

React Integration

Last Modified on 02/27/2023 6:11 pm EST

Select from the tabs below to view the indicated information. Read and follow all of the instructions to ensure that your integration goes smoothly.

Introduction

Introduction

The React implementation is meant to be for websites running Single Page Applications (SPA) using React.

If you have an SPA site using Angular, Vue or similar framework please refer to our [Event-based Implementation](#).

If you have a regular HTML site please refer to our [Standard Implementation](#).

The documentation provided is an example of what your code may look like. Please use the tags provided by your Onboarding Specialist or Customer Service Manager to ensure that your site-specific tags are used.

Overview of changes to implement Freestar's Component

1. Install the React Component.
2. Create your code using the Component Code provided on the Component Tab and the site-specific information provided in your tag file.
3. PR your changes to Master.
4. Once approved, merge your branch to master.
5. Switch to the master branch and run the build.
6. Publish the package using np.
7. Place the CMP CCPA and GDPR links on your site.
8. Set up Freestar's Dynamic ads.txt redirect at least 2 days before your site goes live.

See the Component and Ads.txt tabs of this document for more information on how to place this code on your page.

Additional considerations

There are a few additional considerations that may need to be captured as we proceed with the

integration:

- Tags are built on standard Google Publisher Tag practices.
- If you already use a header bidder wrapper solution, disable it when Freestar is called. This ensures that header bidders are only called by one platform.
- In order for us to perform optimally, our code needs to be on the root page, allowing bidding partners access to a user's cookies and other identifiers.
- If you are using a different URL for your test site where the root domain does not match the domain that Google approved for your site you will need to add additional code to the head of your page. See the Test Page tab for more details.
- Freestar's code does not work with prototype.js, this will need to be removed from your site to add Freestar.

Ad Code

Freestar Ad Code

Where to get your Freestar ad code

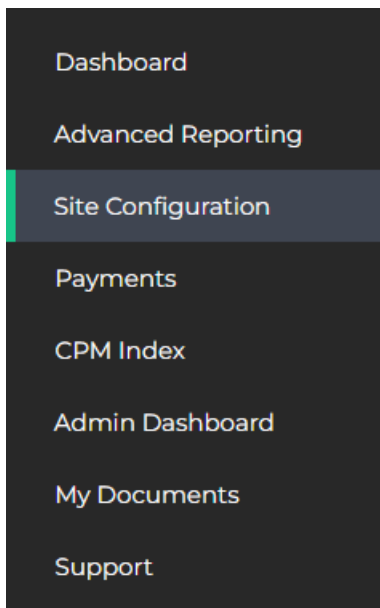
- You can download your tags from the Freestar Dashboard. Your unique ad tags will be in a .txt file. You can open this file in any document or code editor. The file will be unique for each of your domains. The Freestar tags will replace your current provider's tags on page.

- **How to find your ad tags**

1. Login to the [Freestar Dashboard](#).
2. Select the **3 lines icon** in the top left to open the left menu.



3. Select **Site Configuration** in the left menu.



4. Select the Ad Tags Tab at the top of the page.
5. Select the type of tags based on your site architecture. This will download your tags.
 - a. Default - Use this for standard HTML tags for a site that is not using Single Page Architecture (SPA). Please use the [Standard integration guide](#) for these types of tags.
 - b. React - Use this for React tags using our React component for sites using Single Page Architecture (SPA).
 - c. Event-Based - Use this for Event-Based tags for sites using Single Page Architecture (SPA) that are not React like Angular or Vue. Please use the [Event-Based integration guide](#) for these types of tags.

