Integrate Optimizely with Adobe Analytics

Skip Ahead

1. Enable in Optimizely
2. Check the custom report for your Optimizely experiment
3. Set up a custom conversion variable
Troubleshoot your integration

This article will help you:

• Integrate Adobe Analytics (Omniture SiteCatalyst) to read experiment data from Optimizely
• Set up custom conversion variables (eVars) and traffic variables (prop)
• Solve common troubleshooting issues with integration

Optimizely's integration with Adobe Analytics (formerly Omniture SiteCatalyst) lets you view the analytics details for visitors who've:

• Seen a specific variation of your Optimizely experiment
• Seen a specific experience of your Optimizely Personalization campaign

This article walks you through how to set up this integration, both in Optimizely and Adobe Analytics.

Tip:

Using Adobe Tag Manager? Some customers have provided a third-party solution for completing this integration using Dynamic Tag Management DTM. This isn't a supported use case, but some customers use it to integrate Optimizely experiments through DTM.
1. Enable in Optimizely

1. Enable the integration for your Optimizely project, if you haven't already.

2. Navigate to an individual experiment or Personalization campaign that you'd like to track in Adobe Analytics.

3. Under Manage Campaign, click Integrations.

4. Fill out the Adobe Analytics integration form by specifying a Conversion Variable and Traffic Variable.

![Integration Form](image)

Under Conversion Variable, select the eVar that you want Optimizely to use. The eVar should not be in use by another part of your site or a concurrently running Optimizely experiment.

The sVariable is optional. Some sites use custom Adobe Analytics implementations with special sVariable names. The sVariable contains all of the tracking tags to be sent to Adobe Analytics for a given visitor. If you want to specify a custom sVariable, you can do so in the corresponding field when setting up the integration on an experiment level. If the parameter is omitted, Optimizely will use s by default.

Custom "one eVar" Integration is available here, but we do not officially support it: [https://optimizely.atlassian.net/wiki...pageId=2719795](https://optimizely.atlassian.net/wiki...pageId=2719795)

2. Check the custom report for your Optimizely experiment

Once you've enabled the integration, you can check the custom Adobe Analytics report for your experiment to make sure that it's working as expected.

When you log in to your Adobe Analytics account, you should see a dashboard that looks something like this (the example below shows SiteCatalyst 15):
1. To get to the Optimizely custom variable, click **Custom Conversion > Custom Conversion XX-XX > Optimizely (vXX)**.

The XX represents the eVar you selected when you enabled the Optimizely integration.

2. Click the Optimizely custom conversion to go to the custom report.
3. To check how specific custom events performed across variations, click Add Metrics.

4. Drag the custom event you want to display from the Available Metrics list to the Report Data Columns Canvas list to add it to the report.
5. When you're finished adding events to the report, click OK. You should see a modified report that shows the conversion data for the custom event.

If you're using Optimizely Web Personalization, filter the report for the words "holdback" to see only the data from visitors that were included in your Optimizely campaign.

Currently, Personalization tracks events also for visitors who are included in the campaign holdback. Read more about the updated naming conventions and analytics strings Optimizely is using.

To drill down the report by other parameters, click Broken Down by: None.
3. Set up a custom conversion variable

With custom conversion variables (“eVars”) in Adobe Analytics, you can capture conversion events or other attributes specific to your webpage. Just like custom variables in Google Analytics, you can use custom conversion variables to segment your reports for fine-grained analysis. When combined with integration in Optimizely, eVars let you segment your reports by your experiments and identify the best-performing variation.

To reserve an eVar for your Optimizely experiment:

2. Click Edit Settings > Conversions > Conversion Variables.
3. Click Add New eVar.

After adding your new eVar, you'll need to choose a unique name, allocation type, cookie expiration, data type, and relation status. Not sure which values to choose? Here are the default values we recommend:
• Allocation: Most Recent (Last)

• Expire After: Visit

• Type: Text String

• Status: Basic Subrelations

Allocation is used to determine which eVar value should be associated with a given visitor. If the visitor happens to trigger multiple events, you might want to attribute this to the last value they were assigned so that the revenue and participation metrics are attributed to this last value. Choosing "Linear" allocation divides the revenue and success goals evenly over all events. If you want to view your success events across multiple eVar values, you can do that with participation.

Check out this article for more about how allocation differs from participation and to learn how to set up participation.

Troubleshoot your integration

If you’re having trouble with Optimizely’s Adobe Analytics integration, here are a few things to investigate that may help resolve the issue.

Tools to help verify things are working correctly

Here are some tools to help debug a Adobe Analytics integration:

• A Chrome Browser extension that detects when a tracking call has been made and what values have been sent along with it

• An Adobe DigitalPulse Debugger that uses a JavaScript bookmarklet to pop-up information about the Adobe Analytics implementation on a page when run

Identify the Adobe Analytics tracking call

The default implementation of Adobe Analytics uses this code to send a tracking call:

```javascript
s.t();
```

It’s a small bit of code, but this function call sends out all of the Adobe Analytics tracking information for the pageview. By default, Optimizely assumes this will be s, but this tracking call may have a different leading variable that isn’t s if you’re using a custom tracker name. If you can’t find a s.t(); call on your page, search for anything resembling .t() in case another call exists with a different leading variable name before the period.

When a decision event is made, Optimizely sends that event to Adobe Analytics as soon as possible, polling for the
availability of the Adobe Analytics object. If the Adobe Analytics object isn't ready when a decision is made, Optimizely
checks back every 200 ms for 10 seconds to see if Optimizely can send the Adobe Analytics string. This helps with
delayed activation experiments (and if they fall under delayed activation, it helps with geo-targeted experiments too).

**Identify a Custom Tracker (sVariable)**

This step is optional if you’re using a default Adobe Analytics implementation. But if you notice a different variable name
other than s before the period for the .t() tracking call, that will be the name of your custom sVariable. For example:

```javascript
foo.t();
```

The custom s variable name in this example is foo, so you'll need to modify your integration settings to reflect this:

Here is an example of an integration that’s working for a custom tracker named omntag:

```javascript
[...]
572        omntag.prop6 = "Home Page";
573        omntag.prop14 = "Home Page";
574        var s_code = omntag.t();
575    s_code & document.write(s_code)
576    }
577    }
catch (e) {

Here is an example of the s object omntag when Optimizely populates eVar1 correctly:

```javascript
> omntag
  > Object {_il: Array[3], _in: 2, _ci: "s_c", version: "H.26.2", am: "0123456
  > omntag.eVar1
  "Optimizely_HompageTest: Variation1"
```

**Access to eVar/prop variables 50 through 100**

Some older versions of Adobe Analytics do not work with all 100 eVar and prop variables that are available for
assignment within Optimizely.

Access to variable slots 51 through 75 was added in Adobe Analytics Code Version H22 (pre-SiteCatalyst 15). You can
find your Adobe Analytics Code Version number with the DigitalPulse Debugger tool. If you have an older version, limit
your eVar and prop variables to 1 through 50 within Optimizely.
Access to variable slots 76 through 100 was added later, but if you’re having trouble populating them, try an eVar or prop in the range of 1 to 50 to see if that helps.