

百度流量交易服务接入规范

版本：2.5

最后更新时间：2015-7-2

此文档仅供 DSP 公司与百度流量交易服务对接使用，不得向第三方传播

目录

一、百度流量交易服务接口协议	1
1、百度流量交易服务接口说明	1
2、百度流量交易服务向 DSP 发送的广告请求接口(BID REQUEST)	2
3、DSP 向百度流量交易服务返回的竞价广告接口（BID RESPONSE）	16
4、百度流量交易服务向 DSP 发送的竞价结果接口（WIN NOTICE）	22
二、COOKIE MATCHING	23
三、示例	25
附录 1：扩展广告 SDK 脚本文档	26

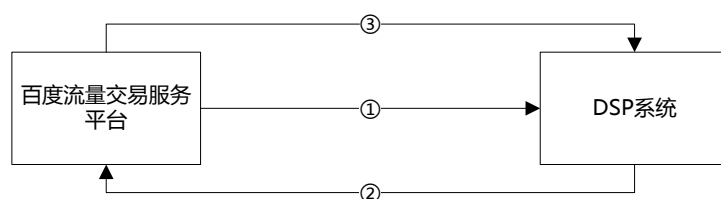
一、百度流量交易服务接口协议

1、百度流量交易服务接口说明

百度流量交易服务协议包括三个接口：

- ① 百度流量交易服务向 DSP 发送的广告请求接口(Bid Request)
- ② DSP 向百度流量交易服务返回的竞价广告接口 (Bid Response)
- ③ 百度流量交易服务向 DSP 发送的竞价结果接口 (Win Notice)

关系如下：



2、百度流量交易服务向 DSP 发送的广告请求接口(Bid Request)

广告请求接口使用 http post 方式进行交互，消息内容使用 protobuf 封装，序列化之后传送，访问超时设置为 130ms。Content-Type 设为 application/octet-stream。每个请求进行一个广告位的竞价。

