEN>
```

### JSON 数据示例

```
{
  "data": requestData
}
```

## 4.1.5 API 响应

### HTTP 状态码

支持 HTTP 标准状态码, 具体如下:

状态码	名称	描述
200	成功	当 API 请求被正确处理, 且能按设计获取结果时, 返回该状态码; 亦适用于批量接口返回部分结果
3xx	跳转	在特定情况下, API 可能会返回这些状态码, 建议调用方按照 HTTP 标准来处理
4xx	客户端错误	由客户端原因造成的错误
5xx	服务器端错误	API 或其下层服务发生内部错误

其中, 4xx 和 5xx 的状态码仅用于辅助调用方快速识别问题, 不作为包含实际语义的错误码, 若有调整也不另行通知, 实际操作结果以 API 返回的数据为准。调用方也应能够识别和处理由于网络异常等因素导致的, 由非 API 服务返回的 HTTP 状态码, 如 504 Gateway Timeout 等。

### HTTP Header

响应会根据接口要求设置 Content-Type。本章节中 Content-Type 均设置为 application/json。

## HTTP Body

响应的 JSON 数据中包含三部分内容，分别为返回码、返回信息和数据，如下表所示：

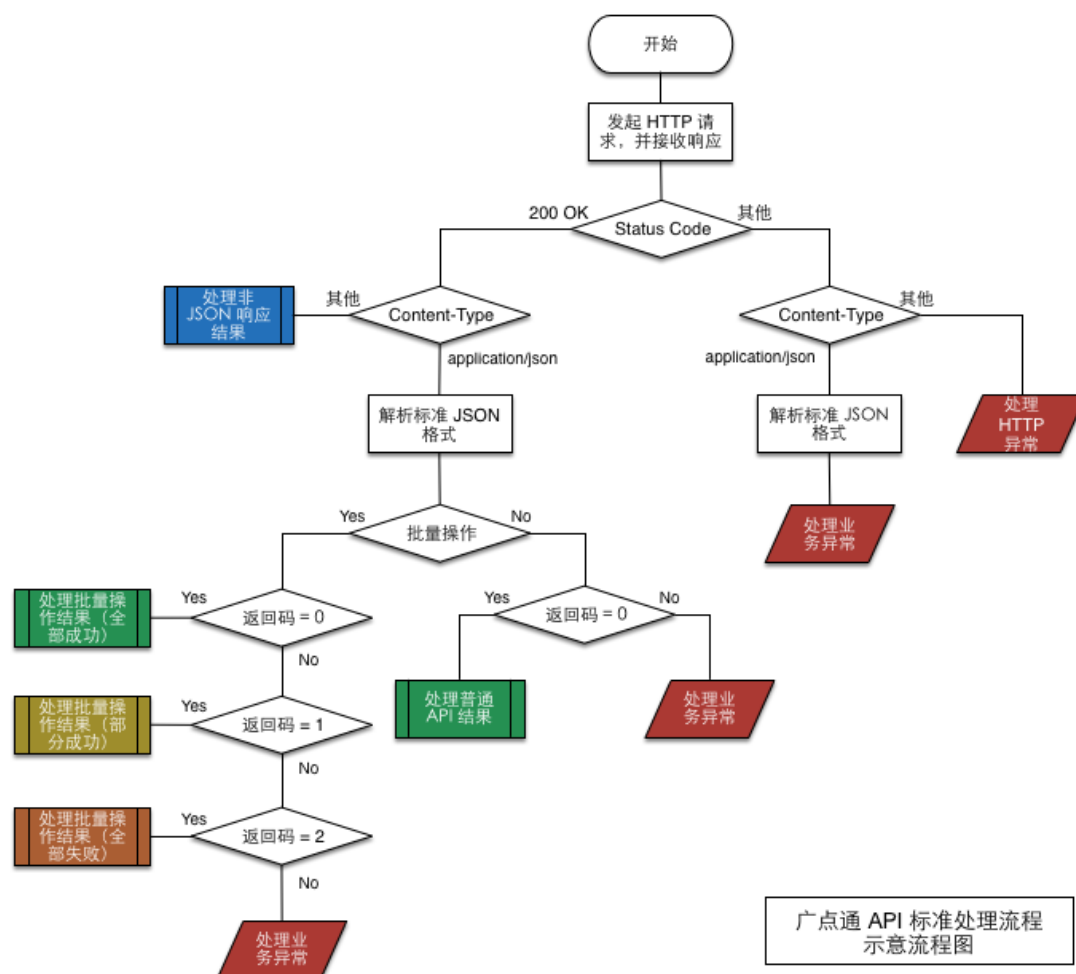
名称	类型	必填	描述
code	int	是	返回码： 0 表示成功，1 表示部分成功（仅用于批量接口）， 2 表示全部失败（仅用于批量接口）。 具体见 <a href="#">返回码</a> 列表
message	string	否	返回信息： 若有错误，此字段为详细的错误信息。
data	json array 或 json object	否	返回结果： 针对批量接口，若无特殊说明，结果将按请求数组的顺序全量返回

对数组形式的 API 请求（批量接口），返回数据如下所示：

```
{
  "code": 1,
  "message": "partial success",
  "data": [
    {
      "code": 0,
      "message": "success",
      "data": {
        "advertiser_id": "205"
      }
    },
    {
      "code": 2001,
      "message": "advertiser already exists",
      "data": {
        "advertiser_id": "206"
      }
    }
  ]
}
```

### 4.1.6 标准处理流程

建议您按照如下流程图所示对响应进行处理：



## 4.2 广告主

广告主存在两种资质：开户资质和投放资质，其区别如下：

资质名称	资质含义	举例
开户资质	广告主开户需要的基本资质：营业执照、ICP 证、行业资质等	
投放资质	投放过程中上传的广告物料为特殊行业或者特殊种类时，需上传相应的投放资质	投放化妆品类广告时需上传化妆品生产企业卫生许可证、化妆品生产许可证等资质

其中开户资质是在广告主创建及审核环节必要的，关系到账户是否会被审核通过。投放资质只有在上传特殊物料时才需要，关系到具体投放的广告是否会被审核通过。两种资质上传的接口是分开的（分别见 4.2.1 及 4.2.3），开户资质的更新会触发账户进入待审核状态，投放资质的更新不会触发账户的审核流程。

## 4.2.1 广告主创建

请求 URL: <https://api.e.qq.com/adx/v1/advertiser/add>

请求字段:

类型	是否数组类型	描述
Advertiser[]	是	最多 5 个

请求类型 Advertiser:

名称	类型	必填	限制	描述
advertiser_id	string	是	小于 128 字节	广告主唯一 id
name	string	是	小于 128 字节	客户名称, 请确保此名称与客户营业执照上的名称一致, 否则会影响该广告主的审核。
homepage	string	是	小于 512 字节	客户网址
brand_name	string	否	小于 16 字节	品牌名称, 16 字以内, 在 Feeds 广告中使用。品牌名称和品牌 logo 绑定上传, 两者缺一将被审核拒绝。
brand_logo_image_url	string	否	小于 1024 字节	品牌 logo, 80*80 png/jpg, 不超过 30KB, 在 Feeds 广告中展示, 品牌名称和品牌 logo 绑定上传, 两者缺一将被审核拒绝。
industry_id	int64	是		行业 id, 见《广告主行业分类》; 当某一级行业下含有二级行业时, 字段中须填写二级行业 id, 而非一级行业 id
qq	int64	否		投放 Android 下载广告时必填。取自广告主(开发者)在腾讯开放平台(open.qq.com)登录时的账户 qq

qualifications	AdvertiserQualification[]	是		开户资质列表，必须包括 ICP 和营业执照，如果特殊行业还需要行业资质。最多可上传 10 个资质。
----------------	---------------------------	---	--	---

请求类型 AdvertiserQualification:

名称	类型	必填	限制	描述
name	string	否	小于 128 字节	资质名称
file_url	string	是	小于 1024 字节	开户资质图片文件 url，图片为 jpg、png、gif 格式，每个文件不超过 1MB

应答字段:

名称	类型	描述
advertiser_id	string	广告主 id

请求示例:

```
{
  "data": [
    {
      "advertiser_id": "12345",
      "name": "腾讯公司",
      "homepage": "http://ads.tencent.com/",
      "brand_name": "腾讯社交广告",
      "brand_logo_image_url": "http://a3.qpic.cn/psb?/V11TAqUh2oXy5u/zJMTID9mj6G5pr5EiSoI108*nAvD1xgMmce*8yFzKA!/b/dPh1vWo3HAAA&bo=UABQAAAAAADByI!&rf=viewer_4",
      "industry_id": 21474836581,
      "qualifications": [
        {
          "name": "腾讯社交广告",
          "file_url": "http://a3.qpic.cn/psb?/V11TAqUh2oXy5u/zJMTID9mj6G5pr5EiSoI108*nAvD1xgMmce*8yFzKA!/b/dPh1vWo3HAAA&bo=UABQAAAAAADByI!&rf=viewer_4"
        }
      ]
    }
  ]
}
```

应答示例:

```
{
```

```

"code": 0,
"message": "操作成功",
"data": [
  {
    "code": 0,
    "message": "操作成功",
    "data": {
      "advertiser_id": "12345"
    }
  }
]
}

```

## 4.2.2 广告主修改

请求 URL: <https://api.e.qq.com/adx/v1/advertiser/update>

请求字段:

类型	是否数组类型	描述
Advertiser[]	是	最多 5 个

请求类型 Advertiser（未修改字段也需填写原值，可修改字段更新时会触发再审核）:

名称	类型	必填	限制	描述
advertiser_id	string	是	小于 128 字节	不可修改，广告主 id 不存在将修改失败
name	string	是	小于 128 字节	可修改
homepage	string	是	小于 512 字节	可修改
brand_name	string	否	小于 16 字节	可修改
brand_logo_image_url	string	否	小于 1024 字节	可修改
industry_id	int64	是		可修改
qq	int64	否		可修改
qualifications	AdvertiserQualification[]	是		可修改，最多 10 个。覆盖之前所有资质。

请求类型 AdvertiserQualification:

名称	类型	必填	限制	描述
----	----	----	----	----

name	string	否	小于 128 字节	资质名称
file_url	string	是	小于 1024 字节	资质图片文件 url，图片为 jpg、png、gif 格式，每个文件不超过 1MB

应答字段：

名称	类型	描述
advertiser_id	string	广告主 id

请求示例：

```
{
  "data": [
    {
      "advertiser_id": "12345",
      "name": "腾讯公司",
      "homepage": "http://ads.tencent.com/",
      "brand_name": "腾讯社交广告",
      "brand_logo_image_url": "http://a3.qpic.cn/psb?/V11TAqUh2oXy5u/zJMTID9mj6G5pr5EiSoI108*nAvD1xgMmce*8yffZKA!/b/dPh1vWo3HAAA&bo=UABQAAAAAADByI!&r=viewer_4",
      "industry_id": 21474836581,
      "qualifications": [
        {
          "name": "腾讯社交广告",
          "file_url": "http://a3.qpic.cn/psb?/V11TAqUh2oXy5u/zJMTID9mj6G5pr5EiSoI108*nAvD1xgMmce*8yffZKA!/b/dPh1vWo3HAAA&bo=UABQAAAAAADByI!&r=viewer_4"
        }
      ]
    }
  ]
}
```

应答示例：

