Analytics Integrations

API Guide and Reference
Updated: Sep 24, 2019
Version: 1.0

Integrations

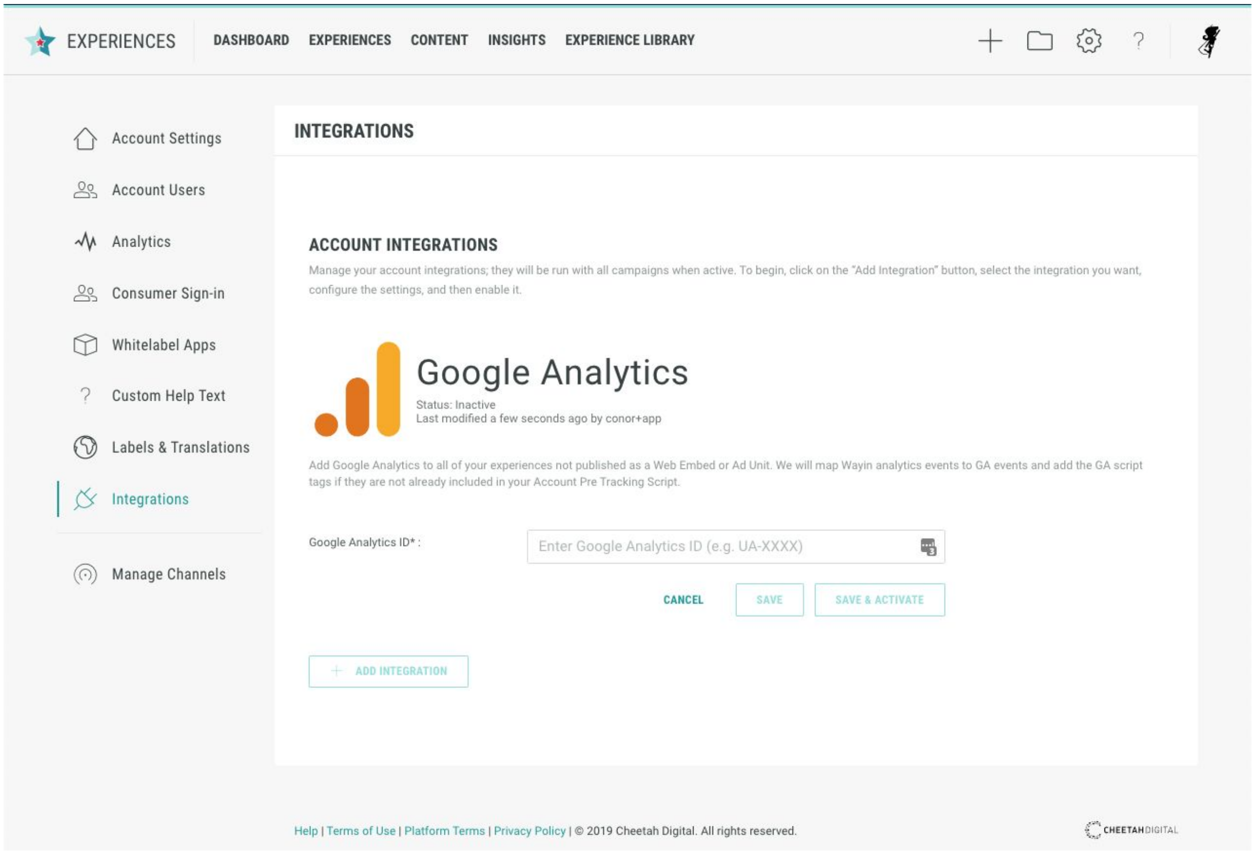
Cheetah Experiences has developed several out of the box integrations with our customers' preferred analytics and tag management platforms. These integrations will get you hooked into our event system with minimal setup.

