

Letv Ad Exchange 对接文档

版本记录

版本号	完成时间	备注
1.0.0	2014-12-22	1. 对接文档初始版本。
1.0.1	2015-01-04	1. 修改 Win-notice 宏，大括号改成小括号。
1.0.2	2015-01-06	1. 删除了业务接口中 103,108,109 三个和尺寸相关的错误信息。 2. 广告上传接口中增加了广告尺寸这个必填项。
1.0.3	2015-01-09	1. 业务接口 107 错误注释添加，增加 115,116,404，请求参数 dspid 改为数字。 2. win-notice 接口中增加了对成交价格的加密。
1.0.4	2015-01-14	1. 增加了对 DSP 不参与竞价时返回格式的说明。
1.0.5	2015-01-15	1. 修改广告类型 type 为必填项
1.0.6	2015-01-16	1. 增加 AES 算法的具体细节描述
1.0.7	2015-01-20	1. AES 加密后的密文要进行 url encode 才能放到 url 中 2. 竞价请求中的 Content-Type 为 application/json 3. 增加了对竞价和监测发生流程的说明
1.0.8	2015-01-27	1. 删除 bid-response 中的 mid（素材 ID）字段
1.0.9	2015-02-04	1. 加入和媒体方接口

概述

本文档描述了接入 Letv Ad Exchange 的各接口的详细含义，包括离线数据同步接口和实时竞价接口。

其中离线数据同步接口中的广告位同步接口传输的数据和实时竞价接口中的数据是一致的（广告位底价只通过实时竞价接口给出），DSP 可以自行选择使用离线接口或者实时竞价接口获取广告位信息。

对 DSP 接口

业务数据接口

概述

Letv exchange 和 DSP 间接口使用 HTTP POST 方式，数据以 json 格式组织。下表中所列出的

dspid 和 token 字段在每次请求中是必填字段，其他字段在下述具体接口中分别说明。所有数据都使用 UTF-8 编码。

请求参数

参数名称	是否必须	含义
dspid	是	DSP 在 ADX 系统中的 ID ，用来标识使 API API 的是哪个 DSP。
token	是	DSP 对应的 token 值，由 ADX 生成，用来进行权限验证。

返回参数

参数名称	是否必须	含义
result	是	0: 执行成功 1: 系统认证失败 2: 请求参数校验错误
message	是	如果 API 执行成功，返回结果数据或成功提示信息；如果执行失败，返回具体的错误提示信息
code	否	如果 API 执行失败，来标识具体的错误代码。请参考下表中的错误代码说明

错误代码

- 101~116 同步广告时的错误类型，即接口后缀：/ad/sync
- 201 获取审核未通过的广告信息时的错误类型，即接口后缀：/ad/denylist
- 301~302 获取指定的广告审核结果时的错误类型，即接口后缀：/ad/getstatus
- 401~404 DSP 报表查询时的错误类型，即接口后缀：/report/detail

ERROR_CODE	错误注释	处理说明
101	广告加载失败	广告加载经常是因为根据广告 url 地址获取信息时超时，确保 url 正常的情况下重试
102	必填项，不支持的文件格式，目前支持的文件格式： jpg,gif,png,swf,flv,mp4	
104	执行插入过程异常	该错误经常是因为将广告记录到 Adx 超时,确保 url 正常的情况下重试
105	广告所属的广告主不能为空	添加广告的 advertiser 属性
106	广告生效时间为空或者不能解析	注：时间格式为 YYYY-mm-dd
107	广告失效时间为空或者不是有效格式或者广告失效时间小于当前时间	注：时间格式为 YYYY-mm-dd

110	参数中存在广告地址为空	
111	第三方监测地址错误	核对监测地址后重试
112	同步广告物料的 ad 参数必须要有	添加 ad 参数列表
113	广告物料的跳转地址 landingpage 参数不能为空	添加可用的 landingpage 参数
114	广告时长不能解析	将时长值修改为整数值
115	广告的宽度必填且为整数类型值	将宽度修改为整数值
116	广告的高度必填且为整数类型值	将高度修改为整数值
201	上传时间不能解析	注：时间格式为 YYYY-mm-dd
301	adurl 不能为空	添加 adurl 属性
302	该广告 url 并未同步到 Adx 中	确保 url 正常的情况下调用同步操作同步广告到 Adx
401	报表查询的开始时间为空或者不能解析	注：时间格式为 YYYY-mm-dd
402	报表查询的结束时间为空或者不能解析	注：时间格式为 YYYY-mm-dd
403	报表查询时间跨度超过 7 天	日期段跨度最大允许 7 天
404	报表查询时的结束时间早于开始时间	核对时间后重试