广告请求接口组成



protobuf 接口

```
message BidRequest {
  // 请求 ID, 唯一标识本次请求, 明文字符串
  required string id = 1;

  // **** 用户信息 ****
  // 用户 IP 地址, 点分十进制字符串
  optional string ip = 2;
  // User-Agent
  optional string user_agent = 3;
  // 用户 ID, 已于 2015 年 4 月 30 日停止使用
  optional string baidu_user_id = 4;
  // 用户 ID 版本, 已于 2015 年 4 月 30 日停止使用
  optional int32 baidu_user_id_version = 5;
  // 用户 ID
```

```
// 在 2015 年 5 月 1 日后, cookie 的版本号为 2
message BaiduId {
    optional string baidu_user_id = 1;
    optional int32 baidu_user_id_version = 2;
}
repeated BaiduId baidu_id_list = 8;
// 用户兴趣
repeated int64 user_category = 6;
// 用户性别, 在 2015 年 7 月 10 日停止使用, 性别信息在用户兴趣字段中提供
enum Gender {
    UNKNOWN = 0;
    MALE = 1;
    FEMALE = 2;
}
optional Gender gender = 7;
// 页面语言
optional string detected_language = 9;
// flash 版本
optional string flash_version = 10;

// **** 位置信息 ****
message Geo {
    // 经纬度信息
    message Coordinate {
        // 经纬度坐标标准
        enum Standard {
            // 百度地图的经纬度坐标标准
            BD_09 = 0;
            // 国测局制定的经纬度坐标标准
            GCJ_02 = 1;
            // 国际经纬度坐标标准
            WGS_84 = 2;
            // 百度地图的墨卡托坐标标准, 以米为单位
            BD_09_LL = 3;
        }
        // 地图坐标标准
        optional Standard standard = 1;
        // 纬度
        optional float latitude = 2;
        // 经度
        optional float longitude = 3;
    }
    repeated Coordinate user_coordinate = 1;
```

```
// 用户位置信息
message UserLocation {
    // 省份，如北京市
    optional string province = 1;
    // 城市，如北京市
    optional string city = 2;
    // 区县，如海淀区
    optional string district = 3;
    // 街道，如中关村大街
    optional string street = 4;
}
optional UserLocation user_location = 2;
}
optional Geo user_geo_info = 28;

// **** 页面信息 ****
// 当前页面 URL
// 当流量来源是 IOS 的 APP 时，IOS 的应用下载链接会在 URL 字段中传输
optional string url = 11;
// 请求的 referer
optional string referer = 12;
// 网站分类
optional int32 site_category = 13;
// 网站质量类型
optional int32 site_quality = 14;
// 页面类型
optional int32 page_type = 15;
// 页面关键词
repeated string page_keyword = 17;
// 页面内容质量
optional int32 page_quality = 18;
// 页面分类
optional int32 page_vertical = 21;
// 发布者不允许的广告行业，2015 年 7 月 15 日后信息同时通过 API 中的媒体过滤设置文件
提供，该字段停止使用时间待定
repeated int32 excluded_product_category = 19 [packed=true];

// **** 移动信息 ****
message Mobile {
    // **** 移动设备信息 ****
```

```
// 百度唯一标识移动设备的编号
optional string DEPRECATED_device_id = 1;
// 新版移动设备序列号标识字段。允许同时存储多个序列号
message MobileID {
    // 移动设备 ID 的类型
    enum IDType {
        // 未知
        UNKNOWN = 0;
        // IMEI
        IMEI = 1;
        // MAC 地址
        MAC = 2;
    }
    optional IDType type = 1; // 序列号类型
    optional string id = 2;    // 序列号
}
repeated MobileID id = 13;
// 设备类型定义
enum MobileDeviceType {
    UNKNOWN_DEVICE = 0;
    HIGHEND_PHONE = 1;
    TABLET = 2;
}
// 设备类型
optional MobileDeviceType device_type = 2;
// 移动平台名，例如 android, iphone 等等
enum OS {
    UNKNOWN_OS = 0;
    IOS = 1;
    ANDROID = 2;
    WINDOWS_PHONE = 3;
}
optional OS platform = 3 [default = UNKNOWN_OS];
// 移动操作系统版本号
// 例如 Android 2.1, major, minor 分别是 2,1
// 例如 Iphone 4.2.1, major, minor, micro 分别是 4,2,1
message DeviceOsVersion {
    optional int32 os_version_major = 1;
    optional int32 os_version_minor = 2;
    optional int32 os_version_micro = 3;
}

optional DeviceOsVersion os_version = 4;
// 设备品牌
```

```
optional string brand = 5;
// 设备机型
optional string model = 6;
// 设备屏宽
optional int32 screen_width = 7;
// 设备屏高
optional int32 screen_height = 8;
// 运营商
optional int64 carrier_id = 9;
// 无线网络类型
enum WirelessNetworkType {
    UNKNOWN_NETWORK = 0;
    WIFI = 1;
    MOBILE_2G = 2;
    MOBILE_3G = 3;
    MOBILE_4G = 4;
}
optional WirelessNetworkType wireless_network_type = 10;
// 移动设备上为广告控制提供的 ID
optional string DEPRECATED_for_advertising_id = 11;
// 新版移动设备用户识别号字段，允许多种类型同时存在
// 推荐使用本字段获取用户 ID，原字段会逐步废弃。
message ForAdvertisingID {
    enum IDType {
        // 未知
        UNKNOWN = 0;
        // Android 适用