Out of the Box Integrations

Google Analytics

From within the Cheetah Experiences Platform:

1. Navigate to your account integration management area `/admin/account/integrations`
2. Click the “Add Integration” button
3. Select “Google Analytics” from the available options
4. Configure your Google Analytics integration
    a. Provide the Google Analytics ID
5. Click “Save & Activate”



INTEGRATIONS

ACCOUNT INTEGRATIONS

Manage your account integrations; they will be run with all campaigns when active. To begin, click on the “Add Integration” button, select the integration you want, configure the settings, and then enable it.

Google Analytics

Status: Inactive
Last modified a few seconds ago by conor+app

Add Google Analytics to all of your experiences not published as a Web Embed or Ad Unit. We will map Wayin analytics events to GA events and add the GA script tags if they are not already included in your Account Pre Tracking Script.

Google Analytics ID* : Enter Google Analytics ID (e.g. UA-XXXX)

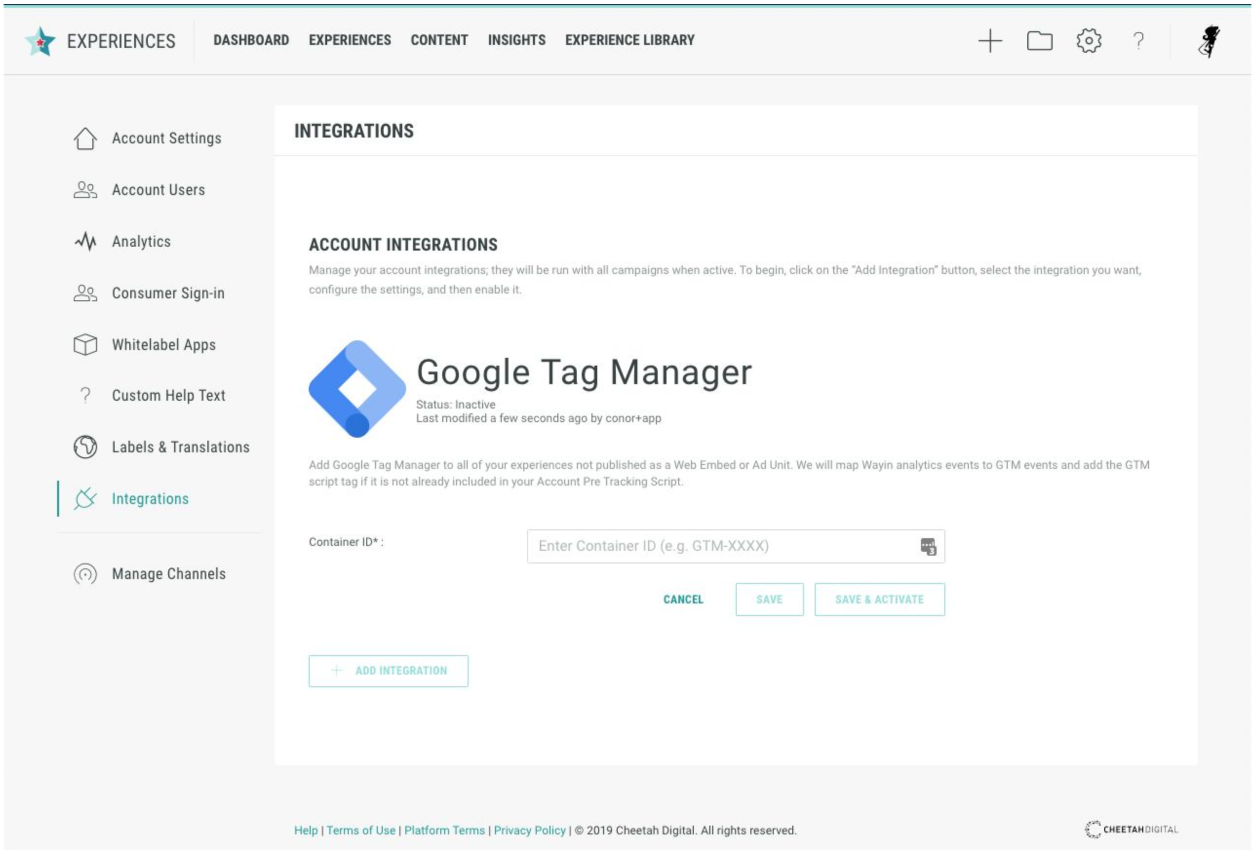
Help | Terms of Use | Platform Terms | Privacy Policy | © 2019 Cheetah Digital. All rights reserved.

