Provision of this Specification does not guarantee connection to the BidSwitch Platform and/or to any Supply Side Partners and/or any Demand Side Partners that may be connected to the BidSwitch Platform from time to time.
# Table of Contents

Introduction .................................................................................................................. 4
Change history ............................................................................................................... 4
User Matching ............................................................................................................. 5
  DSP Initiated User Matching ..................................................................................... 6
  Bid Switch Initiated User Matching ......................................................................... 7
  Opting out users ........................................................................................................ 7
Bid Request .................................................................................................................. 8
  Bid Request Object .................................................................................................. 8
  Impression Object .................................................................................................... 9
  Impression Ext Object ............................................................................................... 10
    Impression Ext Rubicon Object ............................................................................. 10
    Impression Ext Google Object ............................................................................. 10
    Impression Ext YieldOne Object .......................................................................... 11
  Banner Object .......................................................................................................... 11
  Video Object ............................................................................................................ 12
  Video Ext Object ...................................................................................................... 13
  Device Object .......................................................................................................... 13
  Geo Object ................................................................................................................ 14
  User Object .............................................................................................................. 15
  Site Object ............................................................................................................... 15
  Site Ext Object ......................................................................................................... 16
  App Object .............................................................................................................. 16
  Private Marketplace Object ..................................................................................... 17
  Deal Object ............................................................................................................... 17
  Publisher Object ....................................................................................................... 17
  Ext Object ................................................................................................................ 18
    Google Object ....................................................................................................... 18
    Detected_vertical Object ....................................................................................... 18
    Adtruth Object ....................................................................................................... 19
    Ghostery Object ..................................................................................................... 19
    Creative Parameters Object .................................................................................. 20
  Regulations Object ................................................................................................... 20
Examples ..................................................................................................................... 21
  Simple Banner Ad .................................................................................................... 21
  Private Deal and Creative Parameters ..................................................................... 22
  Video Ad .................................................................................................................... 24
  In-App Ad .................................................................................................................. 25
Bid Response ................................................................................................................. 27
  Top Level Bid Response Object .............................................................................. 27
  Seat Bid Object ....................................................................................................... 28
  Bid Object ................................................................................................................ 28
  Ext Object ............................................................................................................... 29
### SSP Specific Fields

- LiveIntent Object ................................................................. 30
- Google Object ........................................................................... 30
- Yieldone Object ........................................................................ 31

### Nbid Response

- Examples ...................................................................................... 31
- Simple Bid Response .................................................................... 31
- Image-only Bid Response .......................................................... 32
- In-App Bid Example ..................................................................... 33
- Video Bid Example ...................................................................... 33
- No Bid Example ............................................................................ 34
- Response to Private Bid Request ................................................. 34

### Macros

- Win price macro ............................................................................ 35
- SSP click tracking url macro ........................................................ 35

### BidSwitch Categories

- Creative Preview ........................................................................... 36
- Creative Approval ......................................................................... 36
- Creative Approval Parameters ...................................................... 37
- Sample Creative Status XML ......................................................... 37

### Additional Features

- QPS Management .......................................................................... 37
- Bid Targeting ............................................................................... 38
- User Sync Targeting ..................................................................... 38
Introduction

Bid Switch is an innovative technology developed at IPONWEB to simplify integration of Demand Side Platforms (DSP) with Sell Side Platforms (SSP). The key idea behind the service is to enable fast and cheap SSP integration without many rounds of specification refinements and development between DSP and SSP. Bid Switch acts like an intermediate between DSP and SSP which adapts custom SSP protocols to OpenRTB 2.0 standards. The present specification defines the protocol for interaction between DSP and Bid Switch.

Change history

<table>
<thead>
<tr>
<th>Version</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>v1.0 - 1.3</td>
<td>Initial version</td>
</tr>
<tr>
<td>v1.4</td>
<td>Added support of bcat, battr and badv fields, ext.openx section containing pub_blocked_content field.</td>
</tr>
<tr>
<td>v1.5</td>
<td>Updated wording of bid response bid.nurl field description. Added bid request ext.aol section containing denied_categories field.</td>
</tr>
<tr>
<td>v1.6</td>
<td>Updated wording of fields descriptions. Added bid response ext.agency_name and ext.advertiser_name fields.</td>
</tr>
<tr>
<td>v1.7</td>
<td>Added bid request imp.bidfloor field. Added support for HTTP 204 as a valid nobid response.</td>
</tr>
<tr>
<td>v1.8</td>
<td>Added bid request site.cat field.</td>
</tr>
<tr>
<td>v1.9</td>
<td>Added SSP click macro support. Added support for video traffic. Added bid request PMP section. Deprecated ssp-specific bid request fields: ext.deals, ext.rubicon, ext.aol, ext.openx.</td>
</tr>
<tr>
<td>v1.10</td>
<td>Changed video.protocol type to array of integer, fixed video request example. Added bid request ext.is_secure field. Added bid response ext.language and ext.cat fields.</td>
</tr>
<tr>
<td>v2.0</td>
<td>Added application traffic support. Bid request site object is optional now.</td>
</tr>
</tbody>
</table>
Added ext.google.excluded_attribute and ext.google.allowed_vendor_type bid request fields.
Added seatbid.bid.ext.google bid response object.
Added seatbid.bid.cid bid response field.
Removed fields and objects deprecated in v 1.9.

v 2.1
Added seatbid.bid.ext.js_url bid response field.
Bid response seatbid.bid.nurl field is optional now.
Added pmp.deals.wseat, imp.banner.expdir and “at” bid request fields.

v 2.2
Added imp.video.protocols, imp.banner.api, site.publisher.name, app.publisher.name, app.name, video.companionad.id, device.os, device.osv, device.make and device.model bid request fields.
Added imp.ext.google and ext.ghostery bid request objects.
Deprecated imp.video.protocol, ext.google.excluded_attribute and ext.google.allowed_vendor_type bid request fields.
Added extended sensitive categories, used in bcat field.

v 2.3
Added site.ext.mobile_site and imp.banner.mimes bid request fields.
Added imp.ext.yieldone, imp.pmp and ext.creative_params bid request objects.
Added seatbid.bid.ext.duration, seatbid.bid.ext.img_url and seatbid.bid.ext.click_url bid response fields.
Deprecated seatbid.bid.ext.liveintent and pmp objects.
Updated btype field description. Important: usage of nurl, ext.js_url or ext.liveintent.img_url bid response fields depends on btype bid request field value.
The field imp.video.linearity is now an optional field.
Added video bid example.
Updated User Matching section.
Added Creative Approval section.

v 2.4
Added tmax, app.paid, regs.coppa, device.dnt, device.ipv6, device.js, device.flashver, site.name, imp.ext.inventory_class, imp.displaymanager, imp.displaymanagerver, imp.ext.yieldone.allowed_creatives, imp.video.pos and imp.instl bid request fields.
Added seatbid.bit.ext.yieldone.creative_type bid response field.
Deprecated imp.ext.yieldone.inventory_class bid request field.
The field device.ip is now optional for in-app traffic.
Changed imp.pmp.deals.at values to comply with OpenRTB 2.2.