Freestar

Site Configuration

Ads.txt App-Adstxt **Ad Tags** API Tokens

Ad Tags

Status	Name	Domain	Default Tags	Event Based Tags	React Tags
●	Andy Test Site	andysite.com	Standard Tags	Event Based Tags	React Tags
●	Demo Ads	demoads.com	Standard Tags	Event Based Tags	React Tags
●	Headphone Review	headphonereview.com	Standard Tags	Event Based Tags	React Tags
●	Lost and Taken	lostandtaken.com	Standard Tags	Event Based Tags	React Tags
●	Publisher Test Site	freestar.com	Standard Tags	Event Based Tags	React Tags
●	Web Design Ledger	webdesignledger.com	Standard Tags	Event Based Tags	React Tags

Notes about your Freestar ad code

- Inside the file you will see pieces of information:

- Variable: eg. `const publisher = 'yoursite-com'`
- Placement Names: eg. `placementName = "yoursite_leaderboard_atf"`
 - There will be a different `placementName` for each ad unit created in Freestar's system.
- These will give you the name of your site variable and the placement tags that will be used with the component. Use the Component tab in this document for the Component code and more information on how to place this code on your pages.
- If you need additional tags please ask your Onboarding Specialist or Customer Service Manager.
- If you receive an error message when pulling your tags, please try disabling your Ad Blocker. If the issue continues to persist, please ask your Onboarding Specialist or Customer Service Manager to send them along.

Component

Component

To create the component you will need the tags provided by your Onboarding Specialist or Customer Service Manager. This will replace the code you had with your previous provider.

1. Install the React Component.

```
npm install --save @freestar/pubfig-adslot-react-component
```

2. Create your code using the Component Code provided below and the site-specific information provided in your tag file.
 - a. Open the site-specific tag file provided by Freestar from your Onboarding Specialist or Customer Service Manager.
 - b. Inside the file look for the following at the top of the file (where `yoursite-com` represents your domain in the Freestar system) and copy the variable value.

```
const publisher = 'yoursite-com'
```

- c. Paste the value over the placeholder '`publisherName`' in the Component Code provided below.

```
const publisher = 'publisherName'
```

- d. Inside the file look for the placement name (where yoursite represents your domain in the Freestar system) and copy the value.

```
placementName = "yoursite_leaderboard_atf"
```

- e. Paste the value over the placeholder 'PublisherName_970x250_728x90_320x50' in the Component Code provided below.

```
const placementName = 'PublisherName_970x250_728x90_320x50'
```

- f. Repeat d-e for each Placement being used on page.
3. PR your changes to Master.
4. Once approved, merge your branch to master.
5. Switch to the master branch and run the build.

```
npm run build
```

6. Publish the package using np.

```
np major|minor|patch
```

7. If you have not done so already follow the instructions on the Ads.txt tab of this document to place your ads.txt lines.

Please use the code provided by your Onboarding Specialist or Customer Service Manager for your site-specific changes.

Component Code

```

import React, { Component } from 'react'

import FreestarAdSlot from '@freestar/pubfig-adslot-react-component'

import './demo.css'

class Demo extends Component {
  render() {
    const placementName = 'PublisherName_970x250_728x90_320x50'
    const slotId = 'in_content_ad_1'
    const publisher = 'publisherName'
    const targeting = { key1: 'value1', key2: 'value2' }
    const { adRefresh } = this.state

    return (
      <div>
        <FreestarAdSlot
          publisher={publisher}
          placementName={placementName}
          slotId={slotId}
          targeting={targeting}
          channel='custom_channel'
          classList={['m-30', 'p-15', 'b-thin-red']}
          onNewAdSlotsHook={({placementName}) => console.log('creating ad', placementName)}
          onDeleteAdSlotsHook={({placementName}) => console.log('destroying ad', placementName)}
          onAdRefreshHook={({placementName}) => console.log('refreshing ad', placementName)}
        />
        <button onClick={this.onAdRefresh}>Trigger Refresh</button>
      </div>
    )
  }
}

export default Demo

```

CCPA & GDPR Links

CCPA & GDPR Links

Why do you need CCPA & GDPR links?

