
YG

中华人民共和国广告行业标准

YG/B 10.24-2015

中国移动互联网广告标准

第五部分：移动互联网视频广告

Mobile Internet Advertising Standard

Part5: Mobile Video Advertising

2015年10月24日 实施

中国广告协会 发布

目 次

前 言.....	5
1. 适用范围.....	6
2. 引用文件.....	6
3. 术语、定义和缩略.....	6
3.1. RTB.....	6
3.2. API.....	6
3.3. APP.....	6
3.4. DSP.....	6
3.5. SDK.....	6
3.6. Campaign_ID.....	7
4. 移动视频广告程序化购买标准.....	8
4.1. 移动视频 RTB 协议.....	8
4.1.1. 接口接入标准.....	8
4.1.2. 竞价请求格式 (Bid Request)	8
4.1.3. 竞价响应格式 (Bid Response)	11
4.1.4. 兼容性扩展.....	13
4.2. 播放器 VAST 协议.....	15
4.2.1. 协议版本.....	16
4.2.2. 广告请求字段.....	16
4.3. 首选交易模式.....	16
4.3.1. 首选交易(prefer deal)模式说明.....	16
4.3.2. 优先交易权.....	17
4.4. 信息传输安全与数据加密.....	17
4.4.1. 权限认证.....	18
4.4.2. 需要加密的数据字段.....	18
4.4.3. 加密解密规则.....	18
4.5. DSP 资质以及系统性能要求.....	18
4.5.1. DSP 资质要求.....	18

4.5.2. DSP 系统性能要求.....	19
4.6. 程序化购买对接流程.....	19
4.7. 程序化广告购买数据交换标准.....	19
4.7.1. 数据对齐.....	19
4.7.2. 数据接口.....	21
4.7.3. 数据存储.....	21
5. 移动富媒体互动广告标准.....	22
5.1. 背景说明.....	22
5.2. 广告形式.....	22
5.3. 接口标准.....	22
5.3.1. 接口概述.....	22
5.3.2. IAB MRAID v2.0.....	23
5.3.3. 初始化.....	23
5.3.4. 事件处理.....	24
5.3.5. 控制广告显示.....	25
5.3.6. 改变广告尺寸.....	29
5.3.7. 检测屏幕和广告的位置及大小.....	37
5.3.8. 访问本地功能.....	40
5.3.9. 视频播放器控制 (可选).....	43
5.3.9.1. 概述.....	43
5.3.9.2. 事件.....	44
5.3.9.3. 播放器控制 mainVideo.....	44
5.3.9.4. HTML5 富媒体浮层 mainAd.....	46
5.3.10. 设备传感器 (可选).....	47
5.3.11. 获取本协议版本.....	50
5.4. 超出范围.....	51
6. 离线广告.....	52
6.1 离线广告定义.....	52
6.2 离线广告形式.....	52
6.3 离线广告播放流程.....	52

6.4 离线广告播放规则.....	53
6.5 离线广告数据上传标准.....	54
6.6 离线数据有效期.....	54
7. 移动视频广告素材规范.....	54
7.1 移动视频广告素材规格规范.....	54
7.2 移动视频广告素材内容规范.....	55
8. 移动视频广告常规监测参数.....	55
9. 移动视频广告常规监测指标.....	56
10. 移动视频广告常规监测方法.....	58
10.1 SDK 监测.....	58
10.2 API Client to Server.....	58
10.3 Server to Server.....	58

前 言

本标准是《中国移动互联网广告标准》第 5 部分——《移动互联网视频广告标准》主要涵盖了移动互联网视频广告的程序化购买、移动富媒体互动广告、离线广告、监测指标等方面，为各方在移动互联网视频广告投放、监测过程中的相关技术问题，提供统一的参考依据。

本标准参照照 GB/T1.1——2009 规则起草。

本标准是在中国广告协会互动网络分会主持下，由众多成员企业共同合作的结晶。成员企业涵盖了互联网数字广告各个领域。本章主要起草单位（排名不分先后）：华扬联众、腾讯、搜狐、优酷土豆集团、爱奇艺、秒针、腾云、新浪。主要起草人（排名不分先后）：郝涛、杨超、唐旭、王佐、程钦召、管延放、高琦、吕博文、张超、刘沛、卢振飞、陈星霖、周洋、马立丽。

本标准已经通过了小范围测试和专业论证，并根据行业和相关单位的反馈意见进行了修改，具备了行业试行的基础，于 2015 年 10 月 24 日正式执行。

1. 适用范围

本标准针对基于 Mobile Web 和 Mobile App 的移动互联网视频广告投放，制定的视频广告售卖模式、系统对接、广告形式、监测实施等方面的技术规范。

2. 引用文件

下列文件对于本文件的应用是必不可少的。凡是注版本号引用文件，仅标注版本适用于本文件。

IAB OpenRTB API Specification (Version2.3)

IAB MRAID v2.0

3. 术语、定义和缩略

3.1. RTB

Real Time Bidding (实时竞价)，是一种利用第三方技术在数以百万计的网站或移动端针对每一个用户展示行为进行评估以及出价的竞价技术

3.2. API

Application Programming Interface，是第三方服务开放给调用者的程序接口。

3.3. APP

Application 的缩写，指移动设备上的第三方应用程序

3.4. DSP

Demand Side Platform (需求方平台)，需求方平台允许广告客户和广告机构更方便地访问，以及更有效地购买广告库存

3.5. SDK

Software Development Kit (软件开发工具包), 通常移动应用开发者可根据自己的应用类型选择适当的广告 SDK 嵌入, 通过 SDK 可以获取相应的数字广告

3.6. Campaign_ID

泛指广告活动 ID, 主要用于区别广告主不同的投放活动。各方对 Campaign_ID 的具体约束规范可自行约定。

4. 移动视频广告程序化购买标准

4.1. 移动视频 RTB 协议

4.1.1. 接口接入标准

竞价请求和竞价响应接口参考 IAB OpenRTB_API_Specification(V2.3), 协议采用 HTTP POST, 开启 Keep-Alive, 建议支持 Pipeline。(同一连接在上一条消息未收到回复时可再发送下一条信息)。

消息格式	请求头 Content-Type
ProtoBuf	application/x-protobuf
JSON	application/json

4.1.2. 竞价请求格式 (Bid Request)

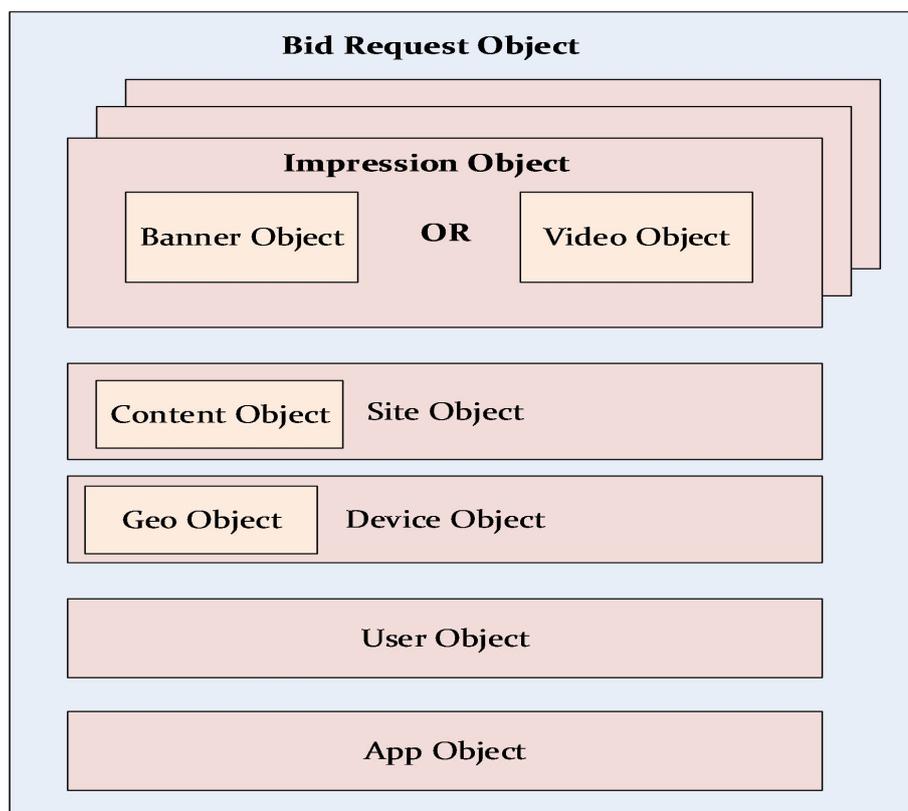


图 1 Bid Request 结构图

其中：

1) 竞价请求 (Bid request) 字段如下所示：

字段描述	建议名称	建议类型	备注
请求 ID	Id	string	
曝光对象	imp	array of objects	单次 bid request 中可包含多个 imp; 单个 bid request 中可同时包含若干 banner object 和 video object。
媒体站点对象	site	object	
用户设备对象	device	object	
用户对象	user	object	
移动应用对象	app	object	

