# ADX-RTB详细设计文档

|  |  |  |  |
| --- | --- | --- | --- |
| **修订记录** | | | |
| 日期 | 修改原因 | 描述 | 作者 |
| 2017.03.28 | 初稿 | ADX-RTB设计文档 | 许炜硕 |
| 2017.03.28 | 修改 | 补充内容 | 许炜硕 |
| 2017.05.04 | 修改 | 修改QPS计算方式 | 许炜硕 |

# 一、文档概述

本文档介绍ADX-RTB系统中，相关功能模块的详细技术设计。

# 功能模块及详细设计

## 1. RTB总体流程

ADX Request

Bid Request前

Bid Request

Bid Response

Bid Response后

ADX Response

1. ADX Request： ADX流量入口
2. Bid Request前： 筛选流量，计算可参与竞价的DSP
3. Bid Request： 发送竞价请求给DSP
4. Bid Response： DSP返回竞价响应
5. Bid Response后：计算此次竞价结果
6. ADX Response： 返回此次结果，作为ADX Request的响应

## 2. Bid Request前

本模块中完成对ADX Request的参数解析、筛选符合竞价条件的DSP。

### 2.1 ADX Request参数解析

1. 根据ADX Request请求，解析出广告位id、ip、ua、设备等信息；
2. 根据广告位id查询出广告位类型（PC视频、app视频、app banner等）及其他信息。

### 2.2 筛选DSP

匹配ADX Request参数与ADX CMS中设定的规则，筛选符合竞价条件的DSP。

筛选过程：

1. 判断广告位是否开启，未开启则退出；
2. 根据本次ADX请求的ip获取地域，从redis的地域-DSP列表中获取所有可用DSP；
3. 循环步骤2中的可用DSP列表，开始过滤所有设定的过滤规则；
4. 根据DSP过滤规则进行过滤，包括：广告位、端、尺寸、广告类型等；
5. 根据媒体过滤规则进行过滤，包括：是否屏蔽了DSP；
6. 根据DSP响应性能进行过滤，包括：DSP的QPS，出价比率，按照出价比率排序DSP列表；
7. 根据RTB的请求性能进行DSP数量限制，例如：限定DSP数量上限为50。
8. 得到最终需要发送Bid Request的DSP列表。

## 3. Bid Request

本模块根据接收的ADX Request信息、流量所属媒体信息、用户设备信息等，依据OpenRTB协议规则，生成Bid Request请求。

Bid Request示例 - Banner：

{

"id": "80ce30c53c16e6ede735f123ef6e32361bfc7b22",

"at": 1,

"cur": [ "USD" ],

"imp": [

{

"id": "1",

"bidfloor": 0.03,

"banner": {

"h": 250, "w": 300, "pos": 0

}

}

],

"site": {

"id": "102855",

"cat": [ "IAB3-1" ],

"domain":"www.foobar.com",

"page": "http://www.foobar.com/1234.html ",

"publisher": {

"id": "8953",

"name": "foobar.com",

"cat": [ "IAB3-1" ],

"domain": "foobar.com"

}

},

"device": {

"ua": "Mozilla/5.0 (Macintosh; Intel Mac OS X 10\_6\_8) AppleWebKit/537.13 (KHTML, like Gecko) Version/5.1.7 Safari/534.57.2",

"ip": "123.145.167.10"

},

"user": {

"id": "55816b39711f9b5acf3b90e313ed29e51665623f"

}

}

详细设计参见OpenRTB模块的设计。

## 4. Bid Response

本模块根据接收到的Bid Response信息，校验其格式，并解析出竞价相关信息，包括：竞价请求id、报价、物料、win notice url等。

Bid Response示例：

{

"id": "1234567890",

"bidid": "abc1123",

"cur": "USD",

"seatbid": [

{

"seat": "512",

"bid": [

{

"id":"1",

"impid":"102",

"price":9.43,

"nurl":"http://adserver.com/winnotice?impid=102", "iurl":"http://adserver.com/pathtosampleimage", "adomain":["advertiserdomain.com"],