The CCPA & GDPR links are required for privacy regulations in California and for GDPR. CCPA is an opt-out regulation so there is no initial prompt. Users are required to hit a link/button that reads “Do Not Sell My Personal Information” to opt out in California. You are required to add a link to the footer of your website that gives users the ability to opt-out. GDPR is an opt-in regulation so a prompt is required. Users are required to consent to all the different ad partners on page to serve ads. You are required to add a link to the footer of your website that gives users the ability to change their permissions.

If you are using Freestar's Quantcast CMP you will need to place this code on your page to ensure the links appear. Once the user clicks on the link(s) this will pop-up their options on the Quantcast CMP.

If you are using your own CMP you will need to follow the instructions provided by your CMP provider to place your links.

Where should you place the CCPA & GDPR links?

Your CCPA & GDPR links should be placed in the footer of your page.

When should you add the CCPA & GDPR links?

The CCPA & GDPR links should be added when you are adding Freestar's code to your page. We recommend doing this on your test page so we can test to ensure that it is working as expected before going live. Even though the CCPA & GDPR protections are for California and the EEA we ask that you add the link for all users to ensure you remain compliant.

How to add the CCPA & GDPR links

Add the code below to the footer of your page.

```
<footer>
<a class="change-consent" onclick="window.__uspapi('displayUspUi');">Do Not Sell My Personal Information</a>
<a class="change-consent" onclick="window.__tcfapi('displayConsentUi', 2, function() {});">Change Consent</a>
</footer>
```

Ads.txt

Ads.txt

Why we use Dynamic Ads.txt?

We utilize a dynamic ads.txt product so that you will always have the most up-to-date version of ads.txt. To add this you will need to create a redirect to a file on our servers. Using our Dynamic Ads.txt will remove the monthly task of you having to update manually.

When should your Ads.txt file be updated?

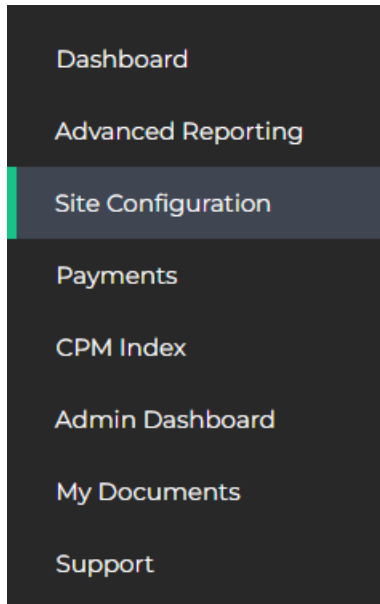
Your ads.txt lines should be updated on your root domain (<https://yoursite.com/ads.txt>) a **minimum of 2 days before your expected launch date**. Freestar's Google ads.txt line needs to be in place before you can test our code on your test environment or page. **For best results, we recommend that you update to use our dynamic ads.txt with all of Freestar's lines during testing as we already added your current provider's lines.**

How to find your dynamic ads.txt redirect URL

1. Login to the [Freestar Dashboard](#).
2. Select the **3 lines icon** in the top left to open the left menu.



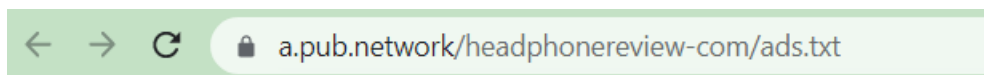
3. Select **Site Configuration** in the left menu.



4. Next to the site that you would like to edit select the "**View Fs Ads.txt**" button in the main section of the page.

Ads.txt Status				
Status ▾	Name ▾	Domain ▾	Type	Operations
●	Headphone Review	headphonereview.com	Dynamic	View FS Ads.txt Edit FS Ads.txt ...

5. Copy the URL in your address bar.



6. Follow the steps below in "**How to set up Dynamic Ads.txt**" to redirect your ads.txt from <https://yoursite.com/ads.txt> to the redirect URL you copied.