广告位信息获取 API

DSP 通过该接口获得 Exchange 拥有的库存信息，包括广告位总量，每个广告位的 ID、名称、尺寸以及支持的物料格式，目前系统支持的物料格式请查看 Ad Exchange 广告素材规格说明章节。

API 接口：<http://ark.letv.com/api/adzone/list>

请求示例：

POST: http://ark.letv.com/api/adzone/list
JSON: {"dspid":11268,"token":"92205dff8f9d48e1b7a26b0b88af7dc1"}

正常返回示例：

<pre> { "result":0, "message": { "total":2, "records":[{ "adzoneid": "101", "adzonename": "汽车-首页-banner", "size": "300*250", "duration":"15-60", //广告位最小-最大时长，单位：秒(s) "allowAdTypes": ["image/jpeg", "video/x-flv", "image/png"] //允许播放的广告素材类型 }, { </pre>

```
        "adzoneid": "102",
        "adzonename": "新闻-视频",
        "size": "400*300",
        "duration": "15-75",
        "allowAdTypes": ["video/x-flv"]
    }
]
```

广告信息同步 API

广告信息同步 API 包括上传广告、获取审核结果、批量获取结果三个部分。

DSP 向 exchange 上传物料，由 exchange 审核，通过审核的物料才能在 exchange 平台上投放出来。审核的结果通过获取审核结果的接口提供给 DSP。

上传广告信息 API

API 接口：<http://ark.letv.com/api/ad/sync>

业务参数：本次同步需要新增的广告信息，是 JSON 格式的数据，具体参考请求示例数据。
请求示例：

POST: <http://ark.letv.com/api/ad/sync>

```
JSON:
{
  "dspid": 11268,
  "token": "92205dff8f9d48e1b7a26b0b88af7dc1",
  "ad": [
    {
      "url": "http://somecdn.com/media/v1/00.jpg",
      // 广告的地址，必填
      "landingpage": "http://www.some.com",
      // 点击跳转落地页的地址，必填
      "advertiser": "广告主的中文名称",
      // 必须填写
      "startdate": "2014-11-16",
      // 物料生效时间，格式要求：YYYY-mm-dd，必须填写
      "enddate": "2014-12-30",
      // 物料失效时间，格式要求：YYYY-mm-dd，必须填写
      "monitor": [
        { "url": "http://impression.com/1", "type": "impression" },
        { "url": "http://click.com/2", "type": "click" } ],
      // 曝光监测地址：非必填，若 "type": "", 则默认为 impression，可以有多个
    }
  ]
}
```

```
"type": "jpg",
//物料类型：必填。
"duration": "",
//时长，非必填
"width": 40,
"height": 60
},
{
"url": "http://somecdn.com/media/v1/01.png",
"landingpage": "http://www.some.com",
"advertiser": "广告主的中文名称",
"startdate": "2014-11-16",
"enddate": "2015-2-26",
"monitor": [],
"type": "gg",
"duration": "",
"width": 20,
"height": 30
}]
}
```

返回示例：

```
{

  "result": 2,

  "message": {

    "102": ["http://somecdn.com/media/v1/00.png"],

    "111": ["http://somecdn.com/media/v1/01.jpg"],

    "114": [

      "http://somecdn.com/media/v1/03.jpg",

      "http://somecdn.com/media/v1/04.png"]

  }

}
```

如果返回信息是：{"result":0,"message":{}}，说明物料上传成功 即物料成功上传到我们的 exchange 系统,物料上传之后的状态是待审核。