User Matching

Bid Switch has an integrated user matching functionality to simplify user sync operations for DSP. User matching information is managed within Bid Switch database, so there is no need for DSP to implement user matching logic to store the mappings.

While making bid request to DSP, Bid Switch will send both DSP user ID and Bid Switch user ID
in \textit{buyeruid} and \textit{id} fields of User Object respectively (see Bid Request). If DSP user ID is not available, only Bid Switch user ID will be sent. For these purposes Bid Switch needs to update User Object in bid request while proxying it from an exchange to the DSP.

User synchronization can be initiated either by DSP or by Bid Switch itself. For both synchronization types Bid Switch supports secure and insecure communications with DSP using HTTP and HTTPS protocols.

**DSP Initiated User Matching**

To sync a particular user with Bid Switch, the DSP should initiate the user to call for 1x1.gif image by the following GET URL:

\begin{verbatim}
http://x.bidswitch.net/sync
\end{verbatim}

providing the following HTTP request parameters:

<table>
<thead>
<tr>
<th>Param</th>
<th>Value Type</th>
<th>Default</th>
<th>Example</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>dsp_id</td>
<td>string</td>
<td>-</td>
<td>123</td>
<td>ID assigned to DSP by Bid Switch.</td>
</tr>
<tr>
<td>user_id</td>
<td>string</td>
<td>-</td>
<td>1234567890</td>
<td>User ID in DSP system. If the value is an empty string, the user will be considered not matched and no further sync requests would be send for this user for the “expires” period.</td>
</tr>
<tr>
<td>expires*</td>
<td>integer</td>
<td>90</td>
<td>30</td>
<td>Expiration time in days for user matching. Should be less or equal to 365.</td>
</tr>
</tbody>
</table>

* This field is optional.

Bid Switch matching URL also supports secure HTTPS connections.

There is an option to exclude a particular user from syncing for a certain time interval. To make it happen DSP can perform 302 redirect to the matching URL leaving \textit{user_id} parameter blank.

**Example 1.** DSP initiated user matching:

\begin{verbatim}
https://x.bidswitch.net/sync?dsp_id=123&user_id=1234567890&expires=30
\end{verbatim}

**Example 2.** User opt out:
Redirecting a user to the following URL will not sync this specific user for the next 5 days and will consider bids from this user as user-unmatched for the given DSP:

http://x.bidswitch.net/sync?dsp_id=123&user_id=&expires=5

**Bid Switch Initiated User Matching**

BidSwitch will also be able to initiate user synchronization. The partner will issue Bid Switch a matching URL that will redirect to the pixel URL described in DSP Initiated User Matching, with properly populated `dsp_id`, `user_id` and `expires` parameters. Note that this URL should be for a pixel that performs a 302 redirect, rather than for an iframe with a script, etc. within it. The only redirect from the DSP's matching URL should be to the Bid Switch pixel.

The matching redirect URL, issued by a DSP should contain `${SSP}` macro. This macro will be substituted with the SSP ID on Bid Switch side before sending a user to the partner. The value of the macro should be sent back to Bid Switch in the `ssp` parameter. The example of the matching URL, provided by partner is shown below:

http://www.dsp-example.com/bsw_sync?bidswitch_ssp_id=${SSP}

Here, `bidswitch_ssp_id` parameter name can be changed to any other name chosen by DSP. On receiving call to the URL above the partner should respond with 302 redirect to Bid Switch matching URL with valid `dsp_id`, `user_id` and `expires` parameters as described in DSP Initiated User Matching and providing one additional `ssp` parameter with the value of `${SSP}` macro. So given a call to the partner matching URL of the form

http://www.dsp-example.com/bsw_sync?bidswitch_ssp_id=ru

DSP server should respond with 302 redirect to Bid Switch matching URL with the following sample `dsp_id`, `user_id` and `expires` parameters:

http://x.bidswitch.net/sync?dsp_id=123&user_id=123&expires=5&ssp=ru

In addition to HTTP protocol the matching redirect URL, provided by the partner should also support secure HTTPS connections. If Bid Switch initiate a user call using HTTPS protocol the partner should redirect the user to Bid Switch matching URL using secure HTTPS connection.

After Bid Switch initiated user matching Bid Switch will associate the value of `user_id` parameter with the user’s browser and make it available in any later bid requests from this user’s browser. This will be sent in the `buyeruid` field of the JSON bid request. The `user_id` should not exceed 36 characters.

**Opting out users**
The DSP may differentiate between valuable and not valuable users. The latter are usually users that don’t have any retargeting or 3p data recorded for them. It is highly recommended that in case of a sync request for a not valuable user the DSP responds with a 302 redirect to BidSwitch sync url with an empty user_id param, e.g.:

http://x.bidswitch.net/sync?dsp_id=123&user_id=&expires=5&ssp=rubicon

**Bid Request**

**Bid Request Object**

This is the top level object that is being sent to DSP. Each bid request sent from Bid Switch to DSP will contain the following fields.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>string</td>
<td>Unique ID of the bid request, provided by the exchange.</td>
</tr>
<tr>
<td>imp</td>
<td>array of objects</td>
<td>Array of Impression Objects.</td>
</tr>
<tr>
<td>device</td>
<td>object</td>
<td>Device Object which specifies the device the ad impression will be delivered to.</td>
</tr>
<tr>
<td>user</td>
<td>object</td>
<td>User Object which describes the user.</td>
</tr>
<tr>
<td>tmax*</td>
<td>integer</td>
<td>Maximum amount of time in milliseconds to submit a bid (e.g., 120 means the bidder has 120ms to submit a bid before the auction is complete).</td>
</tr>
<tr>
<td>at*</td>
<td>integer</td>
<td>Auction Type. If “1”, then first price auction. If “2”, then second price auction.</td>
</tr>
<tr>
<td>site*</td>
<td>object</td>
<td>Site Object describing the site. Either site or app object should be present.</td>
</tr>
<tr>
<td>app*</td>
<td>object</td>
<td>App Object describing the mobile application. Either site or app object should be present.</td>
</tr>
<tr>
<td>bcat*</td>
<td>array of strings</td>
<td>Blocked Advertiser Categories, using IAB taxonomy, extended by additional sensitive categories listed in BidSwitch Categories section. E.g. {“IAB10-1”, “IAB25”, “BSW3”}. Creatives belonging to at least one of the listed categories are not permitted for</td>
</tr>
</tbody>
</table>
### Impression Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>string</td>
<td>Unique ID of the impression being shown.</td>
</tr>
<tr>
<td>banner*</td>
<td>object</td>
<td>Banner Object describing the ad properties. Required for banner impressions.</td>
</tr>
<tr>
<td>video*</td>
<td>object</td>
<td>Video Object describing the ad properties. Required for video impressions.</td>
</tr>
<tr>
<td>bidfloor*</td>
<td>float</td>
<td>Bid floor in CPM as set by the SSP. The currency is the same as specified in cur field.</td>
</tr>
<tr>
<td>instl*</td>
<td>integer</td>
<td>1 if the ad is interstitial or full screen; else 0 (i.e., no).</td>
</tr>
<tr>
<td>pmp*</td>
<td>object</td>
<td>Private Marketplace object.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be sent in each request.
<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>displaymanager*</td>
<td>string</td>
<td>Name of ad mediation partner, SDK technology, or native player responsible for rendering ad (typically video or mobile).</td>
</tr>
<tr>
<td>displaymanagerversion*</td>
<td>string</td>
<td>Version of ad mediation partner, SDK technology, or native player responsible for rendering ad (typically video or mobile).</td>
</tr>
<tr>
<td>ext*</td>
<td>object</td>
<td>Impression Ext Object</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be sent in each request.

**Impression Ext Object**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>rubicon*</td>
<td>object</td>
<td>Impression Ext Rubicon Object.</td>
</tr>
<tr>
<td>google*</td>
<td>object</td>
<td>Impression Ext Google Object.</td>
</tr>
<tr>
<td>yieldone*</td>
<td>object</td>
<td>Impression Ext YieldOne Object.</td>
</tr>
<tr>
<td>inventory_class*</td>
<td>integer</td>
<td>Inventory class according to SSP classification.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be sent in each request.

**Impression Ext Rubicon Object**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>site_size_session_count*</td>
<td>integer</td>
<td>User session counter for this site/size. A &quot;user session&quot; here is 24 hours.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be sent in each request.

**Impression Ext Google Object**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>excluded_attribute*</td>
<td>array of integer</td>
<td>List of excluded creative attributes as defined by Google.</td>
</tr>
</tbody>
</table>
**allowed_vendor_type**  
array of integer  
List of allowed vendor types as defined by Google. Note that if site.publisher.id or app.publisher.id field value equals to "google_1" then the vendors listed in https://commondatastorage.googleapis.com/adx-rtb-dictionaries/gdn-vendors.txt are also allowed for bidding.

* Fields marked with asterisk are optional and may not be sent in each request.

**Impression Ext YieldOne Object**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>inventory_class*</td>
<td>integer</td>
<td>Inventory class according to YieldOne classification. DEPRECATED: use imp.ext.inventory_class instead.</td>
</tr>
<tr>
<td>allowed_creative_types</td>
<td>array of integer</td>
<td>List of allowed creative types as defined by YieldOne. For example [1, 3, 6].</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be sent in each request.

**Banner Object**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id*</td>
<td>string</td>
<td>Unique identifier for this banner object. Used for tracking multiple banner objects (e.g., in companion banner array)</td>
</tr>
<tr>
<td>w</td>
<td>integer</td>
<td>Width of the impression in pixels.</td>
</tr>
<tr>
<td>h</td>
<td>integer</td>
<td>Height of the impression in pixels.</td>
</tr>
<tr>
<td>battr*</td>
<td>array of integers</td>
<td>Blocked creative attributes as defined in OpenRTB.</td>
</tr>
</tbody>
</table>
| btype*  | array of integers | Blocked creative types as defined in OpenRTB.  
Depending on the value the following bid response fields are disallowed:  
1: N/A  
2: ext.img_url and ext.click_url  
3: ext.js_url  
4: nurl |
<p>| pos*    | integer  | Ad Position as defined in OpenRTB.                                          |</p>
<table>
<thead>
<tr>
<th><strong>Field</strong></th>
<th><strong>Type</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>topframe*</td>
<td>integer</td>
<td>If the banner is delivered in the top frame this parameter is “1”, it there’s an iframe it will be omitted.</td>
</tr>
<tr>
<td>mimes*</td>
<td>array of strings</td>
<td>Whitelist of content MIME types supported. Possible values include &quot;text/html&quot; and &quot;application/x-shockwave-flash&quot;.</td>
</tr>
<tr>
<td>expdir*</td>
<td>array of integers</td>
<td>Possible expansion directions for an expandable ad: 1: Left 2: Right 3: Up 4: Down 5: Fullscreen If the field is not present expandable creatives are not allowed.</td>
</tr>
<tr>
<td>api*</td>
<td>array of integers</td>
<td>List of supported API frameworks for this impression as defined in OpenRTB.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be sent in each request.

**Video Object**

<table>
<thead>
<tr>
<th><strong>Field</strong></th>
<th><strong>Type</strong></th>
<th><strong>Description</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td>mimes</td>
<td>array of strings</td>
<td>Content MIME types supported.</td>
</tr>
<tr>
<td>pos*</td>
<td>integer</td>
<td>Ad Position as defined in OpenRTB.</td>
</tr>
<tr>
<td>linearity*</td>
<td>integer</td>
<td>1: Linear/In-stream 2: Non-Linear/Overlay</td>
</tr>
<tr>
<td>minduration</td>
<td>integer</td>
<td>Minimum video ad duration in seconds</td>
</tr>
<tr>
<td>maxduration</td>
<td>integer</td>
<td>Maximum video ad duration in seconds</td>
</tr>
<tr>
<td>protocol*</td>
<td>array of integers</td>
<td>Accepted video bid response protocols as defined in OpenRTB. DEPRECATED: use imp.video.protocols field instead.</td>
</tr>
<tr>
<td>protocols</td>
<td>array of integers</td>
<td>Accepted video bid response protocols as defined in OpenRTB.</td>
</tr>
<tr>
<td>w*</td>
<td>integer</td>
<td>Width of the player in pixels.</td>
</tr>
<tr>
<td>Field</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>h</td>
<td>integer</td>
<td>Height of the player in pixels.</td>
</tr>
<tr>
<td>startdelay</td>
<td>integer</td>
<td>Indicates the start delay in seconds or generic values below: 0: Pre-roll -1: Generic mid-roll -2: Generic post-roll</td>
</tr>
<tr>
<td>batt*</td>
<td>array of integers</td>
<td>Blocked creative attributes as defined in OpenRTB.</td>
</tr>
<tr>
<td>minbitrate</td>
<td>integer</td>
<td>Minimum bit rate in Kbps.</td>
</tr>
<tr>
<td>maxbitrate</td>
<td>integer</td>
<td>Maximum bit rate in Kbps.</td>
</tr>
<tr>
<td>api*</td>
<td>array of integers</td>
<td>List of supported API frameworks for this impression as defined in OpenRTB.</td>
</tr>
<tr>
<td>companionad*</td>
<td>array of objects</td>
<td>If companion ads are available, they can be listed as an array of banner objects. See Banner Object.</td>
</tr>
<tr>
<td>ext*</td>
<td>object</td>
<td>Video Ext Object</td>
</tr>
</tbody>
</table>

* Fields marked with an asterisk are optional and may not be sent in each request.

