

Publisher Verification

Implementation Guide

Last updated: 9/2020

Table of Contents

IAS Publisher Verification Pixel Implementation Overview	3
Publisher Verification Pixel - Google Ad Manager	3
Google Ad Manager API Integration for Reporting	6
Publisher Google Ad Exchange	8
Publisher Verification Pixel - AppNexus Open AdStream (OAS)	8
Third Party Pixel Implementation	10
Piggyback Pixel Implementation	11
Publisher Verification Pixel - Smart AdServer	13
Publisher Verification Pixel - Other Ad Servers	14
ID Attribution Specifications	15
Native Ad Implementation Overview	17
Step 1 - Use a unique CSS ID selector with the Placement.	17
Step 2 - Append the ias_adpath parameter to IAS's pixel.	17
Using the Native Ad Solution with HTML5 Ads	18

IAS Publisher Verification Pixel Implementation Overview

The Integral Ad Science (IAS) Publisher Verification solution uses a monitoring pixel to interact with the browser at the time of ad delivery. The pixel collects verification and viewability signals, which feed into rating models powered by IAS's technology. These models power the viewability, invalid traffic detection, and brand safety metrics available in the Integral Platform.

IAS's Publisher Verification solution works across desktop and mobile web display, mobile app display, and desktop video.

Publisher Verification Pixel - Google Ad Manager

IAS's Publisher Verification solution natively integrates with Ad Manager. The Publisher Verification Pixel is served via a JavaScript tag that loads asynchronously to avoid blocking either the loading of the publisher's content or the ad creative.

IAS uses several Ad Manager macros to automatically pull information such as the ad unit, line item, and order that's being served for each ad so that information is available in IAS reports. IAS's Publisher Verification pixel works in all Ad Manager tag types: Google Publisher Tags (GPT) or legacy DART, synchronous or asynchronous, JavaScript or iframe.

Specification for display tags (desktop web, mobile web, and mobile app):

```
<script id='ias-%%CACHEBUSTER%%'>
(function() {
    var iasScriptUrl, hiddenFrame, hiddenDoc, where, domain;
    iasScriptUrl = '//pixel.adsafeprotected.com/jload?anId=[Client
ID]&campId=%%WIDTH%%x%%HEIGHT%%&pubId=%eadv!&chanId=%epid!&placementId=%eaid!
&pubCreative=%ecid!&pubOrder=%ebuy!&custom=[OPTIONAL]&custom2=
[OPTIONAL]&custom3=[OPTIONAL]';
    hiddenFrame = document.createElement('iframe');
    (hiddenFrame.frameElement || hiddenFrame).style.cssText
    = "width: 0; height: 0; border: 0; display: none;";
    hiddenFrame.src = 'javascript:false';
    where = document.getElementById('ias-%%CACHEBUSTER%%');
    where.parentNode.insertBefore(hiddenFrame, where);
    try {
        hiddenDoc = hiddenFrame.contentWindow.document;
    } catch (e) {
        domain = document.domain;
        hiddenFrame.src="javascript:var d=document.open();d.domain='"+ domain
+ "';void(0);";
        hiddenDoc = hiddenFrame.contentWindow.document;
```

```

    }
    hiddenDoc.open().write('<body onload="' +
    'window.__IntegralASUseFIF = true;' +
    'var js = document.createElement(\'script\');'+
    'js.src = \'' + iasScriptUrl + '\';'+
    'document.body.appendChild(js);">');
    hiddenDoc.close();
  })();
</script>

```

Specification for video wrapper (desktop and mobile web):

[https://vast.adsafeprotected.com/vast?anId=\[Client ID\]&pubId=%adv!&chanId=%epid!&placementId=%eaid!&pubCreative=%ecid!&pubOrder=%ebuy!&cb=%CACHEBUSTER%&custom=\[OPTIONAL\]&custom2==\[OPTIONAL\]&custom3==\[OPTIONAL\]&originalVast=INSERT_VAST_URL](https://vast.adsafeprotected.com/vast?anId=[Client ID]&pubId=%adv!&chanId=%epid!&placementId=%eaid!&pubCreative=%ecid!&pubOrder=%ebuy!&cb=%CACHEBUSTER%&custom=[OPTIONAL]&custom2==[OPTIONAL]&custom3==[OPTIONAL]&originalVast=INSERT_VAST_URL)

anId	Client ID assigned and provided by IAS Professional Services during implementation.
campId	The size of the trafficked creative in the ad server (for example, "728x90" or "300x250").
pubId	An identifier for the advertiser that purchased the current ad.
chanId	An identifier for the served ad unit.
placementId	An identifier for the line item / flight for the current ad.
pubCreative	An identifier for the creative served for the current ad.
pubOrder	An identifier for the order / campaign for the current ad.
custom / custom2 / custom3	Optional key/value pairs to track in IAS reports; these are often replaced with Ad Manager pattern macros such as %%PATTERN:pos%%, %%PATTERN:trafficSource%%, etc. Omit the parameter if it is not used. Use these for things that have <= 200 unique values, so no IP addresses, user Ids, zip codes, etc.
originalVast	URL pointing to the advertiser's VAST resource.