"cid": "campaign111",

"crid": "creative112",

"attr": [ 1, 2, 3, 4, 5, 6, 7, 12 ]

}

]

}

]

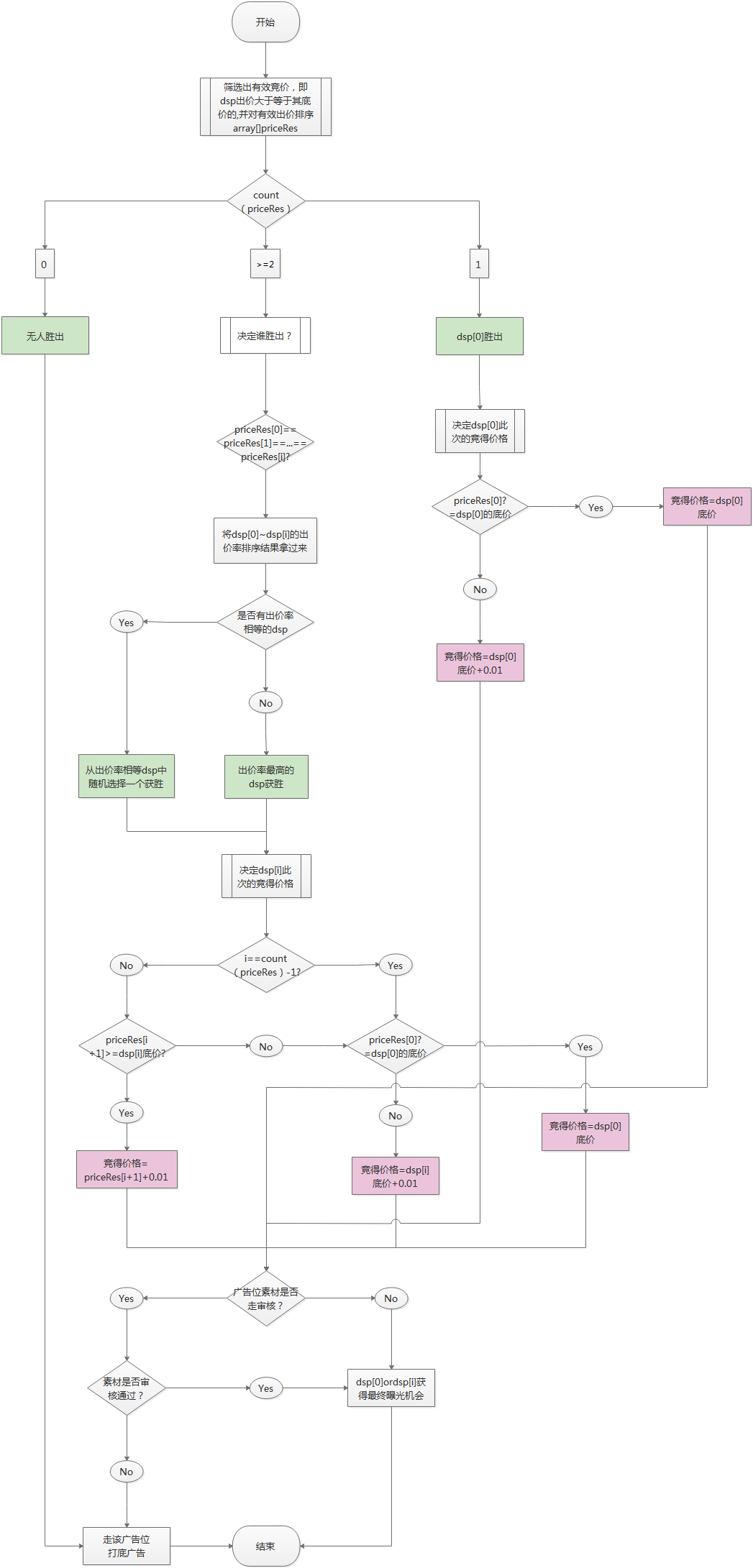
}

详细设计参见OpenRTB模块的设计。

## 5. Bid Response后

### 5.1 计算竞价结果

计算流程参照RTB需求文档中的计算竞价结果流程图。



### 5.2 发送win notice

对于竞价成功的DSP：

1. 读取Bid Response中的nurl；
2. 如果1中nurl为空，则读取ADX CMS中DSP设置的win notice url；
3. 如果2中也为空，则不发送win notice；
4. 对nurl中的参数进行宏替换；
5. 发送win notice给相应的DSP。