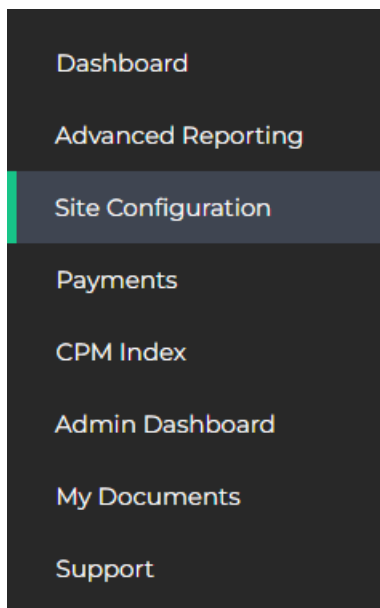
How to set up Dynamic Ads.txt

Find instructions in the Freestar Dashboard

1. Login to the [Freestar Dashboard](#).
2. Select the **3 lines icon** in the top left to open the left menu.



3. Select **Site Configuration** in the left menu.

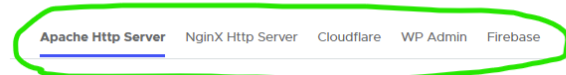


4. At the bottom of the page, you will find instructions for the different methods for creating a redirect. Select the tab that is appropriate for your setup.

Implementing Dynamic Ads.txt

You need to install a redirect to the freestar hosted Ads.txt file at the **root level** of your domain.

For additional questions please contact your Account Manager [here](#)



Apache Http Server Instructions

1. Select the "**Apache .htaccess**" option in the "**Copy Redirect**" menu for the site you want to set up.
2. If your site does not have an **.htaccess** file:
 1. Create a text file and paste the copied text in the first line.
 2. Name the text file **.htaccess**
 3. Login to your server via **FTP/SFTP** or whatever method you use to upload files to your server.
 4. Upload the file to the root directory.
3. If your site already has an **.htaccess** file:
 1. Using your preferred method of editing files on your server open the existing **.htaccess** file
 2. Insert the copied text into the first line of the **.htaccess** file. The text should be on its own line.
 3. Save the file.
4. To verify it uploaded to the correct spot, you should be able to navigate to **https://yourdomain.com/ads.txt** and verify that the url redirected to **a.pub.network**.

If you are unable to log in you can find the instructions on how to implement this [here](#).

How to edit your custom Ads.txt lines when using Dynamic Ads.txt

For the lines that are specific to your site and outside Freestar's portfolio, you can either edit this in the Freestar Dashboard or ask your Onboarding Specialist or Customer Service Manager to add them to your custom file.

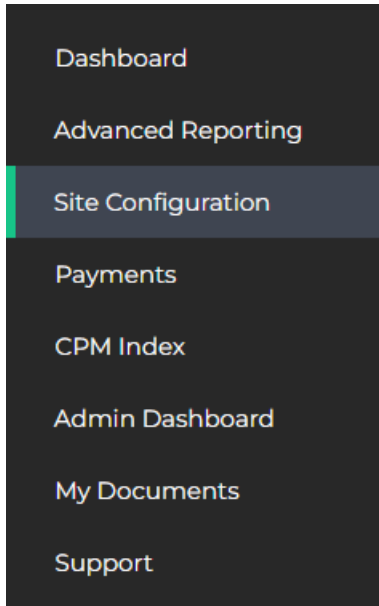
To update in the Freestar Dashboard

Watch a demo of how this is done [here](#).

1. Login to the [Freestar Dashboard](#).
2. Select the **3 lines icon** in the top left to open the left menu.



3. Select **Site Configuration** in the left menu.



4. Next to the site that you would like to edit select the "**Edit Fs Ads.txt**" button in the main section of the page.

Status	Name	Domain	Type	Operations
●	Headphone Review	headphonereview.com	Dynamic	View FS Ads.txt Edit FS Ads.txt

5. Add a new line to the top or bottom of your file.

Headphone Review

Entries to be added to the end of your ads.txt:

```
#my new line  
abc.com 123456789, DIRECT
```

```
Infolinks.com, 3126932, DIRECT  
rubiconproject.com, 20322, RESELLER, Obfd66d529a55807  
google.com, pub-6373315980741255, RESELLER, f08c47fec0942fa0  
google.com, pub-4299156005397946, RESELLER, f08c47fec0942fa0  
Adtech.com, 11409, RESELLER  
annexus.com, 3251, RESELLER
```

[Save Ads.txt](#)

- a. Use # at the beginning of any comment lines.