        ANDROID_ID = 4;
        // IOS 适用
        IDFA = 5;
    }
    optional IDType type = 1; // ID 类型
    optional string id = 2; // ID 内容
}
repeated ForAdvertisingID for_advertising_id = 14;

// **** 移动应用信息 ****
message MobileApp {
    // 百度移动联盟为该 App 分配的 app id
    optional string app_id = 1;
    // App 应用的 package 全名
    optional string app_bundle_id = 2;
    // App 应用分类
```



```
optional int32 app_category = 3;
// App 开发者 ID
optional int32 app_publisher_id = 4;
// App 允许的交互类型定义
// 电话、下载
enum AppInteractionType {
    TELEPHONE = 0;
    DOWNLOAD = 1;
}
// App 允许的交互类型
repeated AppInteractionType app_interaction_type = 5;
}
// 移动应用信息，当流量来自 App 时该字段非空
optional MobileApp mobile_app = 12;
}
// 移动信息，当流量来自移动设备时该字段非空
optional Mobile mobile = 29;

// **** 视频信息 ****
Message video {
    //视频的标题
    optional string title = 1;
    //视频的标签
    repeated string tags = 2;
    //视频的播放时长
    optional int32 content_length = 3;
    //视频的频道 ID
    repeated int64 channel_id = 4;
}
Optional Video video = 30;

// **** 广告位信息 ****
message AdSlot {
    // 广告位 ID，全局唯一 ID
    optional uint64 ad_block_key = 1;
    // 当前页面广告位顺序 id，同一页面从 1 开始
    optional int32 sequence_id = 2;
    // 展示类型
    optional int32 adslot_type = 3;
    // 宽
    optional int32 width = 4;
```

```
// 高
optional int32 height = 5;
// 展示位置
optional int32 slot_visibility = 6;
// 发布者允许的创意类型
repeated int32 creative_type = 7 [packed=true];
// 发布者不允许的 landing page url
//2015 年 7 月 15 日后信息同时通过 API 中的媒体过滤设置文件提供,该字段停止使用时间
待定
repeated string excluded_landing_page_url = 8;
// 媒体过滤设置信息的 ID
repeated fixed64 publisher_settings_list_id = 14;
// 发布者设置的底价,单位分
optional int32 minimum_cpm = 9;
// 视频广告的最大时长
// 该字段已于 2015 年 6 月 3 日停止使用,目前使用 VideoInfo 中的字段
optional int32 max_video_duration = 10;
// 视频广告的最小时长
// 该字段已于 2015 年 6 月 3 日停止使用,目前使用 VideoInfo 中的字段
optional int32 min_video_duration = 11;
// 视频广告的播放延时, 0 及正整数表示前贴, -1 表示中贴, -2 表示后贴
// 该字段已于 2015 年 6 月 3 日停止使用,目前使用 VideoInfo 中的字段
optional int32 video_start_delay = 12;

// 视频广告信息
message VideoInfo {
    // 视频广告的最大时长
    optional int32 max_video_duration = 1;
    // 视频广告的最小时长
    optional int32 min_video_duration = 2;
    // 视频广告的播出延时, 0 及正整数表示前贴, -1 表示中贴, -2 表示后贴
    optional int32 video_start_delay = 3;
}
optional VideoInfo video_info = 15;

message LinkUnitInfo {
    // 链接单元广告候选样式
    // 在 request 中, BES 会计算在此次请求中最优的广告样式集合, 并将此集合
    // 发送给 DSP, 最后使用的样式将根据 DSP 返回的关键词组合进行确定。
    repeated int32 style_type = 1;
```

```
// 链接单元广告可以返回的关键词个数
optional int32 req_keyword_num = 2;
// BES 推荐使用的关键词, dsp 可选择使用, 也可以使用自己认为更好的关键词
// 使用 utf-8 编码
repeated string proposed_keyword = 3;
}
optional LinkUnitInfo link_unit_info = 16;

// 优先交易信息
message PreferredOrderInfo {
  message PreferredOrder {
    // 优先交易 ID
    optional int64 order_id = 1;
    // 该优先交易的交易价格
    optional int64 fixed_cpm = 2;
  }
  // 一次请求可能包含多个优先交易
  repeated PreferredOrder preferred_orders = 1;
  // 允许优先交易的同时, 是否允许竞价
  optional bool allow_auction = 2 [default=true];
}
optional PreferredOrderInfo preferred_order_info = 13;
}

// 可扩展创意信息
message ExpandCreativeInfo {
  // 触发动作类型
  enum ActionType {
    // 悬停
    HOVER = 1;
    // 点击
    CLICK = 2;
    // 加载
    LOAD = 3;
  }
  repeated ActionType expand_action_type = 1;
  // 扩展方向
  enum Direction {
    EXPAND_NONE = 0;
    // 向上扩展
    EXPAND_UP = 1;
    // 向下扩展
```

```

        EXPAND_DOWN = 2;
        // 向左扩展
        EXPAND_LEFT = 3;
        // 向右扩展
        EXPAND_RIGHT = 4;
    };
    optional Direction expand_direction = 2;
    // 扩展创意允许的创意类型
    repeated int32 expand_creative_type = 3 [packed=true];
    // 扩展创意的宽度
    optional int32 expand_width = 4;
    // 扩展创意的高度
    optional int32 expand_height = 5;
    // 扩展创意展现时长
    optional int32 expand_duration = 6;
}
optional ExpandCreativeInfo expand_creative_info = 17;
}

// 默认每次请求一个广告位
repeated AdSlot adslot = 20;

// ***** 系统使用 *****
optional bool is_test = 26 [default=false];
optional bool is_ping = 27 [default=false];
}
    
```