**Video Ext Object**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>skippable*</td>
<td>integer</td>
<td>Indicates whether the video ad may be skippable (i.e. contain a skip button). The values are: 0: allow skippable 1: require skippable 2: deny skippable</td>
</tr>
</tbody>
</table>

* Fields marked with an asterisk are optional and may not be sent in each request.

**Device Object**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ip*</td>
<td>string</td>
<td>IPv4 address closest to the device. Bids containing</td>
</tr>
<tr>
<td>Field</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------</td>
<td>-----------</td>
<td>-----------------------------------------------------------------------------</td>
</tr>
<tr>
<td>invalid IP addresses will be discarded. The field is optional for in-app traffic.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>geo</td>
<td>object</td>
<td>Geo Object as derived from the device’s location services or supplied by the SSP if device IP is missing.</td>
</tr>
<tr>
<td>ua*</td>
<td>string</td>
<td>Browser or application user agent string.</td>
</tr>
<tr>
<td>language*</td>
<td>string</td>
<td>Alpha-2/ISO 639-1 code of browser language.</td>
</tr>
<tr>
<td>carrier*</td>
<td>string</td>
<td>Carrier or ISP derived from the IP address.</td>
</tr>
<tr>
<td>connectiontype*</td>
<td>integer</td>
<td>Connection type as defined in OpenRTB.</td>
</tr>
<tr>
<td>didsha1*</td>
<td>string</td>
<td>SHA1 hashed device ID.</td>
</tr>
<tr>
<td>didmd5*</td>
<td>string</td>
<td>MD5 hashed device ID.</td>
</tr>
<tr>
<td>dpidsha1*</td>
<td>string</td>
<td>SHA1 hashed platform-specific ID.</td>
</tr>
<tr>
<td>dpidmd5*</td>
<td>string</td>
<td>MD5 hashed platform-specific ID.</td>
</tr>
<tr>
<td>ifa*</td>
<td>string</td>
<td>Native identifier for advertisers; an opaque ID assigned by the device or browser for use as an advertising identifier (e.g. Apple’s IFA, Android’s Advertising ID, etc).</td>
</tr>
<tr>
<td>make*</td>
<td>string</td>
<td>Device make (e.g., “Apple”).</td>
</tr>
<tr>
<td>model*</td>
<td>string</td>
<td>Device model (e.g., “iPhone”).</td>
</tr>
<tr>
<td>os*</td>
<td>string</td>
<td>Device operating system (e.g., “iOS”).</td>
</tr>
<tr>
<td>osv*</td>
<td>string</td>
<td>Device operating system version (e.g., “3.1.2”).</td>
</tr>
<tr>
<td>dnt*</td>
<td>integer</td>
<td>If “0”, then do not track Is set to false, if “1”, then do no track is set to true in browser.</td>
</tr>
<tr>
<td>devicetype*</td>
<td>integer</td>
<td>Device type as defined by OpenRTB.</td>
</tr>
<tr>
<td>ipv6*</td>
<td>string</td>
<td>IP address in IPv6.</td>
</tr>
<tr>
<td>js*</td>
<td>integer</td>
<td>“1” if the device supports JavaScript; else “0”.</td>
</tr>
<tr>
<td>flashver*</td>
<td>string</td>
<td>Flash version detected.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be sent in each request.

**Geo Object**
The information provided in Geo Object is based on MaxMind database, except lat and lon values.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>lat*</td>
<td>float</td>
<td>Latitude from -90 to 90. South is negative.</td>
</tr>
<tr>
<td>lon*</td>
<td>float</td>
<td>Longitude from -180 to 180. West is negative.</td>
</tr>
<tr>
<td>country*</td>
<td>string</td>
<td>Country using ISO-3166-1 Alpha-2.</td>
</tr>
<tr>
<td>region*</td>
<td>string</td>
<td>Region using ISO-3166-2 or FIPS region codes.</td>
</tr>
<tr>
<td>city*</td>
<td>string</td>
<td>City name as provided by MaxMind.</td>
</tr>
<tr>
<td>zip*</td>
<td>string</td>
<td>Zip/postal code.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be sent in each request.

### User Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id*</td>
<td>string</td>
<td>Unique customer ID of this user on the exchange. It may not be present if DSP opted for unknown users.</td>
</tr>
<tr>
<td>buyeruid*</td>
<td>string</td>
<td>DSP user ID for this user as mapped by Bid Switch for the DSP.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be sent in each request.

### Site Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id*</td>
<td>string</td>
<td>Site ID on the exchange prefixed with exchange name followed by underscore. E.g. if Rubicon SSP sends bid request for site ID “123”, the field becomes “rubicon_123”.</td>
</tr>
<tr>
<td>name*</td>
<td>string</td>
<td>Site name (may be masked by publisher request).</td>
</tr>
<tr>
<td>domain*</td>
<td>string</td>
<td>Domain of the site, used for advertiser side blocking. For example, “foo.com”.</td>
</tr>
</tbody>
</table>
### cat*  array of strings
Array of IAB content categories for the publisher site.

### page*  string
URL of the page where the impression will be shown.

### publisher  object
Publisher Object.

### ext*  object
Site Ext Object

* Fields marked with asterisk are optional and may not be sent in each request.

## Site Ext Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>mobile_site*</td>
<td>integer</td>
<td>A value of 1 in this field indicates the user is accessing a mobile-optimized site.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be sent in each request.