We will include GA scripts in your experiences, except for the noted publishing destinations, and dispatch all Cheetah Experiences analytics events to your GA account. If you want greater control over your analytics (e.g. you only want to receive a subset of the events or wish to customize the Category, Action, or Label) you can create your own analytics script in the Account Analytics section of the Experiences Console `/admin/account/tracking`.

Google Tag Manager

From within the Experiences Console:

1. Navigate to your account integration management area `/admin/account/integrations`
2. Click the “Add Integration” button
3. Select “Google Tag Manager” from the available options
4. Configure your Google Tag Manager integration
    a. Provide the Container ID
5. Click “Save & Activate”



INTEGRATIONS

ACCOUNT INTEGRATIONS

Manage your account integrations; they will be run with all campaigns when active. To begin, click on the “Add Integration” button, select the integration you want, configure the settings, and then enable it.

Google Tag Manager

Status: Inactive
Last modified a few seconds ago by conor+app

Add Google Tag Manager to all of your experiences not published as a Web Embed or Ad Unit. We will map Wayin analytics events to GTM events and add the GTM script tag if it is not already included in your Account Pre Tracking Script.

Container ID* : Enter Container ID (e.g. GTM-XXXX)

We will include GTM scripts in your experiences, except for the noted publishing destinations, and dispatch all Cheetah Experiences analytics events to your container for you to manage.