广告请求接口说明

- [1] id：请求 ID，明文字符串，唯一标识本次请求。
- [2] ip：用户 ip 地址，点分十进制字符串，utf8 格式。
- [3] user_agent：用户 user agent 信息，utf8 格式。
- [4] baidu_user_id：百度用户 id，用于标记访问的用户。
- [5] baidu_user_id_version：百度用户 ID 版本，用于用户 ID 升级时使用。
- [6] user_category：用户兴趣信息，按照权重从高到低最多发送 4 个兴趣信息。对应关系见用户兴趣字典（2015 年 4 月 16 日开始使用新人群标签 1.0）。
- [7] gender：用户性别。该信息为用户行为分析结果，不保证百分百准确。

值	说明
UNKNOWN	未识别性别
MALE	男
FEMALE	女

[8] detected_language : 页面语言。

[9] flash_version : flash 版本。当前浏览器或者设备上的 flash 播放器版本。

[10] coordinate : 坐标信息。根据能获取的数据提供百度坐标体系数据或国测局标准、国际经纬度坐标标准的数据。

[11] userLocation : 用户位置信息。提供省份、城市、区县、街道 4 级数据。

[12] url : 广告位所在的 url 信息。utf8 格式。当流量来源是 IOS 的 APP 时, IOS 的应用下载链接会在 URL 字段中传输。

[13] referer : referer url 信息。utf8 格式。

[14] site_category : 网站分类 ID, ID 对应关系在网站分类字典中提供。

[15] site_quality : 网站质量类型

值	说明
0	未知
1	流量质量较低
2	流量质量较高
3	流量质量很高

[16] page_type : 页面类型

值	说明
0	未定义
1	正文页
2	索引页
3	视频页
4	论坛帖子
5	图片页
6	漫画页
7	主页
8	介绍页
9	小说页
10	下载页
11	问答页
12	搜索结果页
13	广告页

[17] page_keyword : 页面关键词。可以提供最多 3 个, 每个词最多 5 个字, utf8 格式。

[18] page_quality : 页面内容质量。

值	说明
0	正常页面
1	包含低俗内容页面

[19] page_vertical: 页面分类, ID 对应关系见页面分类字典。

[20] excluded_product_category: 不允许的广告行业, ID 对应关系见广告行业字典。

[21] device_id: 设备 ID。百度唯一标识移动设备的编号。该字段会逐渐被 MobileID 字段代替。

[22] MobileID: 新版的移动设备 ID, 可以解决之前 device_id 只提供 IMEI 或者 MAC 地址并且不说明类型的问题, 该字段提供设备 ID 的类型及对应的值, 并且不同类型可以同时提供。

[23] device_type: 设备类型。

值	说明
0	未识别设备
1	手机
2	平板

[24] platform: 操作系统

值	说明
0	其他及未识别
1	iOS
2	Android
3	Windows_Phone

[25] os_version: 移动操作系统版本号。

[26] brand: 设备品牌。

[27] model: 设备机型。

[28] screen_width: 设备屏宽。

[29] screen_height: 设备屏高。

[30] carrier_id: 运营商。MCC+MNC:比如 4600x, 前三位为 MCC (Mobile Country Code), 后两位为 MNC (Mobile Network Code)。

[31] wireless_network_type: 网络类型。

值	说明
0	未识别
1	WIFI
2	2G
3	3G
4	4G

[32] for_advertising_id：操作系统为广告精准投放所提供的 ID。

[33] app_id：APP ID。百度为该 App 分配的唯一 ID。

[34] app_bundle_id：App 应用的 package 全名。

[35] app_category：APP 应用分类。ID 对应关系见分类词典。

[36] app_publisher_id：APP 开发者 ID。百度为开发者分配的唯一 ID。

[37] app_interaction_type：APP 允许的交互类型。

值	说明
0	电话直拨
1	点击下载

[38] title：视频的标题。

[39] tags：视频的标签。

[40] content_length：视频的播放时长。

[41] channel_id：视频的频道 ID。

[42] ad_block_key：广告位 ID，对广告位进行唯一标识。

[43] sequence_id：广告位顺序 id，为了同一页面多个广告位请求使用，从 1 开始计数。

[44] adslot_type：展示类型。

值	说明
0	固定
1	悬浮
11	插屏
12	开屏
21	视频前贴片
22	视频中贴片
23	视频后贴片
26	视频暂停

[45] width：广告位宽度，像素。

[46] height：广告位高度，像素。

[47] slot_visibility：广告位位置信息。

值	说明
0	其它屏
1	第一屏
2	第二屏

[48] creative_type：允许的创意类型

值	说明
0	文本
1	图片
2	FLASH
4	图文
5	链接单元 (linkunit)
7	Video

[49] excluded_landing_page_url : 发布商不允许的 landing page url 列表。

[50] publisher_settings_list_id : 媒体保护设置信息的 ID。媒体保护设置信息通过 API 进行获取 , API 规范详见《百度流量交易服务 API 接口规范》。

[51] mininum_cpm : 发布商设置的底价 , 单位分。

[52] max_video_duration : 视频广告的最大时长 , 仅在 adslot_type 为 21、22、23 的时候有值 , 单位是秒。

[53] min_video_duration : 视频广告的最小时长 , 仅在 adslot_type 为 21、22、23 的时候有值 , 单位是秒。

[54] video_start_delay : 视频广告的播放延时。0 及正整数表示前贴 , -1 表示中贴 , -2 表示后贴。仅在 adslot_type 为 21、22、23 的时候有值 , 单位是秒。

[55] style_type : 链接单元广告候选样式。在媒体允许投放链接单元类的广告时(creative_type 为 5) , BES 会在 request 中计算在此次请求中最优的广告样式集合 , 此集合用该字段标识 , 最后使用的样式将根据 DSP 返回的关键词组合进行确定。