```
{
  "code": 0,
  "message": "操作成功",
  "data": [
    {
      "code": 0,
      "message": "操作成功",
      "data": {
        "advertiser_id": "12345"
      }
    }
  ]
}
```



}

### 4.2.3 广告投放资质

请求 URL: [https://api.e.qq.com/adx/v1/advertiser/update\\_ad\\_qualifications](https://api.e.qq.com/adx/v1/advertiser/update_ad_qualifications)

请求字段:

类型	是否数组类型	描述
AdvertiserAdQualification[]	是	最多 30 个

请求类型 AdvertiserAdQualification:

名称	类型	必填	限制	描述
advertiser_id	string	是	小于 128 字节	不可修改, 广告主 id, 不存在将修改失败
qualifications	AdQualification[]	是		可修改, 最多可上传 30 个投放资质。覆盖之前所有投放资质。

请求类型 AdQualification:

名称	类型	必填	限制	描述
name	string	否	小于 128 字节	投放资质名称
file_url	string	是	小于 1024 字节	投放资质图片文件 url, 图片为 jpg、png、gif 格式, 每个文件不超过 1MB

应答字段:

名称	类型	描述
advertiser_id	string	广告主 id

请求示例:

```
{
  "data": [
    {
      "advertiser_id": "12345",
      "qualifications": [
        {
          "name": "腾讯社交广告 1",
          "file_url": "http://a3.qpic.cn/psb?/V11TAqUh2oXy5u/zJMTID9mj6G5pr5EiSoI108*nAvD1xgMmce*8yffzKA!/b/dPh1vWo3HAAA&bo=UABQAAAAAADByI!&rf=viewer_4"
        }
      ]
    }
  ]
}
```

```
]
}
```

应答示例:

```
{
  "code": 0,
  "message": "操作成功",
  "data": [
    {
      "code": 0,
      "message": "操作成功",
      "data": {
        "advertiser_id": "12345"
      }
    }
  ]
}
```

#### 4.2.4 广告主审核状态获取

请求 URL: [https://api.e.qq.com/adx/v1/advertiser/get\\_review\\_status](https://api.e.qq.com/adx/v1/advertiser/get_review_status)

请求类型:

名称	是否数组类型	描述
advertiser_ids	是	advertiser_id 列表, 最多 20 个

应答类型 AdvertiserReviewStatus:

名称	类型	描述
advertiser_id	string	
review_status	string	具体值参见附录 2 <a href="#">审核状态</a>
review_msg	string	拒绝原因
industry_id	int64	审核员校正后的行业 ID
union_only	boolean	广告主是否被判定为仅能在腾讯广告联盟投放

请求示例:

```
{
  "data": [
    "10010001",
    "10010002",
    "10010003"
  ]
}
```

```
}
```

应答示例:

```
{
  "code": 0,
  "message": "操作成功",
  "data": [
    {
      "code": 0,
      "message": "操作成功",
      "data": {
        "advertiser_id": "10010001",
        "review_status": "APPROVED",
        "review_msg": "",
        "industry_id": 21474836482,
        "union_only": "true"
      }
    },
    {
      "code": 0,
      "message": "操作成功",
      "data": {
        "advertiser_id": "10010002",
        "review_status": "PENDING",
        "review_msg": "",
        "union_only": "false"
      }
    },
    {
      "code": 0,
      "message": "操作成功",
      "data": {
        "advertiser_id": "10010003",
        "review_status": "REJECTED",
        "review_msg": "资质不合法",
        "union_only": "false"
      }
    }
  ]
}
```

## 4.3 广告

如果广告主审核不通过将无法成功上传广告信息。

### 4.3.1 广告创建

请求 URL: <https://api.e.qq.com/adx/v1/creative/add>

请求字段:

类型	数组类型	描述
creative[]	是	最多 10 个

请求类型 Creative（具体由不同样式带来的字段填充要求参见“广告样式说明”）:

名称	类型	必填	限制	描述
advertiser_id	string	是	小于 128 字节	物料对应的广告主 id
contract_code	string	否	小于 255 字节	PMP 合同号, 一个广告只能带一个 contract_code
creative_id	string	是	小于 128 字节	物料唯一 id, 字节串类型, 请确保该 id 为 DSP 层级下唯一, 不会出现不同的广告主有相同的 creative_id
product_type	int32	否		1-普通链接广告（默认值） 2-android 应用下载 3-ios 应用下载 4-微信品牌页（仅针对微信流量）
app_id	int64	否		下载类的 app id。如果 product_type 为 2 或 3, 则必填
pkgname	string	否	小于 256 字节	应用宝包名。如果 product_type=2, 则该字段不填默认投放主线包, 填写包名则投放对应的渠道包; 如果 product_type 是其它值, 可不填此字段
landing_page	string	否	小于 1024 字节	落地页, 仅供审核, ADX 不使用该地址跳转。product_type 为 1 或 4, 必填; product_type 为 2 或 3, 不用填写该字段
target_url	string	是	小于 1024 字节	点击监测地址。product_type 为 1 或 4, 为 ADX 实际使用的跳转地址, DSP 须在点击监测后 302 至落地页 (landing page); product_type 为 2 或 3, 为 ADX 后

				台上报点击 ping 的地址
monitor_url1	string	是	小于 1024 字节	DSP 曝光监测地址，可支持的宏参见 5.6.5 DSP 监测部分
monitor_url2	string	否	小于 1024 字节	第三方曝光监测地址 1，可支持的宏参见 5.6.5 第三方曝光监测部分
monitor_url3	string	否	小于 1024 字节	第三方曝光监测地址 2，可支持的宏参见 5.6.5 第三方曝光监测部分
begin_date_included	date	否		广告播放起始日期。格式 yyyy-MM-dd。 广告不带合同号 (contract_code)：字段未填写，默认为上传当天；字段填写，日期需大于等于上传当天。广告带合同号 (contract_code)：字段未填写，默认为合同起始时间；字段填写，日期需大于等于合同起始时间。
end_date_included	date	否		广告播放结束日期。格式 yyyy-MM-dd。广告不带合同号 (contract_code)：字段未填写，默认为上传日期起 60 天后失效；字段填写，最长不能超过上传时间 180 天。广告带合同号 (contract_code)：字段未填写，默认为合同结束时间；字段填写，CPD 合同必须小于等于合同结束时间，preferred deal 和合约 CPM 类型的广告最长不能超过上传时间 180 天。
creative_spec	int32	是		广告样式，见《广告样式说明》文档
title	string	否	小于 30 字节	标题，根据 creative_spec 选填
description	string	否	小于 256 字节	描述，根据 creative_spec 选填
multimedia1_file_url	string	否	小于 1024 字节	媒体文件 url1，根据 creative_spec 选填
multimedia2_file_url	string	否	小于 1024 字节	媒体文件 url2，根据 creative_spec 选填
call_to_action	string	否	4 字以内	原生广告的行动语，普通链接类默认“查看详情”，应用下载类默认“下载”。非原生广告填写无效。
deep_link_uri	string	否	小于 1024 字节	应用直达 URI。应用直达功能即用户点击广告后，“直接到达”广告主

				<p>“应用”内广告指定的商品页。填写该字段时需指定广告商品类型（<code>product_type</code>），若为普通链接（<code>product_type=1</code>），则用户未安装广告主的 App 时会直接通过内置浏览器打开商品页；若为 Android 应用下载（<code>product_type=2</code>），App 本身需要在腾讯开放平台上架，用户未安装 App 时会直接下载该 App；若为 iOS 应用下载（<code>product_type=3</code>），则用户未安装该 App 时会跳转到 App Store 应用页面。目前仅支持<b>移动联盟资源</b>。</p>
--	--	--	--	--

应答字段：

名称	类型	描述
creative_id	string	

请求示例：

```
{
  "data": [
    { //广告基本信息
      "advertiser_id": "1347",
      "creative_id": "112",
      "contract_code": "", //PMP 投放中的合同号，普通竞价广告不需上传。如 1df3ed49a11c458c9b907ad81b1ec988
      "begin_date_included": "2016-01-01", //默认为上传时间
      "end_date_included": "2016-03-31", //默认为上传时间后 60 天

      //广告监测信息
      "landing_page": "http://ads.tencent.com/",
      "target_url": "http://click.dsp1.com?bidid=%%BID_REQUEST_ID%%&clickparam=%%CLICK_PARAM%",
      "monitor_url1": "http://view.dsp1.com?bidid=%%BID_REQUEST_ID%%&win=%%WIN_PRICE%%&impparam=%%IMPRESSION_PARAM%",
      "monitor_url2": "http://view.third1.com?os=__OS__&imei=__IMEI__&idfa=__IDFA__&ip=__IP__&ua=__UA__&ts=__TS__",
      "monitor_url3": "http://view.third2.com?os=__OS__&imei=__IMEI__&idfa=__IDFA__&ip=__IP__&ua=__UA__&ts=__TS__",

      //广告样式信息（参考“Tencent Ad Exchange 广告样式说明”文档）
      "creative_spec": 148,
      "product_type": 1,
    }
  ]
}
```

```

        "multimedia1_file_url": "http://www.images1.jpg",
        "multimedia2_file_url": "http://www.images2.jpg",
        "app_id":, //当为下载广告时 (product_type=2/3), 需填写此字段
        "pkgname": "", //当希望投放 Android 渠道包下载广告时 (product_type=2), 需填写
此字段
        "title": "腾讯社交广告", //根据 “Tencent Ad Exchange 广告样式说明” 填写
        "description": "欢迎大家访问腾讯社交广告官网", //根据 “Tencent Ad Exchange 广
告样式说明” 填写
        "call_to_action": "敬请浏览", //根据 “Tencent Ad Exchange 广告样式说明” 填写
        "deep_link_uri": "" //当投放 DeepLink 广告时填写
    }
]
}

```

#### 注:

- 1、Feeds(creative\_spec=65)需要在广告主层级上传 brand\_logo\_image\_url 和 brand name 字段
- 2、动态创意广告只需上传“广告基本信息”、“广告样式信息”中的 creative\_spec(220/222) 和 product\_type (1) 字段
- 3、App 下载广告无需上传“广告监测信息”中的 landing\_page 字段
- 4、Android App 下载广告需要在广告主层级填写 qq 字段 (要求见 4.2.1 广告主创建), 若未填写, 广告上传会提示错误
- 5、广告样式部分需要参考“Tencent Ad Exchange 广告样式说明”。其中标明为拉取的字段为 Tencent Ad Exchange 自行拉取的内容, 请勿填写

#### 应答示例:

```

{
  "code": 0,
  "message": "操作成功",
  "data": [
    {
      "code": 0,
      "message": "操作成功",
      "data": {
        "creative_id": "112"
      }
    }
  ]
}

```

### 4.3.2 广告修改

请求 URL: <https://api.e.qq.com/adx/v1/creative/update>

请求字段:

类型	数组类型	描述
Creative[]	是	最多 10 个

请求类型 Creative（未修改字段也需填写原值，可修改字段更新时会触发再审核）:

名称	类型	必填	限制	描述
advertiser_id	string	是	小于 128 字节	不可修改
contract_code	string	否	小于 255 字节	可修改
creative_id	string	是	小于 128 字节	不可修改
product_type	int32	否		可修改
app_id	int64	否		可修改
pkgname	string	否	小于 256 字节	可修改
landing_page	string	是	小于 1024 字节	可修改
target_url	string	是	小于 1024 字节	可修改
monitor_url1	string	是	小于 1024 字节	可修改
monitor_url2	string	否	小于 1024 字节	可修改
monitor_url3	string	否	小于 1024 字节	可修改
begin_date_included	date	否		可修改
end_date_included	date	否		可修改
creative_spec	int32	是		可修改
title	string	否	小于 30 字节	可修改
description	string	否	小于 256 字节	可修改
multimedia1_file_url	string	否	小于 1024 字节	可修改
multimedia2_file_url	string	否	小于 1024 字节	可修改
call_to_action	string	否	4 字以内	可修改
deep_link_uri	string	否	小于 1024 字节	可修改

应答字段:

名称	类型	描述
creative_id	string	

请求示例:



```
{
  "data": [
    {
      "advertiser_id": "1347",
      "creative_id": "112",
      "contract_code": "",
      "begin_date_included": "2016-01-01",
      "end_date_included": "2016-03-31",
      "landing_page": "http://ads.tencent.com/",
      "target_url": "http://click.dsp1.com?bidid=%%BID_REQUEST_ID%%&clickparam=%%CLICK_PARAM%%",
      "monitor_url1": "http://view.dsp1.com?bidid=%%BID_REQUEST_ID%%&win=%%WIN_PRICE%%&impparam=%%IMPRESSION_PARAM%%",
      "monitor_url2": "http://view.third1.com?os=__OS__&imei=__IMEI__&idfa=__IDFA__&ip=__IP__&ua=__UA__&ts=__TS__",
      "monitor_url3": "http://view.third2.com?os=__OS__&imei=__IMEI__&idfa=__IDFA__&ip=__IP__&ua=__UA__&ts=__TS__",
      "creative_spec": 148,
      "product_type": 1,
      "multimedia1_file_url": "http://www.images1.jpg",
      "multimedia2_file_url": "http://www.images2.jpg",
      "app_id": ,
      "pkgname": "",
      "title": "腾讯社交广告",
      "description": "欢迎大家访问腾讯社交广告官网",
      "call_to_action": "访问",
      "deep_link_uri": ""
    }
  ]
}
```

应答示例:

```
{
  "code": 0,
  "message": "操作成功",
  "data": [
    {
      "code": 0,
      "message": "操作成功",
      "data": {
        "creative_id": "112"
      }
    }
  ]
}
```

```
}
```

### 4.3.3 广告审核状态获取

请求 URL: [https://api.e.qq.com/adx/v1/creative/get\\_review\\_status](https://api.e.qq.com/adx/v1/creative/get_review_status)

请求字段:

类型	数组类型	描述
AdvertiserCreative[]	是	最多 20 个

请求类型 AdvertiserCreative:

名称	类型	限制	描述
advertiser_id	string	小于 128 字节	广告主 Id
creative_id	string	小于 128 字节	广告 id

返回类型 CreativeReviewStatus:

名称	类型	描述
advertiser_id	string	
creative_id	string	
review_status	string	具体值参见附录 2 <a href="#">审核状态</a>
review_msg	string	拒绝原因
industry_id	int64	审核员校正后的行业 ID

请求示例:

```
{
  "data": [
    {
      "advertiser_id": "10010001",
      "creative_id": "100199991"
    },
    {
      "advertiser_id": "10010001",
      "creative_id": "100199992"
    },
    {
      "advertiser_id": "10010001",
      "creative_id": "100199993"
    }
  ]
}
```

应答示例:

```
{
  "code": 0,
  "message": "操作成功",
  "data": [
    {
      "code": 0,
      "message": "操作成功",
      "data": {
        "advertiser_id": "10010001",
        "creative_id": "100199991",
        "review_status": "APPROVED",
        "review_msg": "",
        "industry_id": 21474836482
      }
    },
    {
      "code": 0,
      "message": "操作成功",
      "data": {
        "advertiser_id": "10010001",
        "creative_id": "100199992",
        "review_status": "PENDING",
        "review_msg": ""
      }
    },
    {
      "code": 0,
      "message": "操作成功",
      "data": {
        "advertiser_id": "10010001",
        "creative_id": "100199993",
        "review_status": "REJECTED",
        "review_msg": ""
      }
    }
  ]
}
```

## 4.4 Pretargeting

很多 DSP 会基于自己的带宽以及业务的侧重点对 ADX 的流量进行筛选, 这一操作称为 Pretargeting (流量预定位)。通过 Buyer API 提供的接口, DSP 可以管理自己的

Pretargeting 配置。

### 4.4.1 创建 Pretargeting 配置

请求 URL: <https://api.e.qq.com/adx/v1/pretargeting/add>

请求类型 PretargetingConfig:

名称	类型	必填	限制	描述
pretargeting_id	string	是	小于 128 字符	DSP 为 Pretargeting 配置分配的唯一 ID。可以为 ID，也可以填写便于阅读的文本名称，DSP 需保证其唯一性
is_active	int32	是	只允许 0 和 1 两个值	该 Pretargeting 配置是否生效。1 表示有效，0 表示无效，0 以外的值被认为是 1，默认为 1
targeting_definition	object	是		Pretargeting 配置，Tencent Ad Exchange 根据该字段所指定的定向条件进行流量预定位

通过 Pretargeting 创建接口可以创建一条 Pretargeting 配置，当该 Pretargeting 配置被命中时，BidRequest 将会转发给 DSP。

targeting\_definition 对象描述了 DSP 对流量的期望，其是一个如下格式的对象：

```
{
  "area":
  {
    "included": ["710000", "540000"],
    "excluded": ["710200"]
  },
  "device_type":
  {
    "included": ["DEVICE_TYPE_MOBILE"]
  }
}
```

示例表明：地域定向台湾省(710000)或西藏自治区(540000)，排除高雄市(710200)，并且设备类型定向移动设备(DEVICE\_TYPE\_MOBILE)。

targeting\_definition 对象中的每个属性（如 area）代表一个定向维度，称为定位器。每个定位器都支持对其定向值进行 included（包含）和 excluded（排除）操作。included 和 excluded 的值即是定向值，定向值都为 string 类型。定位器内为或关系，定位器之间为

与关系。ADX 目前支持的定向维度和定向值请见如下表格：

定位器	定位器名称	定向值取值	定向值说明
area	地域	国标地域编码	详见“Tencent Ad Exchange 地域码”
device_type	设备类型	DEVICE_TYPE_OTHER	其他设备类型
		DEVICE_TYPE_MOBILE	移动设备
		DEVICE_TYPE_PC	PC
operating_system	操作系统	OPERATING_SYSTEM_OTHER	其他操作系统
		OPERATING_SYSTEM_IOS	iOS 系统
		OPERATING_SYSTEM_ANDROID	Android 系统
		OPERATING_SYSTEM_WINDOWS	Windows 系统
siteset	站点集	SITESET_QZONE	QQ 空间（PC）
		SITESET_QQ_CLIENT	QQ 客户端（PC）
		SITESET_QQ_MUSIC	QQ 音乐
		SITESET_MOBILE_UNION	腾讯广告联盟
		SITESET_WECHAT	微信
		SITESET_MYAPP_MOBILE	应用宝（移动）
		SITESET_MYAPP_PC	应用宝（PC）
		SITESET_OTHER_INNER_MOBILE	腾讯社交广告内部移动站点
placement_type	广告样式	PLACEMENT_TYPE_BANNER	Banner 广告
		PLACEMENT_TYPE_INLINE	插屏广告
		PLACEMENT_TYPE_SPLASH	开屏广告
		PLACEMENT_TYPE_FEEDS	Feeds 广告
		PLACEMENT_TYPE_NATIVE	原生广告

请求示例：

```
{
  "data" : {
    "pretargeting_id" : "123"//也可以填写如"台湾和西藏的移动设备"
    "is_active" : 1,
    "targeting_definition":{
      "area": {
        "included": ["710000", "540000"],
        "excluded": ["710200"]
      },
      "device_type": {
        "included": ["DEVICE_TYPE_MOBILE"]
      }
    }
  }
}
```

```
}  
}  
}  
}
```

应答示例:

```
{  
  "code": 0,  
  "message": "操作成功",  
  "data" : {}  
}
```

## 4.4.2 更新 Pretargeting 配置

请求 URL: <https://api.e.qq.com/adx/v1/pretargeting/update>

请求类型 PretargetingConfig:

名称	类型	必填	限制	描述
config_id	string	是	小于 128 字符	DSP 为 Pretargeting 配置分配的唯一 ID, 必须是已存在的 ID
is_active	int32	是	只允许 0 和 1 两个值	该 Pretargeting 配置是否生效.1 表示有效,0 表示无效, 0 以外的值被认为是 1, 默认为 1
targeting_definition	object	是		格式要求见 4.4.1 创建 Pretargeting 配置

更新 Pretargeting 配置时, 字段要求与创建 Pretargeting 时要求相同, 但是 pretargeting\_id 必须是已存在的。另外, 更新采用的是全覆盖式更新,即更新时每个定位器都要填值,否则定位器将会被设置为系统默认值。另外, 通过 is\_active 的值可以控制当前这一条 Pretargeting 的配置是否生效。

请求示例:

```
{  
  "data" : {  
    "pretargeting_id" : "123">//也可以填写如"台湾和西藏的移动设备"  
    "is_active" : 1,  
    "targeting_definition":{  
      "area": {  
        "included": ["710000", "540000"],  
        "excluded": ["710200"]  
      }  
    }  
  }  
}
```

```
    },
    "device_type": {
      "included": ["DEVICE_TYPE_MOBILE"]
    }
  }
}
```

应答示例:

```
{
  "code": 0,
  "message": "操作成功",
  "data": {}
}
```

### 4.4.3 删除 Pretargeting 配置

请求 URL: <https://api.e.qq.com/adx/v1/pretargeting/delete>

请求字段:

名称	类型	必填	描述
pretargeting_id	string	是	要删除的 Pretargeting 配置的 id

请求示例:

```
{
  "data": {
    "pretargeting_id": "123"//也可以填写如"台湾和西藏的移动设备"
  }
}
```

应答示例:

```
{
  "code": 0,
  "message": "操作成功",
  "data": null
}
```

### 4.4.4 根据 ID 获取 Pretargeting 配置

请求 URL: <https://api.e.qq.com/adx/v1/pretargeting/get>

请求字段：

名称	类型	必填	描述
pretargeting_id	string	是	要获取相应 Pretargeting 配置的配置 id

返回类型为 PretargetingConfig。

若提供的 id 不存在,接口同样会返回“操作成功”，但 data 字段的值为 null。

请求示例：

```
{
  "data" : {
    "pretargeting_id" : "123"//也可以填写如"台湾和西藏的移动设备"
  }
}
```

应答示例：

```
{
  "code": 0,
  "message": "操作成功",
  "data" : {
    "pretargeting_id" : "123",
    "is_active" : 1,
    "targeting_definition":{
      "area": {
        "included": ["710000","540000"],
        "excluded": ["710200"]
      },
      "device_type": {
        "included": ["DEVICE_TYPE_MOBILE"]
      }
    }
  }
}
```

#### 4.4.5 获取所有的 Pretargeting 配置

请求 URL: <https://api.e.qq.com/adx/v1/pretargeting/list>

请求字段：

名称	类型	必填	描述
无			



返回类型为 PretargetingConfig 数组。

将会返回所有的 Pretargeting 配置，不支持按条件过滤。请求体必须是{"data":null}，请求体中没有 data 字段将会被认为是错误的。

请求示例：

```
{
  "data" : null
}
```

应答示例：

```
{
  "code": 0,
  "message": "操作成功",
  "data" : [
    {
      "pretargeting_id" : "123",
      "is_active" : 1,
      "targeting_definition":{
        "area": {
          "included": ["710000", "540000"],
          "excluded": ["710200"]
        },
        "device_type": {
          "included": ["DEVICE_TYPE_MOBILE"]
        }
      },
      "pretargeting_id" : "124",
      "is_active" : 1,
      "targeting_definition":{
        "area": {
          "included": ["710000", "540000"],
          "excluded": ["710200"]
        },
        "siteset": {
          "included": ["SITESET_QQ_CLIENT"]
        }
      },
      "pretargeting_id" : "125",
      "is_active" : 1,
      "targeting_definition":{
        "area": {
          "included": ["110000", "310000"]
        }
      },
    }
  ]
}
```