Note: Mobile app video requires AVID and OMID nodes in the tag and doesn't require the video wrapper. For example:

```

<Extensions>
  <Extension type="AdVerifications">
    <AdVerifications>
      <Verification vendor="company.com-omid">
        <JavaScriptResource apiFramework="omid" browserOptional="true"><![CDATA[
https://company.com/omid.js ]]></JavaScriptResource>
      </Verification>
    </AdVerifications>
  </Extension>
</Extensions>

```

Verification Configuration Field	Description	IAS Specific Notes
VendorKey	The key that the vendor acquires from the IAB. The vendor provides this during configuration.	integralads.com
JavaScriptResource	URI Location of the vendor's OM SDK script (should be prefixed with protocol).	Add the IAS Campaign Monitoring (CM) display pixel.
VerificationParameters / URL Parameters	A free text string of data that needs to be passed to the verification node in the OM SDK. The structure of the string is specific to the vendor's implementation (some could be ordinal based, while others are key-value based).	Ignored/not required.
Rejection tracker URL	The URL that should be pinged if the verification script cannot be run.	Ignored/not required

Note: For mobile app, you can assign a viewability provider to an Ad Manager line item. See the [Ad Manager help website](#) for implementation instructions.

Inside Ad Manager, the IAS Publisher Verification Pixel is trafficked either by:

- Using [creative wrappers](#) to automatically include the pixel on all or a subset of inventory.
- Pasting the pixel at the bottom of a [third party](#) or [custom creative](#) that's already trafficked.

Note: Google Ad Manager Small Business does not support creative wrappers at this time. Publishers using Ad Manager Small Business can either paste the IAS Publisher Verification Pixel at the bottom of each creative manually, or set up a [creative template](#) that includes the IAS Publisher Verification Pixel. An example creative template for an image:

```
<a href="%%CLICK_URL_UNESC%%"DEST_URL%%">
  
</a>
<script src="//pixel.adsafeprotected.com/jload?anId=[Client
Id]&campId=%%WIDTH%%x%%HEIGHT%%&pubId=%eadv!&chanId=%epid!&placementId=%eaid!
&pubCreative=%ecid!&pubOrder=%ebuy!">
</script>
```

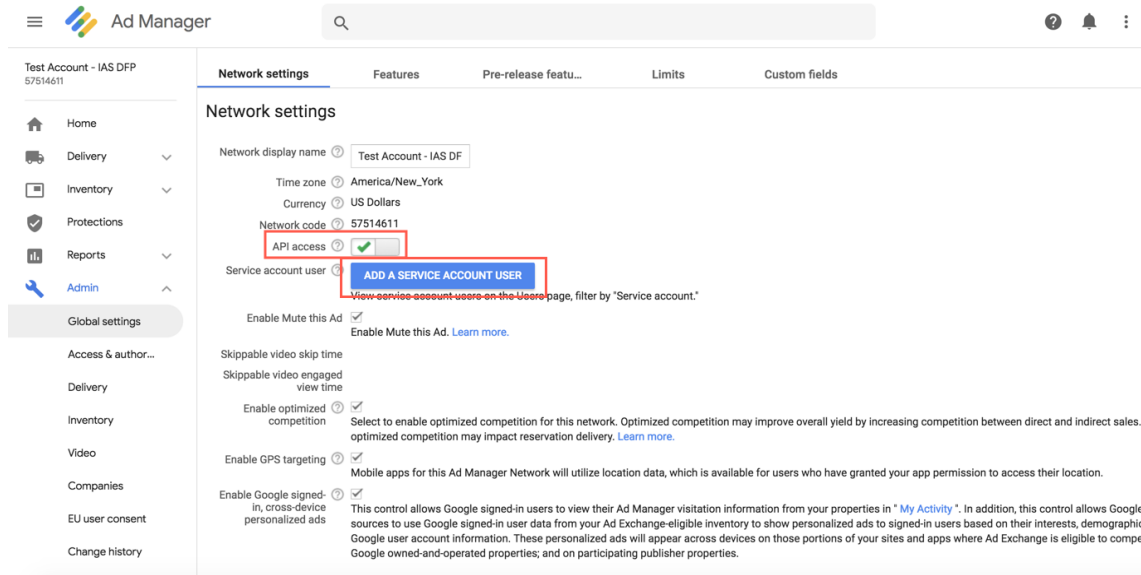
Note: Do not apply more than one creative wrapper with the same wrapping order (other than "No preference") to a given ad unit. This can create conflicts between

creative wrappers applied to your whole network and those applied to individual ad units, or between creative wrappers applied to parent and child ad units.