## App Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id*</td>
<td>string</td>
<td>Application ID on the exchange prefixed with exchange name followed by underscore. E.g. if Rubicon SSP sends bid request for site ID “123”, the field becomes “rubicon_123”.</td>
</tr>
<tr>
<td>name*</td>
<td>string</td>
<td>Application name.</td>
</tr>
<tr>
<td>domain*</td>
<td>string</td>
<td>Domain of the application, for example, “mygame.foo.com”.</td>
</tr>
<tr>
<td>cat*</td>
<td>array of strings</td>
<td>Array of IAB content categories for the application.</td>
</tr>
<tr>
<td>bundle*</td>
<td>string</td>
<td>Application bundle or package name (e.g., com.foo.mygame).</td>
</tr>
<tr>
<td>publisher</td>
<td>object</td>
<td>Publisher Object.</td>
</tr>
<tr>
<td>storeurl*</td>
<td>string</td>
<td>App store’s URL for the mobile application.</td>
</tr>
<tr>
<td>ver*</td>
<td>string</td>
<td>Application version.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be sent in each request.
### Private Marketplace Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>private_auction</td>
<td>integer</td>
<td>A value of 1 indicates that only bids submitted inside pmp.deals will take part in the auction. A value of 0 indicates that bids without deal information may also be considered for serving.</td>
</tr>
<tr>
<td>deals</td>
<td>array of objects</td>
<td>Array of Deal objects.</td>
</tr>
</tbody>
</table>

### Deal Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>string</td>
<td>Deal id.</td>
</tr>
<tr>
<td>wseat*</td>
<td>array of string</td>
<td>Array of buyer seats allowed to bid on this Direct Deal. Seats are an optional feature of an exchange. For example, [&quot;4&quot;, &quot;34&quot;, &quot;82&quot;, &quot;45&quot;] indicates that only advertisers using these exchange seats are allowed to bid on this direct deal.</td>
</tr>
<tr>
<td>bidfloor*</td>
<td>float</td>
<td>Deal price in CPM. If deal.at = 3 then this is the exact price of the deal, otherwise this is the bid floor of the deal. The currency is the same as specified in cur field.</td>
</tr>
<tr>
<td>at</td>
<td>integer</td>
<td>Auction type. ‘1’ for first price auction, ‘2’ for second price auction, ‘3’ for fixed price deal.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be sent in each request.

### Publisher Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>string</td>
<td>Publisher ID prefixed with SSP name followed by underscore, e.g. if Rubicon passes publisher ID 25, then the present field will be equal to “rubicon_25”. For Google AdX bid requests the Google seller network ID is used.</td>
</tr>
<tr>
<td>name*</td>
<td>string</td>
<td>Publisher name.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be sent in each request.
Ext Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ssp</td>
<td>string</td>
<td>SSP identification string. E.g., “rubicon” or “openx”.</td>
</tr>
<tr>
<td>is_secure*</td>
<td>integer</td>
<td>0 for non-secure pages; 1 for secure pages. Creatives for secure pages should be SSL-compliant.</td>
</tr>
<tr>
<td>google*</td>
<td>object</td>
<td>Google object. This field is optional and is used only for integrations with Google AdX.</td>
</tr>
<tr>
<td>adtruth*</td>
<td>object</td>
<td>Adtruth object. This field is optional and is present if the corresponding data is provided by the SSP.</td>
</tr>
<tr>
<td>ghostery*</td>
<td>object</td>
<td>Ghostery object. This field is optional and is present if the feature is enabled for the trading partner.</td>
</tr>
<tr>
<td>creative_params*</td>
<td>array of objects</td>
<td>SSP-specific parameters indicating which creatives are valid for bidding in correspondence with creative approval feedback data.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be sent in each request.

Google Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>detected_vertical*</td>
<td>array of objects</td>
<td>List of detected vertical values as defined by Google.</td>
</tr>
<tr>
<td>excluded_attribute*</td>
<td>array of integer</td>
<td>List of excluded creative attributes as defined by Google. DEPRECATED: use imp.ext.google object instead.</td>
</tr>
<tr>
<td>allowed_vendor_type*</td>
<td>array of integer</td>
<td>List of allowed vendor types as defined by Google. DEPRECATED: use imp.ext.google object instead.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be sent in each request.

Detected_vertical Object
<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>integer</td>
<td>The vertical id as defined by AdX docs.</td>
</tr>
<tr>
<td>weight</td>
<td>float</td>
<td>Weight for this vertical, in the (0.0, 1.0] range. More relevant verticals have higher weights.</td>
</tr>
</tbody>
</table>

### Adtruth Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>tdl_millis*</td>
<td>int64</td>
<td>Time Difference Linking, which allows for differentiation between similarly-configured devices that might share a DeviceInsight identifier.</td>
</tr>
<tr>
<td>RECIPE VERSION_x</td>
<td>string</td>
<td>The specific AdTruth device identification recipe version and the corresponding DeviceInsight identifier. There might be multiple AdTruth recipe versions. E.g. &quot;WEB_APP_BRIDGE_4_0&quot;:&quot;4FD87B97751E9C305FD5314A665D2AF1C8FAA2D3&quot;</td>
</tr>
</tbody>
</table>

### Ghostery Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>status</td>
<td>string</td>
<td>Indicates whether the bid request is associated with a single detected domain. Possible values are: “verified” : the publisher domain of the bid request is verified by Ghostery; “multi” : there are more than one domains associated with the bid request; “unverified” : the detected domains do not</td>
</tr>
</tbody>
</table>
match those provided in the bid request; “unknown”: there is not enough data available for the current bid request.

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>domain*</td>
<td>string</td>
<td>True publisher domain as detected by Ghostery.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be sent in each request.

### Creative Parameters Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>type</td>
<td>string</td>
<td>Possible values:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- “whitelist”: only creatives with at least one of the listed parameter values are allowed;</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- “blacklist”: creatives with at least one of the listed parameter values are not allowed</td>
</tr>
<tr>
<td>name</td>
<td>string</td>
<td>Parameter name as defined by the SSP</td>
</tr>
<tr>
<td>value</td>
<td>array of strings</td>
<td>Parameter values eligible for bid current request.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be sent in each request.

### Regulations Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>coppa</td>
<td>integer</td>
<td>Flag indicating whether or not this request falls under the COPPA regulations established by the USA FTC, where 0 = no, 1 = yes.</td>
</tr>
</tbody>
</table>
Examples

Simple Banner Ad

Following is a basic example of a bid request for a banner ad.

```json
{
    "id": "1234534625253",
    "imp": [{
        "id": "1",
        "banner": {
            "w": 300,
            "h": 250,
            "pos": 1,
            "topframe": 0,
            "mimes": ["text/html", "application/x-shockwave-flash"]
        },
        "ext": {
            "rubicon": {
                "site_size_session_count": 5
            }
        }
    }],
    "site": {
        "id": "234563",
        "domain": "siteabcd.com",
        "page": "http://siteabcd.com/page.htm",
        "publisher": {
            "id": "rubicon_25"
        },
        "ext": {
            "mobile_site": 1
        }
    },
    "device": {
        "ip": "64.124.253.1",
        "geo": {
            "lat": 33.94940185546875,
            "lon": -116.97750091552734,
            "county": "US",
            "region": "NY",
            "city": "White Plains",
            "zip": "10601"
        }
    }
}
```
Private Deal and Creative Parameters