```

    "operating_system": {
      "included": ["OPERATING_SYSTEM_IOS"]
    }
  }
]
}

```

## 4.5 频次控制

Tencent Ad Exchange 提供的频次控制功能，支持对同一个广告主下的一个或多个广告进行联合频次控制。通过频次控制接口，DSP 可以管理自己的频次控制的配置（简称“频次配置”）。

### 4.5.1 创建频次配置

请求 URL: [https://api.e.qq.com/adx/v1/frequency\\_cap/add](https://api.e.qq.com/adx/v1/frequency_cap/add)

请求类型 FrequencyCap:

名称	类型	必填	限制	描述
frequency_cap_id	string	是	小于 128 字符	DSP 为频次配置分配的唯一 ID，需保证其唯一性
advertiser_id	string	是	小于 128 字符	需要进行频次控制的广告主 ID
creative_id_list	string[]	是		广告 ID 数组，广告 ID 个数不大于 100 个。且每个广告只能关联一个频次配置
period_type	enum	是		频次控制周期。取值见附录 2 <a href="#">频控周期类型 (PeriodType)</a>
expire_date	date	是		频次配置有效时间（格式为 yyyy-MM-dd）。开始时间为配置生效当天，结束时间为 expire_date（中间 <b>持续时间需要在 30 天内</b> ）
frequency_limit	int32	是	小于 1000	频次控制数。 creative_id_list 涉及到的广告展示总和的上限

通过频次配置新增接口，可以为某个广告主下的一个或者多个广告进行频次控制配置。频次配置生效后，Tencent Ad Exchange 将会控制该频次配置下的所有广告，在频控周期内频次数不超过 frequency\_limit 次。

请求示例：

```
{
  "data" : {
    "frequency_cap_id" : "123",
    "advertiser_id" : "advertiser_123",
    "creative_id_list" : ["creative1","creative2","creative3"],
    "period_type" : "ALWAYS",
    "expire_date" : "2016-06-28",
    "frequency_limit" : 10
  }
}
```

应答示例:

```
{
  "code": 0,
  "message": "操作成功",
  "data" : null
}
```

## 4.5.2 根据 ID 获取频次配置

请求 URL: [https://api.e.qq.com/adx/v1/frequency\\_cap/get](https://api.e.qq.com/adx/v1/frequency_cap/get)

请求字段:

名称	类型	必填	限制	描述
frequency_cap_id	string	是	小于 128 字符	DSP 为频次配置分配的唯一 ID

返回类型为 FrequencyCap。

若提供的 id 不存在,接口同样会返回“操作成功”,但 data 字段的值为 null。

请求示例:

```
{
  "data" : {
    "frequency_cap_id" : "123"
  }
}
```

应答示例:

```
{
  "code": 0,
```

```
"message": "操作成功",
"data": {
  "frequency_cap_id": "123",
  "advertiser_id": "advertiser_123",
  "creative_id_list": ["creative1", "creative2", "creative3"],
  "period_type": "ALWAYS",
  "expire_date": "2016-06-28",
  "frequency_limit": 10
}
```

### 4.5.3 删除频次配置

请求 URL: [https://api.e.qq.com/adx/v1/frequency\\_cap/delete](https://api.e.qq.com/adx/v1/frequency_cap/delete)

请求字段:

名称	类型	必填	限制	描述
frequency_cap_id	string	是	小于 128 字符	DSP 为频次配置分配的唯一 ID

该接口用以从 Tencent Ad Exchange 服务器中删除该 id 指定的配置。

请求示例:

```
{
  "data": {
    "frequency_cap_id": "123"
  }
}
```

应答示例:

```
{
  "code": 0,
  "message": "操作成功",
  "data": null
}
```

## 4.6 报表

通过数据报表接口可以获取竞价报表和消耗报表,所有报表都是按照天维度进行统计的。查询操作当日上午 8 点之后可以获取到前一天的相关报表数据(如遇特殊情况会有所延迟),目前暂时无法给出查询当日的相关数据。一次最多可查询连续 30 天的数据。

## 4.6.1 竞价报表

请求 URL: <https://api.e.qq.com/adx/v1/report/bid>

请求字段:

类型	数组类型	描述
ReportQuery	否	

请求类型 ReportQuery:

名称	类型	必填	描述
start_date	date	是	查询的开始日期, 必须为查询操作当日之前一年以内的日期, 格式为 yyyy-MM-dd, 例如 2015-04-08
end_date	date	是	查询的结束日期, 必须为查询操作当日之前一年以内的日期, 且必须在查询开始日期之后的 30 天以内, 格式为 yyyy-MM-dd, 例如 2015-04-18

应答字段:

名称	类型	描述
date	int64	竞价统计的日期 (如 20150408)
bid_request	int64	竞价请求数
bidding	int64	参与竞价数
bid_success	int64	竞价成功数
bid_fail	int64	竞价失败数
non_bidding	int64	未参与竞价数
timeout	int64	超时数
parse_fail	int64	解析失败数

请求示例:

```
{
  "data": {
    "start_date": "2015-04-08",
    "end_date": "2015-04-09"
  }
}
```

应答示例:

```
{
  "code": 0,
```

```

"message": "操作成功",
"data": [
  {
    "date": 20150408,
    "bid_request": 100000000,
    "bidding": 95000000,
    "bid_success": 30000000,
    "bid_fail": 65000000,
    "non_bidding": 2000000,
    "timeout": 2000000,
    "parse_fail": 1000000
  },
  {
    "date": 20150409,
    "bid_request": 110000000,
    "bidding": 100000000,
    "bid_success": 40000000,
    "bid_fail": 60000000,
    "non_bidding": 8000000,
    "timeout": 1000000,
    "parse_fail": 1000000
  }
]
}

```

## 4.6.2 竞价失败原因统计报表

请求 URL: [https://api.e.qq.com/adx/v1/report/bid\\_fail](https://api.e.qq.com/adx/v1/report/bid_fail)

请求字段:

类型	数组类型	描述
ReportQuery	否	

请求类型 ReportQuery:

名称	类型	必填	描述
start_date	date	是	查询的开始日期, 须为查询当日之前一年以内的日期, 格式为 yyyy-MM-dd, 例如 2015-07-01
end_date	date	是	查询的结束日期, 须为查询当日之前一年以内的日期, 且必须在查询开始日期之后的 30 天以内, 格式为 yyyy-MM-dd, 例如 2015-07-01
advertiser_id	string	否	广告主 id, 不为空时, 接口返回指定广告主的统计结果, 否则返回 DSP 的整体统计结果

creative_id	string	否	广告 id，指定要统计竞价失败原因的广告。当此字段不为空时， advertiser_id 字段也不能为空
reason_code	int32	否	竞价失败原因编号，用来指定查询某个失败原因的数据；当此字段为空时，默认返回所有竞价失败原因的统计数据。

应答字段：

名称	类型	描述
date	int64	竞价失败原因统计的日期（如 20150701）
bid_count	int64	竞价失败数
reason_code	int32	竞价失败原因编号
reason_message	string	竞价失败原因

竞价失败原因列表：

编号	竞价失败原因
0	其他
1	素材状态异常
2	广告主状态异常
3	合约状态异常
4	素材规格与流量不符
5	出价低于底价
6	被合约的广告位白名单过滤
7	非最高价
8	返回的 app 素材信息缺失
9	被广告位行业黑名单过滤
10	原生广告素材要素不符合广告位要求
11	原生广告素材商品类型不符合广告位要求
12	素材文件格式（JPG、GIF 等）不对
13	被广告位关键字黑名单过滤
14	素材准播模块过滤
15	广告位广告主白名单过滤
16	广告位广告主黑名单过滤
17	返回的 impression_param 过长
18	获取广告素材失败
19	动态创意的 bid response 不合法
20	被流量侧曝光频次规则过滤
21	App 已安装过滤
22	DSP 点击地址超长
23	曝光次数到达广告主侧要求的频次上限
24	优先级低于 PMP 广告

请求示例：

```
{
  "data": {
    "start_date": "2015-01-01",
    "end_date": "2015-01-09",
    "advertiser_id": "123"
  }
}
```

应答示例：

```
{
  "code": 0,
  "message": "操作成功",
  "data": [
    {
      "bid_count": 11,
      "date": 20160101,
      "reason_code": 2,
      "reason_message": "广告主状态异常"
    },
    {
      "bid_count": 56,
      "date": 20160102,
      "reason_code": 1,
      "reason_message": "素材状态异常"
    }
  ]
}
```

### 4.6.3 消耗报表

请求 URL: <https://api.e.qq.com/adx/v1/report/exposure>

请求字段：

类型	数组类型	描述
ReportQuery	否	

请求类型 ReportQuery:

名称	类型	必填	描述
----	----	----	----



start_date	date	是	查询的开始日期，必须为查询当日之前一年以内的日期，格式为 yyyy-MM-dd，例如 2015-04-08
end_date	date	是	查询的结束日期，必须为查询当日之前一年以内的日期，且必须在查询开始日期之后的 30 天以内，格式为 yyyy-MM-dd，例如 2015-04-18

应答字段：

名称	类型	描述
date	int64	曝光统计的日期（如 20150408）
pv	int64	展示数，未经反作弊系统的原始曝光量
click	int64	点击数
avg_cpm	int64	平均 cpm 单价。单位：分。 $\text{avg\_cpm} = \text{cost} / \text{valid\_pv} * 1000$
cost	int64	当日消耗。单位：分
valid_pv	int64	反作弊系统判定为有效的曝光量
valid_click	int64	反作弊系统判定为有效的点击量

请求示例：

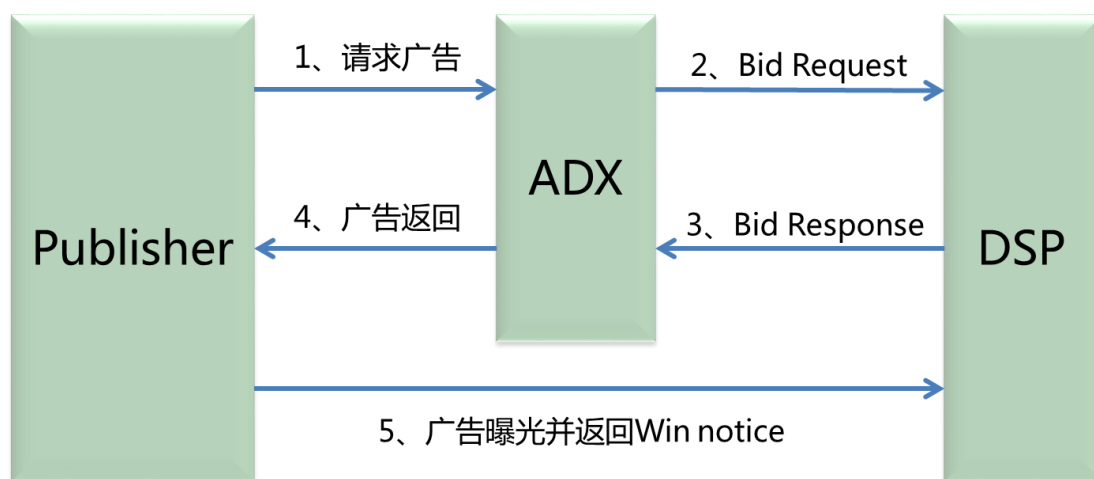
```
{
  "data": {
    "start_date": "2015-04-08",
    "end_date": "2015-04-09"
  }
}
```

应答示例：