Google Ad Manager API Integration for Reporting

IAS supports connecting to your Ad Manager account to pull in the names of your line items, ad units, and advertisers into IAS reports. To give IAS access to your Ad Manager instance:

1. Ensure that the "API access" switch is **on** in the Ad Manager Admin section.
2. Click the "[Add a service account user](#)" button.



3. On the "New service account user" screen, enter the following information:

New service account user

Name	<input type="text" value="Integral Service Account"/>
Email [?]	<input type="text" value="719622553906-armvjd283ahnotp0sajm19c"/>
Confirm email	<input type="text" value="719622553906-armvjd283ahnotp0sajm19c"/>
User's language	<input type="text" value="English (US)"/>
Role [?]	<input type="text" value="Executive"/>

Executives can only access the reports tab, where they can run reports on orders.

Name	Integral Service Account	This name will show up on the "Users" page in Ad Manager.
Email	719622553906-armvjd283ahnotp0sajm19dfafus9dd2@developer.gserviceaccount.com	Use this email address.
Role	Executive or Custom:	<p>If you are using the default, built-in Ad Manager roles, "Executive" is the most limited role applicable to IAS.</p> <p>If you have custom roles, IAS requires the following permissions:</p> <ol style="list-style-type: none"> 1. View companie

		<p>s and contacts</p> <ol style="list-style-type: none"> 2. View ad units, placements, and custom targeting 3. View all orders and line items
Teams	All Entities	<p>If your Ad Manager account is configured to use Teams, you should select "All Entities" as the team for the Integral Service Account.</p>

Note: Provide your Ad Manager network code (for example, "1234") to your IAS representative.

Publisher Google Ad Exchange

IAS can measure ads served by Google Ad Exchange on a publisher page. In order to set this up:

1. Replace `anId` in the script below with your specific `anId` as a number.
2. Add the script in the `<head>` of your parent page (it cannot run from inside an ad), as high up above other scripts as possible so it loads before ads, minimizing discrepancies.

```
<script src="https://static.adsafeprotected.com/iasADX.js" onload="window.__iasADX.start(anId)"></script>
```

Note: This is currently in closed beta. If you are interested, reach out to your IAS representative.

Publisher Verification Pixel - AppNexus Open AdStream (OAS)

IAS's Publisher Verification Pixel integrates natively into Open AdStream with the following tag.

Specification:

```
<script id='ias-%%RAND%%'>
  (function() {
    var iasScriptUrl, hiddenFrame, hiddenDoc, where, domain, page;
    page = '%PAGE%';
    while (page.length > 50) page = page.substring(0,page.lastIndexOf('/'));
    iasScriptUrl = '//pixel.adsafeprotected.com/jload?anId=
[ClientId]&campId=%%WIDTH%
%%x%%HEIGHT%%&pubId=%%ADV%%&placementId=%%CAMP%%&pubCreative=%%ADID%%&cb=%%RAN
D%%&chanId=' + page;
    hiddenFrame = document.createElement('iframe');
    (hiddenFrame.frameElement || hiddenFrame).style.cssText = "width: 0;
height: 0; border: 0; display: none;";
    hiddenFrame.src = 'javascript:false';
    where = document.getElementById('ias-%%RAND%%');
    where.parentNode.insertBefore(hiddenFrame, where);
    try {
      hiddenDoc = hiddenFrame.contentWindow.document
    } catch (e) {
      domain = document.domain;
      hiddenFrame.src = "javascript:var d=document.open();d.domain='" + domain
+ "';void(0);";
      hiddenDoc = hiddenFrame.contentWindow.document
    }
    hiddenDoc.open().write('<body onload="' + 'window.__IntegralASUseFIF =
true;' + 'var js = document.createElement(\'script\');' + 'js.src = \'' +
iasScriptUrl + '\';' + 'document.body.appendChild(js);">');
    hiddenDoc.close()
  })();
</script>
```

Where:

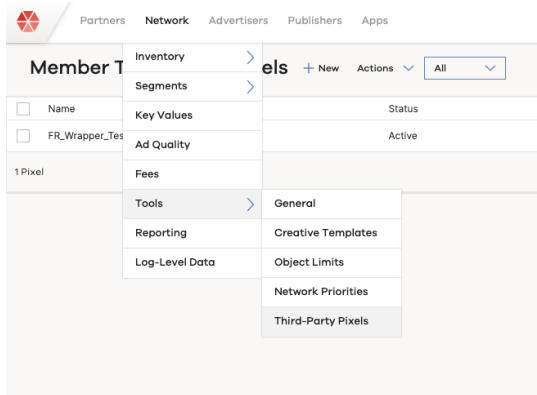
anId	Client ID assigned and provided by IAS Professional Services during implementation.
campId	The size of the trafficked creative in the ad server (for example, "728x90" or "300x250").
pubId	An identifier for the advertiser that purchased the current ad.
chanId	An identifier for the served ad unit.
placementId	An identifier for the line item / flight for the current ad.