表 4-1 竞价请求 (Bid request) 字段

2) 曝光对象 (imp) 字段如下所示：

字段描述	建议名称	建议类型	备注
曝光 id	id	string	唯一
广告位 id	tagid	string	
广告位底价	bidfloor	string	单位为分/千次曝光
banner 广告	banner	object	一个 imp object 中只可包含一个 banner 对象
视频广告	video	object	一个 imp object 中只可包含一个 video 对象
首选交易	pmp	object	
扩展字段	ext	object	

表 4-2 曝光对象 (imp) 字段

3) 视频广告(imp.video object)字段如下所示：

字段描述	建议名称	建议类型	备注
物料格式	mimes	array of string	flv/mp4/jpg/png
广告展现形式	linearity	int	1: in-stream linear ads, 2: overlay
广告最短时长	minduration	int	单位是秒
广告最长时长	maxduration	int	单位是秒
广告位宽度	weight	int	
广告位高度	height	int	

表 4-3 视频广告(imp.video object)字段

4) 首选交易 (imp.pmp object) 字段如下所示:

字段描述	建议名称	建议类型	备注
首选交易 deal id	id	string	符合条件的 deal
竞价方式	at	int	第一竞价法: 即挑选最高的 deal 价格者赢得竞价

表 4-4 首选交易 (imp.pmp object) 字段

5) 媒体站点对象 (site object) 字段如下所示:

字段描述	建议名称	建议类型	备注
站点名称	name	string	
页面 URL	page	string	
来源 URL	ref	string	
视频内容对象	content	object	

表 4-5 媒体站点对象 (site object) 字段

6) 视频内容对象 (site.content) 字段如下所示:

字段描述	建议名称	建议类型	备注
视频标题	title	string	参照各自视频网站的标签字典
视频标签	tag	string	多个标签用分号分割
所属一级频道 id	channel	string	参照各自视频网站的频道字典
其他扩展信息	ext	object	

表 4-6 视频内容对象 (site.content) 字段

7) 设备对象 (device object) 字段如下所示

字段描述	建议名称	建议类型	备注
浏览器 ua	ua	string	
IP 地址	ip	string	
地理信息	geo	string	参照 OpenRTB 接口标准
设备生产商	make	string	如: "Apple""Huawei"
设备型号	model	string	如: iphone6p
操作系统	os	string	如: "iOS""WP"
操作系统版本	osv	string	如: "8.1"
运营商 id	carrier	string	
语言	lang	string	标准为: ISO_3166_2
网络类型	connectiontype	int	0-未知; 1-Ethernet; 2-wifi; 3-未知代蜂窝网络; 4-2G 网络; 5-3G 网络; 6-4G 网络
设备类型	devicetype	int	0-手机; 1-平板; 2-PC; 3-OTT 设备

IDFA 编码方式	idfa-enc	int	0-明文;
IDFA	idfa	string	
Openudid 编码方式	openudid-enc	int	0-明文; iOS 7 之后版本无 openudid
openudid	openudid	string	
安卓广告 ID	aaID	string	
IMEI 编码方式	imei-enc	int	1-加密 (原文统一 MD5 加密)
IMEI	imei	string	
安卓 ID	AndroidID	string	原文统一 MD5 加密
MAC 地址	Mac	string	MAC 地址转大写后 MD5 加密。
其他扩展字段	ext	object	

表 4-7 设备对象 (device object) 字段

8) 移动应用对象 (app object) 字段如下所示:

字段描述	建议名称	建议类型	备注
APP 名称	appname	string	
APP 商店	market	int	0-iOS appstore; 1-Google Play; 2-其他
APP 在相应应用市场内的 ID	appid	string	
APP 分类	appcat	string	参见 IAB OpenRTB API Specification (Version2.3) 协议中, 内容分类表单 5.1 部分。
其他扩展信息	ext	object	

表 4-8 移动应用对象 (app object) 字段

4.1.3. 竞价响应格式 (Bid Response)

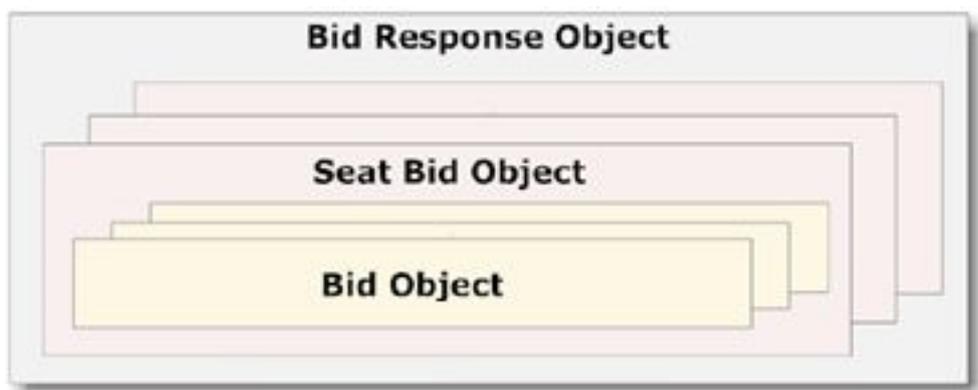


图 4-3 竞价响应结构图

其中：

1) 竞价响应(Bid Response object) 字段如下所示：

字段描述	建议名称	建议类型	备注
竞价请求 id	id	string	bid request id
DSP 竞价 ID	bidid	string	dsp 自行维护的竞价 id
出价对象	seatbid	array of objects	

表 4-9 竞价响应(Bid Response object) 字段

2) 出价对象 (seatbid object) 字段如下所示：

字段描述	建议名称	建议类型	备注
针对曝光 (imp) 的竞价	bid	array of objects	

表 4-10 出价对象 (seatbid object) 字段

3) 针对曝光的竞价 (seatbid.bid object) 字段如下所示：

字段描述	建议名称	建议类型	备注
DSP 对应该次曝光的出价 id	id	string	必填
曝光 id	impid	string	必填；需对应 bid request 中 impid
出价价格	price	float	必填。单位：分
win notice url	nurl	string	
物料类型	mtype	string	
广告物料 url	adm	string	必填；需支持两种宏替换，%%CLICK_URL_ESC%% (Exchange 的点击监测地址) 和%%WINNING_PRICE%% (竞价最终价格)
创意 id	cid	string	dsp 系统中的创意 id
首选交易 deal id	deal	string	
扩展字段	ext	object	规定监测和落地页等

表 4-11 曝光的竞价 (seatbid.bid object) 字段

4) bid.ext object 字段如下所示：

字段描述	建议名称	建议类型	备注
目标页 url	ldp	string	
曝光监测 url	pm	array of strings	可有多于个曝光监测 url
点击监测 url	cm	array of strings	可有多于个点击监测 url

表 4-12 bid.ext object 字段

4.1.4. 兼容性扩展

针对新的移动视频广告形式，预留部分字段名称，暂不在实际中使用。

字段描述	建议名称	建议类型	备注
广告类型	adslottype	int	0-linear ads; 1-skippable linear ads; 2-companion ads; 3-nonlinear ads; 4-ad pods

表 4-13 预留部分字段

Bid Request 示例如下：

```
{
  "device" : {
    "ip" : "192.168.0.1",
    "ua" : "mozilla, firefox",
    "geo" : {
      "lat" : 23.458335,
      "lon" : -50.273971,
      "ext" : {
        "accuracy" : 50
      }
    }
  },
  "didmd5" : "86582ed374218d769fa365bf29dc70e8",
  "dpidmd5" : "965e60bd2a95e3c350066f56a44420d40822df96",
  "make" : "Apple",
  "model" : "iPhoen5%2C1",
  "os" : "iOS",
  "osv" : "7.0",
  "carrier" : "460-00",
  "language" : "zh_CN",
  "js" : 1,
  "connectiontype" : 2,
  "devicetype" : 0,
  "ext" : {
    "idfa" : "1E2DFA89-496A-47FD-9941-DF1FC4E6484A",
```

```

        "mac" : "F0B4799A8CC9",
        "ssid" : "home-wifi",
        "w" : 960,
        "h" : 640,
    "brk" : 1,
        "ts" : 1374225975,
        "interstitial": 1
    }
},
"id" : "requestid123",
"imp" : [
    {
        "banner" : {
            "h" : 250,
            "w" : 300,
            "mimes":["image/jpg","image/png","image/gif"],
        },
        "bidfloor" : 0.1000000014901161,
        "id" : "impid1230",
        "tagid" : "portal_01"
    },
    {
        "banner" : {
            "h" : 250,
            "w" : 300
        },
        "bidfloor" : 0.1000000014901161,
        "id" : "impid1231",
        "tagid" : "portal_01"
    }
],
"app" : {
    "name" : "app1",
    "ext" : {
        "sdk" : "yk_v1",
        "market" : 3,
        "appid" : "1001",
        "cat" : "2",
        "tag" : "news,system"
    }
},
"user" : {
    "id" : "ykid123456"
}
}