```
{
  "code": 0,
  "message": "操作成功",
  "data": [
    {
      "date": 20150408,
      "pv": 100000000,
      "click": 1000000,
      "avg_cpm": 2632,
      "cost": 250000000,
      "valid_pv": 95000000,
      "valid_click": 900000
    },
    {
      "date": 20150409,
      "pv": 104000000,
```

```
"click": 1040000,  
"avg_cpm ": 2704,  
"cost ": 270400000,  
"valid_pv": 100000000,  
"valid_click": 1000000  
}  
]  
}
```

## 5. 实时竞价 RTB 接口



1、实时竞价部分包含如下两部分接口：

- 1) Tencent Ad Exchange 向 DSP 发送的广告请求消息（Bid Request）；
- 2) DSP 向 Tencent Ad Exchange 返回的竞价广告消息（Bid Response）。

2、竞价流程如图：

- 1) 媒体方向 Tencent Ad Exchange 发起广告请求；
- 2) Tencent Ad Exchange 收到媒体方的广告请求后向 DSP 发送 Bid Request(竞价请求)；
- 3) DSP 收到 Bid Request 后根据其中携带的信息及需求决定是否参与竞价及出价策略，将这些信息写入 Bid Response 并返回给 Tencent Ad Exchange；
- 4) Tencent Ad Exchange 选择出价最高的广告回复给媒体方；
- 5) 广告曝光并返回 Win notice（包括成交价等信息）。

### 5.1 通信方式

Tencent Ad Exchange 和 DSP 之间的基础通信协议采用 HTTP 协议，使用 POST 方法发送 Bid Request 消息，数据使用 Protobuf 格式，Bid Request 序列化之后作为 HTTP 请求的 Content 来发送。响应的时候 Bid Response 序列化之后作为 HTTP 响应的 Content 来

返回。

请在您的服务器上启用 **HTTP 长连接服务（必须为长连接）**，连接的超时时间至少设置为 10s，避免为每个请求建立 tcp 连接而消耗时间。

## 5.2 编码方式

若无特殊说明，所有输入参数、返回结果中的字节编码全部使用 UTF-8（无 BOM 头）。

## 5.3 HTTP 请求头

Content-Type: application/x-protobuf

## 5.4 HTTP 响应

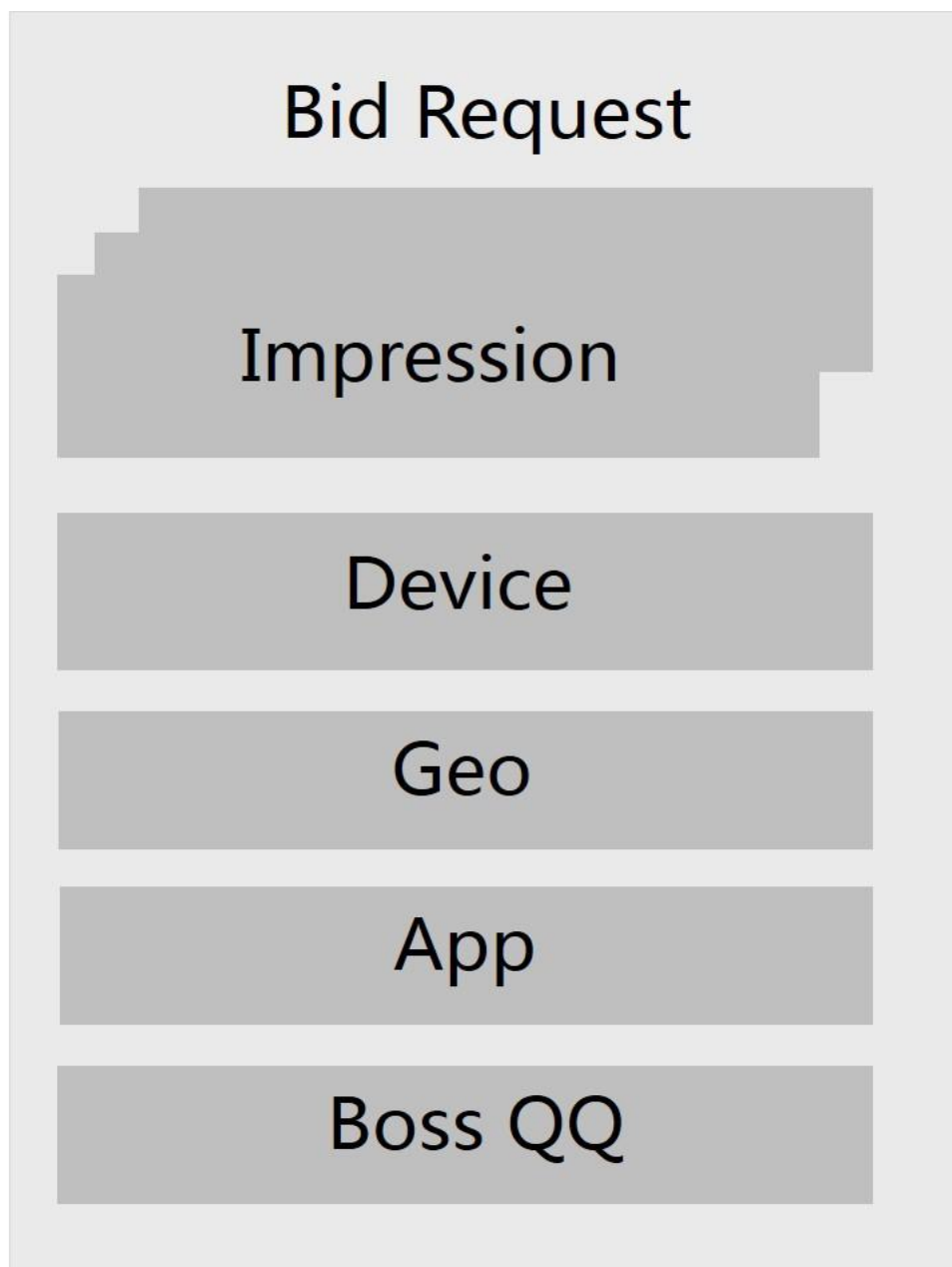
无论是否参与竞价，返回码都为 200，Content-Type: application/x-protobuf，content 为 Bid Response 序列化之后的串。如果不参与竞价，仅填充 Bid Response 的 id 返回即可。

## 5.5 超时时间

Tencent Ad Exchange 向 DSP 发送请求到 DSP 返回响应的时间：Feeds 流量（creative\_spec=65 或者 Pretargeting 中 placement\_type=PLACEMENT\_TYPE\_FEEDS）为 100ms，其他流量为 150ms。超时部分视为 DSP 放弃本次出价。

## 5.6 消息定义

### 5.6.1 Bid Request



#### 5.6.1.1 Bid Request 类型

名称	类型	必填	限制	描述
id	string	是	小于 128 字节	ADX 提供的 Bid Request 唯一标识
is_ping	bool	否		true 表示探测网络延

				迟，不触发竞价逻辑。 DSP 设置 request_id 和 processing_time_ms 后立即返回 Bid Response。DSP 必须支持此特性。
is_test	bool	否		true 表示测试请求，竞价成功的广告不会被展示和计费。DSP 对此类请求的处理完全同普通请求。DSP 必须支持此特性。
impressions	array of Impression	否		每个 impression 代表一个广告位的请求
device	device	否		设备信息
ip	string	否	小于 64 字节	设备联网 ip
area_code	int32	否		部分流量上可能没有有效 IP，此时用地域码做定向。取值见《广点通地域码》
user	user	否		用户信息，仅对部分 DSP 开放。与 Device 类型中的 id 不会同时出现
geo	geo	否		地理位置信息
app	app	否		App 信息
boss_qq	uint64	否		填充 Boss QQ 号。 Boss QQ 功能：该功能满足广告主希望在投放期内看到自己的广告需求（目前仅在腾讯内部流量中适用）。Boss QQ 配置在广告主层级。每个广告主 ID 最多绑定 10 个 QQ 号，不同广告主 ID 可绑定相同的 QQ 号。当某个 Boss QQ 号访问 Tencent Ad Exchange 的流量时，Bid Request 中 boss_qq 字段中会填充相应的 QQ 号，若 DSP 希望与此

				QQ 号绑定的广告主 (ID) 的广告能以较高的几率展示, 则需回复该广告主的广告, 否则正常回复其他未做绑定操作的广告主的广告。详细信息参见《DSP 常见问题解答》中 6.1.8
support_deep_link	bool	否		广告位是否支持 deep link, 默认为不支持
wx_flow_class	array of uint64	否		微信公众号分类 ID 信息。暂未启用。

### 5.6.1.2 Impression 类型

名称	类型	必填	限制	描述
id	string	是	小于 128 字节	Bid Request 范围内, Impression 唯一标识
placement_id	int64	是		ADX 提供的广告位 id
creative_specs	array of int32	是		广告位支持的素材规格, 取值见《广告样式说明》
bid_floor	int32	是		广告位的 CPM 底价。单位: 分
blocking_industry_id	array of int64	否		广告位过滤的行业 ID 列表, 取值见《广告主行业分类》。
natives	native	否		原生广告位信息
contract_code	string	否	小于 255 字节	广告位上配置的合同号。当此字段有值时, DSP 需回复符合相应合同的广告。
multimedia_type_white_list	string	否	小于 256 字节	广告位支持的素材类型。取值范围为"gif", "jpeg", "png", "swf", "psd", "bmp", "tiff", "tiff", "jpc", "jpx", "jb2", "swc", "iff", "wbmp", "xbm", "webp", "flv", "wav", "mp3", "mp4", "avi"。字段为空时, 格式不做限制。
blocking_keyword	array of string	否		广告位过滤的关键词, <b>模糊匹配</b>

advertiser_whitelist	array of string	否		广告位的广告主白名单。 当 advertiser_whitelist 非空时，该广告位只接受白名单广告主的广告。
advertiser_blacklist	array of string	否		广告位过滤的广告主
pretargeting_ids	array of uint64	否		请求命中的 Pretargeting 配置包 ID

## Native 类型

名称	类型	必填	限制	描述
required_fields	uint64	否		原生广告位强制需要的组件，DSP 返回的广告至少需要满足此要求
type	enum Type	否		原生广告位要求的广告类型参见附录 2 <a href="#">原生广告类型</a> 。应用类广告需根据 Device 类型中的 os 字段判断操作系统

## 5.6.1.3 Device 类型

名称	类型	必填	限制	描述
id	string	否	小于 32 字节	设备唯一标示。 Android 用 IMEI md5sum，iOS 用 IDFA md5sum。微信流量不传递该字段。
device_type	enum DeviceType	否		设备类型。具体值参见附录 2 <a href="#">设备类型</a>
os	enum OperatingSystem	否		设备运行的操作系统。具体值参见附录 2 <a href="#">移动设备操作系统</a>
os_version	string	否	小于 128 字节	操作系统版本,如 "4.2.1"。
user_agent	string	否	小于 1024 字节	user agent, 如 "Mozilla/5.0"。
screen_width	int32	否		屏幕宽度。单位：像素
screen_height	int32	否		屏幕高度。单位：像素
dpi	int32	否		屏幕每英寸像素
carrier	enum Carrier	否		运营商。具体值参见

				附录 2 <a href="#">设备运营商</a>
connection_type	enum ConnectionType	否		设备联网方式。具体值参见附录 2 <a href="#">设备联网方式</a>
brand_and_model	string	否	小于 128 字节	设备品牌型号
language	string	否	小于 32 字节	设备语言，如 zh en 等
idfa	string	否		iOS 设备 IDFA 原文
manufacturer	string	否		设备制造商信息
android_id	string	否		Android 设备 Android ID 密文，采用 md5sum 方式加密

### 5.6.1.4 User 类型

名称	类型	必填	限制	描述
id	string	否	小于 128 字节	Cookie Mapping ID（具体参见文档《腾讯广点通 Cookie Mapping 服务对接说明文档》）。与 user_demographic 不会同时给出。
user_demographic	Demographic	否		人口学信息。与 id 不同时给出。
verticals	array of Vertical	否		用户商业兴趣

### Demographic 类型

名称	类型	必填	限制	描述
gender	enum Gender	否		用户性别信息。具体值参见附录 2 <a href="#">用户性别</a>
education	enum Education	否		用户学历信息。具体值参见附录 2 <a href="#">用户学历</a>
age_low	int32	否		用户年龄范围中的最小值（含）
age_high	int32	否		用户年龄范围中的最大值（含）
consume_level	enum ConsumptionAbilityLevel	否		用户消费能力信息。具体值参见附录 2 <a href="#">用户消费能力</a>



house_price	uint32	否		用户住房价格。单位：元/平方米
-------------	--------	---	--	-----------------

## Vertical 类型（暂未启用）

名称	类型	必填	限制	描述
id	uint32	否		用户兴趣 ID
weight	float	否		用户兴趣权重

## 5.6.1.5 Geo 类型

名称	类型	必填	限制	描述
latitude	int32	否		纬度 * 1 000 000。采用火星坐标系
longitude	int32	否		经度 * 1 000 000。采用火星坐标系
accuracy	double	否		经纬度精度半径。单位：米

## 5.6.1.6 App 类型

名称	类型	必填	限制	描述
industry_id	int64	否		App 所属行业 id，取值见《APP 行业分类》
app_bundle_id	string	否	小于 256 字节	App 的唯一标识。Android 应用是 package name，iOS 应用是 bundle id

## 5.6.1.7 Bid Request 示例

## 常规请求示例