There are two ways to implement this tag, through a third party pixel or a piggyback pixel.

Third Party Pixel Implementation

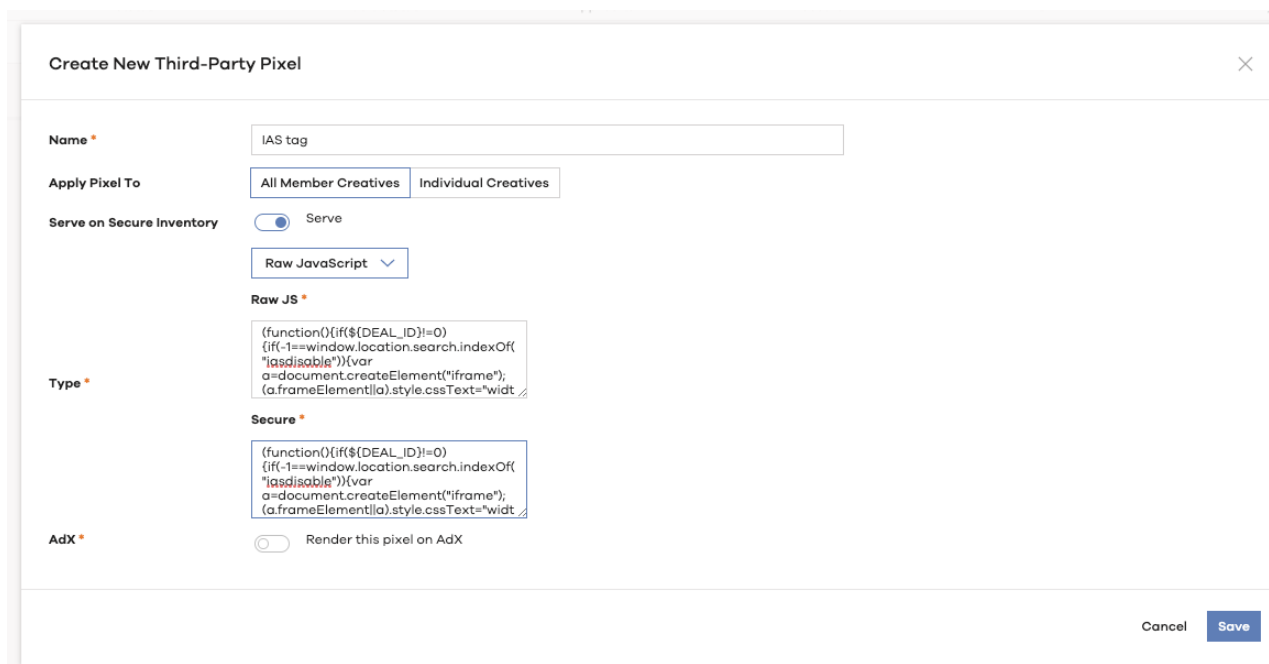
This method is only for a managed campaign. Third party pixel inventory in Open AdStream is similar to creative wrappers in Ad Manager.

1. Minify the tag within the 1000 character limit.
2. In the top menu bar, click "Network" > "Tools" > "Third-Party Pixels". The "Create New Third-Party Pixel" window appears.



3. Fill in the fields and then click the "Save" button.

Note: In the dropdown menu, you must select the Raw JavaScript for the custom tag and JavaScript logic.



The screenshot shows the 'Create New Third-Party Pixel' window. It has a close button (X) in the top right corner. The form contains the following fields and options:

- Name ***: A text input field containing 'IAS tag'.
- Apply Pixel To**: Two radio buttons, 'All Member Creatives' (selected) and 'Individual Creatives'.
- Serve on Secure Inventory**: A toggle switch labeled 'Serve'.
- Type ***: A dropdown menu showing 'Raw JavaScript'.
- Raw JS ***: A text area containing the following JavaScript code:


```
(function(){if(!DEAL_ID){if(-1==window.location.search.indexOf("iasdisable")){var a=document.createElement("iframe");(a.frameElement||a).style.cssText="width:100%;height:100%;border:none;"}else{a=document.createElement("script");(a.frameElement||a).style.cssText="width:100%;height:100%;border:none;"}a.src="https://www.iasads.com/ads/iasdisable.js";a.onload=function(){a.remove();}}})
```
- Secure ***: A text area containing the following JavaScript code:


```
(function(){if(!DEAL_ID){if(-1==window.location.search.indexOf("iasdisable")){var a=document.createElement("iframe");(a.frameElement||a).style.cssText="width:100%;height:100%;border:none;"}else{a=document.createElement("script");(a.frameElement||a).style.cssText="width:100%;height:100%;border:none;"}a.src="https://www.iasads.com/ads/iasdisable.js";a.onload=function(){a.remove();}}})
```
- AdX ***: A toggle switch labeled 'Render this pixel on AdX'.