其他情况当 result 为 2，message 是一个 object，前面的数值是错误的 code 类型，后面的数组是该错误类型下的物料 URL 列表

获取审核未通过的广告信息 API

API 接口： <http://ark.letv.com/api/ad/denylist>

业务参数：

字段名称	类型	描述
uploaddate	string	上传的时间：将返回该时间之后的所有未审核通过的广告列表 可以为空，为空时返回所有的审核未通过的广告列表 格式要求：YYYY-MM-DD

请求示例：

POST: http://ark.letv.com/api/ad/denylist
JSON: { "dspid":11268, "token":"92205dff8f9d48e1b7a26b0b88af7dc1", "uploaddate":"2014-12-04" }

返回示例：

正常返回
{ "result":0, "message": { "total":2, "records": [{ "url": "http://material.client.com/123.swf", //广告 url "reason": "显示异常" //审核不通过原因 }, { "url": "http://material.client.com/134.swf", "reason": "跳转地址异常" }] } }
校验出请求参数存在错误时的返回示例
{ "result":2, "message": "上传时间不能解析" }

批量获取指定广告的审核状态

API 接口：<http://ark.letv.com/api/ad/getstatus>

业务参数：

字段名称	类型	描述
adurl	array of string	需要查询的一个或多个广告的 URL

请求示例：

POST: http://ark.letv.com/api/ad/getstatus
JSON: { "dspid":11268, "token":"92205dff8f9d48e1b7a26b0b88af7dc1", "adurl":["http://material.client.com/134.swf","http://material.client.com/135.s wf","http://material.client.com/136.swf"] }

返回示例

正常返回
{ "result":0, "message": { "total":3, "records": [{ "url": "http://material.client.com/134.swf", "result":"不通过" "reason": "默认素材不需要上传了" }, { "url": "http://material.client.com/135.swf", "result":"通过" "reason": "" }, { "url": "http://material.client.com/136.swf", "result":"待审核" "reason": "" }] } }
业务参数校验出错误的示例
{

<pre>"result":2, "message":{"301":"adurl 不能为空"} }</pre>
或者
<pre>{ "result":2, "message": { "302" : ["http://material.client.com/135.swf","http://material.client.com/136.swf"] } }</pre>

报表获取 API

DSP 接入 ADX 平台后，可以通过 API 获取相关的报表数据，传入参数是一个时间段：开始时间和结束时间，返回的信息是[每天、每个广告位] 上的参与竞价数、成功竞价数、曝光数、点击数、计费的费用。

特别说明：为了保证数据返回的速度，一次获取的时间间隔建议不要超过一周。对于时间跨度超过一周的请求，我们将直接返回异常。

API 接口：<http://ark.letv.com/api/report/detail>

业务参数：

字段名称	类型	描述
startdate	string	报表查询的开始日期，要求是 YYYY-MM-dd 的格式，如 2012-02-01，不能为空
enddate	string	报表查询的结束日期，要求是 YYYY-MM-dd 的格式，如 2012-02-07，不能为空

请求示例：

POST: http://ark.letv.com/api/report/detail
JSON:
<pre>{ "dspid":11268, "token":"92205dff8f9d48e1b7a26b0b88af7dc1", "startdate":"2014-12-05", "enddate":"2014-12-11 13:14:15" }</pre>

返回示例：

<pre>{ "result":2, "message": { "402": "报表查询的结束时间为空或者不能解析" } }</pre>

```
}
```

或者

```
{
  "result":2,
  "message":
  {
    "403": "报表查询时间跨度超过 7 天"
  }
}
```

或者

```
{
  "result":0,
  "message":
  {
    "total" :2,
    "records" :
    {
      "2014-12-06": [
        {
          "adzoneid": 1,
          "bid": 12,
          "winbid": 8,
          "pv": 8,
          "click": 2,
          "expend": 1000
        }
      ]
      "2014-12-07": [
        {
          "adzoneid": 1,
          "bid": 162,
          "winbid": 61,
          "pv": 61,
          "click": 10,
          "expend": 3000
        },
        {
          "adzoneid": 2,
          "bid": 5445,
          "winbid": 554,
          "pv": 365,
          "click": 14,
          "expend": 9000
        }
      ]
    }
  }
}
```

```
    }
  }
}
```