```

id: "6dfv7gekg5f7w"
is_ping: false
is_test: false
impressions {
  id: "0"
  placement_id: 4563245454546
  creative_specs: 135
  bid_floor: 20
  blocking_industry_id: 21474839087
}

```

```

device {
  id: 76fe209711707e9fa3f7d3073623aa6b //id 示例，实际请求中 id 和用户标签不会同时提
供
  device_type: DEVICETYPE_MOBILE
  os: OS_ANDROID
  os_version: "4.2.1"
  carrier: CARRIER_UNKNOWN
  connection: CONNECTIONTYPE_WIFI
  brand_and_model: Redmi
  language: zh
  manufacturer: Xiaomi
}
ip: "212.21.1.45"
user {
  user_demographic {
    gender: GENDER_MALE
    education: EDUCATION_DOCTOR
    age_low: 31
    age_high: 40
    consume_level: CONSUMPTION_ABILITY_LEVEL_HIGH
    house_price: 50000
  }
}
app {
  industry_id: 21474840082
  app_bundle_id: "com.tencent.mobileqq"
}
boss_qq: 123456789

```

### 原生广告请求示例

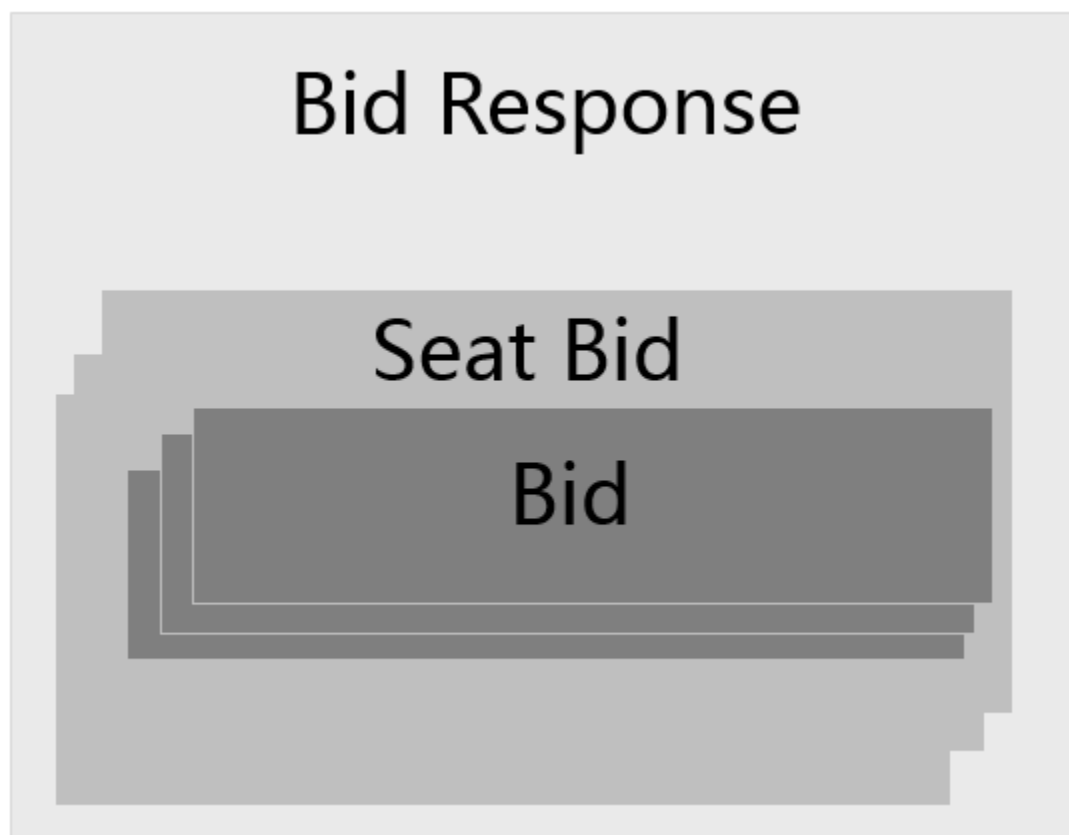
```

id: "6dfv7gekg5f7y"
is_ping: false
is_test: false
impressions {
  id: "0"
  placement_id: 4563245454547
  creative_specs: 147
  bid_floor: 20
  blocking_industry_id: 21474839087
  natives {
    required_fields: 15
    type= APP
  }
  natives {

```

```
    required_fields: 15
    type= LINK
  }
}
device {
  id: 76fe209711707e9fa3f7d3073623aa6b
  device_type: DEVICETYPE_MOBILE
  os: OS_ANDROID
  os_version: "4.2.1"
  carrier: CARRIER_UNKNOWN
  connection: CONNECTIONTYPE_WIFI
  brand_and_model: Redmi
  language: zh
  manufacturer: Xiaomi
}
ip: "212.21.1.45"
user {
  user_demographic {
    gender: GENDER_MALE
    education: EDUCATION_DOCTOR
    age_low: 31
    age_high: 40
    consume_level: CONSUMPTION_ABILITY_LEVEL_HIGH
    house_price: 50000
  }
}
geo {
  latitude: 100000000
  longitude: 1000000000
  accuracy: 15.0
}
app {
  industry_id: 21474840082
  app_bundle_id: "com.ss.android.essay.joke"
}
boss_qq: 123456789
```

## 5.6.2 Bid Response



### 5.6.2.1 Bid Response 类型

名称	类型	必填	限制	描述
request_id	string	是	小于 128 字节	ADX 提供的 Bid Request 唯一标识。
seat_bids	array of seatbid	否		DSP 的出价集合, 每个 seat_bid 对应 Bid Request 中的一个 impression
processing_time_ms	int32	否		收到 Bid Request 至发送完 Bid Response 的用时。单位: 毫秒

### 5.6.2.2 Seat Bid 类型

名称	类型	必填	限制	描述
impression_id	string	是	小于 128 字节	Bid Request 中对应

				Impression 的 id。该字段必填。
bids	array of bid	否		DSP 返回的广告结果。

### 5.6.2.3 Bid 类型

名称	类型	必填	限制	描述
creative_id	string	是	小于 128 字节	DSP 侧的素材 id。
bid_price	int32	是		DSP 的 CPM 出价。单位：分。
impression_param	string	否	移动联盟流量需保证做完宏替换的曝光监测地址 base64 encode 之后小于 1024 字节；其他流量需保证做完宏替换的曝光监测地址 url encode 之后小于 1024 字节	替换 DSP 曝光监测 URL（monitor_url1）里的宏%%IMPRESSION_PARAM%%。随素材一起上传的 DSP 曝光监测 URL 形如： http://imp.dsp1.com?bid_request_id=%%BID_REQUEST_ID%%&impression_param=%%IMPRESSION_PARAM%%&win_price=%%WIN_PRICE%% 通过宏%%IMPRESSION_PARAM%%，DSP 可以把内部参数从竞价模块透明传输到曝光监测模块。DSP 负责参数加解密
click_param	string	否	移动联盟流量需保证做完宏替换的点击跳转地址 base64 encode 之后小于 1024 字节；其他流量需保证做完宏替换的点击跳转地址 url encode 之后小于 1024 字节	替换 DSP 点击监测 URL 里的宏%%CLICK_PARAM%%。随素材一起上传的 DSP 点击监测 URL 形如： http://click.dsp1.com?bid_request_id=%%BID_REQUEST_ID%%&click_param=%%CLICK_PARAM%% 通过宏%%CLICK_PARAM%%，DSP 可以把内部参数从竞价模块透明传输到点击监测模块。DSP 负责参数加解密
dynamic_creative	DynamicCreative	否		若回复的广告是动态创意，需在该结构中填写相应的值
app_filter	bool	否		是否需要 Tencent Ad Exchange 进行 App 已安装过滤的标识，默认为 false。若 DSP 需要针对该次回复的广告做 App 已安装过滤，此字段的值为 true；否则为 false

### 5.6.2.4 DynamicCreative 类型

名称	类型	必填	限制	描述
html_snippet	string	否		动态创意代码片段
support_gdt_helper_js	bool	否		<p>动态创意代码片段是否内嵌了 Tencent Ad Exchange 提供的辅助 JS 代码。辅助代码详见</p> <p><a href="http://qzonestyle.gtimg.cn/qzone/biz/gdt/mob/sdk/v2/dyc/gdt_dynamic_creative.js">http://qzonestyle.gtimg.cn/qzone/biz/gdt/mob/sdk/v2/dyc/gdt_dynamic_creative.js</a></p> <p>DSP 以 iframe 代码片段的形式提供动态创意，需要调用该 JS 代码提供的接口来实现两个功能：</p> <ol style="list-style-type: none"> <li>1. 通知流量方 iframe 内广告相关素材加载完成，可以展示；</li> <li>2. 通知流量方发生了用户点击,流量方负责使用浏览器打开跳转地址。</li> </ol>
product_infos	Array of ProductInfo	否		动态创意对应的商品信息
click_url_domain_whitelist	Array of string	否		预留，暂未启用

### 5.6.2.5 ProductInfo 类型

名称	类型	必填	限制	描述
product_id	string	否		DSP 的商品 ID
product_industry	int64	否		商品一级行业 ID。取值见《广告主行业分类.xlsx》。
product_category	int64	否		商品二级行业 ID。取值见《广告主行业分类.xlsx》。
product_name	string	否		商品名称
product_description	string	否		商品描述
product_image_urls	string	否		商品图片地址

### 5.6.2.6 Bid Response 示例