值	说明	示例
0	排行榜样式	
1	配图样式	
2	标签云样式	

[56] req_keyword_num : DSP 竞价时可以返回的最大关键词个数。

[57] proposed_keyword : BES 推荐使用的关键词。DSP 可以选择使用 , 也可以使用自己认为更好的关键词。使用 UTF-8 编码。

- [58] order_id：优先交易的 ID。
- [59] fixed_cpm：优先交易的交易价格。
- [60] allow_auction：本次请求在允许优先交易的同时，是否允许竞价，默认值为 true。
- [61] ActionType：触发动作类型，指通可以触发扩展创意的动作。
- [62] Direction：扩展方向，创意扩展时的方向。
- [63] expand_creative_type：扩展创意允许的创意类型，可能与原始创意允许的创意类型不一致。
- [64] expand_width：扩展创意的宽度。
- [65] expand_height：扩展创意的高度。
- [66] expand_duration：扩展创意展现时长，以扩展创意完全展开的时间点开始计时。
- [67] adslot：广告位，目前百度流量交易服务只支持一个广告位的竞价请求。
- [68] is_test：是否是测试请求。接入测试使用，正式上线后设置为 false。
- [69] is_ping：是否是心跳请求。接入测试使用，正式上线后设置为 false。

3、DSP 向百度流量交易服务返回的竞价广告接口 (Bid Response)

竞价结果返回接口以 http post 方式进行返回，消息内容使用 protobuf 进行封装，序列化之后返回。响应的 Content-Type 设为 application/octet-stream。不参与竞价，也必须在超时时限内返回 protobuf 格式的响应，超时或者返回信息无法解析被视为访问失败。

竞价结果返回接口有几种方式：Static、HTML Snippet。其中 Static 方式是最基本的接口，所有的 DSP 都可以采用。HTML Snippet 方式使用必须得到百度的确定，并且采用 HTML Snippet 方式返回的 Bid Response 必须小于 15K。

protobuf 接口

```
message BidResponse {  
    // 返回 ID，将请求中的 id 复制给返回 id，便于 session trace  
    required string id = 1;  
  
    // **** 竞价广告信息 ****  
    message Ad {  
        // 广告位顺序 ID  
        optional int32 sequence_id = 1;  
        // 创意 ID  
        optional int64 creative_id = 2;  
        // 代码段  
        optional string html_snippet = 7;  
        // 竞价链接单元广告  
        message LinkUnitKeyword {  
            // 竞价链接单元广告的关键词，utf-8 编码  
            optional string keyword = 1;  
            // 关键词竞价的样式  
            repeated int32 selected_style_type = 2;  
        }  
        optional LinkUnitKeyword link_unit_keyword = 18;  
        // 广告主 id  
        optional uint32 advertiser_id = 8;  
        // 创意宽  
        optional int32 width = 9;  
        // 创意高  
        optional int32 height = 10;  
        // 创意所属行业  
        optional int32 category = 11;  
        // 创意类型  
        optional int32 type = 12;  
        // 到达页面
```

```

optional string landing_page = 13;
// 点击链接
repeated string target_url = 14;
// 展示监测
repeated string monitor_urls = 17;
// 扩展创意信息
message ExpandCreativeInfo {
    // 扩展创意宽
    optional int32 creative_width = 1;
    // 扩展创意高
    optional int32 creative_height = 2;
    // 扩展创意着陆页
    optional string creative_landing_page = 3;
    // 扩展创意类型
    optional int32 creative_type = 4;
}
optional ExpandCreativeInfo expand_creative_info = 19;
// 最高竞价, 单位分
optional int32 max_cpm = 3;
// 扩展参数
optional string extdata = 5;
// 是否进行 cookie matching
optional bool is_cookie_matching = 6;
// 如果选择优先交易, 该交易的 ID
optional int64 preferred_order_id = 15;

}
repeated Ad ad = 2;

// **** 系统使用 ****
// debug 接口
optional string debug_string = 3;
// DSP 处理时间
optional int32 processing_time_ms = 4;
}

```

广告返回接口说明

- [1] id : 返回信息的 ID , 这个 ID 要与 request id 一致 , 表明这个 response 与 request 的对应关系。
- [2] sequence_id : 广告位顺序 ID , 与请求中的顺序 ID 对应 , 从 1 开始。
- [3] creative_id : 对于将创意上传到百度 , 由百度进行广告渲染的方式 , creative_id 是 DSP 内部的广告 ID , 这个 ID 唯一标识 DSP 上传到百度的创意。对于使用 html snippet , DSP 自己进行广告渲