## 6. ADX Response

根据ADX Request请求的广告位类型，以及竞价成功的DSP的Bid Response（或者打底广告），返回不同形式的ADX Response。

视频位返回VAST3.0格式的内容，Banner位返回JSON格式的内容。

## 7. 日志生成模块

本模块用于在ADX请求生命周期的不同阶段，生成不同的日志数据。在ADX请求结束前，将本次生命周期产生的日志打包后，发送给日志代理模块。

详细的日志生成阶段及报数内容参见【adx报数规则】文档。

## 8. Cookie Mapping

1. 在接收到ADX Request时，生成adxcookieid；

2. 发送Bid Request时，携带adxcookieid；

3. 返回ADX Response时，加入ADX的cookie mapping url，url中加入dspid参数。js代码创意的广告位放在js代码中，其他广告位放在pv节点中；

4. 客户端上报ADX的cookie mapping url；

5. ADX的cookie mapping服务器收到cm请求，解析dspid参数，302到相应dsp的cookie mapping url，携带adxcookieid；

6. DSP接收到ADX的302请求，完成cookie mapping映射。

## 9. 竞价代理

在一次ADX请求的生命周期中，需要发送多个不同的Bid Request到不同的DSP，等待DSP响应返回Bid Response。出于效率和扩展性的考虑，独立出一个竞价代理的功能模块，完成竞价请求的转发和接收功能。

DSP

Nginx代理

竞价代理

RTB主流程

Nginx代理

竞价代理

DSP

DSP

Nginx代理

竞价代理

竞价代理流程：

1. RTB主流程中，将所有参与竞价的DSP的Bid Request数据及对应的DSP URL映射好，选择一个竞价代理发送Bid Request数据；
2. 竞价代理接收到竞价数据，按DSP进行拆分，通过本地的Nginx代理，并行的向多个DSP发送Bid Request；
3. 竞价代理在规定的等待时间内（100毫秒），等待DSP的Bid Response，忽略超时的DSP响应；
4. 竞价代理映射好DSP和Bid Response数据，返回给RTB主流程。

## 10. 日志代理

在一次ADX请求的生命周期中，在多个环节中都会产生日志数据。在ADX请求完成之前，需要将本次生命周期中的日志数据打包，发送给DT的报数API。与竞价代理类似，也设置一个日志代理模块，用于完成日志数据的转发功能。

## 其他

### 11.1 QPS计算方式

1. 根据ADX CMS设置的DSP的QPS值，换算成每1分钟的请求数上限，记录到Redis中。例如某DSP在ADX CMS中设置的QPS上限是1000，则换算成1分钟的请求数上限是60000。

Redis中的key类似：dsp\_qps\_{dspid}，

Redis中的value为DSP允许ADX发送的竞价请求数上限；

1. 发送Bid Request前：

先读取DSP设置的QPS值，如果为空则说明未设置QPS，允许发送竞价请求；

如果不为空，则读取相应DSP在Redis中dsp\_qps\_{dspid}的值，大于0则说明未达到QPS上限，允许ADX向相应DSP发送竞价请求；否则说明达到QPS上限，不允许发送竞价请求；

1. 发送Bid Request后，将相应DSP的dsp\_qps\_{dspid}的值减1；
2. ADX CMS设置定时任务，每1分钟更新所有DSP的dsp\_qps\_{dspid}的值，按照CMS中设置的QPS值换算成1分钟的请求数上限。