```
request_id: "6dfv7gekg5f7w"
seat_bids {
  impression_id: "0"
  bids {
    creative_id: "1234"
    bid_price: 70
    impression_param: "r1fM_Koi2p.HFdWgJMcGqSb3MjzUW.DCrGen5"
    click_param: "r1fM_Koi2p.HFdWgJMcGqSb3MjzUW.CrGen5"
  }
}
processing_time_in_ms:40
```

### 5.6.3 Win Notice

目前，win notice 会随着曝光上报一起发送，不做单独的 win notice。如果要记录成交价格，需要在曝光监控地址里面包含宏%%WIN\_PRICE%%，Tencent Ad Exchange 会将该宏替换为加密后的成交价格串。

**说明：**在部分流量上曝光由 Tencent Ad Exchange 后台 sever 上报，不支持用户侧上报。

### 5.6.4 成交价加解密

原始价格字节串后面用空格补齐为 16 字节的倍数，并采用 **AES128**（ECB 模式关闭自动 padding）加密，经过 **base64 encode**（详见 rfc4648: <http://tools.ietf.org/html/rfc4648>）后，作为加密后的成交价格串。加密 KEY 是 Tencent Ad Exchange 分配给 DSP 的授权令牌（TOKEN）的前 16 字节（byte）。

### 5.6.5 宏替换

#### 1. DSP 监测

宏名称	描述
%%BID_REQUEST_ID%%	ADX 提供的 Bid Request 唯一标识
%%WIN_PRICE%%	加密后的 CPM 成交价。单位：分
%%IMPRESSION_PARAM%%	DSP 自定义的展示监测 URL 动态部分，该宏用 Bid Response 中的 impression_param 替换，ADX 不对 impression_param 做任何处理
%%CLICK_PARAM%%	DSP 自定义的点击监测 URL 动态部分，该宏用 Bid Response 中的 click_param 替换，ADX 不对 click_param 做任何处理

宏使用示例如下：

DSP 曝光监测 (monitor\_url1):

http://view.dsp1.com?bidid=%%BID\_REQUEST\_ID%%&win=%%WIN\_PRICE%%&impparam=%%IMPRESSION\_PARAM%%

DSP 点击监测:

http://click.dsp1.com?bidid=%%BID\_REQUEST\_ID%%&clickparam=%%CLICK\_PARAM%%

## 2. 第三方曝光监测

宏名称	描述
__OS__ %%OS%%	客户端操作系统的类型 0-未知 1-iOS 2-Android 3-Windows 4-Symbian 5-Java
__IMEI__ %%IMEI%%	基于 IMEI 密文的用户唯一识别码
__IDFA__ %%IDFA%%	基于 IDFA 密文的用户唯一识别码
__IP__ %%IP%%	用户终端的公网 IP 地址
__UA__ %%UA%%	用户终端设备的 User Agent
__TS__ %%TS%%	Tencent Ad Exchange 的曝光时间 (UTC 时间戳, 自 1970 年起的毫秒数)

第三方曝光监测 (monitor\_url2 和 monitor\_url3) 宏使用示例如下:

http://view.third.com?os=\_\_OS\_\_&imei=\_\_IMEI\_\_&idfa=\_\_IDFA\_\_&ip=\_\_IP\_\_&ua=\_\_UA\_\_&ts=\_\_TS\_\_

原有的%%宏仍然支持:

http://view.third.com?os=%%OS%%&imei=%%IMEI%%&idfa=%%IDFA%%&ip=%%IP%%&ua=%%UA%%&ts=%%TS%%



# 附录

## 附录 1、Buyer API 基础数据类型定义

类型名称	描述	传输方法
boolean	布尔型	JSON 中使用 boolean 类型
double	双精度浮点型	JSON 中使用 number 类型
int32	32 位有符号整数	JSON 中使用 number 类型
int64	64 位有符号整数	JSON 中使用 number 类型，注意，某些开发语言中可能会丢失精度
string	字符串	使用 string 传输
datetime	日期时间，形如 2015-12-08 16:08:00	使用 string 传输，yyyy-MM-dd HH:mm:ss 格式（时间采用 24 小时制），若无特殊说明，时区均为“北京 (UTC+08:00)”
date	日期，形如 2015-01-31	使用 string 传输，yyyy-MM-dd 格式
time	时间，形如 16:00:59，精确到秒	使用 string 传输，HH:mm:ss 格式（24 小时制）
enum	枚举类型	使用 string 传输，其中字节只可以是英文字母和下划线，不得有其他特殊字节
timestamp	unix timestamp	JSON 中使用 number 类型传输
file	仅用于 form 表单上传文件的模式	N/A

## 附录 2、枚举值定义

### Buyer API

#### 商品类型（ProductType）

值	描述
1	普通链接
2	Android 应用下载
3	iOS 应用下载
4	微信品牌页（仅针对微信流量）

#### 审核状态（ReviewStatus）

值	描述
PREPARING	准备中
PENDING	待审核
APPROVED	通过
REJECTED	拒绝

#### 频控周期类型（PeriodType）

值	描述
ALWAYS	整个生命周期（expire_date）都做频次控制
BY_DAY	以 1 天为周期做频次控制

### 实时竞价 RTB 接口

#### 原生广告组件（Impression.Native.Fields）

值	描述
HEADLINE	标题/应用名称
ICON	图标/应用图标

IMAGE	主图
DESCRIPTION	描述

## 原生广告类型（**Impression.Native.Type**）

值	描述
LINK	普通链接
APP	应用下载

## 设备类型（**Device.DeviceType**）

值	描述
DEVICETYPE_UNKNOWN	未知
DEVICETYPE_PC	PC
DEVICETYPE_MOBILE	移动（手机）
DEVICETYPE_PAD	平板电脑

## 设备运营商（**Device.Carrier**）

值	描述
CARRIER_UNKNOWN	未知
CARRIER_CHINAMOBILE	中国移动
CARRIER_CHINAUNICOM	中国联通
CARRIER_CHINATELECOM	中国电信

## 设备联网方式（**Device.ConnectionType**）

值	描述
CONNECTIONTYPE_UNKNOWN	未知
CONNECTIONTYPE_WIFI	WiFi
CONNECTIONTYPE_2G	2G
CONNECTIONTYPE_3G	3G
CONNECTIONTYPE_4G	4G

## 移动设备操作系统（Device.OperatingSystem）

值	描述
OS_UNKNOWN	未知
OS_IOS	iOS
OS_ANDROID	Android
OS_WINDOWS	Windows
OS_SYMBIAN	Symbian
OS_JAVA	Java

## 用户性别（User.Demographic.Gender）

值	描述
GENDER_UNKNOWN	未知
GENDER_MALE	男性
GENDER_FEMALE	女性

## 用户学历（User.Demographic.Education）

值	描述
EDUCATION_UNKNOWN	未知
EDUCATION_DOCTOR	博士
EDUCATION_MASTER	硕士
EDUCATION_COLLEGE	大学生
EDUCATION_HIGH_SCHOOL	高中生
EDUCATION_MIDDLE_SCHOOL	初中
EDUCATION_PRIMARY_SCHOOL	小学

## 用户消费能力（User.Demographic.ConsumptionAbilityLevel）

值	描述
CONSUMPTION_ABILITY_LEVEL_UNKNOWN	未知
CONSUMPTION_ABILITY_LEVEL_HIGH	高
CONSUMPTION_ABILITY_LEVEL_LOW	低

## 附录 3 返回码（Buyer API）

### 系统返回码

返回码	返回描述
0	全部成功
1	部分成功，仅用于批量接口
2	全部失败，仅用于批量接口
10000	未知系统错误
10001	系统维护中
10002	接口不存在
10003	该版本接口已废止
10004	HTTP Method 错误

### 认证错误

返回码	返回描述
11000	未知认证错误
11001	授权令牌（TOKEN）已过期
11002	IP 不在白名单内
11003	没有权限
11004	访问频率太高

### 通用业务错误

返回码	返回描述
12000	未知请求错误
12001	参数为空
12002	参数非法
12003	文本太短
12004	文本太长
12105	文件数据传输失败
12106	文件数据保存失败
12107	文件数据 Mime Type 不符合要求
12108	文件数据容量超限
12109	文件数据（多媒体格式）高度不符合要求
12110	文件数据（多媒体格式）宽度不符合要求

## 广告主

返回码	返回描述
23001	广告主不存在
23002	广告主已经存在
23003	广告主 id 为空
23004	广告主 id 非法
23005	广告主 homepage 空
23006	广告主 name 为空
23010	广告主行业 id 为空
23011	品牌名称为空
23012	品牌图片为空
23013	品牌名称太长
23015	品牌图片高度不符合要求
23016	品牌图片宽度不符合要求
23017	品牌图片格式错误
23018	广告主行业 id 错误
23019	品牌图片太大
24001	资质为空
24002	资质转存失败
24003	资质太多
24004	资质文件太大

## 广告创意

返回码	返回描述
26001	广告不存在
26002	广告已存在
26003	过期日期格式不正确
26004	过期日期超限
26005	该客户广告太多
26006	广告的 landing page 为空
26007	广告的 monitor url 为空
26008	广告 id 为空
26010	规格不正确
26011	规格和文本或多媒体材料不匹配
26012	品牌信息为空
26013	开始日期格式不正确
26014	开始日期超限
26021	广告的 pmp contract code 不存在
26031	广告的 landing page 中不能包含端口号

26032	广告的 landing page 不能使用 ip
26041	广告的商品类型不正确
26042	app id 未填写或不合法
26043	无法获得 iOS APP 相关信息
26044	无法获得 Android APP 相关信息
26045	渠道包信息错误
26051	广告所属广告主不在合同允许的廣告主列表之内
26052	广告的投放日期范围超出了合同规定的范围
26053	不允许新增或者更新 CPD 广告
26061	Call To Action 为空
26062	Call To Action 超过了四个字

## 附录 4、功能使用说明

### 应用直达 (Deep Link)

什么是 DeepLink 广告？

用户点击广告后，“直接到达”广告主“应用”内广告指定的落地页，叫做应用直达。具体来说，应用直达广告允许广告主在投放广告时，指定一个在广告主 App 内的落地页。用户点击第三方流量上展现的广告后，系统根据当前广告和用户的情况作出判断，如果条件符合，唤起广告主的 App，并在该 App 内打开广告主指定的页面（非应用直达的广告最多只能打开 App 的首页）。

DeepLink 广告可以提升用户体验，并提升广告的转化效果。移动端屏幕比较小，App 比 web 页更适合用户浏览，能实现更多的功能。唤起广告对应的 APP，能提升用户的浏览体验和操作体验。对电商等注重转化的广告主，可以将您想推广的商品做进广告素材里，用户看到广告产生兴趣并点击后，在 App 里直接打开商品对应的页面，缩短了转化路径，提升了转化效果。

如何投放 DeepLink 广告？

我们在广告层级增加一个 `deep_link_uri` 字段，若希望投放 DeepLink 广告则在上传广告时需以 DeepLink 地址填写该字段。

在进入正式投放前，需确认以下几个问题：

- 1) 了解客户的 App 是否已经支持了 DeepLink 唤起，如果不支持，需要先支持。目前市场上的主流 App 基本上已经全部支持了 DeepLink 唤起。
- 2) 如果客户推广的是 Android App，需先将该 App 在腾讯开放平台上架。
- 3) 确保应用直达地址的正确性。

DSP 需要向广告主获取其应用直达地址。目前支持 Deeplink 地址，并兼容了 Applink 地址。需要注意的是 DeepLink 地址需按写入 scheme 的格式填写，且该 URI 与推广 App 和操作系统相关。形如：`example://showProduct?Id=330885`

请与待推广 App 的广告主及其技术人员了解您所推广的 App 的 scheme 格式，并进行测试，以确保直达页地址能被正确打开。

若推广的是应用类广告（`product_type=2` 和 `product_type=3`）推广商品，当用户点击广告后，如果该用户已安装了对应的 App，会直接打开 App 并跳转到指定的落地页，而不是 App 首页。如果该用户没有安装 App，则会引导用户下载 App 并安装。对于 Android 应用，安装完成并打开后依然会跳转到指定的落地页；对于 iOS 应用，安装完成并打开后只能到应用首页。



对“网站链接”推广商品（`product_type=1`），当用户点击广告后，如果该用户已安装了对应的 App，会直接打开 App 并跳转到指定的落地页，而不是 App 首页。如果该用户没有安装 App，则会打开网站链接所对应的网页。

## 动态创意（Dynamic Creative）

什么是动态创意？

顾名思义，动态创意是指可以根据不同用户的具体情况，动态的给用户展示个性化内容的一种广告形式。动态创意一般由广告模板和动态元素两部分组成：

广告模板用来描述动态元素在一条广告上的布局、配色、静态背景、静态文案等信息，通常是一段固定的 html 片段；

动态元素描述广告主需要推广的商品或服务的具体信息，通常包含图片、文字等；

在实践中，动态元素通常取自广告主的商品库。商品库里有成千上万甚至上百万种商品及商品信息，动态创意通过对目标人群进行拆分，并打上对应标签，根据人群身上所带的标签自动从商品库中选取对应的动态元素，再通过模板将动态元素进行固定模式的渲染，组合成动态创意进行推送。

Tencent Ad Exchange 如何支持 DSP 投放动态创意？

目前 Tencent Ad Exchange 支持在移动联盟插屏流量上投放动态创意。

已在 Tencent Ad Exchange 开户的 DSP 投放动态创意的流程如下：

- 1、当 DSP 下某个已在 Tencent Ad Exchange 开户的广告主需投放动态创意时，需以邮件的形式向 Tencent Ad Exchange 提出申请，提供相关信息，Tencent Ad Exchange 审核通过后，为该广告主开通投放动态创意功能；
- 2、DSP 通过 Buyer API 创建动态创意广告，详见“[如何上传动态创意？](#)”；
- 3、DSP 接收 Bid Request 并判断可以出动态创意时，进行 Bid Response，并在 Bid Response 中携带动态创意的相关信息。详见“[如何在竞价时回复动态创意？](#)”；
- 4、如果竞价成功，动态创意将进行展示，并支持点击和跳转。

如何上传动态创意？

Buyer API 通过新增规格的方式支持动态创意的上传，本次新增了以下 2 种动态创意的素材规格，对应 2 种尺寸。上传和更新动态创意的接口与普通广告一致。动态创意只支持广告创建，不支持更新。

尺寸	代码段类型	素材规格
300*250	iframe	220
600*500	iframe	222

需要注意的是，上传过程中，无需提交广告的 html 片段，html 片段将在 Bid Response 里实时回复（详见 [5.6.2.4 DynamicCreative 类型](#)）。同时，由于动态创意的第三方监测需由

DSP 自行完成, Tencent Ad Exchange 无需提供第三方监测功能, 因此上传素材时与第三方监测地址相关的各个字段自动失效, 可以不填。

广告创建示例如下:

1、接口地址: <https://api.e.qq.com/adx/v1/creative/add>

2、Request Body 实例:

```
{
  "data": [
    {
      "advertiser_id": "10010001",
      "creative_id": "67890",
      "product_type": 1,
      "begin_date_included": "2015-01-01",
      "end_date_included": "2015-01-31",
      "creative_spec": 220
    }
  ]
}
```

如何在竞价时回复动态创意?

当 BidRequest::impressions::creative\_specs 中包含上述的动态创意规格的时候, 那么 DSP 可以针对本次请求返回动态创意的广告。

接口示例:

Bid Request:

```
id: "lwt4yksk52iru"
is_ping: false
is_test: false
impressions: {
  id: "0"
  placement_id: 4563245454546
  creative_specs: 220
  bid_floor: 6000
}
device: {
  id: "121212121212121212121212121212121"
  device_type: DEVICETYPE_MOBILE
  os: OS_ANDROID
  os_version: "4.2.1"
  user_agent: "user agent xxxxxx"
  screen_width: 768
  screen_height: 1024
  dpi: 320
  carrier: CARRIER_CHINAMOBILE
  connection: CONNECTIONTYPE_WIFI
}
```

```
}
ip: "212.21.1.45"
```

如果确认参与竞价并希望投放动态创意，DSP 需要在 BidResponse::seat\_bids::bids::dynamic\_creative 中填充动态创意的相关信息，具体接口说明见 [5.6.2.4 DynamicCreative 类型](#)。

需要说明的是，html\_snippet 内填写的是以 iframe 形式存放的 html 片段，Tencent Ad Exchange 在支持动态创意的移动流量上用 webview 方式展示该 html 片段，并按照约定的方式协助完成广告点击和跳转。

在该 html 片段里，DSP 需要完成如下处理：

1. 完成广告的拼装，将拼装好的广告以 iframe 方式放置到 html 片段中；
2. 嵌入 Tencent Ad Exchange JS（见“[Tencent Ad Exchange 辅助 JS 接口](#)”），并使用该 JS 向 Tencent Ad Exchange 上报曝光和点击；
3. 处理点击后的落地页，Tencent Ad Exchange 会支持广告点击后的跳转，DSP 需要设定好落地页，并确认该页面的正确性。为提高广告效果，建议使用更适合移动端浏览的页面作为广告落地页；
4. 进行第三方监测上报（如需）。

接口示例：

Bid Response:

```
request_id: "lwt4yysk52iru"
seat_bids {
  impression_id: "0"
  bids {
    creative_id: "83524"
    bid_price: 5000
    dynamic_creative {
      html_snippet: "<iframe src=\"http://bidtest.dynamiccreative.com?clickm=%%CLICK_URL_ESC%%&p=%%WIN_PRICE%%\"></iframe>"
      support_gdt_helper_js: true
      product_infos {
        product_id: "53572178"
        product_industry: 21474836481
        product_category: 21474836581
        product_name: "product_name_1"
        product_multimedia_url: "http://www.product1.multimedia.url.com"
      }
      product_infos {
        product_id: "53487241"
        product_industry: 21474836481
        product_category: 21474836581
        product_name: "product_name_2"
        product_multimedia_url: "http://www.product2.multimedia.url.com"
      }
    }
  }
}
```

```

    }
  }
  processing_time_ms: 10

```

Bid Response 中的 iframe 代码段:

```

<iframe frameborder="0" scrolling="no" marginwidth="0"
marginheight="0" padding="0" width="300" height="250"
src="http://bidtest.dynamiccreative.com/gdt?info=everything_you_need&w=300&h=250&
p=%%WIN_PRICE%%"/>

```

iframe 最终展示代码:

```

<html>
  <head>
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
    <meta http-equiv="Cache-Control" content="no-cache" />
    <meta name="viewport" content="width=device-width,user-scalable=0,initial-scal
e=1.0, minimum-scale=1.0, maximum-scale=1.0" />
    <meta name="MobileXX" content="320" />
    <style>{*{margin:0;padding:0;}}</style>
  </head>
  <body>
    // add anything you need here
    // .....
    // call gdt helper js
    <script src="http://qzonestyle.gtimg.cn/qzone/biz/gdt/mob/sdk/v2/dyc/gdt_dynam
ic_creative.js">
    </script>
    <script>
      window.onload=function(){gdt.onExposure();}
      // suppose you add one image for each product
      var products =document.getElementsByTagName("img");
      for(var i=0;i<products.length;i++){
        products[i].onclick=function(){
          gdt.onClick(this.getAttribute("click-url"));
        }
      }
    </script>
  </body>
</html>

```

代码段支持的宏

%%WIN\_PRICE%%: 成交价加密串

## Tencent Ad Exchange 辅助 JS 接口

GDT.onExposure():

通知 SDK iframe 内广告相关素材加载完成，可以展示。

GDT.onClick(clickURL):

通知 SDK 发生了用户点击，SDK 负责使用浏览器访问 clickURL(目标落地页)。

具体代码见：[http://qzonestyle.gtimg.cn/qzone/biz/gdt/mob/sdk/v2/dyc/gdt\\_dynamic\\_creative.js](http://qzonestyle.gtimg.cn/qzone/biz/gdt/mob/sdk/v2/dyc/gdt_dynamic_creative.js)

## 流量预定位（Pretargeting）

什么是流量预定位？

很多 DSP 会基于自己的带宽以及业务的侧重点对 ADX 的流量进行筛选。这一操作称为流量预定位（Pretargeting）。流量筛选的某一个维度（如地域）被称为定位选择器。DSP 对所有定位选择器的选择结果称为一个配置包。不同的配置包之间是或关系，不同定位选择器之间是与关系，同一定位选择器中的选项一般为或关系。

Tencent Ad Exchange 如何支持 Pretargeting？

Tencent Ad Exchange 支持的定位选择器有如下几种：

定位选择器	选项
设备类型	PC、Mobile
操作系统	Windows、iOS、Android、其他
地域	见“腾讯社交广告地域码”
站点集	腾讯社交广告内部移动站点、微信、QQ 客户端、腾讯广告联盟等
广告样式	Feeds、Banner、开屏、原生、插屏

DSP 对流量的选择（配置包）通过 Buyer API 上传（具体接口见 [4.4 Pretargeting](#)），另外 Buyer API 还实现了对配置包的获取列表、查询、更新和删除功能，且可以对配置包进行上下线处理，线上同时生效的配置包不超过 10 个：

项目	配置包 ID	是否在线	定位器选项
获取列表	不填	不填	不填
查询	必填	不填	不填
创建	必填	必填	必填（至少在一个定位器中有选择项）
更新	必填	必填	必填（至少在一个定位器中有选择项）
删除	必填	不填	不填

当请求至少命中 DSP 的一个配置包时，该请求可发送给相应的 DSP，并且会在 Bid Request 中的 impression.targeting\_ids 字段将所有命中的配置包 ID 发送给 DSP；当未命中任何配置包时，不发送请求；当线上无配置时，在满足整体 QPS 限制的前提下发送全部种类的请求。

## 频次控制（Frequency Capping）

Tencent Ad Exchange 的频次控制包括哪些功能？

品牌广告主在程序化购买投放中有频次控制（常规）、跨媒体频控和跨屏频控的需求。

DSP 目前面临的问题是：1、在 PC 和 Mobile 无统一的用户身份识别码；2、在需要标签的流量上会拿不到用户 ID（Tencent Ad Exchange 的政策是用户 ID 和用户标签不会同时给出）。

由于上述原因，DSP 无法满足品牌广告主频次控制（常规）、跨媒体频控和跨屏频控的需求。

为了解决上述问题，Tencent Ad Exchange 支持根据 DSP 上传的频次控制配置进行相应的频次控制。

如何进行频次控制配置？

每个广告主可以创建多条频控配置，在每条频控配置中需要配置广告列表、有效期、类型和次数：

每条频控配置中的广告列表中可以包含一个或者多个广告，而每个广告最多只能出现在一个频控配置中。如果列表包含多个广告，则对列表中的所有广告的总展示次数进行频控。

有效期指此条频控的有效期，频控配置创建后立即生效，到有效期截止时失效，整个频控的有效期最长 **30** 天。

频控类型有两种模式，按天频控和全周期频控。按天频控指控制广告列表中广告每天的总展示次数。全周期频控会控制所有广告在频控有效期内总的展示次数。

频控次数就是所有广告展示总和的上限。包括 **PC** 端和 **Mobile** 端。

DSP 可以通过 **Buyer API** 提供的接口创建、查询和删除频控配置，具体接口见 [4.5 频次控制](#)