```

Bid Response 示例

```
{
```

```

"bidid" : "bidid_request_response123",
"id" : "requestid123",
"seatbid" : [
  {
    "bid" : [
      {
        "adm" : "http://adserver.com/a.jpg",
        "id" : "bidid_impression_responseid1230",
        "impid" : "impid123",
        "nurl" : "nurl123",
        "price" : 2000,
        "crid" : "123",
        "ext" : {
          "ldp" : "http://advertiser.com",
          "pm" : ["http://pv.dsp.com/?a=3"],
          "cm" : ["http://click.dsp.com/?a=3"]
        },
      },
      {
        "adm" : "<iframe frameborder=\"0\" scrolling=\"no\"
src=\"http://cr.dsp.com/ad?clickm=%%CLICK_URL_ESC%%\" ></iframe>",
        "id" : "bidid_impression_responseid11231",
        "impid" : "impid123",
        "nurl" : "nurl123",
        "price" : 0.2000000029802322,
        "crid" : "123",
        "ext" : {
          "type" : "c",
          "ldp" : "http://advertiser.com",
          "pm" : ["http://pv.dsp.com/?a=3"],
          "cm" : ["http://click.dsp.com/?a=3"]
        }
      }
    ]
  }
]
}

```

4.2. 播放器 VAST 协议

VAST 提供了一个通用的协议，使广告投放机构只需采用一种广告响应格式即可在不同的媒体或不同的视频播放器上投放广告。IAB 于 2008 年引入了第一版 VAST，并从那时候开始，VAST 逐步被业界广泛采用和发展，目前已经更新到第三版。VAST 3.0 对之前的版本向前兼容，增加了更多特性和更丰富的数据报告。

4.2.1. 协议版本

支持 VAST 2.0 及以上版本。关于 VAST 格式的详细说明，参考 <http://www.iab.net/vast>。

4.2.2. 广告请求字段

字段描述	建议名称	建议类型	备注
广告位 ID	l	string	
广告所在页 url	u	string	
来源页 url	f	string	
视频最小时长	minduration	string	
视频最长时长	maxduration	string	
频道 id	channel	int	
视频标题	title	string	
视频标签	tag	string	
运营商	carrier	string	
联网类型	connectiontype	string	
app 名称	appname	string	
appid	appid	string	
app 应用市场	market	string	
imei 编码方式	imei-enc	int	0- 明文； 1- 加密（自行设置加密规则）
imei	imei	string	
idfa	idfa	string	
openudid	openudid	string	
安卓广告 id	aaaid	string	
安卓 id	aid	string	默认加密，自行设置加密规则
mac	mac	string	默认加密，自行设置加密规则

表 4-14 广告请求字段

4.3. 首选交易模式

4.3.1. 首选交易(prefer deal)模式说明

当一个广告展示请求发生时，ADX（广告交易平台）首先把展示机会交给拥有“First Look”权利的广告主进行挑选，挑选完毕后才会有机会进入 RTB（私有，公开）。

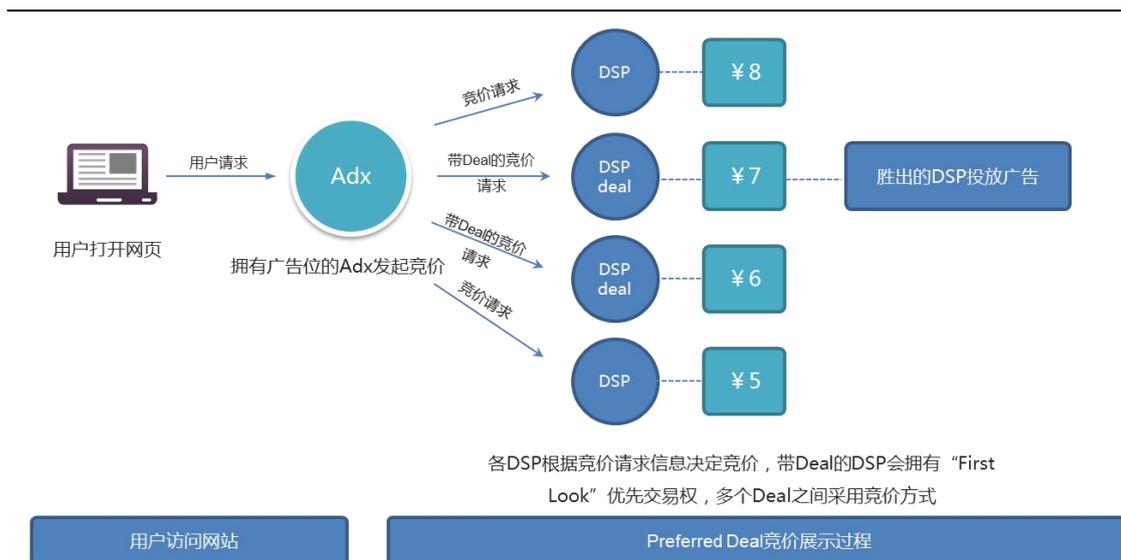


图 4-4 Preferred Deal 竞价展示过程

Preferred Deal

Preferred Deal又叫“Unreserved Fixed Rate” 固定价格无预订，采用“First Look”优先交易权交易。具体来说就是买方和卖方平台议定好CPM价格然后通过系统的方式，可以使自己的数据对每一个广告优先挑选，按需出价购买。

Auction

Auction平等竞价体系，价高胜出，次高结算

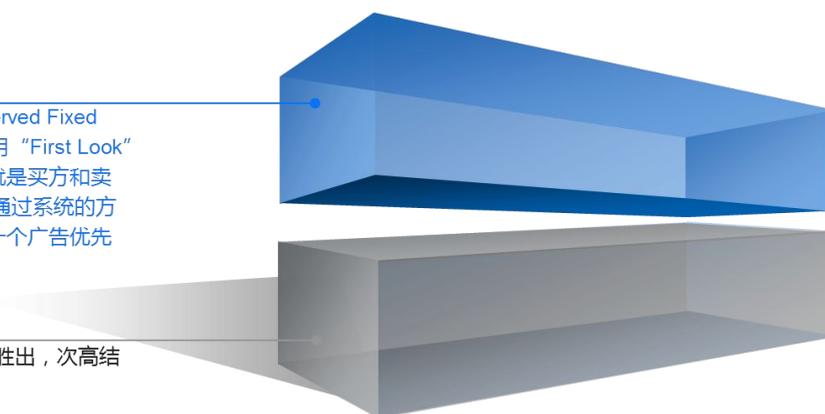


图 4-5 Preferred Deal 和 Auction 竞价区别

4.3.2. 优先交易权

优先交易权（First Look）当一个广告展示请求发生时，特定的买家可以优先判断是否需要此次展示，如果需要则按照指定的 CPM 价格购买，如果不需要，则开放此次展示机会进行拍卖竞价，这样特定的买家就有了绝对的控制权，在需要投放时，买家优先获得理想的广告展示，轻松控制占比，在需要精准投放时，借助数据和算法，确保每一分的预算都用在目标受众的展示上

4.4. 信息传输安全与数据加密

4.4.1. 权限认证

权限认证体系，参考本标准第一部分之《互联网数字广告基础标准》第九章“身份权限认证”内容。

4.4.2. 需要加密的数据字段

最终成交价格(winprice)。

4.4.3. 加密解密规则

- 1) 最终成交价格使用 AES 加密，广告系统方提供 AES（128 位）密钥。
- 2) AES 密钥传输采用不对称方式传输。
- 3) 建议采用线上自助环境验证提供密钥。

4.5. DSP 资质以及系统性能要求

4.5.1. DSP 资质要求

- 1) 广告交易平台验证

广告需求方平台（DSP）以下材料（复印件需加盖公司红色公章）：

- A) 法人代表身份证；
- B) 税务登记证副本；
- C) 营业执照副本；
- D) 开户许可证副本
- E) 组织机构代码证副本
- F) 其他证明公司资质的材料。

- 2) 其他建议考核指标

其它考核指标包括成立日期、月均 RTB 预算、客户数量、与客户结算方式、是否进行

过移动端投放、是否进行过视频广告投放等。

4.5.2. DSP 系统性能要求

- 1) 响应时间上限：80-120ms，建议通用上限为 100ms。特殊情况（例如预留库存、首选交易等非竞价模式，或涉及多次网络通信）可根据双方约定酌情放宽。
- 2) 最小 QPS 要求：5000。
- 3) IDC 部署地：至少有一处部署地与所合作 ADX 的既有部署满足竞价响应时间要求。

4.6. 程序化购买对接流程

- 1) 商务接触及相关资质验证
- 2) 签署保密协议，并提供对接文档，测试环境开户
- 3) DSP 内部开发
- 4) 测试环境联调测试（静态 API、RTB 接口）、服务器部署以及网络性能测试
- 5) 签订正式合同，并完成相关保证金支付或进行充值
- 6) 开通正式环境投放账号，开启流量

4.7. 程序化广告购买数据交换标准

4.7.1. 数据对齐

在一个广告主的一次投放计划中，会有多个不同的广告 campaign，每个 campaign 分别对应一个监测点（或者说一条监测代码）。媒体和 DSP 对每个 campaign 都会分配一个私有的 campaign_id，两者之间是一一对应的。