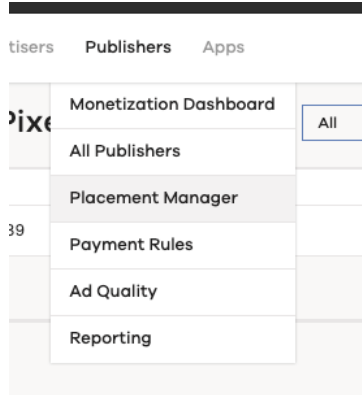
At the bottom right, there are 'Cancel' and 'Save' buttons.

Piggyback Pixel Implementation

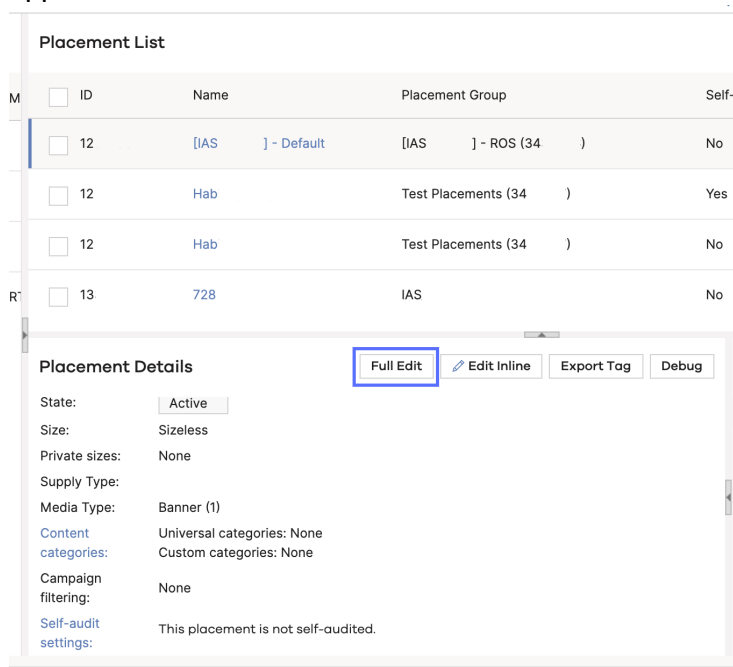
Open AdStream can work on an API to plug the iasScriptUrl on all placements. Piggyback pixels in Open AdStream are similar to creative templates in Ad Manager.

Note: You cannot add JavaScript logic since it is a URL and not a script.

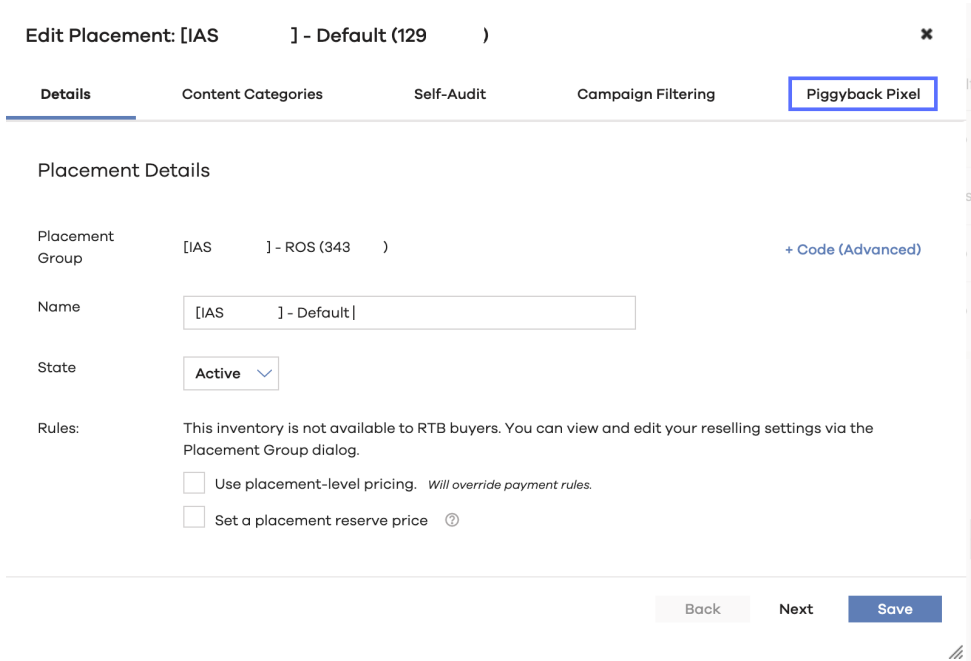
1. In the top menu bar, click "Publishers" > "Placement Manager". The "Placement List" window appears.