实时竞价接口

概述

实时竞价发生时，exchange server 和 DSP server 间使用 HTTP POST 方式，数据格式为 json，Http 请求头中 Content-Type 为 application/json，启用 keep-alive，单次请求超时时间设置为 120ms。

竞价请求（bid-request）

字段名称	类型	描述
id	string	用来表示本次竞价请求的唯一 ID, 对应 DSP 响应中的 id 字段，由 exchange 生成。
imp	array of objects	曝光对象数组
site		
device	object	设备对象，目前仅使用了 ip 字段
user	object	用户对象
app	object	应用对象，包含移动应用的信息，目前字段为空

imp 对象

字段名称	类型	子字段	子字段类型	描述
id	string			针对每一个曝光的唯一 ID，对应 DSP 响应中 bid.impid 字段，由 exchange 生成。
adzoneid	int			广告位 id
bidfloor	double			底价
banner	object	banner 类型广告位，该字段和 video 字段有且只有一个		
		w	int	广告位宽度
		h	int	广告位高度
video	object	video 类型广告位		
		mime	array of string	支持播放的视频格式，目前有 video/x-flv application/x-shockwave-flash

				video/mp4 三种类型。
		display	int	广告位类型，参看广告位类型表。
		minduration	int	视频广告最小时长
		maxduration	int	视频广告最大时长
		w	int	广告位宽度
		h	int	广告位高度

site 对象

字段名称	类型	子字段	子字段类型	描述
name	string			媒体网站名称
page	string			当前页面 url
ref	string			Referer url
content	object	视频的内容相关信息，只有视频贴片类广告才有这个字段 目前整个字段为空		
		title	string	视频标题
		keywords	string	视频标签关键字，多个关键字用英文逗号分隔
		ext	object	扩展字段，具体描述见下方

site.content.ext 对象

字段名称	类型	描述
channel	string	视频频道 ID
cs	string	二级频道 ID

user 对象

字段名称	类型	子字段	子字段类型	描述
id	string			exchange 用户 ID
gender	string			性别，M 为男性，F 为女性，空表示未知，目前字段为空
yob	int			出生年份，4 位数字，目前字段为空
ext	object	目前字段为空		
		models		用数字形式标记的 DMP 信息标签

app 对象

该对象目前没有使用，所有字段为空

字段名称	类型	子字段	子字段类型	描述
name	string			App 的名称
ext	object	扩展信息		
		sdk	string	投放 SDK 的版本，例如“91_v1”
		market	int	应用商店编号
		appid	string	应用在应用商店内的 ID。
		cat	string	应用在应用商店内的分类 ID。
		tag	string	app 在上述应用商店内的标签(英文或中文 UTF8-urlencode 编码) 多个标签使用半角逗号分隔。
content	object			视频的内容相关信息。只有视频贴片类型的广告位才会有这个字段，同 site.content 对象

device 对象

该对象目前只填充了 ip

字段名称	类型	子字段	子字段类型	描述
ua	string			user agent
ip	string			ip
geo	object	设备的当前地理位置信息		
		lat	float	纬度（-90~90）
		lon	float	经度（-180~180）
		ext	object	扩展属性，参见 Geo.ext 详细字段解释。
didmd5	string			使用 MD5 哈希的 Device ID，也就是 IMEI 的 md5 值。
dpidmd5	string			使用 MD5 哈希的平台相关 ID，即 Android ID 或 iOS 的 UDID，或者 windows phone 的 id
make	string			设备生产商，如“Apple”
model	string			设备型号，如“iPhone”
os	string			操作系统，如“Android”
osv	string			操作系统版本号，如“4.1”
carrier	string			运营商的 ID，参见 OpenRTB 协议及 http://en.wikipedia.org/wiki/Mobile_Network_Code 。
language	string			目前使用的国家—语言，如“zh_CN”