Following is a basic example of bid request for private auction.

```json
{
    "id": "1234534625254",
    "imp": [
        {
            "id": "1",
            "bidfloor": 0.426,
            "banner": {
                "w": 300,
                "h": 250,
                "pos": 1,
                "topframe": 0,
            }
        }
    ]
}
```
"expdir": [1, 3],
"pmp": {
    "private_auction": 0,
    "deals": [
        {
            "id": "deal-1",
            "wseat": ["4", "15"],
            "bidfloor": 2.5,
            "at": 1
        },
        {
            "id": "deal-2",
            "bidfloor": 2.0
            "at": 2
        }
    ]
},
"site": {
    "id": "234563",
    "domain": "siteabcd.com",
    "page": "https://siteabcd.com/page.htm",
    "publisher": {
        "id": "rubicon_25"
    }
},
"device": {
    "ip": "64.124.253.1",
    "geo": {
        "county": "US",
        "region": "NY",
        "city": "White Plains",
        "zip": "10601"
    },
    "ua": "Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10.6; en-US; rv:1.9.2.16) Gecko/20110319 Firefox/3.6.16",
    "language": "en"
},
"user": {
    "id": "45asdf987656789adfad4678rew656789"
},
"cur": ["USD"],
"ext": ["USD"]
“ssp”: “rubicon”,
“creative_params”: [{
    “type”: “blacklist”,
    “name”: “ad_type”,
    “value”: [“14011”, “14006”]
}]
}
]

Video Ad

Following is a basic example of a bid request for a video ad.

{
    “id”: “1234534625253”,
    “imp”: [ {
        “id”: “1”,
        “video”: {
            “mimes”: [
                “video/x-flv”,
                “video/mp4”,
                “application/x-shockwave-flash”,
                “application/javascript”
            ],
            “linearity”: 1,
            “minduration”: 5,
            “maxduration”: 30,
            “protocol”: [2, 3, 5, 6],
            “w”: 640,
            “h”: 480,
            “startdelay”: 0,
            “battr”: [13,14],
            “minbitrate”: 300,
            “maxbitrate”: 1500,
            “api”: [1],
            “companionad”: [{
                “w”: 300,
                “h”: 250,
                “id”: 1
            }],
            “companionstype”: [1, 2, 3],
            “ext”: {

```
"skippable": 1
}
]
"site": {
  "id": "234563",
  "domain": "siteabcd.com",
  "page": "http://siteabcd.com/page.htm",
  "publisher": {
    "id": "google_25"
  }
},
"device": {
  "ip": "64.124.253.1",
  "geo": {
    "county": "US",
    "region": "NY",
    "city": "White Plains",
    "zip": "10601"
  },
  "ua": "Mozilla/5.0 (Macintosh; U; Intel Mac OS X 10.6; en-US; rv:1.9.2.16) Gecko/20110319 Firefox/3.6.16",
  "language": "en"
},
"user": {
  "id": "45asdf987656789adfad4678rew656789",
  "buyeruid": "1234567890"
},
"cur": ["USD"],
"ext": {
  "ssp": "google"
}
}

In-App Ad

Following is an example of a bid request for a mobile application banner ad.

{"app": {
  "bundle": "302324249",
  "cat": [}
"IAB14",
"IAB24"
],
"id": "mopub_af8f72a9eaae43d6b1330cada08f86f6",
"publisher": {
    "id": "mopub_agltb3B1Yi1pbmNyEAsSB0FjY291bnQY7cCnEgw"
},
"ver": "4.2"
},
"at": 2,
"badv": [
    "badsite.com"
],
"bcat": [
    "IAB7-39",
    "IAB8-18",
    "IAB14-1",
    "IAB25",
    "IAB26",
    "BSW3"
],
"cur": [
    "USD"
],
"device": {
    "carrier": "234-30",
    "connectiontype": 3,
    "dpidmd5": "c43fba88be23abcc348227dd1d7c044a",
    "dpidsha1": "a17b984e3431e967230c1fc27cb3fa4029bcf4de",
    "geo": {
        "city": "",
        "country": "GB",
        "region": "",
        "zip": ""
    },
    "ifa": "3F6F224C-5F05-4269-B0A3-81637B860A2A",
    "ip": "31.121.161.211",
    "language": "en",
    "make": "samsung",
    "model": "GT-I9505",
    "os": "Android",
    "osv": "4.4.2",
    "ua": "Mozilla/5.0 (Linux; Android 4.4.2; GT-I9505 Build/KOT49H) AppleWebKit/537.36 (KHTML, like Gecko) Version/4.0 Chrome/30.0.0.0 Mobile Safari/537.36"
"ext": {
    "ssp": "mopub"
},
"id": "bc9b529e-511a-41c2-a768-dcb139ad4a6a",
"imp": [{
    "battr": [
      9,
      10,
      14,
      6
    ],
    "btype": [
      4
    ],
    "bidfloor": 0.70,
    "id": "1"
}]
"user": {
  },
"wseat": [
    "18"
]}

Bid Response

Top Level Bid Response Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>string</td>
<td>ID of the bid request.</td>
</tr>
<tr>
<td>seatbid</td>
<td>array of objects</td>
<td>Array of Seat Bid objects. The length of the array can be either 1 (for yes-bid) and 0 (for no-bid).</td>
</tr>
<tr>
<td>cur*</td>
<td>string</td>
<td>Bid currency using ISO-4217 alphabetic codes. If not</td>
</tr>
</tbody>
</table>
* Fields marked with asterisk are optional and may not be present in each response.

**Seat Bid Object**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>bid</td>
<td>array of objects</td>
<td>Array of Bid Objects. The only reason the length of this object can be greater than 1 is that there are several slots in the bid request.</td>
</tr>
<tr>
<td>seat*</td>
<td>string</td>
<td>ID of the bidder seat on whose behalf this bid is made. The field is REQUIRED if the wseat field is present in the bid request.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be present in each response.

Formally there can be multiple bids within Seat Bid for two reasons. The first one is that there can be several slots in request. The second one is when there are more than one bid for a single slot. The BidSwitch allows no more than two bids for a single ad slot.

**Bid Object**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>id</td>
<td>string</td>
<td>ID for the bid object chosen by the bidder for tracking and debugging purposes.</td>
</tr>
<tr>
<td>impid</td>
<td>string</td>
<td>ID of the impression object from bid request to which this bid applies.</td>
</tr>
<tr>
<td>price</td>
<td>float</td>
<td>Price as a float value, CPM. All prices assumed to be in USD, if cur parameter is omitted.</td>
</tr>
<tr>
<td>nurl*</td>
<td>string</td>
<td>Win notice URL. For banner impressions the ad markup should be returned via this URL. Ad markup should be in HTML format. For video impressions the url should contain the url of impression pixel. The url may contain substitution macros, see Macros section below. For application banner inventory this field will be IGNORED, use ext.js_url instead.</td>
</tr>
<tr>
<td>adid</td>
<td>string</td>
<td>ID that references the ad to be served if the bid wins.</td>
</tr>
<tr>
<td>Field</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>adomain</td>
<td>array of strings</td>
<td>Advertiser’s primary or top-level domain for advertiser checking. This can be a list of domains if there is a rotating creative. Note that some SSPs allow only one domain. To such SSPs we only send the first domain from the list.</td>
</tr>
<tr>
<td>cid*</td>
<td>string</td>
<td>Campaign ID or similar that is used by the bidder to track and organize their campaigns. The field is REQUIRED for some SSPs, including Nexage and MoPub.</td>
</tr>
<tr>
<td>ext*</td>
<td>object</td>
<td>Ext Object. This field is optional.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be present in each response.

**Ext Object**

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>deal*</td>
<td>string</td>
<td>This is the ID of the deal between a publisher and a seat. It is used only if an exchange supports private auctions. If this field is present it should be equal to one of the elements in pmp.deals field in the bid request object.</td>
</tr>
<tr>
<td>advertiser_name*</td>
<td>string</td>
<td>The name of the advertiser serving the creative. Required in bids responses to AdScale, Centro, BRX and YieldOne bids.</td>
</tr>
<tr>
<td>agency_name*</td>
<td>string</td>
<td>The name of the agency representing the advertiser. Required in bids responses to AdScale and Centro bids.</td>
</tr>
<tr>
<td>cat*</td>
<td>string</td>
<td>IAB category of the creative. Required in bid responses to RMX and YieldOne bids.</td>
</tr>
<tr>
<td>liveintent*</td>
<td>Object</td>
<td>Required field for bid responses to LiveIntent bid requests. DEPRECATED: use seatbid.bid.ext.img_url and seatbid.bid.ext.click_url instead.</td>
</tr>
<tr>
<td>google*</td>
<td>Object</td>
<td>Contains additional information for Google bids. The field is recommended.</td>
</tr>
<tr>
<td>yieldone*</td>
<td>Object</td>
<td>Contains additional information for YieldOne bids. The field is required for YieldOne bids.</td>
</tr>
</tbody>
</table>
### SSP Specific Fields

#### LiveIntent Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>img_url</td>
<td>string</td>
<td>URL of the creative image. In order to receive user cookie and win price this url should point to the DSP handler and be later redirected to the actual creative location. The url may contain macros, e.g. <code>${AUCTION_PRICE}</code>. If this field is present, the nurl field of the response will be ignored.</td>
</tr>
</tbody>
</table>
click_url | string | Creative click URL.

Google Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>attribute*</td>
<td>array of integer</td>
<td>List of creative attributes as defined by Google.</td>
</tr>
<tr>
<td>vendor_type*</td>
<td>array of integer</td>
<td>List of creative vendor types as defined by Google.</td>
</tr>
</tbody>
</table>

* Fields marked with asterisk are optional and may not be present in each response.

Yieldone Object

<table>
<thead>
<tr>
<th>Field</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>creative_type</td>
<td>integer</td>
<td>Creative type as defined by YieldOne.</td>
</tr>
</tbody>
</table>

Nobid Response

If the DSP opts to pass on the bid request, the seatbid field in the bid response should contain an empty array of bids. The BidSwitch also accepts HTTP 204 responses as valid nobid responses.

Examples

Simple Bid Response

Following is an example of a bid response with the ad served on win notice. The bid for this impression is a $9.43 CPM.