- b. ■ Note: It is very important that you do not delete any of your existing lines without checking with your Freestar representative as some lines are very important and may cause ads to stop serving if removed.

6. Select **Save Ads.txt**.

Testing

Creating a Test Page

Using the tags provided by your Onboarding Specialist or Customer Service Manager follow the instructions provided on the Component and ads.txt tabs in this document to make the changes on your test environment or page.

Test page set-up

- We would prefer if a test environment can be provided so we can test several pages with different layouts before you go live. If this is not possible a test page is acceptable.
- Use a test environment or page with the same code as a regular page on your site, using all the same plugins and third-party scripts (except your previous provider's) so we can check for code conflicts.
- We prefer that your test environment or page not have live traffic. This allows us the opportunity to test everything before you set it live.
- Replace your current provider's tags on page with the Freestar tags.
- Please do not place our code on any pages where you do not want ads to run or pages with no content like a Login page.
- Do not push tags to production until Freestar gives the green light as we like to ensure everything is set up properly and we do not disturb user experience.
- Freestar's code does not work with prototype.js, this will need to be removed from your site to add Freestar.

Code changes if using a different root domain for your test environment

- If you are using a different URL for your test environment or page where the root domain does not match the domain that Google approved for your site you will need to add an additional line of code to the head of your page. Add the following line of code to the head of your page.
 - Code to add (replace `https://yoursite.com` with the root domain we had approved with

Google)

```
var freestar = freestar || {};  
freestar.queue = freestar.queue || [];  
freestar.queue.push(function () {  
  googletag.pubads().set('page_url', 'https://yoursite.com');  
});
```

Code changes if using subdomains with Quantcast's CMP

- If you are using a subdomain for your test environment, page or live site where your Quantcast Choice CMP was only set up with the root domain you will need to edit the host line of code provided by Quantcast to allow ads to serve on all subdomains. This allows you to use the same code on each subdomain while telling Quantcast to use the root domain.
 - If you did set up each subdomain in the Quantcast UI for your CMP you can skip this step, otherwise, follow the steps below.
 - Code to adjust (replace yoursite.com with the root domain you added in the Quantcast UI)

```
var host = "yoursite.com";
```

- Full Example:

```
(function() {  
  var host = "yoursite.com";  
  var element = document.createElement('script');  
  var firstScript = document.getElementsByTagName('script')[0];  
  var url = 'https://quantcast.mgr.consensu.org'  
    .concat('/choice/', '1_3qeSjJG-hKn', '/', host, '/choice.js')  
  var uspTries = 0;  
  var uspTriesLimit = 3;  
  element.async = true;  
  element.type = 'text/javascript';  
  element.src = url;  
  firstScript.parentNode.insertBefore(element, firstScript);  
});
```

Code changes if you would like a 3rd party script to wait for Freestar's Quantcast CMP consent before loading

- If you need to load a script after we receive consent from our Freestar Quantcast CMP you can use the Freestar Queue as this will only load if consent is given.
- First, you will need to make sure the code is added after our Queue is initialized.

```
freestar.queue = freestar.queue || [];
```

- Second, you will add the code to push information to our queue.

```
<script>freestar.queue.push(function() {  
  });  
</script>
```

- Third, you will add your third party code inside our push function.

```
<script>freestar.queue.push(function() {  
  //enter your code here  
});  
</script>
```

How to Test

- To test that the ads are loading as expected you can use a URL parameter `?dfpkey=fsads` to load our House Campaigns. This will allow you to see the ads filling the slots and ensure you have placed the code in the correct locations.
 - Example
 - URL: <https://yourtestdomain.com/?dfpkey=fsads>
- Double-check that you made all the required code replacements by searching for your previous provider in our code.