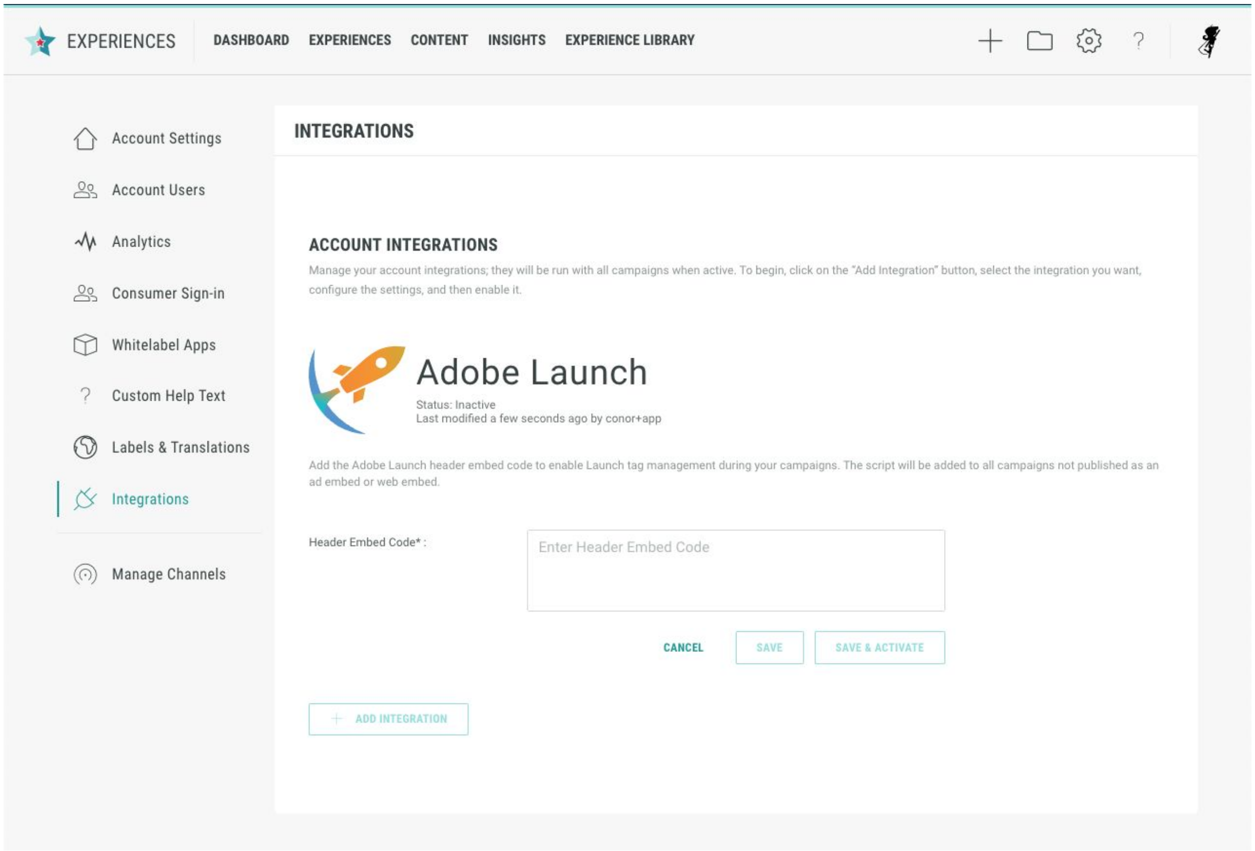
Adobe Launch

As part of our partnership with Adobe, we have developed an integration with the Adobe Launch tag management system. You can find us listed in Adobe Exchange.

After you have configured your Adobe Launch instance to listen for Experiences events, you'll need to configure your Cheetah Experiences account to send events to Launch.

From within the Experiences console:

1. Navigate to your account integration management area `/admin/account/integrations`
2. Click the "Add Integration" button
3. Select "Adobe Launch" from the available options
4. Configure your Launch integration
    a. Provide the Header Embed Code generated by Launch
5. Click "Save & Activate"



INTEGRATIONS

ACCOUNT INTEGRATIONS

Manage your account integrations; they will be run with all campaigns when active. To begin, click on the "Add Integration" button, select the integration you want, configure the settings, and then enable it.

Adobe Launch

Status: Inactive
Last modified a few seconds ago by conor+app

Add the Adobe Launch header embed code to enable Launch tag management during your campaigns. The script will be added to all campaigns not published as an ad embed or web embed.

Header Embed Code* : Enter Header Embed Code

We will include the embed script in your experiences automatically, except for the noted publishing destinations, and dispatch all Wayin analytics events to Launch for you to manage.

Publishing Destination Differences

Web Embeds

For Cheetah Experiences published as web embeds, we do **not** include analytics as part of the experience. This is to avoid double counting as most customers will already have analytics tracking setup on the parent page and choose to manage Experiences events there. See Custom Integrations for how to do this.

Ads

For Cheetah Experiences published as ad units, we do **not** include analytics as part of the experience. This is so customers don't incur excessive costs from the potentially high viewership of ad units. Additionally, ad providers already provide many analytic capabilities. You can always add custom analytics to any experience. See Custom Integrations for how to do this.

Custom Integrations

Some customers use analytics tools that we do not currently integrate with; please let us know at support@wayin.com if you are using a service that we don't have an out-of-the-box integration with, so we can work it into our roadmap. Others prefer greater control over their analytics and choose to write their own integration.

Subscribing to the event bus

Most of the time you will want to listen for events from within the experience embed, but if you are publishing your experience as a web embed, then you will likely want to setup your listeners in the parent page so you can combine your Experiences analytics events with those arising from the rest of the page. In all other publishing destinations, you will need to setup your listeners inside the embed.

When inside the Experiences embed

Typically you will want to setup your custom analytics at the account level so that all experiences you publish are measured. Navigate to /`admin/account/tracking` and add your JavaScript to the post-tracking script.