```json
{
    "id": "1234567890",
    "seatbid": [
        {
            "bid": [
                {
                    "id": "1",
                    "price": 9.43
                }
            ]
        }
    ]
}
```
"impid" : "102",
"price" : 9.43,
"adid" : "314",
"cid" : "42",
"nurl" : "http://adserver.com/winnote?impid=102&winprice=${AUCTION_PRICE}&click_url=${CLICK_URL:URLENCODE}",
"adomain" : ['advertiserdomain.com']
"ext" : {
   "liveintent" : {
      "click_url" : "http://adserver.com/click?impid=102"
   },
   "advertiser_name" : "Coca-Cola",
   "language" : "en",
   "cat" : "IAB12",
   "agency_name" : "CC-advertising"
}
}]
}]
}]

Image-only Bid Response

Following is an example of a bid response with the ad served on win notice. The bid for this impression is a $9.43 CPM.

{
   "id" : "1234567890",
   "seatbid" : [
      "bid" : [
         "id" : "1",
         "impid" : "102",
         "price" : 9.43,
         "adid" : "314",
         "cid" : "42",
         "img_url" : "http://adserver.com/imp?impid=102&winprice=${AUCTION_PRICE}&click_url=${CLICK_URL:URLENCODE}"
      ],
      "adomain" : ['advertiserdomain.com']
   ]
}
In-App Bid Example

```
{  
  "cur" : "USD",  
  "id" : "e9c3e120-ffcb-4300-9c98-644cb26f95df",  
  "seatbid": [{  
      "bid": [  
        "adid": "3",  
        "adomain": ["nokia.com"],  
        "cid": "11",  
        "ext": {  
          "js_url": "http://adserver.com/winnotice?impid=102&winprice=\${AUCTION_PRICE}&click_url=\${CLICK_URL:URLENCODE}"  
        },  
        "id": "1c3ff810-3623-4b04-8396-9e7ca071cb72",  
        "impid": "1",  
        "price": 4.079077199308326  
      ]  
    }]  
}
```

Video Bid Example