js	integer			是否启用 Javascript, 1:启用 (默认值) 0:未启用
connectiontype	integer			网络连接类型, 和 OpenRTB 一致: 0—未知, 1—Ethernet, 2—wifi, 3—蜂窝网络, 未知代, 4—蜂窝网络, 2G, 5—蜂窝网络, 3G, 6—蜂窝网络, 4G。
devicetype	integer			设备类型, 和 0—手机, 1—平板, 2—PC, 3—互联网电视。
ext	object	扩展信息		
		idfa	string	iOS 的 IDFA 字段, 如: "1E2DFA89-496A-47FD-9941-DF1FC4E6484A"
		mac	string	MAC 地址, 如"F0B4799A8CC9"
		macmd5	string	MD5 哈希的 MAC 地址
		ssid	string	WIFI 的
		w	integer	设备的屏幕宽度, 以像素为单位
		h	integer	设备的屏幕高度, 以像素为单位
		brk	integer	设备是否越狱, 1—已启用 (默认), 0—未启用。
		ts	integer	发送请求时的本地 UNIX 时间戳(秒数, 10 进制)
		interstitial	integer	是否使用全屏/互动方式来展现广告。1—是, 0—否 (默认值)。

geo.ext 对象

该对象目前未使用，填充为空

字段名称	类型	子字段	子字段类型	描述
accuracy	integer			GPS 的精确度, 单位为米。如: 100 表示精确度为 100 米。

示例

```
{
  "id": "a9eedb41f98941e88289f829e238ded0",
  "imp": [
    {
```

```
        "id": "exchange-imp-id1",
        "adzoneid": 123,
        "bidfloor": 1.01,
        "banner": { "w": 980, "h": 90 }
    }
],
"site": {
    "name": "",
    "page": "",
    "ref": "",
    "content": {
        "title": "",
        "ext": { "channel": "", "cs": "" }
        "keywords": ""
    },
},
"device": {
    "ua": "",
    "ip": "10.58.88.39",
    "didmd5": "",
    "dpidmd5": "",
    "make": "",
    "model": "",
    "os": "",
    "osv": "",
    "carrier": "",
    "language": "",
    "js": 0,
    "connectiontype": 0,
    "devicetype": 0,
    "geo": { "lat": 0.000000, "lon": 0.000000, "ext": {"accuracy":0} },
    "ext": {
        "idfa": "",
        "mac": "",
        "macmd5": "",
        "ssid": "",
        "w": 0,
        "h": 0,
        "brk": 0,
        "ts": 0,
        "interstitial": 0
    }
},
"user": {
```

```

        "id": "881996789ad2d2acc3507e435a0497c",
        "gender": "",
        "yob": 0,
        "ext": { "models": [] }
    },
    "app": {
        "name": "",
        "ext": {
            "sdk": "",
            "market": 0,
            "appid": "",
            "cat": "",
            "tag": ""
        }
    }
}

```

竞价响应（bid-response）

字段名称	类型	描述
id	string	exchange 生成，用来标识这一次 http 请求，会被记录到竞价日志中
bidid	string	DSP 生成，需要填写到 DSP 的 win-notice 地址中
seatbid	array of objects	bid 对象数组的数组，每个 bid 对象代表针对单次曝光的出价信息。由于 seatbid 中包含的对象本身是个(bid 对象的)数组，所以 seatbid 只有一个元素。

bid 对象

seatbid 数组中的元素也是一个数组，其中包含多个 bid 对象。

bid 对象数组中的每一个元素表示一次 HTTP 请求中的一个广告曝光请求，在 exchange 的内部实现中，是用 impid 字段来作为唯一标示的。对每一个曝光请求，如果 DSP 选择不参与竞价，就不需要返回对应的 bid 对象，如果一次 HTTP 请求中的所有广告曝光请求都不参与，就直接返回一个空的 bid 数组。具体看下面示例。

下面是 bid 对象的格式。

字段名称	类型	描述
id	string	DSP 生成，针对每一次曝光，同样需要填写到 DSP 的检测地址中
impid	string	exchange 生成，针对每一次曝光，会被记录到竞价日志中。同时 exchange 用这个 ID 来区别多个曝光请求的响应
price	double	单位为分
nurl	string	win notice url。 win notice 中包含三个宏：