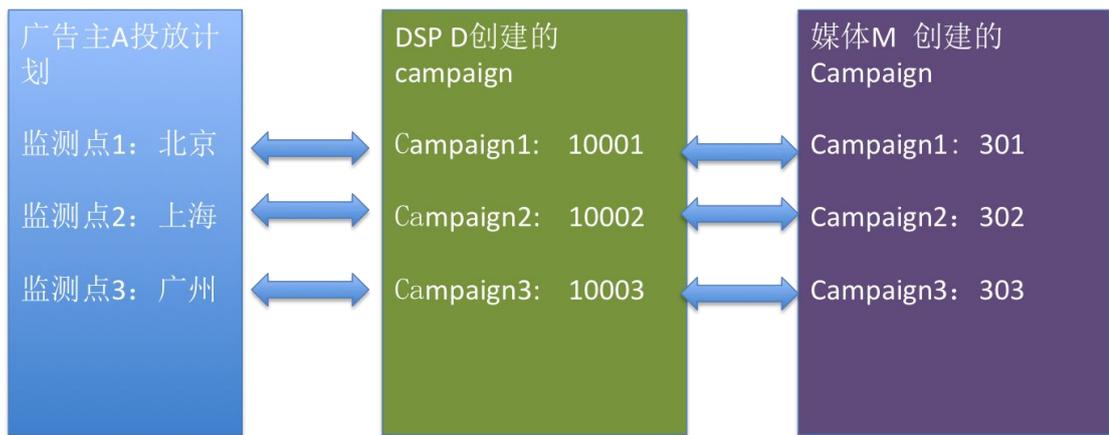


图 4-6 广告主、DSP 和媒体数据对齐流程

在媒体向 DSP 发起广告请求时，会在请求中额外加入 `campaign id` 和 `organization code` 两个参数，（`dex` 前缀是 `data exchange` 的简写，以便与 DSP 自有的参数区分），示例如下：

```
dex_campaign_id=301&dex_org_code=iqiyi
```

DSP 收到请求时，在推送日志中记录推送数据时，需要存储这两个参数，并根据这两个参数计算的推送量，竞价数，退回数。同时，DSP 在返回广告响应时（通常是通过 Vast 协议），需要在 DSP 自己的曝光和点击监测地址上加上这两个参数。曝光和点击监测发送后，DSP 可以根据组织代码和媒体 `campaign_id` 为维度计算出相应的曝光，点击，及 UV 数据。

对于多条点击监测通过 302 跳转方式串行发送，可以在 DSP 控制的监测地址里面通过宏替换方式，完成参数设置，例如：DSP1 的点击监测会 302 跳转到 `miaozhen.com` 的第三方点击监测。

DSP1 可以对 DSP1 自有的监测参数做宏替换，写入 `dex_campaign_id` 和 `dex_org_code` 两个参数,示例如下：

替换前：

```
http://www.dsp1.com?para1=1&para2=2&dex_campaign_id=[DEX_CAMPAIGN_ID]&dex_org_code=[DEX_ORG_CODE]&redirects=http%3A%2F%2Fmiaozhen.com%2F%3Ftp1%3D1%26tp2%3D2
```

替换后:

`http://www.dsp1.com?para1=1¶2=2&dex_campaign_id=301&dex_org_code=iqiyi&redirects=http%3A%2F%2Fmiaozen.com%2F%3Ftp1%3D1%26tp2%3D2`

DSP 需要向媒体提供以媒体 `campaign_id` 为维度的曝光点击和推送量数据。

同时, 媒体方需要向 DSP 提供媒体方自己记录的每个 `campaign` 的曝光点击 UV 及推送量, 退回数, 竞价数及故障数数据, DSP 获得数据后, 可以通过媒体 `campaign` 与 DSP `campaign` 的映射关系, 转为 DSP `campaign` 为维度的数据报表。

4.7.2. 数据接口

为了方便数据交换, DSP 与媒体双向对方提供数据查询接口时, 统一支持以 `org_code` 和 `campaign_id` 为 key 的查询。数据以 http 接口交互汇总的数据。示例如下:

`date=2014-01-22&hour=2&dex_campaign_id=301&dex_org_code=iqiyi`

http 请求参数定义如下:

参数名	数据类型	是否必选	备注
<code>date</code>	Date	是	数据查询日期, 格式如“YYYY-MM-DD”
<code>hour</code>	Int	否	从 0-23。
<code>dex_campaign_id</code>	String	是	广告活动 ID.如果没有, 则默认返回全部有数据的 <code>campaign</code>
<code>dex_org_code</code>	String	是	企业组织机构代码标识

表 4-15 数据接口字段

4.7.3. 数据存储

媒体采集的数据需要以数据采集的原始格式在媒体平台至少保存 2 年。向监测公司传

输的数据需要以数据传输的原始格式在监测公司平台至少保存 2 年。对超过期限的数据，媒体和监测公司可以选择清除。具体参考《移动互联网广告监测标准》第 5 章，第 5.2.3 节《信息保存规范》内容。

5. 移动富媒体互动广告标准

5.1. 背景说明

在过去的几年中，移动应用上的富媒体广告越来越流行，IAB 组织定义的 MRAID 广告协议已经应用广泛，但是 MRAID 协议的应用场景是 Inline 广告及 Interstitial 广告，并未在视频领域广泛使用，本协议在兼容 MRAID 2.0 协议的基础上，对播放器的可交互式广告进行了拓展，另外 MRAID 协议在支持移动终端传感器方面直接依赖了 HTML5 标准，最新的 HTML5 标准虽然对各传感器行为进行了定义，但并不是所有的浏览器内核都可以支持，所以本标准对移动终端传感器事件进行了拓展，让更多的移动设备可以使用丰富的传感器交互方式。

5.2. 广告形式

移动富媒体互动广告的常见形式有：

- 1) 视频前贴片、中插、后贴片
- 2) 视频暂停广告
- 3) 视频框外广告
- 4) 视频角标广告

5.3. 接口标准

5.3.1. 接口概述

- 1) MRAID 2.0 定义了 webview 组件展示的创意素材和 App 之间的交互接口。
- 2) 视频播放器控制 定义了 webview 组件展示的创意素材和视频播放器的交互接口。

3) 设备传感器 定义了 App 和传感器相关功能和 webview 组件展示的创意素材之间的交互接口，其中方法是指创意 js 层调用 App 方法，事件是 App、播放器或系统触发的事件，供 js 层处理。

5.3.2. IAB MRAID v2.0

1) 方法列表

addEventListener	getVersion
createCalendarEvent	isViewable
close	open
expand	playVideo
getCurrentPosition	removeEventListener
getDefaultPosition	resize
getExpandProperties	setExpandProperties
getMaxSize	setResizeProperties
getPlacementType	storePicture
getResizeProperties	supports
getScreenSize	useCustomClose
getState	getOrientationProperties
setOrientationProperties	

表 5-1 IAB MRAID 方法列表

2) 事件列表

error	stateChange
ready	viewableChange
sizeChange	

表 5-2 事件列表

5.3.3. 初始化

1) ready 事件

当容器完全加载、初始化完成并且准备好来自广告创意的任何调用时，ready 事件触发。在执行任何富媒体操作之前，广告应该等待 ready 事件。因为一些时间问题，比如要在广告注册监听之前触发 ready 事件，广告设计者可以配合 getState()方法来使用 ready 事件。

2) getVersion 方法

getVersion 方法允许广告在显示前确认基础功能集。版本号必须符合 MRAID 规范（例如 1.0 或者 2.0），不是供应商的 SDK 版本号。

```
getVersion() -> String
```

参数:

- none

返回值

- String - MRAID 兼容的 SDK 版本号或者 IAB 针对性认证的 SDK 版本号。举个例子，对于本文的版本号，getVersion() 将返回 “2.0”

5.3.4. 事件处理

Web 层和原生层的通信本质上异步的。通过事件处理，广告设计者能够监听指定的事件，并且可以根据需要响应这些事件。

1) addEventListener 方法

使用本方法为指定事件订阅一个指定的处理方法。这样，多个监听器就可以订阅一个指定事件，一个监听器可以处理多个事件。

```
addEventListener(event, listener)
```

参数:

- event - String 要监听的事件名称
- listener - 执行监听的函数

返回值:

- none

2) removeEventListener 方法

使用本方法从指定事件退订处理方法。当事件监听器不再有用时，它们应当被移除以避免出错。如果没有指定监听器方法，那么正在监听事件的所有方法将被移除。

```
removeEventListener(event, listener)
```

参数:

- event - String 事件名称
- listener - 要被移除的函数

返回值:

- none

触发事件:

- none

3) 错误处理

error 事件，每当容器错误发生时，error 事件就被触发。该事件包含错误发生时的描述，以及，在适当的时候，包含导致错误的操作名称。

```
“error” -> function(message, action)
```

参数:

- message: String 错误描述
- action: String 引发错误的动作

5.3.5. 控制广告显示

1) getState 方法

返回广告容器的当前状态，返回广告容器是否处于其默认状态，是否是固定位置，是否处于展开或缩放状态，是否是更大的位置，或者是否是隐藏的。

状态	解释
hidden	插播式广告转变为关闭后的状态。在被支持的情况下，banner 广告转变为关闭后的状态
resized	广告容器已通过 MRAID2.0 resize() 方法改变大小
expanded	广告容器已展开覆盖到应用程序内容的最顶层

default	应用程序和 SDK 设定好广告容器的初始位置和大小
---------	---------------------------

表 5-2 广告容器的当前状态

<p>getState() -> String</p> <p>参数:</p> <ul style="list-style-type: none"> • none <p>返回值:</p> <ul style="list-style-type: none"> • String: "loading"或"default"或"expanded" 或 "resized" 或 "hidden" <p>相关事件:</p>