Ready for Freestar to Test

Once your test environment or page is ready for us to test please provide the URL and any login credentials to your Onboarding Specialist or Customer Service Manager. Our team will run a series of tests to ensure everything is set up as expected and there is no conflicting code on your site. We will let you know if there are any issues or edits that need to be made.

Test Page Passes Testing

Once your test environment or page has passed testing we will then schedule a launch date with your team. Please do not launch without checking with our team first as we want you to be set up for success.

Props

Props

`publisher` A *required* string of the publisher, which will be provided by Freestar.

`placementName` A *required* string of the ad unit placement, which will be provided by Freestar.

`slotId` An *optional* string to specify the element id of the containing div around the adslot. Defaults to the placement.

targeting An *optional* object of key/value pairs for targeting.

channel An *optional* string of a custom channel to use.

classList An *optional* array of strings representing any additional classes that should be applied to the wrapper dom element of the ad slot.

adRefresh An *optional* number bound to the ad refresh. Increment this value to trigger a refresh of the ad slot. Please consult your Freestar support team for more information.

onNewAdSlotsHook An *optional* event hook that returns the `placementName` when the component **mounts** and an ad is requested.

onDeleteAdSlotsHook An *optional* event hook that returns the `placementName` when the component **unmounts**.

onAdRefreshHook An *optional* event hook that returns the `placementName` when the component refreshes an ad.

integrity An *optional* attribute that when passed enables SRI for our pubfig library. The component will use this value for the integrity attribute when loading pubfig

Pageview Tracker

Freestar Pageview Tracker

How to Track Virtual Page Views Through Pubfig

Pubfig version 4.6.0 +

Pubfig.js collects data values such as URL location which is then used in various tables. In order to properly track data sites that are using Single Page Applications (SPAs), or sites with slideshows/carousels that change URLs / URL parameters the publisher must take these new actions to assure the accuracy of the collected data.

When the location and or URL is updated, the lifecycle of the DOM and or Window does not reload the pubfig.js script. In order to address this, the publisher must invoke the `freestar.trackPageview()` method. This will ensure that the new URL is stored and used throughout the data collection for that page or view.

In order to fend off timing issues, the function must be invoked within the Freestar queue. e.g.:

```
// (1) new view / url state updated

// (2) upon completion of the view being mounted invocation of the trackPageview method within the queued method

freestar.queue.push(function(){
  freestar.trackPageview()
});
```

Advanced

Advanced Implementation

API Methods

- **Key-Values**

- **FreestarAdSlot.setPageTargeting** Proxy for the GPT setTargeting call to set page level targeting. See Freestar documentation [here](#).
- **FreestarAdSlot.clearPageTargeting** Proxy for the GRP clearTargeting call to clear page level targeting. See Freestar documentation [here](#).

- **Pageviews**

- **FreestarAdSlot.trackPageView** Proxy for the freestar.trackPageview() method.
 - Freestar collects data values such as URL location which is then used in various tables. In order to properly track data sites that are using Single Page Applications (SPAs), or sites with slideshows/carousels that change urls/url parameters these new actions must be taken by the publisher to assure the accuracy of the collected data. When the location and/or URL is updated the lifecycle of the DOM and/or Window does not reload the pubfig.js script. In order to address this the publisher must invoke the freestar.trackPageview() method. This will ensure that the new url is stored and used throughout the data collection for that page or view. See [instructions for more details](#).

Queuing Freestar Ad Placements

- If you would like to allow the Freestar library to preload but need to hold off on ad delivery until business logic has completed please follow the instructions [here](#)

For more details please see our GitHub:

<https://github.com/freestarcapital/pubfig-adslot-react-component>