		<p>\$(DSP_BID_ID)\$ \$(DSP_BID_IMP_ID)\$ \$(DSP_BID_PRICE)\$</p> <p>其中： \$(DSP_BID_ID)\$ 会被替换成 bid-response 中的 bidid， \$(DSP_BID_IMP_ID)\$会被替换成 bid-response 中 bid 对象的 id， \$(DSP_BID_PRICE)\$会被替换成实际成交价。</p> <p>其中，价格部分会通过 aes 加密，然后通过 base64 编码转换成可打印字符，最后进行 url encode 后放到 url 中，加密时使用 DSP token 作为密钥。解密的时候，需要首先进行 url decode，然后做 base64 解码，最后再用 aes 解密才能得到真实价格。</p> <p>Aes 的算法模式为 ECB，密钥长度为 256 位，补码方式采用 PKCS5Padding。密钥按照字符串来处理（每个字符代表 8bit），超过 32 字节的密钥会被截断。</p>
adm	string	素材 url，这个字段既是素材的实际地址，同时也用来识别素材。 url 必须以"http://"开头。
ext	object	扩展字段

ext 对象

字段名称	类型	描述
ldp	string	点击跳转地址
pm	array of string	曝光检测 url
cm	array of string	点击检测 url
type	string	素材类型，形式为 MIME Type

示例

正常返回
<pre>{ "id": "exchange-id", "bidid": "dsp-id-1111", "seatbid": [{ "bid": [{ "id": "dsp-imp-id1", "impid": "exchange-imp-id1", "price": 11.11,</pre>

<pre> "nurl": "http://www.dsp1.com/winnotice1", "adm": "http://www.cdn.com/a1.jpg", "ext": { "ldp": "http://www.aol.com", "pm": ["http://www.letv.com/pv1", "http://www.dsp.com/pv1",], "cm": ["http://www.letv.com/click1", "http://www.dsp.com/click1",], "type": "image/jpeg" } } } </pre>
<p>不参与竞价</p> <pre> { "id": "exchange-id", "bidid": "dsp-id-1111", "seatbid": [{ "bid": [] }] } </pre>

Win-notice

DSP 在实时竞价过程中通过 bid-response 中的 nurl 字段向 exchange 通知 win-notice 发送地址。nurl 中可以包含三种宏，由 exchange 来替换。

例如：nurl 为：

```
http://www.dsp.com/win-notice?bidid=${dsp_bid_id}&bidimpid=${dsp_bid_imp_id}&price=${dsp_bid_price}
```

其中的 \${DSP_BID_ID} 会被替换成 bid-response 中的 bidid，\${DSP_BID_IMP_ID} 会被替换成 bid-response 中 bid 对象的 id，\${DSP_BID_PRICE} 会被替换成实际成交价。

其中，价格部分会通过 aes 加密，然后通过 base64 编码转换成可打印字符，最后进行 url encode 后放到 url 中，加密时使用 DSP token 作为密钥。

解密的时候，需要首先进行 url decode，然后做 base64 解码，最后再用 aes 解密才能得到真实价格。

Aes 的算法模式为 ECB，密钥长度为 256 位，补码方式采用 PKCS5Padding。密钥按照字符串来处理（每个字符代表 8bit，就是说需要 32 字节长的密钥），超过 32 字节的密钥会被截断。Exchange 向媒体前端返回时，会将 win-notice 当成 DSP 方的曝光监测地址之一，由前端广告播放器向 DSP 发送 win-notice。

Cookie-mapping

Letv exchange 使用 letv.com 的流量资源和 DSP 进行 cookie-mapping，由 DSP 方存储 cookie 数据。具体过程如下：

1. DSP 提供 cookie-mapping 接口地址，例如：
http://www.dsp.com/cookie-mapping/letv?id={letv_user_id}
2. Letv 通过浏览器，APP 等客户端向 cookie-mapping url 发送请求，请求中 letv_user_id 为 letv exchange 中的用户 ID
3. DSP 收到请求后，保存 DSP cookie 中的用户 ID 和 letv_user_id 的对应关系。

竞价和监测发生流程

在正常无用户干预的情况下，竞价可能比播放提前 0 到 60s：

在不考虑用户端网络延迟的情况下，

对于前贴广告，如果第一个前贴广告由 exchange 系统投出，那么竞价和曝光几乎是同时发生（可能有一些网络延迟）；如果是第五支前贴广告由 exchange 系统投出，那么曝光会在竞价 60s 之后发生。