2. Select a placement and then click the "Full Edit" button. The "Edit Placement" window appears.



3. Select "Piggyback Pixel" on the top-right.



Edit Placement: [IAS] - Default (129)

Details Content Categories Self-Audit Campaign Filtering **Piggyback Pixel**

Placement Details

Placement Group: [IAS] - ROS (343) [+ Code \(Advanced\)](#)

Name: [IAS] - Default

State: Active

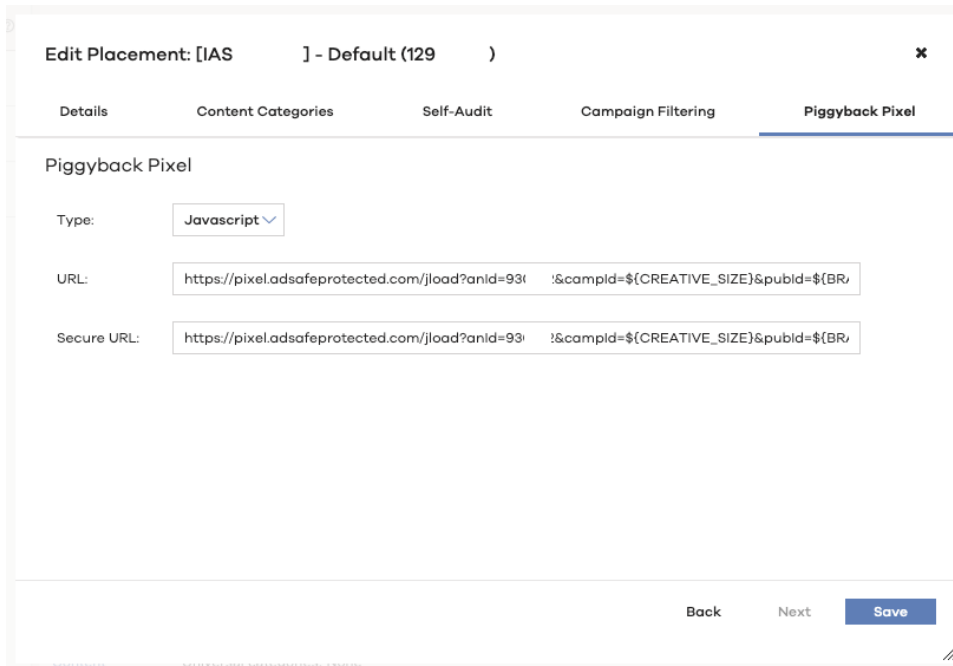
Rules: This inventory is not available to RTB buyers. You can view and edit your reselling settings via the Placement Group dialog.

☐ Use placement-level pricing. Will override payment rules.

☐ Set a placement reserve price ?

Back Next **Save**

4. In the "Type" dropdown menu, select **Javascript**. Fill in the fields and then click the "Save" button.



Edit Placement: [IAS] - Default (129)

Details Content Categories Self-Audit Campaign Filtering **Piggyback Pixel**

Piggyback Pixel

Type: Javascript

URL: https://pixel.adsafeprotected.com/jload?anId=93!&campId=\${CREATIVE_SIZE}&pubId=\${BR}

Secure URL: https://pixel.adsafeprotected.com/jload?anId=93!&campId=\${CREATIVE_SIZE}&pubId=\${BR}

Back Next **Save**

Publisher Verification Pixel - Smart AdServer

IAS's Publisher Verification Pixel integrates natively into the Smart AdServer with the following tag:

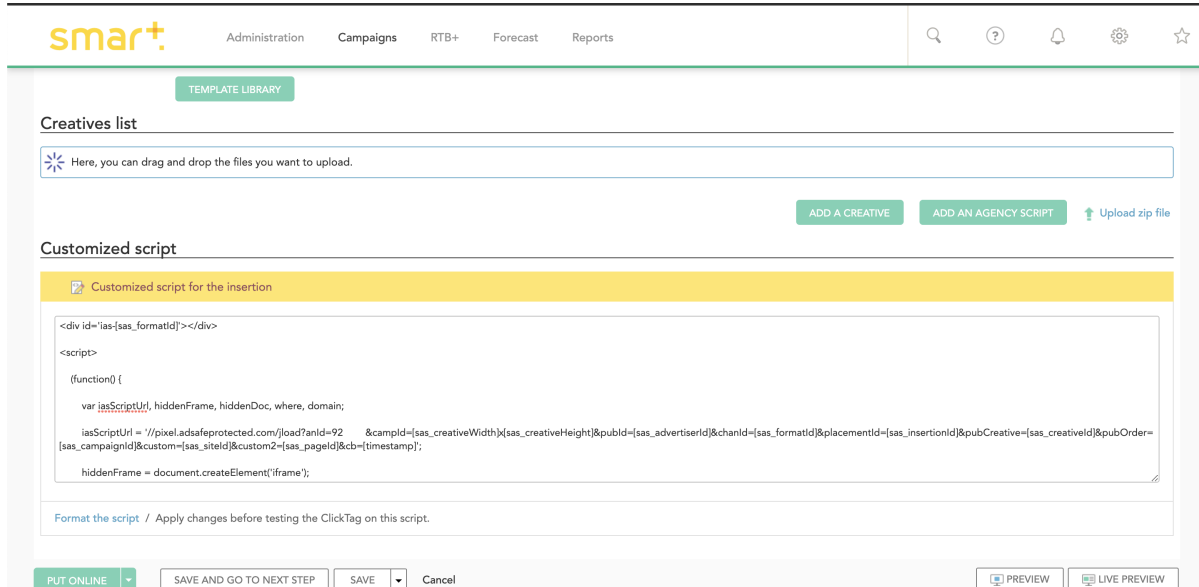
Specification:

```
<div id="ias-[timestamp]"></div>
<script>
(function() {
var ias = document.createElement('script');ias.src =
'//pixel.adsafeprotected.com/jload?anId=[Client ID]&campId=[sas_
creativeWidth]x[sas_creativeHeight]&pubId=[sas_advertiserId]&chanId=[sas_
formatId]&placementId=[sas_insertionId]&pubCreative=[sas_creativeId]&pubOrder=
[sas_campaignId]&custom=[OPTIONAL]&custom2=[OPTIONAL]&custom3=[OPTIONAL]';
document.getElementById('ias-[timestamp]').parentNode.appendChild(ias);
})();
</script>
```

Where:

anId	Client ID assigned and provided by IAS Professional Services during implementation.
campId	The size of the trafficked creative in the ad server (for example, "728x90" or "300x250").
pubId	An identifier for the advertiser that purchased the current ad.
chanId	An identifier for the served ad unit.
placementId	An identifier for the line item / flight for the current ad.
pubCreative	An identifier for the creative served for the current ad.
pubOrder	An identifier for the order / campaign for the current ad.
custom / custom2 / custom3	Key/values to track in IAS reports such as position, traffic source, page type, etc. Omit the parameter if it is not used. Use these for things that have <= 200 unique values, so no IP addresses, user IDs, zip codes, etc.

You can plug this tag on any third party script for any line item. To plug this tag across all the placements, ask your Amazon Transparent Ad Marketplace (Amazon TAM) representative.



smart+ Administration Campaigns RTB+ Forecast Reports

TEMPLATE LIBRARY

Creatives list

Here, you can drag and drop the files you want to upload.

ADD A CREATIVE ADD AN AGENCY SCRIPT Upload zip file

Customized script

Customized script for the insertion

```
<div id="ias-{sas_formatid}"></div>
<script>
(function() {
var iasScriptUrl, hiddenFrame, hiddenDoc, where, domain;
iasScriptUrl = "/pixel.adsafeprotected.com/jload?anId=92 &campId={sas_creativeWidth}x{sas_creativeHeight}&pubId={sas_advertiserId}&chanId={sas_formatid}&placementId={sas_insertionId}&pubCreative={sas_creativeId}&pubOrder={sas_campaignId}&custom={sas_siteId}&custom2={sas_pageId}&cb={timestamp}";
hiddenFrame = document.createElement('iframe');
```

Format the script / Apply changes before testing the ClickTag on this script.

PUT ONLINE SAVE AND GO TO NEXT STEP SAVE Cancel PREVIEW LIVE PREVIEW

Publisher Verification Pixel - Other Ad Servers

The IAS Publisher Verification Pixel also works on other ad servers but requires that publishers insert their ad server's macros in the appropriate places in the IAS JavaScript snippet. The IAS Publisher Verification Pixel should work on all common tag formats, and supports full functionality when embedded directly on a page, in friendly iframe, or in a cross-domain iframe.

The IAS Publisher Verification Pixel as specified below can be used on either HTTP or HTTPS pages.

Specification for display tags (desktop and mobile web):

```
<script src="//pixel.adsafeprotected.com/jload?anId=[Client Id]&campId=[Creative Size]&pubId=[Advertiser]&chanId=[Ad Unit]&placementId=[Line Item]&pubOrder=[Order]&pubCreative=[Creative]&custom=[Optional]&custom2=[Optional]&custom3=[Optional]">
</script>
```

Specification for display tags (mobile app):

```
<script src='mraid.js'></script><script
src="https://pixel.adsafeprotected.com/jload?anId=[Client Id]&campId=[Creative Size]&pubId=[Advertiser]&chanId=[Ad Unit]&placementId=[Line Item]&pubOrder=[Order]&pubCreative=[Creative]&custom=[Optional]&custom2=[Optional]&custom3=[Optional]">
</script>
```