2) stateChange 事件

当状态被广告或者环境以程序方式改变时，会触发 stateChange 事件。当广告视图在 default、expanded、resized、hidden 状态之间改变时（这些状态都是在调用 expand(), resize(), close()产生的）触发 stateChange 事件。

<p>"stateChange" -> function(state)</p> <p>参数:</p> <ul style="list-style-type: none"> • state - String, "loading"或"default"或"expanded"或 "resized"或 "hidden" <p>经由触发:</p> <ul style="list-style-type: none"> • expand() 方法、resize() 方法、close() 方法或者 App

3) getPlacementType()方法

容器返回放置类型的值以便于创意在需要时引发不同的行为。

值描述:

inline 默认的放置类型是嵌入到内容中显示（即 banner）

interstitial 这种广告放置类型是覆盖在内容上面

```
getPlacementType() -> String
```

参数:

- none

返回值:

- String: "inline", "interstitial"

相关事件:

- none

4) isViewable 方法

isViewable 方法返回广告容器当前是否在屏幕上。

```
isViewable() -> boolean
```

参数:

- none

返回值:

- boolean -

true: 容器在屏幕上并且对用户可见;

false: 容器不在屏幕上并且不可见

相关事件:

- viewableChange

5) viewableChange 事件

当广告从屏幕上出现或离开时会触发 **viewableChange** 事件。

```
“viewableChange” -> function(isViewable)
```

参数:

- isViewable - boolean

true: 容器在屏幕上并且对用户可见;

false: 容器不在屏幕上并且不可见

经由触发:

- 应用程序视图控制器的改变

广告在任何情况下都有可能不在屏幕之外加载, 最佳实践是: 广告检查自身可见状态或者在采取任何行动之前注册 viewableChange 监听事件。

示例:

```
// Wait for the SDK to become ready
if (mraid.getState() === 'loading') {
    mraid.addEventListener('ready', onSdkReady);
} else {
    onSdkReady();
}

function onSdkReady() {
    // Wait for the ad to become viewable for the first time
    if (mraid.isViewable()) {
        showMyAd();
    } else {
        mraid.addEventListener('viewableChange', function(viewable) {
            if (viewable) {
                mraid.removeEventListener('viewableChange', arguments.callee);
            }
        });
    }
}
```

```
        showMyAd();
    }
});
}
}
function showMyAd() {
...
}
```

5.3.6. 改变广告尺寸

MRAID2.0 包含三种截然不同的方式，用于广告改变尺寸。

1) resize()

说明：可完成复杂的广告尺寸变化

```
resize()
```

参数：

- none

返回值：

- none

触发事件：

- sizeChange, stateChange

侧面影响：

- 改变状态

2) 控制 resize 属性

resizeProperties 对象

```
resizeProperties object = {  
  "width" : integer,  
  "height" : integer,  
  "offsetX" : integer,  
  "offsetY" : integer,  
  "customClosePosition" : string,  
  "allowOffscreen" : boolean  
}
```

width: integer —— (必须) 创意的像素宽

height: integer —— (必须) 创意的像素高

offsetX: —— (必须) TODO

offsetY: —— (必须) TODO

customClosePosition: string —— (可选) "top-right"、"center"、"bottom-left"、"bottom-right"、
"top-center" 或者 "bottom-center"。表示相对于缩放的创意，容器提供的关闭事件区的原点位置。如果没有指定或者不是这些选项之一，默认取"top-right"。

allowOffscreen: —— (可选) 告诉容器是否应该允许缩放创意绘制完整/部分的超出屏幕。

True (default): 容器不应试图定位缩放创意。

False: 容器应尽量重新调整缩放创意来适应 getMaxSize()的取值。

3) getResizeProperties 方法

说明：使用此方法获取广告缩放后的属性。

`getResizeProperties()` -> JavaScript Object

参数:

- none

返回值:

- { ... } - 包含 `resize` 属性的 JS 对象

触发事件:

- none

4) `setResizeProperties` 方法

说明: 使用此方法来设置广告的缩放属性, 特别是 `width` 和 `height` 属性。

`setResizeProperties()` -> JavaScript Object

参数:

- `properties`: JavaScript Object { ... } —— 这个对象包含要缩放广告的宽和高、关闭位置、偏移方向 (都以 DIP 为单位)、广告是否可以缩放到离开屏幕。更多信息见 `resizeProperties` 对象。

返回值:

- none

触发事件:

- none

5) `expand()`

说明: 简单的, 模态的, 增加广告的尺寸

```
expand([URL])
```

参数:

- URL (可选): 本文中的 URL 会在一个新的遮罩视图中显示。如果传入 null 或者不使用此参数, 当前广告的主体会在同一个 Web View 中使用

返回值:

- none

触发事件:

stateChange

6) 控制 expand 属性

expand 属性是为了给广告设计者提供额外的功能。

```
expandProperties object = {  
  
    "width" : integer,  
  
    "height" : integer,  
  
    "useCustomClose" : boolean,  
  
    "isModal" : boolean (read only)  
  
}
```

7) getExpandProperties 方法

返回完整的含有 expand 属性的 JavaScript 对象。

```
getExpandProperties() -> JavaScript Object
```

参数:

- none

返回值:

- { ... } - this object contains the expand properties

触发事件:

- none

8) setExpandProperties 方法

说明: 设置整个含有 expand 属性的 JavaScript 对象。

```
setExpandProperties(properties)
```

参数:

- properties: JavaScript Object { ... } —— 这个对象包含广告的宽高值, 更多属性参考 properties 对象.

返回值:

- none

触发事件:

- none

9) open()

说明: 在浏览器窗口中打开外部移动站点。

open(URL)

参数:

- URL – String, 网页 URL

返回值:

- None

关闭展开式广告和插播式广告

10) close()

说明: close 方法会使广告容器回到初始状态。对于处在 expanded 或 resized 状态的广告,调用 close 方法会将广告状态变为 default。对于处在 default 状态的插播式广告,调用 close 方法会将广告状态变为 hidden。在本规范中,对于处在 default 状态的横幅广告,调用 close 产生的影响未定义,因此一般不建议广告设计者在横幅广告中调用 mraid.close()方法。

close()

参数:

- none

返回值:

- none

触发事件:

- stateChange

11) useCustomClose 方法

说明: A) 尽管 MRAID 要求所有的实现容器必须提供一个含默认关闭图标的可点击区域,但对于广告创建者来说,完全可以使用他们自己设计的关闭指引。

对于展开式广告，设计者不需要调用此方法，通常在 `setExpandProperties()` 方法中设定 `useCustomClose` 属性即可。

B) 对于那些无法调用 `expand` 方法的独立插播式广告，但是仍然要求具有 MRAID 强制关闭选项，广告设计者应在 `ready` 事件完成后立即调用该方法。

C) 这个方法作为 `expand` 属性中一个同名属性的快捷方法。设置 `useCustomClose` 属性或者调用此方法，两者具有相同的效果，可以互换使用。如果一个广告通过 `expand` 属性和此方法都设置了 `useCustomClose`，任何较晚的调用将覆盖较早的设置。他们通知容器停止使用默认的关闭指引。

```
useCustomClose(isCustomize)
```

参数:

`isCustomize` - boolean

- `true` —— 广告创意提供它们自己设计的关闭指引
- `false` —— 将会显示容器为关闭指引提供的默认图片

返回值:

- `none`

触发事件:

- `none`

注意事项:

当用户在 MRAID 广告中点击一个 HTML 超链接（通过 `` 标签定义），会有两种可能：目标页面可以在现有的 Web View 中载入，或者目标页内容可以打开一个独立的浏览器窗口并在那里载入这个链接。兼容 MRAID 的 SDK 可以选择两种方式的任何一种，因此广告设计者们应避免使用内嵌超链接和 `window.location`。对于一个标准 MRAID 广告，指定链接要在单独的浏览器中打开页面时，`mraid.open()` 才是恰当的方法。在广告视图中加载新网页后可以离开广告并没有被写入 MRAID 规范中，App 此时也许处在不可用状态。

11) 控制方向属性

```
mraid.setOrientationProperties ( {"allowOrientationChange":true} );

mraid.expand()

/* user changes to landscape, starts game */

mraid.setOrientationProperties ( {"allowOrientationChange": false } );

/* user is done with game */

mraid.setOrientationProperties ( {"allowOrientationChange":true} );

orientationProperties object = {
"allowOrientationChange" : boolean,
"forceOrientation" : "portrait|landscape|none"
}
```

`allowOrientationChange`: boolean —— 如果设置为 `true`，广告容器将允许基于设备取向的变化；如果设置为 `false`，则忽略这种变化（例如：即使设备改变方向，`Web View` 也不会改变）。默认值是 `true`。不管 `allowOrientationChange` 如何设定，广告创意通过设置 `forceOrientation` 的值总是可以改变它们的方向。