对于中插广告，竞价发生在广告播放前 30s。

但是在有用户干预的时候，竞价时间和广告播放时间之间的时间差是不确定的，竞价和曝光监测的时间差会拉长，任何时间长度都有可能。

无论任何情况下，曝光监测一定是在广告播放的同时并行发送的，点击监测一定是在点击事件发生时同时并行发送的。

对媒体接口

VAST 3.0 接口

接口 URL

乐视 Ad Exchange 的接口 URL 为

<http://ark.letv.com/adx?adzone=123&res=xml>

其中各个参数以及含义如下表：

参数	说明
adzone	乐视 exchange 广告位 id，可以有多个，用英文逗号分隔。

	同一个广告位 id 允许出现多次，只要有多个广告位 ID，就会返回多个广告。
res	响应的类型，可以是 json，jsonp，xml 三者之一
j	当 res=jsonp 时，这个参数指明前端调用的函数名，在请求发生时由前端填入。 例如： 请求是 res=jsonp&j=fun_name 那么返回的 jsonp 数据就是 <i>fun_name (完整 json)</i> ;

此处 adzone 是乐视 exchange 中定义的广告位 ID，媒体方接入时，需要在发送请求时填入对应的广告位 ID。

VAST 数据格式

VAST 格式分为线性和非线性两种。其中，前贴，中贴，后贴会返回线性格式，暂停广告会返回非线性格式。

示例如下：

线性广告
<pre> <!-- 线性类型素材 --> <?xml version="1.0" encoding="UTF-8"?> <VAST version="3.0"> <Ad id="44781"> <!-- 此处的ID是exchange中的分配的广告位ID --> <InLine> <AdSystem version="1"><![CDATA[letv exchange]]></AdSystem> <!-- 曝光地址有1到多个，id属性可以不用 --> <Impression><![CDATA[http://ark.letv.com/et?mid=182942]]></Impression> <Impression><![CDATA[http://vletv.admaster.com.cn/i/a23351,b200550473,c1760,i200002107,m202,h]]></Impression> <Creatives> <Creative> <Linear> <Duration><![CDATA[00:00:15]]></Duration> <!-- 视频时长 --> <!-- 整个trackingevents字段都暂不支持，只有曝光监测 --> <TrackingEvents></TrackingEvents> <MediaFiles> <!-- id是乐视广告平台中的广告素材ID， delivery 属性暂时不支持，"progressive"参数没有用处， 宽高等属性已经支持，是素材的实际尺寸， apiFramework 现在不支持，"VPAID"参数没有用处 --> </pre>

```

        <MediaFile id="123"
type="application/x-shockwave-flash" delivery="progressive"
width="640" height="360" apiFramework="VPAID">

<![CDATA[http://i1.letvimg.com/gugwl/201311/04/miaoZhenApp.swf]]>
        </MediaFile>
    </MediaFiles>

    <!-- 这个字段可以完全忽略，在后续版本中会删除 -->
    <AdParameters
xmlEncoded="false"><![CDATA[{"tMethod":
"", "reqType":0, "pid":0, "autoPlay":0, "ast":0}]]></AdParameters>
        <VideoClicks>
            <!-- 点击跳转，只能一个 -->

<ClickThrough><![CDATA[http://ark.letv.com/et?mid=182941&u=http%3A%2F
%2Fc.admaster.com.cn%2Cc1760%2Ci200002107%2Cm101%2Ch]]></ClickThrough
>

            <!-- 点击监测，在点击跳转同时请求，可有0到多个 -->

<ClickTracking><![CDATA[http://ark.letv.com/et?mid=182943]]></ClickTr
acking>

<ClickTracking><![CDATA[http://letv.m.cn.miaozhen.com/x.gif?k=1017163
&p=40LJH0&rt=2&ns=[M_ADIP]&ni=[M_IESID]&v=[M_LOC]&o=]]></ClickTrackin
g>

        </VideoClicks>
    </Linear>
</Creative>
</Creatives>
</InLine>
</Ad>
</VAST>