染的方式，creative_id 代表 html snippet 的 ID。当 html snippet 中的创意内容、监测地址以及 landing page 改变时，需要生成新的 creative_id。建议使用 64 位整数，utf8 编码。

说明：移动广告只支持静态创意投放，由百度进行广告渲染。

- [4] html_snippet：仅在使用 html snippet 方式时使用。该字段中除了点击链接有特殊要求外，其他样式无要求。如果 html snippet 中有多个创意时，只需要在第一个创意的展示监测中加入%%PRICE%%宏。点击宏的使用方法如下，请注意：在一次返回中，只能使用一种点击宏方法。

宏	说明
%%CLICK_URL_{N}%%	<ul style="list-style-type: none"> 使用此宏时必须同时填写 target_url 字段 如果 html snippet 中有多个创意时，每个创意的点击链接位置仅需要使用宏%%CLICK_URL_{N}%%，不需要完整的点击地址。完整的点击地址顺序填写到 target_url 字段中。 这里{N}为具体的序号，代表第 N 个创意，N 从 0 开始计数 DSP 的 click url 在页面上展现为被转义的形式。字符的具体转义规则： <ul style="list-style-type: none"> a 0~9, a~z, A~Z 不变； b 空格被转义为+（加号）； c 所有其他字符被转义为对应的 utf-8 编码字符，并以%开头； 比如原始的 URL： http://click.dsp.com?macroA=%%A%%&macroB=##MACRO_B##&macroC=(MacroC.Value)&refer=\$REFER-URL 会被转义为： http%3A%2F%2Fclick%2Edsp%2Ecom%3FmacroA%3D%25%25A%25%25%26macroB%3D%23%23MACRO%5FB%23%23%26macroC%3D%28MacroC%2EValue%29%26refer%3D%24REFER%2DURL
%%CLICK_URL_ESC%%	<ul style="list-style-type: none"> 使用此宏时不需要填写 target_url 字段 只需要返回一个宏即可，DSP 可以将创意的点击链接写为 http://dsp_click_url&url=%%CLICK_URL_ESC%%landing_page_url（要求 landing_page_url 被 encode 两次）。BES 会构造 BES 的 click_url 并 encode 后替换宏。 请注意：DSP 在广告渲染时，需要对 BES 的 click_url 进行 decode，并且所有创意的点击链接都要加入 BES 的 click_url，否则 BES 无法准确记录点击数据，会影响 BES 的报表提供和数据反作弊； 使用 CLICK_URL_ESC 作为 click_url 宏时，html 代码中请勿出现 CLICK_URL_{N}宏。

例如：如 html 中包含两个创意，

- 使用%%CLICK_URL_{N}%%时，则需要使用两个宏%%CLICK_URL_0%%和%%CLICK_URL_1%%。并在 target_url 字段中顺序赋值。

```
<html><body>...<a href="%%CLICK_URL_0%%"/>..<a href="%%CLICK_URL_1%%"/>...</body></html>
```

- 使用%%CLICK_URL_ESC%%时，请参照如下方式：

```
<html><body>...<a href="http://click.dsp.com?some_params&&bes_url=%%CLICK_URL_ESC%%encoded_landing_page_url1"/>...<a href="http://click.dsp.com?some_params&&bes_url=%%CLICK_URL_ESC%%encoded_landing_page_url2"/>
```

当 DSP 返回 html_snippet 时，必须同时设置如下信息：

creative_id: 创意 id
 advertiser_id: 广告主 id
 width : 创意的宽
 height : 创意的高
 category: 关键词所属行业
 type: 创意的物料类型
 landing_page: 着陆页
 target_url: 点击链接
 max_cpm: 关键词报价，单位分

[5] keyword : 竞价链接单元广告的关键词，utf-8 编码。当 DSP 返回关键词竞价时，必须同时设置如下信息：

creative_id: 创意 id
 selected_style_type: 关键词竞价的样式
 advertiser_id: 广告主 id
 category: 关键词所属行业
 type: 创意的物料类型
 landing_page: 着陆页
 target_url: 点击链接
 monitor_urls: 展示监测
 max_cpm: 关键词报价，单位分

[6] selected_style_type :所竞价关键词所使用的样式。为空表示可以适用于 request 中的所有样式，如果 DSP 对样式无特别要求，建议为空，这样可以提升关键词的胜出率。如果选择，该样式的值必须是 bid request 中所列明的候选样式(style_type)中的一种或多种，不能超出所列明的样式。广告最终的样式由 BES 衡量多重因素从候选样式中选择的最优样式进行展现。