`forceOrientation`: string —— 可以设置 `portrait`、`landscape`、`none` 其中一个值。如果设置了 `forceOrientation` 的值，不管设备的方向如何，视图都必须以特定方向打开。也就是说，如果用户在横屏模式下观看广告，并且点击展开它，如果广告设计者把 `forceOrientation` 的属性设置为 `portrait`，那么广告将以竖屏模式打开。默认值为 `none`。

12) `getOrientationProperties` 方法

说明：`getOrientationProperties` 方法返回完整的含有 `orientation` 属性的 JavaScript 对象。

```
getOrientationProperties() -> JavaScript Object
```

参数:

- none

返回值:

- { ... } —— 这个对象包含 orientation 属性。

触发事件:

- none

13) setOrientationProperties 方法

说明: 设置 JavaScript 方向属性对象。

```
setOrientationProperties(properties)
```

参数:

- properties: JavaScript Object { ... } —— 这个对象包含 allowOrientationChange 和 forceOrientation 的值。

返回值:

- none

触发事件:

- none

5.3.7. 检测屏幕和广告的位置及大小

1) getCurrentPosition 方法

说明: getCurrentPosition 方法返回广告视图当前的位置和尺寸, 以 DIP 为单位。

getCurrentPosition() -> JavaScript Object

参数:

- none

返回值:

- JavaScript Object - {x, y, width, height}:

x=从 getMaxSize 方法定义的矩形左边开始的偏移量 (DIP 为单位)

y=从 getMaxSize 方法定义的矩形顶部开始的偏移量 (DIP 为单位)

width=当前容器的宽。

height=当前容器的高。(都是以 DIP 为单位)

关联事件:

- none

2) getMaxSize 方法

说明: getMaxSize 方法返回广告可以展开到或缩放到的最大尺寸 (宽和高都以 DIP 为单位)。

getMaxSize() -> JavaScript Object

参数:

- none

返回值:

- JavaScript Object, {width, height} —— 视图可以增长到的最大宽和高

关联事件:

- none

3) sizeChange 方法

说明：当广告的尺寸在 App 界面内发生改变时会触发 sizeChange 事件。它可能是设备方向改变或调用 resize 或调用 expand 方法产生的结果。都是以 DIP 为单位。

当广告的 Web View 的显示状态发生改变时，触发此事件。

"sizeChange" -> function(width, height)

参数：

- width - Number: 视图的宽。
- height - Number: 视图的高。

经由触发：

- 缩放、展开、关闭、方向改变或者在 App 注册尺寸相关事件监听器后导致广告视图尺寸发生改变时触发。

4) getDefaultPosition 方法

不管视图处于何种状态时调用，getDefaultPosition 方法都将返回默认广告视图的位置和尺寸（DIP 为单位）。

getDefaultPosition() -> JavaScript Object

参数：

- none

返回值：

- JavaScript Object - {x, y, width, height} :

x=从 getMaxSize 左边开始的偏移量（DIP 为单位）；

y=从 getMaxSize 顶部开始的偏移量（DIP 为单位）；

width=当前容器的宽；

height=当前容器的高；

5) getScreenSize 方法

说明: getScreenSize 方法基于当前方向, 以 DIP 为单位, 返回正在运行广告设备的实际像素宽和高。有一点需要注意的是: 如果设备从横屏转为竖屏, ScreenSize 将发生改变 (反之亦然)。另外一点需要注意的是: getScreenSize 返回设备屏幕的总计尺寸, 包含操作系统给状态栏/系统栏的预留区域, 或那些可以被 App 或广告覆盖的其它区域。

```
getScreenSize() -> JavaScript Object
```

参数:

- none

返回值:

- {width, height}

关联事件:

5.3.8. 访问本地功能

1) supports 方法

说明: supports 方法允许广告询问设备支持的指定功能。

sms: 设备支持使用短消息功能, 按照协议发送一个短消息

tel: 设备支持使用电话功能拨打电话: 按协议

calendar: 设备可以创建一个日历项

storePicture: 设备支持 MRAID storePicture 方法

inlineVideo: 设备可以使用<video>标签播放 HTML5 视屏文件, 并且按着 video 标签中指定的尺寸 (宽和高) 播放。这里并不需要以全屏方式播放视频。

supports(feature) -> Boolean

参数:

- String 要检测特性的名称 (sms, tel, calendar, storePicture, inlineVideo)

return values:

- Boolean

true —— 支持该功能, 并且 getter 和事件都是可用的。

false —— 当前设备不支持本功能。

2) 用设备的物理特性工作

理论上 HTML5 的最新标准已经支持, 但由于并非所有设备都能支持最新 HTML5 标准, 所以这部分工作, 本协议由第三部分予以补充, 广告设计人员, 监听本文第三部分的事件, 将可以获得更多设备的支持

3) 存储图片

调用 storePicture 方法会在设备相册中放任一张图片。图片可能是本地或者从网络下载的。为了确保用户注意到图片将添加到他们的设备相册, 在每张图片的添加过程中, MRAID 要求 SDK/容器使用操作系统层级的处理器显示模型对话框询问用户确认或取消添加图片。

storePicture (URI)

参数:

- URI - String: 图片或其他媒体资源的 URI

关联事件:

- none

4) 创建日历事件

说明: 用于在本地创建日历事件。

```
createCalendarEvent(parameters)
```

参数:

- parameters: JavaScript Object {...} —— 对象包含日历事件所需的条目, 根据 W3C 规范指定的日历条目编写。

返回值:

- none

关联事件:

- none

示例:

```
createCalendarEvent  
  
createCalendarEvent({  
  
description: "Mayan Apocalypse/End of World",  
  
location: 'everywhere',  
  
start: '2012-12-21T00:00-05:00',  
  
end: '2012-12-22T00:00-05:00'  
  
})
```

5) 处理视频

说明: 移动设备上的视频播放要么通过行内播放(在当前 Web View 里, App 中或移动网页), 要么通过打开本地播放器播放。在有可能的情况下, MRAID 兼容的容器最好支持行内播放, 同时允许广告设计者指定视频创意是以行内播放还是单独播放。广告设计者可以使用“supports(inlineVideo)”方法确定运行创意的设备是否显示行内视频。

6) inline 播放

说明：为了使内嵌式视频能够播放并且自动播放，MRAID 兼容的 SDK 应向 Web View 中插入必要的开启标签，这依赖于设备的操作系统类型。

对于 iOS 设备，必须使用下列标签：

```
webView.mediaPlaybackRequiresUserAction = NO;
```

```
webView.allowsInlineMediaPlayback = YES;
```

对于 Android 设备（Honeycomb, Ice Cream Sandwich, 还有它们之后的版本），SDK 必须请求硬件加速，这些依赖于问题中的视图和它是如何被添加到 WindowManager 中的：

```
getWindow().setFlags(WindowManager.LayoutParams.FLAG_HARDWARE_ACCELERATED,  
WindowManager.LayoutParams.FLAG_HARDWARE_ACCELERATED);
```

设备 / 操作系统限制可能会阻止 inline 视频播放，广告设计者可以使用“supports(inlineVideo)”方法确定运行创意的设备是否显示行内视频。

7) playVideo 方法

说明：使用此方法通过设备的本机外部播放器在设备上播放视频。

```
playVideo(URI)
```

参数：

- URI - String, 视频地址或视频流

返回值：

- none

5.3.9. 视频播放器控制（可选）

5.3.9.1. 概述

为增加视频前贴、中插、后贴广告的互动能力，需要在视频广告播放器上增加一层 Webview 来展示 HTML5 广告，Webview 层可以不与视频播放器进行交互，但为了给 Webview

层以更多的权限,下面定义的方法及事件可以将原来由原生代码实现的功能可以由 Webview 层来实现,带来更大的灵活性。但这些接口均为可选,实现者可仅通过增加 Webview 来实现互动能力。

5.3.9.2. 事件

1) durationChange 事件

获取当前广告时长

侦听方法:

`mraid.mainVideo.addEventListener(mraid.mainVideo.EVENTS.DURATION_CHANGE, func);`
此事件需配合 `mraid.stageReady()`方法使用,当页面被加载成功后,需调用 `mraid.stageReady()`方法,以上事件才会被触发,传入广告时长

“durationChange” -> `function(duration)`

参数:

- duration - Number

广告剩余时长

经由触发:

- 广告播放器广告时间改变时触发

5.3.9.3. 播放器控制 mainVideo

1) mainVideo.pause 方法

说明: 暂停主广告播放

`mainVideo.pause()`

参数:

- none

返回值:

- none

2) `mainVideo.resume` 方法

说明: 恢复主广告播放

`mainVideo.resume()`

参数:

- none

返回值:

- none

3) `mainVideo.getProgress` 方法

获取当前广告播放进度, 以 $0\sim 1$ 之前的小数表示

`mainVideo.getProgress()` \rightarrow Number

参数:

- none

返回值:

Number $0\sim 1$ 小数

4) `mainVideo.expand` 方法

说明：调用接口，将视频放大到全屏。

`mainVideo.expand()`

参数：

- none

返回值：

- none

5) `mainVideo.shrink` 方法

说明：调用接口，将视频退出全屏

`mainVideo.shrink()`

参数：

- none

返回值：

- none

5.3.9.4. HTML5 富媒体浮层 `mainAd`

1) `mainAd.setObjectViewable` 方法

说明：根据不同参数，隐藏/显示相关 UI

```
mainAd. setObjectViewable(number, boolean)
```

参数

- number - Number: 根据不同数值设置需要隐藏/显示的 UI

1:倒计时数字

2:详情点击按钮

- boolean - 传入参数设置 UI 的显示/隐藏效果; true:显示, false:隐藏;

例:

隐藏倒计时 mraid.mainAd.setObjectViewable(1, false)

展示倒计时 mraid.mainAd.setObjectViewable(1, true)

隐藏点击详情 mraid.mainAd.setObjectViewable(2, false)

展示点击详情 mraid.mainAd.setObjectViewable(2, true)

5.3.10. 设备传感器（可选）

1) 概述

在 HTML5 的最新标准中已经定义了手机传感器的事件,但由于手机内置浏览器内核并不都可以支持最新的 HTML5 标准,基于此,本协议定义了另外一系列事件,用于给移动端更广泛的支持设备传感器事件和方法都在 mraid 协议基础上直接增加,不产生新的命名空间,由 mraid.addEventListener(事件, 处理方法)监听。

2) shake

“shake” -> function()

参数:

- none

经由触发:

- 手机摇动触发

3) tiltChange

“tiltChange” -> function(x, y, z)

参数:

- x, y, z 分别为对应轴的浮点型旋转向量

经由触发:

- 手机倾斜角度变化时触发

4) headingChange

“headingChange” -> function(heading)

参数:

- heading 取值为 0-360 整数角度, 0 为正北方向, 顺时针累加

经由触发:

- 手机朝向变化触发

5) locationChange

“locationChange” -> function(lat,lng,acc)

参数:

均为浮点数

- lat 为经度
- lng 为纬度
- acc 为精确度

经由触发:

- 手机地理位置变化触发

6) networkChange

“networkChange” -> function(network)

参数:

- network - String 取值为 unknown, offline, cell, wifi

经由触发:

- 手机网络状态变化触发

7) orientationChange

“orientationChange” -> function(angle)

参数:

- Angle 为角度, 90 为竖屏, 180 为横屏

经由触发:

- 手机横竖屏变化触发

8) supports 方法

supports 方法允许广告询问设备支持的指定功能。该方法是对 mraid.supports() 进行拓展, 取值为 shake、title、heading、keyboard、location、network、orientation。

supports(feature) -> Boolean

参数:

- String 要检测特性的名称 (shake、title、heading、keyboard、location、network、orientation)

return values:

- Boolean

true —— 支持该功能, 并且 getter 和事件都是可用的。

false —— 当前设备不支持本功能。

5.3.11. 获取本协议版本

1) caamraid.getVersion 方法

getVersion() -> String

参数:

- none

返回值

- String - 本协议的版本号, getVersion()将返回“1.0”

5.4. 超出范围

移动富媒体互动广告标准在实际应用中,有以下七个方面是不适用的,超出了使用范围,具体如下所述:

- 1) 不包含从广告服务器, 广告网络, 本地资源检索广告

本协议仅包含客户端内部对富媒体广告的支持, 不包含和广告服务器交互的协议, 以及与广告网络交互的相关协议, 也不包含客户端如何检索出富媒体广告等本地策略。

- 2) 广告尺寸方面

富媒体广告素材的尺寸规格定义, 不在本协议范围内。

- 3) 不包含缓存或离线下载资源到本地文件系统

本协议不包含对广告物料的缓存及下载定义。

- 4) IDE 集成方面

本协议不提供对 IDE 开发集成环境的定义和支持。

- 5) 安全及隐私方面

本协议不涉及广告安全和隐私部分。

- 6) 国际化方面

本协议不涉及对各个语言版本的支持。

- 7) 错误报告

广告播放失败等问题导致的错误不在本协议范围内

6. 离线广告

6.1 离线广告定义

离线广告是指用户在观看“离线缓存视频”时，在无网络或者有网络（包括 2G/3G/4G，不包括 WIFI）情况下调用缓存在移动设备上的广告素材进行的广告播放。

6.2 离线广告形式

根据广告展现形式不同，离线广告可分为以下几类：

- 1) 离线闪屏（开屏）广告：指在移动设备上启动 APP 的过程中展现的广告。
- 2) 离线贴片广告：指用户在观看本地缓存的视频之前展现的贴片广告。
- 3) 离线暂停广告：指用户在观看本地缓存视频过程中暂停时展现的广告。
- 4) 角标广告：是指用户在观看本地缓存的视频过程中在视频播放区域右下角出现的图片或者互动广告。

6.3 离线广告播放流程

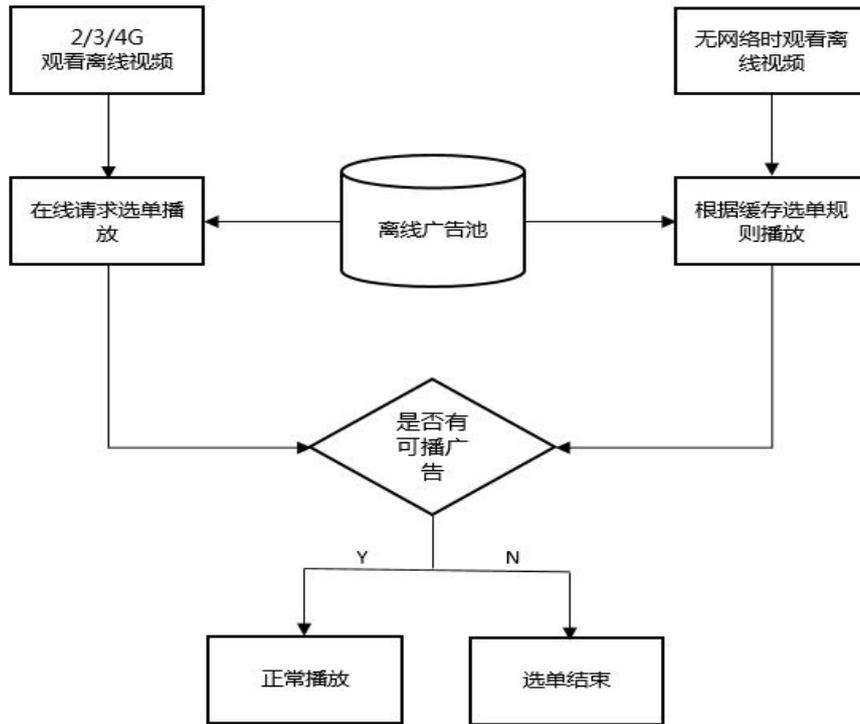


图 6-1 离线广告播放流程

由于广告素材需要能够在用户处于在线（包括有线、WIFI）状态下进行更新，同时也需及时更新广告订单规则，包括广告贴片素材、时长、轮播数等，建议离线广告素材至少提前 3 天以上下载，并缓存在本地。

6.4 离线广告播放规则

离线状态	网络	广告视频	是否可点击
半离线	2G/3G/4G	实时请求广告，播放缓存素材，弹性上报数据	是
完全离线	无网络	提前请求广告，播放缓存素材，延时上报数据	否

表 6-2 离线广告播放规则

注意事项:

1) 在 2G/3G/4G 状态下, 如果广告素材在本地已经缓存, 可以直接播放; 否则不实时拉取素材, 跳过该广告。

2) 在无网络情况下, 如果广告素材在本地已经缓存, 则直接播放, 否则无广告播出

6.5 离线广告数据上传标准

离线广告数据在上传时, 尽量采用实时上传。在 2G/3G/4G 环境下, 因为网络不稳定, 在若干次(建议 3 次以上)实时上传失败重试后, 需要将监测数据(包括上传给第三方的数据)加上时间戳后缓存在本地, 待网络情况良好后再上传。

完全无网络的情况下, 需要将监测数据(包括上传给第三方的数据)加上时间戳之后缓存在本地, 待网络情况良好后上传。在离线环境下, APP 自动将离线广告的实际发生时间记录到 TIME 字段; 如果是实时上报的数据, 则该字段不做记录, 为空或为默认值。

上报给第三方的数据时间戳采用宏定义 `__TIMESTAMP__` (双下划线), 时间戳的格式采用标准 UNIX 时间戳。

6.6 离线数据有效期

数据延时上传的情况, 可能发生在广告排期结束之后, 故需要各方约定延时数据的有效期, 建议默认的延时数据的有效期为投放结束后 7 个自然日, 最大为 14 个自然日。

7. 移动视频广告素材规范

目前在移动视频领域, 由于各视频媒体视频播放器标准的不统一, 在实际广告投放过程中, 广告主、代理商为了满足广告投放的需求, 需要制作多个不同规范的素材, 极大地影响了广告投放的效率, 同时也提高了制作成本, 也间接的增加了广告主、代理商、媒体人员之间的沟通成本。