Note: Do not use a pre-tracking script as the NGX object will not have been initialized by the time this is executed.

```
NGX.api('analytics:listen', <eventName>, function (payload?) {
 // Your custom code here
});
```

`<eventName>` - replace this with the event string you want to listen for. (e.g. “form:submit”)

`payload` - this is an optional parameter in the callback that certain events provide. You can refer to the individual events below to see the payload for each

When outside the Experiences embed (parent page)

```
window.addEventListener('ngx:integration:response', (event) => {
 const { NGX, id } = event.detail;
 NGX.api('analytics:listen', <eventName>, (payload?) => {
   // Your custom code here
 });
});
```

`NGX` - This is an instance of the Display API. You listen for events on it the exact same way as if you were inside the embed.

`id` - This is the experience id. Its primary use is when you have multiple Cheetah Experiences on the same page and need to keep track of multiple `NGX` objects.

```
window.dispatchEvent(
  new CustomEvent('ngx:integration', { detail: { id: <id> } })
);
```

Place this snippet **after** the one above where you add the event listener. This will register your page with our experience(s). Once we receive this event we will reply with the “ngx:interaction:response” event registered above so you can begin receiving events.

Note: In the event that you have multiple experiences embedded on the same page, you should only dispatch this event once.

`<id>` - This is any string that identifies you to us. Your company name would be a good value to provide.

Events

While many of our events are supported in all versions of our API, some features are not yet available in the new API and others, (e.g. “navigate:slide”), are not applicable in the old API.

Payloads

Customers typically use a few different values to uniquely identify their experiences and associate events with them. Each of our analytics events includes the following contextual values to help you associate the events; these values are stored in a container object called `info`.

`trackingReference` - this is tracking code you typically use for external reporting and analytics systems. It can be set in the “Settings” screen of the experience editor.

`experienceId` - this is the unique ID for the experience generated by Cheetah Experiences.

`externalReference` - this is a unique ID/code/reference that you provide to identify the experience in your marketing systems. We will automatically set this for you when you create a new experience which you can replace with your own value in the settings screen.

Each event may also supply additional relevant data which you can find listed below.

Application

Event	Description	Payload
app:viewport	The experience enters the user's browser viewport	

Navigation

Event	Description	Payload
navigate:page	User navigates to `<pageName>`	`pageName`
navigate:slide	User navigates to `<slideId>`	`pageName, slideId`

Forms

Event	Description	Payload

form:submit	User clicks the Submit button and all validation has passed	
form:entry:alreadyentered	The form submission is rejected because the user has already entered	
form:entry:failed	The form submission is rejected for an unspecified reason (network connectivity, suspected spam, etc...)	
form:entry:success	The form submission is accepted	
form:module:interaction	Similar to `form:field:blur`, except it will fire a maximum of one time per module. This means for a module like Name that has more than 1 input, you may see multiple blur events, but a single interaction event	`remoteFieldName`

UGC

Event	Description	Payload
action:share:campaign:twitter	User clicks the campaign Twitter Share button	
action:share:campaign:facebook	User clicks the campaign Facebook Share button	
action:share:campaign:pinterest	User clicks the campaign Pinterest Share button	
action:share:content:twitter	User clicks a Twitter Share button on a piece of content	
action:share:content:facebook	User clicks a Facebook Share button on a piece of content	
action:share:content:pinterest	User clicks a Pinterest Share button on a piece of content	

vote:success	A user's vote is accepted	
action:content:click	User clicks on a piece of social content to show the detail view	`url`

User Actions

Event	Description	Payload
action:cta	User clicks a CTA. Useful for tracking conversions when the CTA occurs after a form submission.	`url`
action:goal	User clicks the goal action	
action:showmore	User initiates a "Show more" event on a collection or entries grid	
action:next	User clicks the Next button on a carousel or form wizard	
action:previous	User clicks the Previous button on a carousel or form wizard	