[7] advertiser_id : 广告主 ID。

[8] width : 创意的宽。

[9] height : 创意的高。

[10] category : 创意所属的行业，必须指定到第 2 级行业。广告分类体系见数据字典。

[11] type : 创意类型。

值	说明
0	文本
1	图片
2	FLASH
4	图文
5	链接单元 (linkunit)
7	Video

[12] landing_page : 到达页面。在 Html snippet 中创意的 LandingPage 必须在同一个域下。LandingPage 只需要提交第一个 "/" 前的域名即可。如 http://www.abc.com/abc.html 这样的只需要提交 http://www.abc.com 就可以。

[13] target_url : 包含 302 跳转的完整点击链接。使用方法说明如下：

- 当返回 html snippet 并且在 html snippet 中使用%%CLICK_URL_{N}%%时使用。
Html_snippet 中含有多个创意的情况下，每个创意点击链接的顺序应与 html snippet 中设定的创意的%%CLICK_URL_{N}%%的顺序一致。如该顺序不正确，将引发点击的统计偏差。

接上例：

```
ad.add_target_url("http://click.buyer.com?idea=ad0...");
```

```
ad.add_target_url("http://click.buyer.com?idea=ad1...");
```

- 当返回关键词时，只需要填写点击链接即可，不需要添加点击宏。

点击链接目前支持的宏包括：

宏	说明
%%ID%%	请求 ID
%%EXT_DATA%%	DSP 提供的扩展参数
%%CREATIVE_ID%%	创意 ID

%%STYLE_TYPE%%	链接单元广告展现时的样式 ID
%%INDEX%%	关键词在链接单元广告中的排序

[14] monitor_urls：展示监测。

[15] creative_width：扩展创意的宽度。

[16] creative_height：扩展创意的高度。

[17] creative_landing_page：扩展创意的落地页。

[18] creative_type：扩展创意的创意类型。

值	说明
0	文本
1	图片
2	FLASH
4	图文
5	链接单元 (linkunit)
7	Video

[19] max_cpm：DSP 提供的广告 CPM 最高竞价信息。

[20] extdata：DSP 提供的扩展参数，用于 DSP 自定义内容。会替换 win notice url 和 click url 中的宏%%EXT_DATA%%。如果本次请求需要 cookie matching，该扩展参数会以 ext_data=的形式附加在 cookie matching 的 url 中。

[21] is_cookie_matching：DSP 根据自身设置，表示是否需要进行 cookie matching。

[22] preferred_order_id：如果选择优先交易，该交易的 ID。

[23] ad：如果放弃竞价，返回 0 个 ad。目前只支持一个广告位的竞价请求返回，也就是说目前每次最多返回一个广告。

[24] debug_string：联调测试用字段，utf8 编码。

[25] processing_time_ms：DSP 内部广告检索时间，单位毫秒。

4、百度流量交易服务向 DSP 发送的竞价结果接口 (Win Notice)

通过对展示监测链接中特定参数的宏替换，将广告的计费价格发送给赢得竞价的 DSP 平台。百度流量交易服务对竞标成功的 DSP 通过 HTTP GET 发送竞价成功通知。

目前支持的宏：

宏	说明
%%PRICE%%	广告成交价格
%%ID%%	请求 ID
%%EXT_DATA%%	DSP 提供的扩展参数
%%CREATIVE_ID%%	创意 ID
%%STYLE_TYPE%%	链接单元广告展现时的样式 ID
%%INDEX%%	关键词在链接单元广告中的排序

其中 DSP 的展现监测链接中必须加入 %%PRICE%%、%%ID%% 宏。

URL Example：

```
http://www.dsp.com/monitor?reqid=%%ID%%&price=%%PRICE%%&ext_data=%%EXT_DATA%%
```

PRICE 基于 SHA-1 HMAC 算法进行加密，C++ 模块可以直接用 openssl 中的接口实现。DSP 注册时获得 32 字节的加密密钥，用于解密。获取 32 字节的完整性检测密钥，用于价格完整性检查。

二、Cookie Matching

Cookie Matching 建立了 Baidu User ID 与 DSP User ID 的映射关系，DSP 负责维护映射关系表。在 RTB 时 DSP 可以通过映射关系获取到 DSP User ID，然后通过 ID 关联的用户属性优化出价决策。

1、Cookie Matching 的准备工作

① DSP 需要提供 Cookie Matching 重定向服务器地址，如下例：

```
http://www.dsp.com/cm
```

② 百度提供的 Cookie Matching 重定向服务器代码，如下例：

```

```

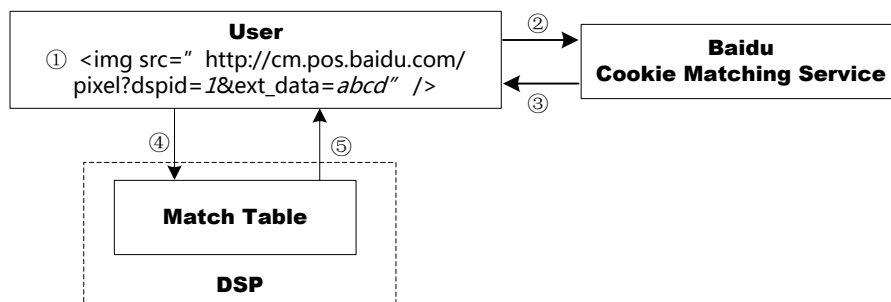
http://cm.pos.baidu.com/pixel 为 Baidu Cookie Matching 服务地址