```
{  
  "cur" : "USD",  
  "id" : "e9c3e120-ffcb-4300-9c98-644cb26f95df",  
  "seatbid": [{  
      "bid": [  
        "adid": "3",  
        "nurl": "http://adserver.com/winnotice?impid=102&winprice=\${AUCTION_PRICE}"  
      ]  
    }]  
}
```
No Bid Example

Following is an example of sending no-bid response to the bid.

{  
    "id": "1234567891",
    "seatbid": []
}

Response to Private Bid Request

{  
    "id": "1234567892",
    "cur": "USD",
    "seatbid": [{  
        "bid": [{  
            "id": "1114125-afaff2f-af2251",
            "impid": "1",
            "price": 9.43,
            "adid": "314",
            "nurl": "http://adserver.com/winnotice?impid=102&winprice=${AUCTION_PRICE}&click_url=${CLICK_URL:urlencode}"
        }, {  
            "id": "1114125-afaff2f-af2251",
            "impid": "1",
            "price": 9.43,
            "adid": "314",
            "nurl": "http://adserver.com/winnotice?impid=102&winprice=${AUCTION_PRICE}&click_url=${CLICK_URL:urlencode}"
        }]
    }]}
Macros

Win price macro may be used in nurl, ext.img_url and ext.js_url fields. Click tracking macro may be used in nurl and ext.js_url fields.

Win price macro

In order for the exchange to convey certain information to the winning bidder (e.g., the settlement price), some substitution macros can be inserted into the win notice URL definition. Prior to calling a win notice URL, the exchange will search the specified URL for any of the defined macros and replace them with the appropriate data. Note that the substitution is simple in the sense that wherever a legal macro is found, it will be replaced without regard for syntax correctness. Furthermore, if the source value is an optional parameter that was not specified, the macro will simply be removed (i.e., replaced with a zero-length string).

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>${AUCTION_PRICE}</td>
<td>Settlement price for the auction. The substituted value will be defined as CPM.</td>
</tr>
</tbody>
</table>

SSP click tracking url macro

Some SSPs require the DSP to add an SSP click url macro in all creatives for click reporting. This is a strict requirement for Google AdX and Pubmatic and is strongly recommended by several other SSPs. DSPs wishing to trade with such SSPs should add the click tracking macro to the win notice URL definition. The BidSwitch will replace the macro with the SSP click tracking url. If the SSP doesn’t support click tracking macros then the tracking macro will be replaced by an empty string.

<table>
<thead>
<tr>
<th>Macros</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>${CLICK_URL}</td>
<td>A placeholder for the SSP click tracking url. Required for bids to SSPs that support click tracking.</td>
</tr>
<tr>
<td>${CLICK_URL:URLENCODE}</td>
<td>SSP click tracking url in urlencoded form.</td>
</tr>
</tbody>
</table>
The SSP click url (if present) should be inserted before the landing page in the creative. The landing page url should be single-escaped.

BidSwitch Categories

For the better coverage of sensitive categories BidSwitch extends the standard IAB list with additional categories. These categories may be used in bcat, site.cat, app.cat fields in bid requests and seatbid.bid.cat field in bid response.

<table>
<thead>
<tr>
<th>ID</th>
<th>Creative Category</th>
</tr>
</thead>
<tbody>
<tr>
<td>BSW1</td>
<td>Alcohol</td>
</tr>
<tr>
<td>BSW2</td>
<td>Gambling</td>
</tr>
<tr>
<td>BSW3</td>
<td>Tobacco and smoking</td>
</tr>
<tr>
<td>BSW4</td>
<td>Firearms and weapons</td>
</tr>
<tr>
<td>BSW5</td>
<td>Sexual &amp; Reproductive Health</td>
</tr>
<tr>
<td>BSW6</td>
<td>Ringtones &amp; Downloadable</td>
</tr>
<tr>
<td>BSW7</td>
<td>Drugs &amp; Supplements</td>
</tr>
<tr>
<td>BSW8</td>
<td>Get Rich Quick</td>
</tr>
<tr>
<td>BSW9</td>
<td>Free Gifts, Quizzes, &amp; Surveys</td>
</tr>
<tr>
<td>BSW10</td>
<td>Nudity</td>
</tr>
<tr>
<td>BSW11</td>
<td>Cosmetic Procedures &amp; Body Modification</td>
</tr>
</tbody>
</table>

Creative Preview

To accomplish the integration with Bid Switch, DSP should be able to provide an offline URL or other system to translate from DSP’s creative_id to a web page for preview purposes. The example of this URL may look like:

http://example_dsp.com/preview_page?creative_id=1234567&other_info=foo

where one creative corresponds to one creative_id. This URL will only be accessed by SSPs and should not have any login authentication.

Creative Approval

BidSwitch provides automatic creative approval support for some of the integrated SSPs. The approval status for each creative is stored on BidSwitch side and is available for downloading via BidSwitch UI in XML format. The file lists all recorded creative IDs, their status at the integrated SSPs and possibly a list of SSP-dependent parameter that may limit usage of this
creative at the target SSP under certain circumstances.

Possible create status values are:
“0”: Creative is not approved;
“1”: Creative is approved with some additional parameters (see below);
“2”: Creative is approved.

Creative Approval Parameters

SSP may list some conditions under which the creative may or may not be used in bidding. E.g. this may be a list of placements on which the creative is allowed. Parameter types are SSP-dependent.
BidSwitch lists the creative parameters in bid requests (where available), the parameter types match those provided in the creative status XML file. It’s recommended that these parameters are used for selecting valid creatives for the bid response.

Sample Creative Status XML

```xml
<creatives>
  <creative id="7177">
    <ssp name="rmx" status="0" />
    <ssp name="yieldone" status="1">
      <parameters>
        <parameter name="placement">
          <value>100</value>
          <value>103</value>
        </parameter>
      </parameters>
    </ssp>
  </creative>
  <creative id="276546">
    <ssp name="rmx" status="2" />
    <ssp name="yieldone" status="0" />
  </creative>
</creatives>
```

Additional Features

Bid Switch comes with a bunch of additional features that can be enabled as per client request. The first group of features is related to bid targeting and can be used to filter out undesired traffic and to keep bidding only on those requests the client is mostly interested in.
QPS Management
Bid Switch allows clients to limit the number of bid requests to a certain level. QPS limits can be applied on per datacenter basis. Please contact the Bid Switch support team for more information on this topic.

Bid Targeting
A comprehensive user targeting system has been one of the core parts of IPONWEB platform for many years. Now it is included into Bid Switch service and allows precise user targeting based on a variety of criteria including geolocation parameters, language preferences, device information, publisher and domain data, date and time targeting and so on. The sections below provide a brief description of targeting system capabilities that can be applied globally for a client.

<table>
<thead>
<tr>
<th>Targeting Type</th>
<th>Targeting Options</th>
</tr>
</thead>
<tbody>
<tr>
<td>Geo Targeting</td>
<td>country, region, city</td>
</tr>
<tr>
<td>Publisher Side Targeting</td>
<td>publisher, domain, page, size</td>
</tr>
<tr>
<td>Operation System</td>
<td>type, version</td>
</tr>
<tr>
<td>Browser</td>
<td>type, version</td>
</tr>
<tr>
<td>Device</td>
<td>type, maker, model</td>
</tr>
</tbody>
</table>

User Sync Targeting
Just like targeting to a specific audiences while bidding there is an option to selectively sync users between exchanges. User sync targeting allows a client to focus on a particular country filtering out all other users for matching. This feature can also be combined with QPS management option thus making it possible to match users from a certain country limited to a desired QPS rate.