Specification for video wrapper (desktop and mobile web):

```
<script src="https://vast.adsafeprotected.com/vast?anId=[Client
ID]&pubId=%eadv!&chanId=%epid!&placementId=%eaid!&pubCreative=%ecid!&pubOrder
=%ebuy!&cb=%CACHEBUSTER%&custom=[OPTIONAL]&custom2==[OPTIONAL] custom3==
[OPTIONAL]&originalVast=INSERT_VAST_URL"></script>
```

Where:

anId	Client ID assigned and provided by IAS Professional Services during implementation.
campId	The size of the trafficked creative in the ad server (for example, "728x90" or "300x250").
pubId	An identifier for the advertiser that purchased the current ad.
chanId	An identifier for the served ad unit.
placementId	An identifier for the line item / flight for the current ad.
pubOrder	An identifier for the order / campaign for the current ad.
pubCreative	An identifier for the creative served for the current ad.
custom / custom2 / custom3	Key/values to track in IAS reports such as position, traffic source, page type, etc. Omit the parameter if it is not used. Use these for things that have <= 200 unique values, so no IP addresses, user IDs, zip codes, etc.

Note: Mobile app video requires AVID and OMID nodes in the tag and doesn't require the video wrapper. For example:

```
<Extensions>
  <Extension type="AdVerifications">
    <AdVerifications>
      <Verification vendor="company.com-omid">
        <JavaScriptResource apiFramework="omid" browserOptional="true"><![CDATA[
https://company.com/omid.js ]]></JavaScriptResource>
      </Verification>
    </AdVerifications>
  </Extension>
</Extensions>
```

ID Attribution Specifications

The following table summarizes the values that clients can populate using the macros available through their ad server. The UI labels are all assigned macro names for the appropriate IAS / UI naming convention. IAS supports up to 40 alphanumeric characters for all client provided ID attributes.

Information Type	Parameter	Appears in UI as	Required?	IAS Specification
Client ID	anId	N/A	Required	anId=Integer Provided by IAS
Advertiser ID	pubId	Advertiser	Required	Alphanumeric 50 character limit Unique Value Limit = 7500
Creative Size	campId	Size	Required	Alphanumeric 50 character limit Unique Value Limit = 500 Note: You may also substitute other values instead of size that a publisher would like to track, such as "page template type", "traffic source code", etc.
Ad Unit ID	chanId	Ad Unit	Required	Alphanumeric 50 character limit Unique Value Limit = 7500
Line Item / Flight ID	placementId	Line Item	Required	Alphanumeric 50 character limit Unique Value Limit = 7500
Order / Campaign ID	pubOrder	Order	Required	Alphanumeric 50 character limit Unique Value Limit = 7500
Creative ID	pubCreative	Creative	Required	Alphanumeric 50 character limit Unique Value Limit = 7500
Custom	custom / custom2 / custom3	Custom (configurable); generally key/values passed to the ad server	Optional	Alphanumeric 50 character limit Unique Value Limit = 200

Note: Publishers can assign any ID values up to 50 alphanumeric to these named value pairs. IAS recommends passing all of the optional attributes listed above, however they are not mandatory. If publishers choose not to specify a parameter, remove it rather than pass it with an empty or dummy value.

Native Ad Implementation Overview

IAS introduced a way to properly locate native ads. In order to trigger the native ad logic, you need to do the following:

1. Use a unique CSS ID selector with the placement.
2. Append the `ias_adpath` parameter to IAS's pixel.

Additional information:

- This functionality works as expected in direct and friendly iframe scenarios
- This is not supported in cross-domain iframe scenarios.

Step 1 - Use a unique CSS ID selector with the Placement.

You need to make sure the placement has a unique CSS ID selector. This ID selector can be either already on the placement's div or dynamically set at runtime. In both cases, it needs to be on the page before IAS's pixel is placed.

Step 2 - Append the `ias_adpath` parameter to IAS's pixel.

Once IAS knows the ID, IAS provides it to the pixel:

```
iasScriptUrl = '//pixel.adsafeprotected.com/jload?anId=[Client ID]&campId=%WIDTH%x%HEIGHT%&pubId=%eadv!&chanId=%epid!&placementId=%eaid!&pubCreative=%ecid!&pubOrder=%ebuy!&custom=[OPTIONAL]&custom2=[OPTIONAL]&custom3=[OPTIONAL]&ias_adpath=%23our-ad'
```

This pixel now attempts to find an HTML element with 'our-ad' CSS ID.

```
<div id = 'our-ad'></div>
```

Notice the %23 in the pixel. This is the URL encoded # symbol that you need to include.

Note: Adding the param in the following way does not work:

`ias_adpath=our-ad`

You need to add %23 here. For example: `ias_adpath=%23our-ad`

Using the Native Ad Solution with HTML5 Ads

The IAS pixel expects to be in the same container as the ad creative. With native ad support, you can specify another ID in a separate container to track the ad.

Using the Native Ad mechanism, IAS can point to a specific div inside the HTML5 structure.