为了改变这种无序的现状, 鼓励引导行业有一个统一的、标准的、通用的广告素材标准, 旨在建立一个被行业多方采纳的移动视频广告素材规范。此次标准中我们增加了移动视频广告素材规格规范、移动视频广告内容审核标准规范两部分内容。以供行业成员参考使用。

7.1 移动视频广告素材规格规范

我们结合目前行业内移动视频广告的主要广告形式、广告尺寸、广告格式等因素，整理并推荐以下素材规格规范标准供行业内使用，具体内容如下表所示：

广告类型	时长	格式	尺寸	宽高比	源文件及码率
贴片广告	5s	FLV	640x360	16: 9	编码方式：H.264 素材码率：建议大于等于 1Mbps
	15s	MP4	1280x720		
	30s		640*480	4:3	
	60s				
暂停广告	—	JPG	400x300	16: 9	JPG、PNG 源文件大小：建议小于等于 100KB。 HTML5 源文件及素材总大小：建议小于等于 1M。
		PNG	640x480		
		HTML5	600x500	4:3 6:5	

表 7-1 移动视频广告素材规格规范

备注：

贴片广告包括三种，分别是：前贴片广告（Pre-Roll Ad）、中贴片广告（Mid-Roll Ad）、后贴片广告（Post-Roll Ad）。

7.2 移动视频广告素材内容规范

移动视频广告素材内容，需要严格遵守国家颁布的《中华人民共和国广告法》相关规定。在此规定基础上，行业成员可根据自身产品、平台的属性特点，出台更为详细的细则规范。

8. 移动视频广告常规监测参数

针对移动视频广告，以下为监测系统需要采集的常规参数定义：

参数	说明	是否必选
OS	客户端操作系统：0-Android, 1-IOS, 2-WP, 3-Others;	是
IMEI	适用于 Android, 原值 MD5 加密;	是
MAC	用户终端的硬件地址, 适用于 Android 和 IOS, 字符转换大写后 MD5 加密;	是
AndroidID	适用于 Android, 需要 MD5 加密;	是
IDFA	适用于 IOS, 保留原始值;	是
OpenUDID	适用于 IOS, 保留原始值;	是

IP	数据需求方识别出的当前用户的访问地址，每一次监测请求，数据需求方将相关信息传输给数据拥有方；	否
IESID	由两部分组成，数据需求方标示 ID 和数据需求方订单 ID；	否
TERM	设备的机型信息，通过调用设备提供的 API 接口获取	否
CHANNEL	展示广告，视频分类，例如：综艺，电视剧	否
VIDEONAME	展示广告具体类名，例如：喜洋洋	否
TIMESTAMP	客户端触发监测代码时间	是
APP	客户端平台类型和名称，比如 IOS_MyAPPHD	否

表 8-1 移动视频广告常规监测参数

备注：

1) iOS 操作系统采集 MAC、IDFA、OpenUDID；Android 操作系统采集 IMEI、MAC、AndroidID；Windows Phone 和其他操作系统采集 MAC；如果获取不到，则保留空值。

2) 媒体和监测公司采用一致的算法计算独立访问者，算法必须透明公开：

3) 对 Android 操作系统，依次选用 IMEI、AndroidID、MAC；对 iOS 操作系统，依次选用 IDFA、OpenUDID、MAC；对 Windows Phone 和其他操作系统，选用 MAC，用于识别独立访问者。

9. 移动视频广告常规监测指标

当前视频营销已进入多屏时代，随着移动视频广告的快速增长，对移动视频广告效果的要求越来越高，因此移动视频广告的监测标准也越来越丰富。为了满足移动端的效果监测需求，其既要符合常规互联网的监测指标，又要具有移动端所特有的移动特性指标，具体包含的常规监测指标定义如下所示：

指标	定义
Impression	是指在某一网站的指定时间周期内，广告被展现的总次数。一般地，对于视频广告，视频素材首帧画面被展现即为曝光（展示），视频播放过程可分为 1/4 展示点、1/2 展示点、3/4 展示点和全部观看完毕。
UV	在指定时间周期内访问网站的一台设备即被记为一个访客，在指定时间周期内相同的设备只会被计算一次。一般地，每个访问者都具有唯一标识，独立访问者即网络唯一访客的标识称呼
Click	在指定时间周期内，来访用户点击广告的总次数。一般地，“广告点击”的广义理解可以扩展到用户与广告的有效交互行为，比

	如：点击、摇晃、划动等，此行为将促成后续的页面行为
Clicker	在指定时间周期内，点击过某按钮或广告的总人数。一般地，每个访问者具有唯一标识，点击者即周期内产生点击行为的唯一标识的数量
OTS	Opportunity To See, 是指接触广告的浏览者或者访问者，平均每个人的接触同一个广告的次数
Reach	在指定时间周期内，用户通过点击互联网广告、搜索引擎广告、其他互联网广告链接、直接输入登录页面地址等方式，成功打开着陆页面的行为并被记录叫做一次到达
Reach N+	特定时间内看过广告 N 次及 N 次以上的设备数
CTR	Click-through Rate, 是指某一广告内容或区域被点击的次数与被展现次数之比，即点击量与曝光量的比值。计算公式为： $CTR=CL/PV$
TA Impression	Target Audience Impression, 在曝光量中属于目标受众（即某类特定人群）产生的展现总次数。
TA UV	Target Audience Unique Visitor, 在独立访问者中属于目标受众（即某类特定人群）的独立访问者
TA Reach	指到达用户中属于目标受众（即某类特定人群）的访问者人数
Ta Reach N+	目标受众到达数 / 目标设备到达数。
TA Click	Target Audience Click, 点击量中属于目标受众（即某类特定人群）产生的点击量
TA Clicker	Target Audience Clicker, 点击者中属于目标受众（即某类特定人群）的点击者
Ta CTR	目标受众点击率
Ta OTS	目标受众的平均广告展示次数
CLICK RATE	广告点击率
EVENT	广告交互后，触发事件次数
EVENT CONVERSION RATE	广告交互后，事件转化率
Average Frequency	平均覆盖频次。是指接触广告的浏览者或者访问者，平均每个人的接触同一个广告的次数。
Target Audience	目标受众。在指定周期内，成功进行各种广告内交互事件的目标用户数。根据每个访客的唯一标识进行统计。
Page View	落地页到达量。指某一广告内容或者区域被点击后进行跳转到指定的承接页面，而承接页面（通常是客户的官方网站或者活动页面）完全被打开之后所累积的访问量。
Leads	用户信息量。指在某个时间段内，某一广告通过广告内互动、后续访问官方网站等方式留下个人信息数量。有些情况下广告主也会对于这些个人信息进行真实性的回访，通过回访的结果来衡量网站用户信息收集的质量。
Forward	转发量。在指定时间周期内，访问者在点击视频转发按钮时，向监测服务器发送一条 http 请求。监测服务器为收到的每一次转发请求记录日志。统计转发日志的总数作为转发提交量。在最终社交平台呈现的成功转发记录为转发量
Comment	评论数。在指定时间周期内，访问者在点击视频广告下的评论的提交按钮时，向监测服务器发送一条 http 请求。监测服务器为收到的

	每一次评论记录日志。统计评论日志的总数作为评论提交量
Like	点赞数。在指定时间周期内，访问者点击视频下方的点赞按钮时，向监测服务器发送一条 http 请求，监测服务器为收到的每一条请求记录点赞日志。统计点赞日志的数量作为点赞提交量

表 9-1 移动视频广告常规监测指标

10. 移动视频广告常规监测方法

移动视频广告监测常规方法有 SDK 监测、API Client to Server 和 Server to Server 三种方式。三种方式的具体介绍如下：

10.1 SDK 监测

SDK 监测是指：需要在媒体播放器中嵌入 SDK，与媒体的合作度深入，需要通过媒体的认证。

在广告媒体应用内集成第三方监测平台提供的监测 SDK，通过 SDK 向第三方监测服务器发出监测请求，完成广告监测的一种监测方法。集成的监测 SDK 需要通过媒体及第三方监测的认证，远程动态加载广告内容和监测参数，并向第三方监测服务器发送监测数据，实现广告数据监测认证的第三方监测平台提供的监测 SDK，集成在广告媒体应用内。

10.2 API Client to Server

API Client to Server 是指：通过用户客户端的监测 URL 发送监测请求，完成监测，属于比较常规的监测合作。

媒体方向监测方采用 API 的方式传输数据，通过在监测 URL 中加入宏定义的方式完成，监测代码需要在用户客户端执行。宏定义如下：
 __OS__, __IMEI__, __MAC__, __ANDROIDID__, __IDFA__, __OPENUDID__（__为连续的两个下划线），详细可参见本文档《移动视频广告常规监测参数》部分的描述。

10.3 Server to Server

Server to Server 是指：服务器端对服务器端传输监测数据的监测方式，对监测的要求不高时使用。

服务器对服务器传输监测数据的监测方法，只在某些特殊情况下使用。如当无法在用户客户端执行监测代码，并且曝光、点击不是判断标准或者结算的 **KPI** 时，为达到监测目的，广告服务器和第三方监测服务器可通过该形式进行监测数据交互，统计数据仅作为参考使用。