说明：

- dspid 为 DSP 接入百度流量交易服务的唯一标识
- ext_data 为 DSP 在 Bid Response 中提供的扩展参数。如果直接使用 cookie matching 服务，该参数可以不提供。

其中，dspid 必须提供，ext_data 可以为空

2、Cookie Matching 流程



① DSP 发起 Cookie Matching 请求（可以通过自有资源将如下代码插入页面，或者通过 Bid Response 告诉百度流量交易服务本次请求需要 cookie matching，百度将如下代码插入广告页面），匹配标记：

```

```

② User 向 Baidu Cookie Matching 服务器发送 GET 请求

③ Baidu Cookie Matching 服务将请求重定向到 DSP 提供的 Cookie Matching 服务

④ 向 DSP Cookie Matching 服务发起请求

```
http://www.dsp.com/cm?baidu_user_id=ID&cookie_version=1&timestamp=TIMESTAMP
&ext_data=abcd
```

- baidu_user_id 表示 Baidu User ID
- cookie_version 为 baidu user id 版本号

- timestamp 为了防止浏览器缓存，嵌入的时间戳
 - ext_data DSP 在 Bid Response 中提供的扩展参数
- ⑤ DSP 生成 DSP User ID，将 Baidu User ID 与 DSP User ID 的映射关系保存到 Match Table 中，然后返回空白 1*1 像素图片给用户。

3、错误处理

当 Cookie Matching 服务遇到错误，会发起如下重定向请求。

注意：在错误时，DSP 仍需将 1*1 像素图片正常返回给用户。

`http://www.dsp.com/cm?baidu_error=1×tamp=TIMESTAMP`

err=1 cookie matching 服务内部错误。

三、示例

示例使用 protocol buffer 序列化库 TextFormat(google/protobuf/text_format.h)序列化而成。示例请见 baidu_realtime_bidding.proto 文件。

附录 1：扩展广告 SDK 脚本文档

1. SDK 使用方式

首先将 ext.js 引入用于广告展示的 iframe 内嵌页面中，

目前支持两个方法：

展开广告：

```
baidudup.expand("u12345", function (err) {  
    // "u12345" 表示需要展开的广告位 slotId  
    if (err) {  
        return; // 如果展开出现问题，则 err 参数则存在  
    }  
});
```

收起广告：

```
baidudup.close("u12345", function (err) {  
  
});
```

允许绑定三类事件：

当鼠标移动到广告容器上时：mouseover

当点击广告时：click

当广告加载完毕时：adloaded

绑定事件：

```
baidudup.on("click", function () {  
});
```

例如，当用户鼠标悬浮广告时展开广告，并且展开后 10 秒自动关闭：

```
baidudup.on('mouseover', function () {  
    baidudup.expand("u12345", function () {  
        setTimeout(function () {  
            baidudup.close("u12345", function () {});  
        }, 1000 * 10);  
    });  
});
```

2. SDK 实现原理

(1) 方法调用:

通过 window.postMessage 方法向父窗口 (window.top) 广播消息, 消息格式为

```
var message = {  
  title: "baidudup", //父窗口通过鉴别消息 title 字段来判断是否来自 sdk  
  type: "invokeMethod", // 消息类型, invokeMethod 表示 API 调用  
  method: methodName, // 方法调用名称  
  parameters: parameters, // 调用方法参数  
  guid: guid // 消息唯一标识, 同时与调用该方法的回调函数关联  
};
```

父窗口调用指定 API 后, 将调用结果 (或者是否报错) 返回给 SDK, 数据结构为:

```
{  
  title: "baidudup", // 同上  
  type: "methodResponse", // 同上  
  result: result, // 方法调用后的结果  
  guid: guid // 保证后端能够找到对应的回调函数  
}
```

(2) 事件机制

父页面向 SDK 广播事件发生, 数据结构:

```
{  
  title: "baidudup", // 同上  
  type: "eventHappen", // 同上  
  event: "mouseover", // 发生事件的事件名称  
  eventInfo: obj // 和事件相关的信息  
}
```

广告内嵌页面需要在 sdk 中注册事件处理函数

```
baidudup.addEventListener('mouseover', function (event) {  
  //Handler this event  
});
```