```

非线性广告

```

<!-- 非线性类型素材 -->
<?xml version="1.0" encoding="UTF-8"?>
<VAST version="3.0" code="" cuid="00000000000000000000000000000000">
    <Ad id="44781"> <!-- 此处的ID是exchange中的分配的广告位ID -->
        <InLine>
            <AdSystem version="1"><![CDATA[letv exchange]]></AdSystem>
            <!-- 有1到多个曝光监测地址 -->

<Impression><![CDATA[http://ark.letv.com/et?mid=191454]]></Impression
>

```

```
<Impression><![CDATA[http://s.cr-nielsen.com/cgi-bin/hat?_t=i&_htsinfo=dT0yJmM9ODAwMDAwNDEmZz0xMDAwMTQ3NSZhPTMwMDE2NDgx&_htspl=]]></Impression>
```

```
<Impression><![CDATA[http://8b85ya.v.admaster.com.cn/i/a17555,b200337509,c1760,i0,m202,h,uhttp://p-cn.acxiom-online.com/pixel/trc]]></Impression>
```

```
<Impression><![CDATA[http://g.cn.miaozhen.com/x.gif?k=1012334&p=4+QS00&rt=2&ns=[M_ADIP]&ni=[M_IESID]&na=[M_MAC]&v=[M_LOC]&o=]]></Impression>
```

```
<Creatives>
```

```
<Creative>
```

```
<!-- 只返回一个素材 -->
```

```
<NonLinearAds>
```

```
<!-- minSuggestedDuration保留, 值为空, apiFramework保留, 值为空 -->
```

```
<!-- width 和 height两个属性需要填上 -->
```

```
<NonLinear minSuggestedDuration="" apiFramework="" width="640" height="960">
```

```
<!-- 多个点击监测地址 -->
```

```
<NonLinearClickTracking><![CDATA[http://ark.letv.com/et?mid=191788]]></NonLinearClickTracking>
```

```
<NonLinearClickTracking><![CDATA[http://vletv.admaster.com.cn/i/a23676,b200565840,c1762,i0,m202,h]]></NonLinearClickTracking>
```

```
<!-- 一个点击跳转地址 -->
```

```
<NonLinearClickThrough><![CDATA[http://ark.letv.com/et?mid=191786&u=http%3A%2F%2Fc.admaster.com.cn%2Fc%2Fa22976%2Cb200535951]]></NonLinearClickThrough>
```

```
<!-- creativeType 中填写素材类型, 值是素材地址 -->
```

```
<StaticResource creativeType="image/jpeg">
```

```
<![CDATA[http://i3.letvimg.com/gugwl/201410/22/23day-640-960.jpg]]>
```

```
</StaticResource>
```

```
<!-- 这个字段可以完全忽略, 在后续版本中会删除 -->
```

```
<AdParameters
```

```
xmlEncoded="false"><![CDATA[{"url":
```

```
"","type":"","duration":""}}]]></AdParameters>
</NonLinear>
</NonLinearAds>
</Creative>
</Creatives>
</InLine>
</Ad>
</VAST>
```

数据字典

Ad Exchange 广告素材规格说明

乐视 Ad Exchange 支持为视频，图片，Flash 动画三种类型的广告素材：

视频文件格式为 flv 和 mp4，mp4 必须为 h.264 编码，

图片文件格式为 jpeg，png，gif，其中 gif 只能用单帧，

普通 Flash 动画（swf）中不要写影响播放器运行的脚本。

对于前贴，中贴，后贴以上格式都支持，分辨率方面，最小 400*300，最大 1920*1080，文件大小方面，建议 15s 不要超过 2M，30s 不要超过 4M。

对于暂停，只支持普通 Flash 动画和图片，分辨率方面，最小 400*300，最大 640*480。文件尺寸不要超过 2M。

物料类型	对应文件类型	MIME	备注
png	*.png	image/png	PNG 图片
jpg	*.jpg,*.jpeg	image/jpeg	JPG 或 JPEG 图片
gif	*.gif	image/gif	GIF 图片
flv	*.flv	video/x-flv	FLV 视频
swf	*.swf	application/x-shockwave-flash	Flash 动画
mp4	*.mp4	video/mp4	MP4 视频

广告位类型表

值	描述
0	页面广告
1	拉幕广告
2	前贴片广告
3	标版广告
4	中贴片广告

5	后贴片广告
6	暂停广告
7	浮层广告